Analyse des algorithmes d'Euclide : une approche dynanique

Benoît Daireaux

Université de Caen Basse-Normandie

U.F.R. Sciences **École doctorale SIMEM**
Département d'Informatique

Analyse des algorithmes d'Euclide : une approche dynamique

THÈSE

présentée et soutenue publiquement le 22 juin 2005

pour l'obtention du

Doctorat de l'Université de Caen
(arrêté du 25 avril 2002)

Spécialité Informatique

par

Benoît Daireaux

Composition du jury

Rapporteurs : Michael DRMOTA, Professeur, Université de Vienne
Jean-Michel MULLER, Directeur de Recherche CNRS, ENS Lyon

Examinateurs : Marie-Pierre BÉAL, Professeur, Université de Marne-la-Vallée
Jean MAIRESSE, Chargé de Recherche CNRS, LIAFA
Véronique MAUME-DESCHAMPS, Maître de Conférences, Université de Dijon
Jean-Michel MULLER, Directeur de Recherche CNRS, ENS Lyon
Bruno SALVY, Directeur de Recherche INRIA, Rocquencourt
Brigitte VALLÉE, Directrice de Recherche, GREYC (Directrice)

Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen
CNRS UMR 6072

Mis en page avec la classe thloria.

Remerciements

Ces remerciements s'adressent en premier lieu à Brigitte Vallée. Ces années passées sous sa direction resteront pour moi comme une expérience enrichissante à tous niveaux, aussi bien scientifiques qu'humains. J'ai pu profiter de son impressionnant savoir scientifique et apprécier son enthousiasme communicatif, sa disponibilité (malgré un agenda de ministre) et son extrême gentillesse. Cette thèse lui doit beaucoup, et pour toutes ces raisons je la remercie.

Je suis également très reconnaissant aux divers relecteurs de ce manuscrit, à commencer par mes trois rapporteurs, Michael Drmota, Jean Mairesse et Jean-Michel Muller. Leurs nombreuses remarques et suggestions m'ont été très utiles, et l'attention qu'ils ont portée à mon travail m'a particulièrement touchée.

J'ai eu la chance de travailler à plusieurs reprises avec Véronique Maume-Deschamps, et de profiter de ses grandes connaissances en systèmes dynamiques. Sa relecture attentive des chapitres 2, 4 et surtout 5 en a grandement amélioré les contenus. Je suis très heureux qu'elle fasse partie de mon jury.

Enfin, Marie-Pierre Béal et Bruno Salvy me font l'honneur d'assister à ma soutenance. J'en suis très flatté.

Ma reconnaissance va bien évidemment à tous les occupants du bureau S2-302, sans exception. Jean-Marie, dont l'infinie sagesse profitera j'en suis sur à plus d'un doctorant, et Philippe, vieux compagnon sans qui commencer une thèse n'est pas raisonnablement envisageable, ont réussi à faire de ce bureau un lieu où la bonne humeur n'a d'égale que... la bonne humeur. Leur présence chaque jour fut garante d'un environnement de travail à la limite de la perfection.

Les (ex)locataires du bureau voisin ne sont pas en reste : je pense à (dans le désordre) Jérémie, Régis, Bertrand, Luc et à tous ceux qui un jour ou l'autre seront amenés à poser leur affaires entre ces quatre murs. Mais d'une manière générale, c'est l'ambiance parmi les doctorants (et les autres) de l'équipe qui fût appréciable, et Bruno, Guillaume, Ali, Marc, Fabien, Aline (un petit $\sum$ pour toutes ces années), Céline, ont largement contribué à l'atmosphère détendue qui règne au Greyc.

Il y a heureusement une vie extra-universitaire. Outre certaines personnes déjà mentionnées plus haut, je tiens à remercier pour les moments de détente Olive, Chinois, Gaelle, JF (et Simon), Bartok, Pouig, les Laurents, Fred(dy), J. and J. Marie, Bob, Elvis, Vince, la bande de Flip (K-ro, Raf, Béné, etc...), et que les oubliés m'excusent. Tout ce petit monde rend les choses bien plus supportables parfois...

Enfin, un grand merci à ma famille. À mes parents, dont le soutient constant au cours de ces années d'étude m'est allé droit au coeur, et à ma soeur Émilie.

Sigrun, livet mitt er rikere etter at jeg møtte deg. Takk, for dette og for alt annet.

Bonne lecture...

Table des matières

Introduction

L'objet de ce travail est l'étude la famille des algorithmes de calcul de pgcd, désignés dans le titre par algorithmes d'Euclide, et plus généralement dans cette thèse par algorithmes euclidiens. On peut se pencher sur ces algorithmes a priori simples pour de nombreuses raisons.

Il y a tout d'abord un évident intérêt historique. Décrit dans le Livre 7 des *Éléments* aux alentours de 300 avant J.C. (et probablement déjà connu deux siècles auparavant), l'algorithme d'Euclide est d'après Knuth [Knu98] le grand père de tous les algorithmes : *"We might call it the granddaddy of all algorithms, because it is the oldest non-trivial algorithm that has survived to the present day"*. C'est probablement le premier algorithme à avoir été précisément étudié, en 1845 par G. Lamé, qui, de manière prophétique, a au passage exhibé la première application de la suite de Fibonacci à l'informatique théorique. Pourtant, la compréhension de son comportement est longtemps restée partielle : ce n'est que très récemment que Doug Hensley, puis Brigitte Vallée et Viviane Baladi ont montré que le nombre d'itérations de l'algorithme suivait une loi gaussienne. Beaucoup de paramètres restent à étudier sur les algorithmes euclidiens.

Mais c'est peut-être dans l'omniprésence du calcul de pgcd en arithmétique que réside la principale motivation pour de telles analyses. On peut citer entre autres le calcul fractionnaire, la cryptographie à clé publique, le calcul d'inverses modulaires ou de bases de Gröbner (d'après Tudor Jebelean [Jeb95], le calcul de pgcd occupe 60 % du temps de calcul d'une base de Gröbner), etc...

L'importance du calcul de pgcd dans ces applications pose la question de l'efficacité de ses différentes techniques de calcul. Il est surprenant de noter qu'il existe très peu d'algorithmes significativement plus efficaces que l'algorithme d'Euclide (par significativement on entend que la complexité en bits dans le pire des cas n'est plus quadratique, mais sous-quadratique). Et encore, ces algorithmes ne le sont vraiment que lorsque la taille de l'entrée dépasse quelques milliers de bits, en particulier parce qu'ils tirent parti de la multiplication rapide, elle même efficace à partir d'un certain rang. Les autres algorithmes sont tous de complexité au moins quadratique, et la bonne approche pour les comparer, complémentaire des expérimentations pratiques, est l'analyse en moyenne.

L'analyse usuelle d'algorithmes, qui consiste à exhiber l'entrée qui maximise le temps de calcul de l'algorithme n'est ici que peu instructive. Il est plus pertinent d'étudier le comportement moyen de l'algorithme, c'est-à-dire d'effectuer une étude probabiliste sur l'ensemble de ses entrées. Cette thèse se situe donc dans le cadre général de l'analyse en moyenne d'algorithmes.

Cette branche de l'informatique, initiée par Knuth dans les années 60, est souvent une analyse pertinente :

- elle est réaliste, dans la mesure où elle décrit le comportement "réel" des algorithmes, tel qu'observé en pratique,
- elle apporte une compréhension fine de l'algorithme, puisque l'analyse requiert une étude très précise des mécanismes sous-jacents à l'algorithme.

Depuis sa création par Knuth, ce domaine s'est enrichi de nombreuses contributions, et les techniques existantes sont de natures très variées. On peut consulter les livres de Flajolet et Sedgewick [FS96, FS] pour une introduction à l'analyse d'algorithmes. Cependant, comme nous allons l'expliquer plus bas, les outils classiques ne sont pas suffisants pour l'analyse des algorithmes euclidiens.

Jusqu'au milieu des années 1990, l'analyse d'algorithmes euclidiens illustrait particulièrement bien la disparité de techniques d'analyse en moyenne. En effet, la première analyse en moyenne de l'algorithme d'Euclide a été faite indépendamment par Heilbronn [Hei69] et Dixon [Dix70] en 1971, avec des approches très différentes. L'étude de Dixon était de nature probabiliste, alors que celle de Heilbronn était basée sur les propriétés combinatoires de certaines décompositions des entiers. Les analyses qui suivirent utilisèrent également des techniques variées. On peut citer celle de l'algorithme Centré par Rieger [Rie78], celle de l'algorithme Soustractif par Knuth et Yao [YK75], celle (partielle) de l'algorithme Binaire par Brent [Bre76] et enfin l'analyse plus récente de Hensley [Hen94] de l'algorithme d'Euclide. Ces auteurs ont en particulier montré que le nombre moyen d'itérations de ces algorithmes est asymptotiquement linéaire en la taille de l'entrée, excepté pour l'algorithme Soustractif pour lequel cette quantité est quadratique. Les travaux d'Hensley ont de plus prouvé que le nombre d'itérations de l'algorithme d'Euclide suit une loi gaussienne.

Il a fallu attendre le milieu de années 1990 pour qu'une méthodologie unifiée d'analyse voie le jour. C'est en 1994 que Brigitte Vallée a pour la première fois utilisé l'approche *dynamique* pour l'analyse d'algorithmes [DFV97], qui s'est étendue à l'algorithme d'Euclide dans [Val97, FV98], puis finalement à l'algorithme Binaire dans [Val98a].

Cette approche lui a permis d'obtenir, dans [Val03] puis [AV00, Val00], une classification des algorithmes euclidiens d'une part en algorithmes "rapides", dont le nombre d'itérations et la complexité en bits sont linéaires et quadratiques en moyenne, et d'autre part en algorithmes "lents", pour lesquels ces quantités sont quadratiques et cubiques. Les algorithmes qui rentrent dans cette classification sont nombreux, on distingue en particulier l'algorithme d'Euclide, les algorithme Centré et Par-Excès, l'algorithme Binaire ou encore les algorithme pseudo-euclidiens.

Cependant, il reste de nombreux algorithmes dont on ignore le comportement : cette thèse en étudie trois, et chacun pose des questions différentes.

Les algorithmes α-euclidiens dépendent d'un paramètre réel $\alpha \in [0, 1]$, et généralisent nombre d'algorithmes connus puisque pour $\alpha = 1, 1/2$ ou 0, on retrouve les algorithmes d'Euclide, Centré et Par-Excès. Les deux premiers sont des algorithmes rapides, alors que le dernier est lent. La question est alors de savoir comment varie le comportement de l'algorithme en fonction du paramètre α. Pour une valeur de α générique, l'algorithme est-il rapide ? lent ?

L'algorithme LSB a été introduit récemment par Stehlé et Zimmermann [SZ04]. Bien qu'à première vue proche de l'algorithme Binaire, il appartient à une classe d'algorithme jamais étudiée jusqu'à présent, puisqu'il n'utilise que les bits de poids faible des entiers pour guider les divisions. On peut donc se demander s'il appartient à la classe des algorithmes rapides, et surtout si les méthodes développées jusqu'ici s'appliquent toujours.

Enfin, seuls les algorithmes à la structure simple ont été étudiés jusqu'à présent. Or, les algorithmes les plus utilisés en pratiques sont souvent plus complexes. Ici encore, se pose la question de l'efficacité des méthodes dynamiques pour analyser de tels algorithmes, comme l'algorithme de Lehmer-Euclide, étudié dans ce mémoire.

Nous avons répondu à ces questions, et en particulier montré que la méthodologie générale de l'analyse dynamique permet de traiter ces problèmes.

Avant d'expliquer en quoi ces analyses ont permis d'étendre le champ d'application de l'analyse dynamique, expliquons en quelques lignes en quoi consiste cette démarche.

L'analyse dynamique d'algorithmes repose sur le mariage assez inattendu de deux domaines a priori distincts : l'analyse d'algorithmes et la théorie des systèmes dynamiques.

L'approche naturelle, quand on veut étudier un algorithme euclidien est de faire l'analyse de la structure sous-jacente à l'algorithme, qui est le développement en fraction continue du rationnel formé avec l'entrée de l'algorithme. Cette approche est, par exemple, celle choisie par Knuth dans l'analyse heuristique de l'algorithme d'Euclide [Knu98]. La principale difficulté rencontrée alors est que le processus qui génère le développement en fraction continue est généralement un processus à mémoire non-bornée : le $n$ème chiffre du DFC dépend des $n-1$ chiffres précédents. Les techniques usuelles d'analyse d'algorithmes sont peu adéquates pour traiter cette situation. Par exemple, une modélisation en chaîne de Markov (donc à mémoire bornée) n'est pas possible. Pour les mêmes raisons, les manipulations usuelles de séries génératrices, à base de dictionnaires, ne sont plus possibles. Une manière élégante et efficace de résoudre ce problème est de voir le processus comme un système dynamique. On accède de cette manière à toute une gamme d'outils appropriés pour traiter les corrélations non bornées. L'utilisation conjointe de ces outils et de ceux plus classiques de l'analyse d'algorithmes permet en général de faire l'étude de l'algorithme.

Cette approche, déjà utilisée par Brent et Hensley, a été développée et formalisée par les travaux successifs de Brigitte Vallée. Dans le cadre des algorithmes euclidiens, il faut citer tout d'abord l'analyse de l'algorithme de Gauss [DFV97], d'Euclide [FV98, Val97], de l'algorithme Binaire [Val98a], puis celle d'une classe entière d'algorithmes [Val03], et enfin les analyses plus fines présentées dans [AV00, Val00], où est étudiée la complexité en bits, paramètre plus pertinent (et bien sûr plus difficile à analyser) que le nombre d'itérations. Enfin, il faut préciser que cette approche a des applications également en théorie de l'information : le système dynamique est alors la source qui émet des symboles, et l'étude de cette source [Val01] permet de traiter des problèmes liés à la recherche de motifs (travaux de Bourdon et Vallée [BNV01, BV02]) ou aux structures (trie, patricia trie, etc...) naturellement associées à ces sources (travaux de Clément, Flajolet, Vallée, Bourdon, [CFV01, CFV98, Bou01]).

Pour préciser les contributions de cette thèse à ce domaine, il faut tout d'abord décrire comment procède une analyse dynamique d'algorithme. On procède usuellement en trois étapes.

Étape 1. La première étape est un étape de modélisation. Le but est d'étendre l'algorithme (discret) étudié en un système dynamique (continu), possédant si possible de bonnes propriétés. On fait cette extension en général en deux temps. Tout d'abord on observe l'action de l'algorithme sur les rationnels : si (u, v) est l'entrée de l'algorithme, et (v, r) la paire obtenue après une itération, alors on considère l'application qui au rationnel v/u associe le rationnel r/v. On étend ensuite cette application à un ensemble continu, et on obtient généralement un système dynamique. Notons que cette dernière extension n'est pas toujours triviale, puisque la dynamique sous-jacente à l'algorithme peut dépendre de notions non-définies dans un monde continu.

Étape 2. Une fois modélisé l'algorithme, on cherche à déterminer le comportement du système dynamique, son évolution au cours du temps. Cette étude, de nature probabiliste, nous éloigne à première vue du problème initial (les exécutions de l'algorithme correspondent aux trajectoires des points rationnels, qui forment un ensemble négligeable du point de vue du comportement global du système). Son intérêt réside essentiellement dans les outils employés ici, à savoir l'opérateur de Perron-Frobenius, et un premier opérateur de transfert, qui joueront un grand rôle

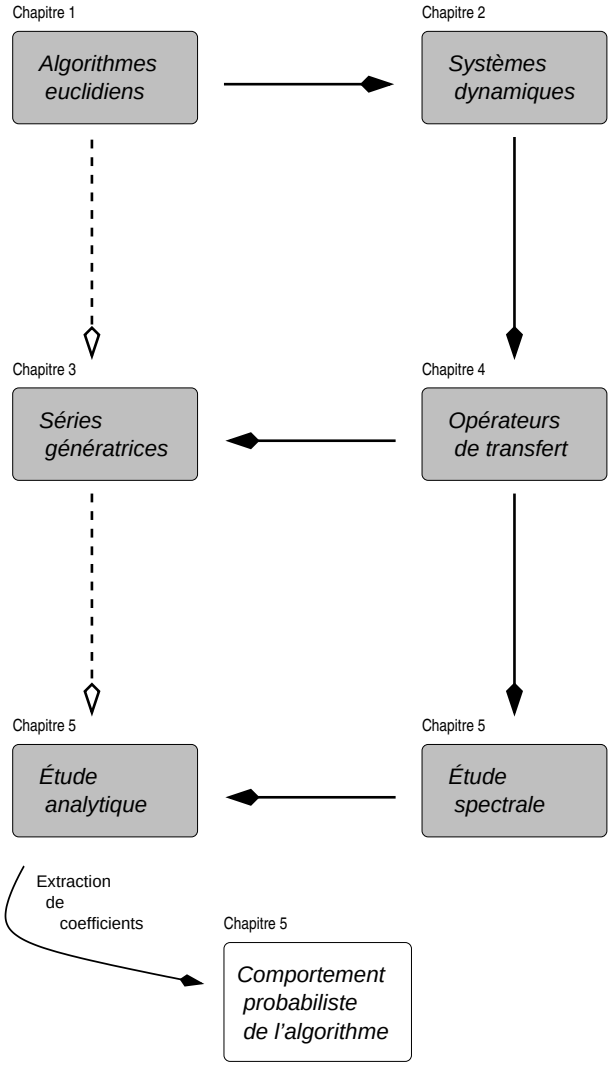

FIG. 1 – Les principales étapes de l'analyse dynamique

dans la dernière étape.

Étape 3. La dernière étape est celle du transfert "du continu au discret". Il s'agit de transposer les résultats de l'étude faite précédemment aux algorithmes. Les opérateurs utilisés à l'étape 2 sont maintenant transformés en un autre opérateur de transfert, qui hérite des propriétés décrites à l'étape précédente. Le retour à l'algorithme se fait maintenant en reliant cet opérateur de transfert aux séries génératrices qui permettent de faire l'analyse de l'algorithme. Les propriétés fonctionnelles de l'opérateur se transforment en propriétés analytiques pour les séries, ce qui permet de conclure l'analyse.

Les analyses présentées dans cette thèse nous ont permis d'élargir le champ d'application de cette méthode, à plusieurs niveaux.

Lors de l'analyse des algorithmes α-euclidiens, menée en collaboration avec Jérémie Bourdon et Brigitte Vallée, les techniques usuelles ne s'appliquaient plus pour les étapes 2 et 3, parce que les

systèmes dynamiques correspondant à ces algorithmes ne satisfont pas les conditions rencontrées précédemment dans les autres analyses. Il a donc fallu modifier en profondeur ces techniques. Nous avons ainsi mis au point des méthodes plus générales et plus maniables, dorénavant utilisées dans notre groupe. Ce travail a donné lieu à une publication dans la revue *Journal of Algorithms* [BDV02].

L'analyse de l'analyse LSB se distingue des précédentes dès la première étape. En effet, l'extension de l'algorithme ne conduit pas à un système dynamique réel, mais à un système dynamique défini sur l'ensemble $\mathbb{Q}_2$ des nombres 2-adiques. Cette situation nouvelle a nécessité un traitement particulier. En effet, les arguments employés dans les étapes 2 et 3 font appel à la théorie des produits de matrices aléatoires, ce qui est nouveau dans le domaine et ouvre la porte à de futures analyses. Ce travail, effectué conjointement avec Véronique Maume-Deschamps et Brigitte Vallée, paraîtra dans la revue *Discrete Mathematics and Theoretical Computer Science* [DMDV05] et a fait l'objet d'un premier rapport technique [Dai04].

Enfin, le travail sur l'algorithme de Lehmer-Euclide a conduit à changer le point de vue des analyses, puisque les paramètres importants ne concernent plus une exécution complète de l'algorithme, mais l'état des principales grandeurs à un moment donné de cette exécution. Nous avons été amenés à analyser ce qu'on appelle l'algorithme d'Euclide interrompu. Ce travail, réalisé en collaboration avec Brigitte Vallée, a été publié dans la revue *Combinatorics, Probabilities and Computing* [DV04].

Organisation du manuscrit

La thèse est organisée de manière à suivre une à une les principales étapes de l'analyse dynamique. À chaque étape, nous indiquons comment traiter les différents cas abordés.

Nous commençons dans le premier chapitre par décrire les algorithmes étudiés ici. Nous décrivons au passage l'ensemble des algorithmes euclidiens, même si certains de ces algorithmes ne seront pas étudiés par la suite. Nous précisons le cadre probabiliste dans lequel nous menons les différentes analyses, et présentons de manière informelle l'ensemble des résultats connus sur ces algorithmes.

Le second chapitre est consacré à la première étape de l'analyse, c'est-à-dire à la modélisation en système dynamique. On présente ici plusieurs modélisations de différentes natures, puisque nous serons amenés à manipuler des systèmes dynamiques de l'intervalle, des systèmes dynamiques probabilistes ou encore des systèmes dynamiques 2-adiques.

Les chapitres 3 et 4 sont consacrés aux principaux outils utilisés en analyse dynamique. Nous décrivons dans le chapitre 3 les séries génératrices, et dans le chapitre 4 les opérateurs de Perron-Frobenius et les opérateurs de transfert. Nous terminons ce chapitre en reliant ces derniers opérateurs aux séries génératrices, ce qui constitue une des pierres de base de la dernière étape de l'analyse.

L'essentiel de l'analyse est fait au chapitre 5. Nous y décrivons quelles sont les méthodes adaptées à chaque type de système, et donc à chaque type d'opérateur. C'est dans ce chapitre que sont énoncés précisément les différents résultats obtenus.

Le chapitre 6 est consacré à l'étude de l'algorithme de Lehmer-Euclide. Il occupe une place à part puisqu'il ne s'agit pas exactement d'une analyse dynamique telle que nous l'avons décrite au paragraphe précédent.

Chapitre 1

Algorithmes Euclidiens

Sommaire

De nombreuses techniques de calcul pgcd ont été développées, depuis l'algorithme d'Euclide dans l'antiquité grecque en passant par les améliorations de Lehmer ou Schönhage, jusqu'au récent algorithme de Stehlé et Zimmermann, par exemple. Nous ne faisons pas ici une présentation exhaustive de tous ces algorithmes, mais décrivons les plus importants d'entre eux et insistons plus particulièrement sur ceux étudiés dans ce mémoire.

Nous présentons deux classifications successives de ces algorithmes. La première distinction se fait selon la structure même de l'algorithme. En effet, les algorithmes les plus basiques fonctionnent comme l'algorithme d'Euclide par divisions successives, et c'est la nature de la division qui différencie les algorithmes. Les algorithmes plus perfectionnés sont d'une structure plus complexe. Ils utilisent certaines propriétés de stabilité des divisions pour accélérer les calculs. L'idée de base est que le quotient produit par une division ne dépend que d'une partie des bits des entiers, et on peut donc simuler une exécution d'un algorithme basique par une suite d'opérations sur des entiers plus petits. A ce niveau, il existe deux approches : celle de l'algorithme

Lehmer-Euclide, itérative, et celle de l'algorithme de Schönhage, basée sur une stratégie Diviser pour Régner. On peut généralement associer à chaque algorithme "basique" sa (ses) version(s) accélérée(s). Les algorithmes présentés dans ce chapitre appartiennent à la première catégorie, ce sont des algorithmes simples, les algorithmes rapides seront présentés dans le chapitre 6.

Comme nous venons de le dire, on distingue les algorithmes basiques selon la nature de la division qu'ils utilisent. Nous considérons essentiellement trois familles de divisions, chacune d'elle étant caractérisée par la partie des entiers qu'elle manipule.

Dans la première, la famille MSB (Most Significant Bits), on trouve l'algorithme d'Euclide : le quotient apparaissant dans la division euclidienne standard dépend principalement des bits dominants des deux entiers, et les divisions successives tendent à faire "disparaître" les bits de poids forts. Les algorithmes de cette classe génèrent une suite de restes décroissante. Outre l'algorithme d'Euclide, et ses variantes les plus connues (algorithmes Centré, Par-Excès, Pair ou Impair), nous introduisons dans cette famille la classe des algorithmes α-euclidien. Ces algorithmes, initialement introduits par Nakada [Nak81] sous la forme d'un développement en fraction continue, dépendent d'un paramètre réel α compris entre 0 et 1, qui détermine la position du reste r engendré par la division par rapport à l'entrée (u, v). Leur étude est instructive, en particulier parce qu'elle généralise certaines analyses antérieures : pour les valeurs $\alpha = 0, 1/2, 1$, l'algorithme correspondant n'est rien d'autre que l'algorithme Par-Excès, Centré ou d'Euclide standard.

La seconde famille est celle des algorithmes LSB (Least Significant Bits). Le principal algorithme de la classe, que nous appelons l'algorithme LSB, a été introduit récemment par Stehlé et Zimmermann [SZ04]. Le but initial était d'utiliser la division LSB pour mettre au point un algorithme de type Diviser pour Régner, qui s'avère être à ce jour le plus simple à décrire, à prouver et à implanter des algorithmes rapides. On peut voir la division LSB comme symétrique de la division MSB : elle est guidée par les bits de poids faibles, et tend faire "disparaître" les bits de poids faible des entiers. Une exécution de l'algorithme génère une suite de restes dont le nombre de zéros à la droite de leur écriture binaire est croissant.

Enfin, la dernière famille contient les algorithmes Mixtes, qui utilisent à la fois les bits de poids fort et les bits de poids faible des entiers. On y retrouve les algorithmes pseudo-euclidiens, ainsi que les dérivés de l'algorithme Binaire. Les algorithmes pseudo-euclidiens sont décrits dans [Knu98, Sha90, Val03] : étant donné un algorithme MSB, on élimine par décalage les puissances de 2 apparaissant au cours de l'exécution de l'algorithme. Les bits de poids forts interviennent donc dans le choix du quotient, et les bits de poids faible dans les décalages successifs. L'algorithme Binaire, introduit par Stein [Ste67] en 1961, procède de manière différente (même si on peut le voir comme la pseudo version de l'algorithme Soustractif) : on fait d'abord apparaître des puissances de 2 par soustractions, qu'on fait ensuite disparaître par décalage. Ce processus se répète et le test d'arrêt dépend d'une comparaison entre entiers : c'est ici qu'interviennent les bits de poids forts. Il existe de nombreuses variantes de cet algorithme : la première est l'algorithme Plus-Moins de Brent et Kung [BK85], et on a ensuite différentes généralisations de ce procédé, dues entre autre à Sorenson [Sor94], Weber [Web95] ou Jebelean [Jeb93b].

Ce chapitre est dans un premier temps consacré à la description des algorithmes de chaque classe. Après avoir décrit, par soucis de clarté et pour introduire les notations, un algorithme euclidien "générique", nous effectuons un léger détour par la cas polynomial, pour expliquer en quoi la distinction en trois classes n'y est pas pertinente du point de vue de l'analyse en moyenne. Puis nous décrivons dans les paragraphes 1.3, 1.4 et 1.5 les algorithmes MSB, LSB et Mixtes. Certains des algorithmes présentés ne seront pas étudiés par la suite. Enfin, le paragraphe 1.6 est consacré à l'analyse en moyenne des algorithmes : nous détaillons le cadre probabiliste dans

lequel nous nous plaçons ainsi que les principaux paramètres que nous étudions, puis présentons les différents résultats obtenus dans le domaine : ceux antérieurs à mes travaux, et ceux décrits dans la thèse.

1.1 Formalisme générique des algorithmes euclidiens

Soient deux entiers u et v. Une division euclidienne générique de u par v s'écrit sous la forme

$$u = vq + \varepsilon 2^k r, \tag{1.1}$$

où le triplet $d(u,v) = (q, \varepsilon, k)$ avec $\varepsilon = \pm 1$ et $k \in \mathbb{N}$ est le quotient, ou chiffre, de la division et r le reste. Si on note $\mathcal{M}_{[d]}$ la matrice

$$\mathcal{M}_{[d]} := \begin{pmatrix} 0 & 1 \\ \varepsilon 2^k & q \end{pmatrix}$$

alors (1.1) s'écrit également

$$\begin{pmatrix} v \\ u \end{pmatrix} = \mathcal{M}_{[d]} \cdot \begin{pmatrix} r \\ v \end{pmatrix}.$$

Un algorithme euclidien est une succession de divisions et d'échanges, qui se termine avec un reste nul (ou égal à un selon les cas, mais il est toujours possible de se ramener à un reste nul). Si pour une entrée (u,v) l'algorithme considéré effectue p divisions, on obtient une suite de la forme

$$u_0 = u, \quad u_1 = v, \quad u_0 = u_1 q_1 + \varepsilon_1 2^{k_1} u_2, \quad u_1 = u_2 q_2 + \varepsilon_2 2^{k_2} u_3, \ldots u_{p-1} = u_p q_p + 0 \tag{1.2}$$

et le dernier reste non nul est alors le pgcd de u et v (éventuellement à une puissance de 2 près). Nous notons (u_i) et (d_i) les suites de restes et de quotients engendrées par l'algorithme. Si on note $\mathcal{U}_i$ et $\mathcal{M}_{[d_i]}$ les vecteurs et matrices suivantes

$$\mathcal{U}_i := \begin{pmatrix} u_{i+1} \\ u_i \end{pmatrix}, \qquad \mathcal{M}_i := \mathcal{M}_{[d_1]} \mathcal{M}_{[d_2]} \cdots \mathcal{M}_{[d_i]},$$

alors on obtient

$$\mathcal{U}_i = \mathcal{M}_{[d_{i+1}]} \mathcal{U}_{i+1}, \qquad \mathcal{U}_0 = \mathcal{M}_i \mathcal{U}_i,$$

et en particulier

$$\mathcal{U}_0 = \mathcal{M}_i \cdot \begin{pmatrix} 0 \\ u_p \end{pmatrix}.$$

Si la matrice $\mathcal{M}_i$ est donnée par

$$\mathcal{M}_i = \begin{pmatrix} b_{i+1} & a_{i+1} \\ b_i & a_i \end{pmatrix}$$

alors a_i et b_i sont les i-èmes continuants, notamment utiles pour approximer le rationnel v/u. On définit de manière symétrique les convergents s_i et t_i en posant $\overline{\mathcal{M}_i} = \mathcal{M}_{[d_{i+1}]} \cdots \mathcal{M}_{[d_p]}$. Les convergents sont alors les coefficients de cette matrice.

Chacun des algorithmes décrits dans les trois sections suivantes sera présenté sous ce formalisme. Nous récapitulons dans la figure (1.8) les caractéristiques des restes et quotients relatifs à chaque type de division.

1.2 Les algorithmes euclidiens sur les polynômes

Avant d'introduire les algorithmes euclidiens sur les entiers, nous allons faire un rapide survol de la situation polynomiale. Étant donnés deux polynômes P_0 et P_1, avec $\deg(P_0) > \deg(P_1)$, la division euclidienne de P_0 par P_1 est définie par la relation

$$P_0 = P_1 Q_1 + P_2, \quad \deg P_2 < \deg P_1 \quad \text{ou} \quad P_2 = 0.$$

Cette division tend donc à faire "disparaître" les termes de plus haut degré des deux polynômes, et se calcule par soustractions et comparaisons entre les degrés des deux polynômes successives. Considérons par exemple les deux polynômes suivants, définis sur $\mathbb{F}_2[X]$,

$$P_0(X) = X^5 + X^4 + X^2 + 1, \qquad P_1(X) = X^3 + X^2 + X. \tag{1.3}$$

La division de $P_0(X)$ par $P_1(X)$ est donnée par

$$(X^5 + X^4 + X^2 + 1) = (X^3 + X^2 + X) \cdot (X^2 + 1) + (X + 1).$$

et a été obtenue en faisant les soustractions suivantes :

$$\begin{aligned} (X^5 + X^4 + X^2 + 1) - X^2 \cdot (X^3 + X^2 + X) &= X^3 + X^2 + 1, \\ (X^3 + X^2 + 1) - (X^3 + X^2 + X) &= X + 1. \end{aligned}$$

Il apparaît clairement ici que le calcul de la division euclidienne est guidé par les termes de plus haut degrés des deux polynômes, qui disparaissent au fur et à mesure que l'on itère les divisions au cours de l'algorithme d'Euclide. On obtient donc une suite de restes de degrés décroissants. Sur l'exemple choisi, cette suite est donnée par :

$$(X^5 + X^4 + X^2 + 1), \quad (X^3 + X^2 + X), \quad (X + 1), \quad 1, \quad 0,$$

et donc les deux polynômes sont premiers entre eux.

Il existe une approche symétrique, qui consiste à faire disparaître les termes de bas degrés. Si on note $\nu(P)$ l'indice du plus petit coefficient non-nul du polynôme P (ou de manière équivalente le plus grand exposant k tel que X^k divise P), alors étant donnés $\nu(P_0) < \nu(P_1)$ on peut définir la division

$$P_0 = P_1 Q_1 + P_2, \quad \nu(P_1) < \nu(P_2) \quad \text{ou} \quad P_2 = 0.$$

Celle-ci se calcule de manière symétrique à la division usuelle, la division des polynômes définis en (1.3) est

$$(X^5 + X^4 + X^2 + 1) = (X^3 + X^2 + X) \cdot \left(\frac{1}{X} + 1\right) + (X^5 + X^4 + X^3 + X^2).$$

L'algorithme obtenu effectue alors les divisions

$$\begin{aligned} (X^5 + X^4 + X^2 + 1) &= (X^3 + X^2 + X) \cdot \left(\frac{1}{X} + 1\right) + (X^5 + X^4 + X^3 + X^2), \quad (1.4) \\ (X^3 + X^2 + X) &= (X^5 + X^4 + X^3 + X^2) \cdot \frac{1}{X} + X^4, \\ (X^5 + X^4 + X^3 + X^2) &= X^4 \cdot \left(\frac{1}{X^2} + \frac{1}{X} + 1\right) + X^5, \\ X^4 &= X^5 \cdot \frac{1}{X} + 0, \end{aligned}$$

dont on déduit pgcd $(P_0, P_1) = \frac{1}{X^5} X^5 = 1$.

On peut donc définir naturellement sur les polynômes deux divisions à priori différentes, une guidée par les termes de plus haut degré, l'autre par les termes de plus bas degré. En réalité, ces deux divisions conduisent à des algorithmes aux comportements probabilistes identiques. En effet, si on note $\bar{P}$ le polynôme miroir de P défini par $\bar{P}(X) = a_n + a_{n-1}X + \cdots + a_0 X^n$ si $P(X) = a_0 + a_1 X + \cdots + a_n X^n$, alors les miroirs des restes produits par l'algorithme d'Euclide usuel sur entrée (P_0, P_1) et les restes produits par le second algorithme sur $(\bar{P}_0, \bar{P}_1)$ sont les mêmes, et inversement. Ainsi l'algorithme usuel appliqué aux polynômes

$$\bar{P}_0(X) = X^5 + X^3 + X + 1, \qquad \bar{P}_1(X) = X^4 + X^3 + X^2$$

effectue les divisions

$$\begin{aligned}
(X^5 + X^3 + X + 1) &= (X^4 + X^3 + X^2) \cdot (X + 1) + (X^3 + X^2 + X + 1), \\
(X^4 + X^3 + X^2) &= (X^3 + X^2 + X + 1) \cdot X + X, \\
(X^3 + X^2 + X + 1) &= X \cdot (X^2 + X + 1) + 1 \\
X &= 1 \cdot X + 0.
\end{aligned}$$

On retrouve ainsi la suite des miroirs des restes de (1.4). Cette propriété repose en particulier sur le coté non-archimédien des normes définies sur les polynômes, et donc sur l'absence de propagation de retenue. La situation est bien sûr différente sur les entiers.

1.3 Les algorithmes MSB

Cette section est dédiée à la description des algorithmes utilisant les bits de poids forts lors du calcul de la division. Nous présentons tout d'abord l'algorithme d'Euclide, puis quelques une de ses variantes, et finalement les algorithmes α-euclidiens qui généralisent plusieurs des algorithmes de cette classe.

1.3.1 L'algorithme d'Euclide $\mathcal{E}$

L'algorithme d'Euclide $\mathcal{E}$ est basé sur la division euclidienne standard. Soient u et v deux entiers positifs et $u > v$. Alors la division euclidienne de u par v est définie par

$$u = vq + r, \qquad 0 \leq r < v. \tag{1.5}$$

Le quotient d est réduit à un seul élément q, qui est bien sûr donné par

$$q = \left\lfloor \frac{u}{v} \right\rfloor$$

où $\lfloor . \rfloor$ désigne la partie entière. Ainsi, la division de $29 = 11101_2$ par $v = 12 = 1100_2$ renvoie le reste $5 = 101_2$:

$$11101_2 = 1100_2 \times 10_2 + 101_2.$$

Il existe de nombreuses méthodes pour calculer cette division, allant de la plus élémentaire, la division "à la main", aux plus sophistiquées utilisant la multiplication rapide de Karatsuba ou la FFT. Mais quelle que soit la méthode utilisée, la division est essentiellement guidée par les bits de poids forts des entiers, que l'on fait par la suite disparaître au fur et à mesure que l'on

Entrée : $(u, v) \in \mathbb{N}^2$, avec $u \geq v$.
Sortie : pgcd (u, v).

$(u_0, u_1) := (u, v)$;
$i := 0$;
`Tant que` $u_{i+1} \neq 0$
$q_{i+1} = d(u_i, u_{i+1})$;
$u_{i+2} := u_i - u_{i+1} q_{i+1}$;
$i := i + 1$;
`Renvoyer` u_i.

FIG. 1.1 – L'algorithme d'Euclide $\mathcal{E}$.

itère les divisions lors de l'exécution de l'algorithme d'Euclide.

Par exemple, si on applique l'algorithme d'Euclide aux entiers $115 = 1110011_2$ et $82 = 1010010_2$ on obtient la suite de restes

$$\begin{aligned} 115 &= 1110011_2 \\ 82 &= 1010010_2 \\ 33 &= 100001_2 \\ 16 &= 10000_2 \\ 1 &= 1_2 \\ 0 &= 0_2 \end{aligned}$$

L'algorithme d'Euclide est souvent utilisé dans sa version étendue, qui non seulement calcule le pgcd des deux entiers mais également les coefficients de Bezout e et f définis par

$$eu + fv = \text{pgcd } (u, v).$$

Ces coefficients sont obtenus par le calcul de deux suites supplémentaires, (e_i) et (f_i), définies par

$$(e_0, f_0) = (1, 0), \quad (e_1, f_1) = (0, 1), \ldots, (e_{i+1}, f_{i+1}) = (e_{i-1}, f_{i-1}) - q_i(e_i, f_i), \ldots$$

ce qui garantit que pour tout i :

$$e_i u + f_i v = u_i.$$

Cette relation est notamment vraie pour $i = p$, et il suffit donc de poser $e = e_p, f = f_p$ pour obtenir les coefficients de Bezout. L'algorithme d'Euclide étendu, $\mathcal{EX}$, est présenté dans la figure (1.2). Remarquons finalement que les suites (e_i) et (f_i) sont étroitement liées aux continuants a_i et b_i. En effet, on vérifie pour tout i :

$$|e_i| = a_i \quad \text{et} \quad |f_i| = b_i.$$

1.3.2 Les algorithmes d'Euclide interrompus

Nous introduisons dès maintenant ce que nous appelons l'algorithme d'Euclide interrompu de paramètre t, noté $\mathcal{E}_t$. Cet algorithme n'est rien d'autre que l'algorithme d'Euclide standard, dans

Entrée : $(u, v) \in \mathbb{N}^2$, avec $u \geq v$.
Sortie : pgcd (u, v) et coefficients de Bezout (e, f).

$(u_0, u_1) := (u, v)$;
$i := 0$;
$(e_0, f_0) := (1, 0)$;
$(e_1, f_1) := (0, 1)$;
`Tant que` $u_{i+1} \neq 0$
    $q_{i+1} := d(u_i, u_{i+1})$;
    $u_{i+2} := u_i - u_{i+1} q_{i+1}$;
    $(e_{i+2}, f_{i+2}) := (e_i, f_i) - q_{i+1}(e_{i+1}, f_{i+1})$;
    $i := i + 1$;
`Renvoyer` u_i et $(e, f) := (e_i, f_i)$.

FIG. 1.2 – L'algorithme d'Euclide étendu $\mathcal{EX}$.

lequel le test d'arrêt $u_{i+1} = 0$ est remplacé par $u_i \leq u^t$ (voir figure (1.3)), t étant un paramètre réel compris entre 0 et 1. Lorsque $t = 0$, on retrouve l'algorithme d'Euclide standard, et lorsque $t = 1$, l'algorithme ne fait rien. Cet algorithme est intrinsèquement lié à la décroissance de la suite des restes, ce qui en fait un objet d'étude privilégié. Il intervient notamment de manière centrale dans l'analyse de l'algorithme de Lehmer-Euclide du chapitre 6.

Entrée : (u, v), avec $u \geq v$, $t \in [0, 1]$.
Sortie : u_i avec $u_i > u^t \geq u_{i+1}$

$(u_0, u_1) := (u, v)$;
$i := 0$;
`Tant que` $u_i > u^t$
    $q_{i+1} := d(u_i, u_{i+1})$;
    $u_{i+2} := u_i - u_{i+1} q_{i+1}$;
    $i := i + 1$;
`Renvoyer` u_i.

FIG. 1.3 – L'algorithme d'Euclide interrompu de paramètre t, $\mathcal{E}_t$.

Enfin nous noterons $\underline{\mathcal{E}}_\delta$ un autre algorithme interrompu, qui s'arrête à la $\lfloor \delta p \rfloor$-ème itération si le nombre total d'itérations est p, δ étant un paramètre réel compris entre 0 et 1. C'est bien sûr un algorithme totalement artificiel, mais son étude permet de mieux comprendre l'évolution des principales quantités engendrées par l'algorithme.

1.3.3 Quelques variantes de l'algorithme d'Euclide

A partir de l'algorithme d'Euclide, il est possible de définir d'autres algorithmes, en imposant des restrictions sur le quotient, comme pour les algorithmes Pair et Impair, ou sur les restes comme pour les algorithmes Centré et Par-Excès.

On peut ainsi imposer au quotient q d'être pair (respectivement impair), et obtenir la division Paire (resp. Impaire) définie par

$$u = qv + \varepsilon r, \quad q \text{ pair (resp. impair)}, \quad \varepsilon = \pm 1, \quad 0 \leq r < v.$$

On calcule le chiffre $d = (q, \varepsilon)$ à partir de celui renvoyé par la division euclidienne standard en posant

$$d := \begin{cases} (q,\ +1) & \text{si } q \text{ est pair (resp. impair)}, \\ (q+1,\ -1) & \text{sinon .} \end{cases}$$

Ainsi le quotient de la division Impaire de 29 par 12 est 3 :

$$29 = 12 \times 3 - 7.$$

En itérant les divisions on obtient les algorithmes $\mathcal{I}$ et $\mathcal{P}$.

De la même manière, les restrictions peuvent porter sur la position du reste r par rapport au diviseur v. C'est le cas des divisions définissant les algorithmes Centré $\mathcal{C}$ et Par-Excès $\mathcal{X}$.
La division Centrée renvoie un reste compris entre 0 et $v/2$:

$$u = vq + \varepsilon r, \quad \varepsilon = \pm 1, \quad 0 \leq r < \frac{v}{2},$$

le chiffre $d = (q, \varepsilon)$ étant calculé à partir de la division standard par

$$d := \begin{cases} (q,\ +1) & \text{si } 0 \leq r < \frac{v}{2}, \\ (q+1,\ -1) & \text{si } \frac{v}{2} \leq r < v. \end{cases}$$

Notons que dans ce cas q n'est rien d'autre que

$$q = \left\lfloor \frac{u}{v} \right\rceil,$$

où $\lfloor x \rceil$ désigne l'entier le plus proche de x. Cette division conduit à l'algorithme Centré $\mathcal{C}$.

L'algorithme Par-Excès $\mathcal{X}$ est basé sur la division

$$u = vq - r, \quad 0 \leq r < v.$$

Le quotient q est maintenant donné par

$$q = \left\lceil \frac{u}{v} \right\rceil,$$

où $\lceil x \rceil$ désigne le plus petit entier supérieur ou égal à x.

1.3.4 Les algorithmes α-Euclidiens $\mathcal{E}_\alpha$

Les algorithmes α-euclidiens ont d'abord été introduits par Nakada [Nak81], ainsi que par Moussa, Cassa et Marmi [MCM99], sous forme de développement en fraction continue. La modélisation sous forme d'algorithme euclidien a été faite dans [BDV02]. Il existe deux types d'algorithmes α-euclidiens : les repliés, notés $\widehat{\mathcal{E}}_\alpha$ et les non-repliés notés $\bar{\mathcal{E}}_\alpha$ (nous notons $\mathcal{E}_\alpha$ lorsqu'il n'est pas nécessaire des différencier les deux types d'algorithmes). Nous commençons par présenter les algorithmes non-repliés, qui peuvent être vu comme une généralisation de variantes des

algorithmes Par-Excès et Centré. Puis nous expliquons comment on replie un algorithme.

Étant donnés deux entiers (éventuellement négatifs) u et v tels que $v/u \in [-1/2, 1/2]$, on peut définir de manière alternative une division centrée par

$$u = \varepsilon(vq + r), \quad -\frac{1}{2} \leq \frac{r}{v} < \frac{1}{2},$$

où ε est le signe de v/u. De même, étant donnés deux entiers u et v tels que $v/u \in [-1, 0]$, on peut définir une division Par-Excès en posant

$$u = vq + r, \quad -1 \leq \frac{r}{v} < 0.$$

Ces divisions sont définies par la position du rationnel r/v qui doit se trouver dans la fenêtre $[-1/2, 1/2[$ ou $[-1, 0[$ (dans le cas de l'algorithme d'Euclide la fenêtre considérée est $[0, 1[$). Il est maintenant naturel de généraliser ces divisions en imposant au reste de se situer dans une fenêtre $[(\alpha - 1), \alpha[$, où α est un paramètre réel compris entre 0 et 1. Quand α vaut 1, 1/2 ou 0 on retrouve donc les divisions Standard, Centrée et Par-Excès telle que présentées ci-dessus.

Plus rigoureusement la division α-euclidienne non-repliée est de la forme

$$u = \bar{\varepsilon}(v\bar{q} + \bar{r}), \quad \alpha - 1 \leq \frac{\bar{r}}{v} < \alpha, \quad \bar{\varepsilon} := \text{ signe de } \frac{v}{u}.$$

Le quotient $\bar{d} = (\bar{q}, \bar{\varepsilon})$ se calcule à partir de la paire (q, r) relative à la division euclidienne de $|u|$ par $|v|$ par

$$\bar{d} := \begin{cases} (q, \text{ signe de } \frac{v}{u}) & \text{si } 0 \leq r < \alpha v, \\ (q+1, \text{ signe de } \frac{v}{u}) & \text{si } \alpha v \leq u < v. \end{cases} \tag{1.6}$$

Il est maintenant possible de "replier" l'intervalle $[(\alpha - 1), \alpha[$ en un intervalle $[0, \alpha^+[$, où $\alpha^+ = \max(\alpha, 1 - \alpha)$, ce qui conduit à la division repliée :

$$u = v\widehat{q} + \widehat{\varepsilon}\widehat{r}, \quad 0 \leq \frac{\widehat{r}}{v} < \alpha^+.$$

Dans ce cas, les entiers considérés sont toujours positifs, et le quotient $\widehat{d} = (\widehat{q}, \widehat{\varepsilon})$ est calculé à partir de la paire (q, r) par :

$$\widehat{d} := \begin{cases} (q, +1) & \text{si } 0 \leq r < \alpha v, \\ (q+1, -1) & \text{si } \alpha v \leq r < v. \end{cases}$$

Les algorithmes obtenus $\widehat{\mathcal{E}}_\alpha$ généralisent les versions classiques des algorithmes Centré et Par-Excès.

Les algorithmes repliés et non-repliés sont très similaires. Considérons les exécutions de ces algorithmes sur une entrée (u, v) vérifiant $0 \leq v \leq \alpha^+ u$:

$$\bar{u}_0 = u, \quad \bar{u}_1 = v, \quad u_0 = \bar{\varepsilon}_1(\bar{u}_1\bar{q}_1 + \bar{u}_2), \quad \bar{u}_1 = \bar{\varepsilon}_2(\bar{u}_2\bar{q}_2 + \bar{u}_3), \quad \ldots \quad , \bar{u}_{p-1} = \bar{\varepsilon}_p(\bar{u}_p\bar{q}_p + 0),$$

$$\widehat{u}_0 = u, \quad \widehat{u}_1 = v, \quad \widehat{u}_0 = \widehat{u}_1\widehat{q}_1 + \widehat{\varepsilon}_1\widehat{u}_2, \quad \widehat{u}_1 = \widehat{u}_2\widehat{q}_2 + \widehat{\varepsilon}_2\widehat{u}_3, \quad \ldots \quad , \widehat{u}_{p-1} = \widehat{u}_p\widehat{q}_p + 0.$$

Alors les suites de chiffres et de restes sont liées par

$$\widehat{\varepsilon}_0 = 1, \quad \bar{\varepsilon}_1 = \text{ signe de } u, \quad \bar{q}_i = \widehat{q}_i, \quad \text{ et } \widehat{\varepsilon}_i = \prod_{j=2}^{i+1} \bar{\varepsilon}_j \quad \text{ pour } i \geq 1.$$

Remarquons de plus que dans les deux cas, q est donné par

$$q = \left\lfloor \frac{u}{v} + 1 - \alpha \right\rfloor,$$

et que le chiffre $d = (q, \varepsilon)$ prend ses valeurs dans l'ensemble suivant :

$$\{(q, -1) \quad q \geq r^-(\alpha)\} \cup \{(q, +1), \quad q \geq r^+(\alpha)\}$$

avec

$$r^+(\alpha) = \left\lfloor 1 + \frac{1-\alpha^2}{\alpha} \right\rfloor, \qquad r^-(\alpha) = \left\lfloor 2 + \frac{\alpha^2}{1-\alpha} \right\rfloor.$$

1.4 Les algorithmes LSB

Nous présentons maintenant l'algorithme LSB, noté $\mathcal{L}$, introduit par Stehlé et Zimmermann dans [SZ04]. De manière similaire à ce qui a été fait sur les polynômes dans la section 1.2, il est possible de définir sur les entiers une division symétrique à la division euclidienne standard. La démarche sera dans la premier temps la même : on définit la valuation 2-adique $\nu(u)$ d'un entier u par

$$\nu(u) := \max \left\{ k, 2^k | u \right\}.$$

La norme 2-adique $|u|_2$ est ensuite donnée par

$$|u|_2 := 2^{-\nu(u)},$$

et dépend donc des bits de poids faible des entiers. Nous cherchons donc à construire une division qui étant donnés deux entiers u et v vérifiant $|u|_2 > |v|_2$ génère un quotient q et un reste r de sorte que

$$u = vq + r, \quad q > 0 \quad \text{ et } \quad |r|_2 < |v|_2. \tag{1.7}$$

Le calcul de cette division est symétrique au calcul de la division euclidienne standard. Cette dernière procède par décalages à gauche et soustractions successives, alors que la division LSB est composée de décalages à droite et soustractions. L'étape de base de cette division est donc l'opération

$$u := u - v \cdot 2^{\nu(u) - \nu(v)},$$

qu'il suffit de réitérer jusqu'à obtenir $|u|_2 < |v|_2$. L'algorithme obtenu produit donc le reste et le quotient satisfaisant (1.7). Notons que dans ce cas, le quotient q est un rationnel de la forme $q = a/2^k$, les entiers a et k étant donnés par

$$k := \nu(v) - \nu(u), \quad a \text{ impair } \text{ et } 0 \leq a < 2^{k+1}.$$

Ainsi les soustractions successives effectuées lors de la division de $29 = 11101_2$ par $12 = 1100_2$ sont

$$11101_2 - 11_2 = 11010_2, \; 11010_2 - 110_2 = 10100_2, \; 10100_2 - 1100_2 = 1000_2,$$

ce qui se résume par

$$11101_2 = 1100_2 \times \frac{1_2 + 10_2 + 100_2}{100_2} + 1000_2, \quad i.e., 29 = \frac{7}{4} \times 12 + 8.$$

Certains problèmes peuvent cependant apparaître lorsqu'on itère ces divisions. Remarquons tout d'abord qu'une telle division peut très bien générer un reste négatif, et que la division d'un entier positif par un entier négatif produit un reste éventuellement supérieur (en valeur absolue) aux deux entiers de départ. On peut de la sorte obtenir une suite infinie de divisions. C'est le cas si on pose $u = -1$ et $v = 2$: la suite des restes est -1, 2, -4, 8, -16 etc... Une manière d'éviter ce problème est de centrer le quotient q pour finalement obtenir une division définie par

$$u = vq + r, \quad |q| < 1 \quad \text{et} \quad |r|_2 < \frac{1}{2}|v|_2, \tag{1.8}$$

le quotient obtenu étant alors de la forme

$$q = \frac{a}{2^{\nu(v)-\nu(u)}}, \quad k := \nu(v) - \nu(u), \quad a \text{ impair et } -2^k < a < 2^k.$$

La division centrée de 29 par 12 est alors

$$11101_2 = 1100_2 \times \frac{-1_2}{100_2} + 100000_2, \quad i.e., \quad 29 = \frac{-1}{4} \times 12 + 32.$$

L'algorithme permettant de calculer le quotient de la division LSB est décrit dans la figure 1.4, et l'algorithme LSB, noté $\mathcal{L}$, est finalement obtenu en itérant les divisions jusqu'à obtenir un reste nul.

Entrée : $(u, v) \in \mathbb{Z}^2$, avec $|u|_2 \geq |v|_2$.
Sortie : $q = d(u, v)$.

$q := 0$;
`Tant que` $|u|_2 \geq |v|_2$
$u := u - v \cdot 2^{\nu(u)-\nu(v)}$;
$q := q + 2^{\nu(v)-\nu(u)+1}$;
`Si` $q \geq 2^{\nu(v)-\nu(u)}$ `alors` $q := q - 2^{\nu(v)-\nu(u)+1}$;
$q := q/2^{\nu(v)-\nu(u)}$;
`Renvoyer` q.

FIG. 1.4 – L'algorithme de division LSB

Remarquons que la propriété de symétrie observée sur les polynômes n'est plus valable ici. Considérons l'algorithme $\mathcal{L}$ appliqué aux entiers $115 = 1110011_2$ et $82 = 1010010_2$. La suite de restes obtenue est

$115 =$	1110011_2
$82 =$	1010010_2
$156 =$	10011100_2
$160 =$	10100000_2
$256 =$	100000000_2
$0 =$	0_2

Entrée : $(u, v) \in \mathbb{Z}^2$, avec $|u|_2 \geq |v|_2$.
Sortie : pgcd (u, v).

$i := 0$;
`Tant que` $u_{i+1} \neq 0$
$q_{i+1} := d(u_i, u_{i+1})$;
$u_{i+2} := u_i - u_{i+1} q_{i+1}$;
$i := i + 1$;
`Renvoyer` $u_i \cdot 2^{\nu(v)-\nu(u)-\nu(u_i)}$.

FIG. 1.5 – L'algorithme de pgcd LSB $\mathcal{L}$

Considérons maintenant l'algorithme d'Euclide appliqué aux miroirs de 115 et 82, c'est à dire $103 = 1100111_2$ et $37 = 0100101_2$. La suite des restes est 103, 37, 29, 8, 5, 3, 2, 1, 0, qui ne correspond pas aux miroirs de la suite précédente.

Le formalisme générique décrit dans la section 1.1 ne s'applique plus exactement ici. En effet, étant donnée une division LSB et le quotient associé $d = (a, k)$, la matrice correspondant à cette division est

$$\mathcal{M}_{[d]} := \frac{1}{2^k} \begin{pmatrix} 0 & 2^k \\ 2^k & a \end{pmatrix}. \tag{1.9}$$

Enfin, remarquons qu'il n'est pas nécessaire de conserver toutes les puissances de 2 apparaissant au cours de l'algorithme. A partir de la division

$$u = vq + r, \quad |q| < 1 \quad \text{et } |r|_2 < \frac{1}{2}|v|_2, \tag{1.10}$$

il est plus judicieux d'itérer le processus avec la paire $(v', r') = (v2^{-\nu(v)}, r2^{-\nu(v)})$ de sorte qu'au cours d'une exécution de l'algorithme, chaque division se fait sur une paire d'entiers (u_i, u_{i+1}) telle que u_i est impair et $\nu(u_{i+1}) > 0$. On est dans ce cadre amené à considérer les matrices $\mathcal{N}_{[d]}$,

$$\mathcal{N}_{[d]} := \begin{pmatrix} 0 & 2^k \\ 2^k & a \end{pmatrix}. \tag{1.11}$$

La suite de restes obtenues sur entrée $(115, 82)$ est maintenant

$$\begin{array}{rr} 115 = & 1110011_2 \\ 41 = & 101001_2 \\ 39 = & 100111_2 \\ 5 = & 101_2 \\ 1 = & 1_2 \\ 0 = & 0_2 \end{array}$$

Finalement, si g est le pgcd de la paire (u, v), si $d_1 = (a_1, k_1), \ldots, d_p = (a_p, k_p)$ est la suite de chiffres engendrée par l'algorithme, si $k = \sum_{i=1}^{p} k_i$ et si on note $\mathcal{M}_p$ et $\mathcal{N}_p$ les matrices

$$\mathcal{M}_p := \mathcal{M}_{[d_1]} \cdots \mathcal{M}_{[d_p]} \quad \text{et} \quad \mathcal{N}_p := \mathcal{N}_{[d_1]} \cdots \mathcal{N}_{[d_p]}$$

alors on vérifie

$$\mathcal{U}_0 = \begin{pmatrix} v \\ u \end{pmatrix} = \frac{1}{2^k} \cdot \mathcal{M}_p \cdot \begin{pmatrix} 0 \\ 2^k g \end{pmatrix} = \mathcal{N}_p \cdot \begin{pmatrix} 0 \\ g \end{pmatrix}. \tag{1.12}$$

1.5 Les algorithmes Mixtes

Finalement, il existe des algorithmes n'appartenant à aucune des deux précédentes catégories, en ce sens que les divisions sur lesquelles ils sont basés utilisent à la fois les bits les plus et les moins significatifs des entiers.

1.5.1 Les algorithmes pseudo-euclidiens

Tout d'abord, il est possible de définir à partir des algorithmes MSB des algorithmes Mixtes, en retirant par décalage les puissances de deux apparaissant dans les restes. On obtient ainsi la famille des algorithmes pseudo-euclidiens, décrite dans [Sha90, Val03]. Par exemple, l'algorithme pseudo-Euclide $\widetilde{\mathcal{E}}$ est basé sur la division

$$u = vq_1 + 2^k r_2, \quad 0 \leq 2^k r_2 < v, \quad \text{et} \quad r_2 \quad \text{impair}.$$

Le reste r_2 de cette division se calcule facilement à partir du reste r de la division standard en posant $k := \nu(r)$. On peut de cette manière associer à chaque algorithme MSB un algorithme dont la division est initialement guidée par les bits de poids forts, mais qui utilise également les bits de poids faibles. On associe à chaque algorithme MSB H un algorithme pseudo-euclidien $\widetilde{H}$. On obtient donc les algorithmes suivants : $\widetilde{\mathcal{E}}$, $\widetilde{I}$, $\widetilde{\mathcal{C}}$, $\widetilde{\mathcal{P}}$, $\widetilde{\mathcal{X}}$, et $\widetilde{\mathcal{E}}_\alpha$, basés sur des divisions qui d'une manière générique s'écrivent

$$u = qv + \varepsilon 2^k r.$$

1.5.2 Les algorithmes Binaire, Plus-Moins et leurs généralisations

Une autre approche pour construire ces algorithmes Mixtes est de retirer les puissances de 2 non pas au reste d'une division MSB, mais au cours même du calcul de cette division. On peut en effet calculer le reste r de la division euclidienne standard en procédant par soustractions successives de la forme $u := u - v$ et en posant finalement $r := u$ dès que u est strictement inférieur à v. On obtient ainsi ce qui est appelé l'algorithme soustractif. La pseudo version de cet algorithme conduit au principe de l'algorithme Binaire de Stein [Ste67], qui est donc une succession de soustractions et décalages du type

$$u := \frac{u - v}{2^k}$$

où $k = \nu(u - v)$. En imposant à u et v d'être impairs, on s'assure de plus que k est supérieur ou égal à 1, et on obtient l'algorithme Binaire, noté $\mathcal{B}$.

Ainsi, entre deux échanges, l'algorithme Binaire effectue les opérations

$$u_1 = \frac{u - v}{2^{k_1}}, \quad u_2 = \frac{u_1 - v}{2^{k_2}}, \quad u_3 = \frac{u_2 - v}{2^{k_3}}, \ldots, \quad u_r = \frac{u_{r-1} - v}{2^{k_r}},$$

et la division Binaire est finalement définie par

$$u = va + 2^k r \tag{1.13}$$

où $a = 1 + 2^{k_1} + 2^{k_1+k_2} + \cdots + 2^{k_1+\cdots+k_{r-1}}$ et $k = k_1 + \cdots + k_r$. L'idée de base de l'algorithme Binaire, qui est que la différence de deux entiers impairs est divisible par deux, se généralise très naturellement. En considérant que soit la somme, soit la différence de ces deux entiers est divisible

```
Entrée : (u, v) ∈ ℕ² impairs avec u ≥ v.
Sortie : pgcd (u, v)

Tant que  u ≠ v
    Tant que u > v
            u := u − v ;
            u := u/2^ν(u) ;
    Échanger u et v ;
Renvoyer  u.
```

FIG. 1.6 – L'algorithme Binaire $\mathcal{B}$.

par 4, on obtient l'algorithme Plus-Moins $\mathcal{PM}$ de Brent et Kung [BK85], dont les opérations de base sont donc de la forme :

$$u := \frac{u + \varepsilon v}{2^k}, \quad \varepsilon = \pm 1,$$

le signe de l'opération étant choisi pour maximiser k. Cet algorithme est décrit dans la figure (1.7), et entre deux échanges les opérations sont maintenant du type

$$u_1 = \frac{u + \varepsilon_1 v}{2^{k_1}}, \quad u_2 = \frac{u_1 + \varepsilon_2 v}{2^{k_2}}, \quad u_3 = \frac{u_2 + \varepsilon_3 v}{2^{k_3}}, \ldots, \quad u_r = \frac{u_{r-1} + \varepsilon_r v}{2^{k_r}},$$

ce qui mène à

$$u = va + 2^k r$$

où $a = 1 - \varepsilon_1 2^{k_1} - \varepsilon_2 2^{k_1+k_2} \cdots - \varepsilon_{r-1} 2^{k_1+\cdots+k_{r-1}}$, $\varepsilon_i = \pm 1$ et $k = k_1 + \cdots + k_r$.

```
Entrée : (u, v) ∈ ℕ² impairs avec u ≥ v.
Sortie : pgcd (u, v)

Tant que  u ≠ v
    Tant que u > v
        Si  4|u − v
            Alors  u := u − v ;
            Sinon  u := u + v ;
        u := u/2^ν(u) ;
    Échanger u et v ;
Renvoyer  u
```

FIG. 1.7 – L'algorithme Plus-Moins $\mathcal{PM}$.

On peut poursuivre la généralisation de l'algorithme Binaire jusqu'à obtenir des algorithmes plus sophistiqués, tels ceux de Sorenson [Sor94], Weber [Web95] ou Jebelean [Jeb93b] (Sorenson a décrit un premier algorithme, amélioré par la suite indépendamment par Weber et Jebelean). Nous présentons brièvement ici les principes qui sous-tendent tous ces algorithmes. L'idée est maintenant de considérer des divisions de la forme

$$\frac{xu + yv}{2^k} = r, \tag{1.14}$$

Algorithme		Entrée	Division	Quotient	Reste
Euclide	$\mathcal{E}$	$0 \leq v < u$	$u = vq + r$	$q \in \mathbb{N}, q \geq 1$	$0 \leq r < v$
Pair	$\mathcal{I}$	$0 \leq v < u$	$u = vq + r$	$q \in \mathbb{N}, q \geq 2$ pair	$0 \leq r < v$
Pair	$\mathcal{I}$	$0 \leq v < u$	$u = vq + r$	$q \in \mathbb{N}, q \geq 1$ impair	$0 \leq r < v$
Centré	$\mathcal{C}$	$0 \leq v < u/2$	$u = vq + \varepsilon r$	$(q, \varepsilon) \in \mathbb{Z} \times \{\pm 1\}$ $(q, \varepsilon) \geq (2, +1)$	$0 \leq r < v/2$
Par-Excès	$\mathcal{X}$	$0 \leq v < u$	$u = vq - r$	$q \in \mathbb{N}, q \geq 2$	$0 \leq r < v$
α-euclidien	$\mathcal{E}_\alpha$	$(1-\alpha)u \leq v < \alpha u$	$u = vq + \varepsilon r$	$\{(q,-1) \quad q \geq r^-(\alpha)\}$ $\cup\{(q,+1), \quad q \geq r^+(\alpha)\}$	$(\alpha - 1) \leq \frac{r}{u} < \alpha$
LSB	$\mathcal{L}$	$\lvert v\rvert_2 < \lvert u\rvert_2$	$u = vq + r$	$q = \frac{a}{2^k}$, $k \geq 1$, $\lvert a\rvert < 2^k$	$\lvert r\rvert_2 < \lvert v\rvert_2$
Binaire	$\mathcal{B}$	v, u impairs, $v < u$	$u := \dfrac{u - v}{2^k}$	$k \geq 1$	$0 \leq r < v$
Plus-Moins	$\mathcal{PM}$	v, u impairs, $v < u$	$u := \dfrac{u \pm v}{2^k}$	$k \geq 2$	$0 \leq r < v$

FIG. 1.8 – Les différents algorithmes euclidiens.

en choisissant x et y de manière à s'assurer que k est supérieur à une valeur k_0 fixée à l'avance. Ainsi, en posant $x = 1$, $y = -1$ on a $k_0 = 1$ et on retrouve l'algorithme Binaire. De même, avec $x = 1$ et $y = \pm 1$ on obtient l'algorithme Plus-Moins. D'une manière générale, le calcul de x et y se fait de la manière suivante : soit c donné par

$$c = (uv^{-1}) \bmod 2^{k_0}.$$

Appliquons l'algorithme d'Euclide étendu à c et 2^k. On obtient une suite de restes x_i et de coefficients e_i, d_i, tous reliés par

$$e_i c + d_i 2^{k_0} = x_i.$$

Le choix de c implique finalement pour tout i

$$e_i u - x_i v = 0 \bmod 2^{k_0}.$$

On peut ainsi choisir d'arrêter l'algorithme d'Euclide étendu à tout moment en posant $y = a_i$ et $x = x_i$. Un choix judicieux est ici d'utiliser l'algorithme interrompu de paramètre $1/2$, ce qui implique, comme nous le montrerons par la suite que les coefficients y et x sont de mêmes tailles, ce qui minimise le coût des multiplications de (1.14).

1.6 Étude probabiliste des algorithmes Euclidiens

L'étude des algorithmes euclidiens est un vieux domaine, bien antérieur à l'apparition de l'informatique. Ainsi, la première analyse dans le pire des cas de l'algorithme d'Euclide remonte au 18ème siècle avec Lamé, qui a exhibé le lien entre ce pire des cas et la suite de Fibonacci. D'une manière générale, sur l'ensemble des entrées de taille binaire inférieure ou égale à N, le nombre maximum d'itérations des algorithmes euclidiens est $O(N)$ et la complexité en bits $O(N^2)$. Obtenir un résultat plus précis sur le pire des cas n'est pas toujours facile. Si pour l'algorithme d'Euclide il est lié à la quantité $\frac{1+\sqrt{5}}{2}$, Stehlé et Zimmermann ont montré que pour l'algorithme

$\mathcal{L}$ il était lié à $\frac{\sqrt{17}-1}{2}$. Pour l'algorithme Binaire, ce pire des cas est donné par les paires $(2^N-1,1)$ alors que pour l'algorithme Plus-Moins il n'a toujours pas été déterminé.

Une étude plus instructive est celle du comportement probabiliste de ces algorithmes. Celle-ci apporte en effet plus d'informations sur le comportement réel des algorithmes, tel qu'observé par expérimentations pratiques. De nombreux paramètres d'intérêt peuvent ainsi être étudiés, allant du nombre d'itérations à la complexité en bits en passant par la longueur moyenne des quotients, des restes ou des continuants. Nous commençons par décrire le cadre probabiliste dans lequel nous étudions les différents paramètres, puis définissons ceux-ci. Nous détaillons ensuite les principaux résultats obtenus dans ce domaine, en particulier ceux obtenus par les méthodes d'analyse dynamiques qui s'expriment tous selon un formalisme unifié. Enfin, nous énonçons les résultats prouvés par la suite dans ce mémoire.

1.6.1 Modèle probabiliste

La première notion à fixer pour décrire notre modèle est celle de taille des entrées de l'algorithme étudié. D'une manière générale, étant donnée une entrée (u,v) d'un algorithme, nous distinguerons sa taille, notée $\ell(u,v)$, de sa norme, notée $||(u,v)||$. Plus précisément, la taille de la paire (u,v) est la longueur binaire de la norme, c'est-à-dire

$$\ell(u,v) = \ell_2(||(u,v)||), \tag{1.15}$$

où ℓ_2 désigne la longueur binaire d'un entier,

$$\ell_2(u) = \lfloor \log_2 u \rfloor + 1.$$

La norme de la paire est définie en fonction du contexte, ou de l'algorithme étudié. Pour les algorithmes MSB ou Mixtes, on sait, étant donnée une entrée (u,v), que l'inégalité $|u| > |v|$ est vérifiée. On choisit donc $||(u,v)|| = |u|$ et la taille $\ell(u,v)$ correspond donc à l'espace mémoire occupé par le plus grand des entiers. La situation est différente pour l'algorithme LSB, puisque v peut être arbitrairement plus grand que u. Nous choisissons dans ce contexte la norme euclidienne, et posons donc

$$||(u,v)|| := \begin{cases} |u| & \text{pour les algorithmes MSB et Mixtes,} \\ (u^2+v^2)^{\frac{1}{2}} & \text{pour l'algorithme LSB.} \end{cases} \tag{1.16}$$

Enfin, toujours dans le cadre de l'étude de l'algorithme LSB, nous serons également amenés à considérer une norme plus exotique. Si pour une entrée (u,v) l'algorithme LSB fait k décalages, nous poserons

$$||(u,v)|| = 2^k. \tag{1.17}$$

D'une manière générale, la norme associée à l'algorithme LSB est la norme euclidienne, nous préciserons explicitement lorsque nous utiliserons la norme définie en (1.17).

Étant donné un algorithme euclidien, nous lui associons l'ensemble $\widetilde{\Omega}$ de ses entrées valides,

$$\widetilde{\Omega} := \{(u,v), \quad (u,v) \text{ est une entrée valide de l'algorithme}\}, \tag{1.18}$$

cet ensemble étant défini comme plus petit ensemble stable par l'opération de division associée à l'algorithme. Ainsi, l'ensemble des entrées valides de l'algorithme d'Euclide est

$$\widetilde{\Omega} := \{(u,v), \quad 0 < v \leq u\},$$

celui des algorithmes Binaires et Plus-Moins est

$$\widetilde{\Omega} = \{(u,v), \quad 0 < v \leq u, \quad u,v \text{ impairs}\},$$

et enfin l'ensemble des entrées valides de l'algorithme LSB est

$$\widetilde{\Omega} := \{(u,v), \quad 0 \leq \nu(u) < \nu(v)\}.$$

Si ℓ désigne la taille associée à un algorithme, nous étudierons le comportement des coûts décrits précédemment sur les sous-ensembles de Ω suivant :

$$\widetilde{\Omega}_N := \left\{(u,v) \in \widetilde{\Omega}, \quad \ell(u,v) = N\right\} \quad \text{et} \quad \widetilde{\Omega}_N^+ := \left\{(u,v) \in \widetilde{\Omega}, \quad \ell(u,v) \leq N\right\}.$$

Enfin, nous serons également amenés à travailler sur les ensembles

$$\Omega := \left\{(u,v) \in \widetilde{\Omega}, \text{pgcd } (u,v) = 1\right\}, \qquad \Omega_N := \left\{(u,v) \in \widetilde{\Omega}_N, \text{pgcd } (u,v) = 1\right\},$$

$$\text{et} \qquad \Omega_N^+ := \left\{(u,v) \in \widetilde{\Omega}_N^+, \text{pgcd } (u,v) = 1\right\}.$$

Nous verrons par la suite qu'étudier le comportement des algorithmes sur les ensembles Ω et $\widetilde{\Omega}$ conduit aux mêmes résultats. L'analyse est cependant plus aisée sur Ω, c'est pourquoi nous introduisons cet ensemble (même si à première vue il peut sembler étrange d'étudier des algorithmes de calcul de pgcd sur des ensembles d'entrées premières entre elles).

Nous pouvons maintenant définir sur ces ensembles des probabilités et des espérances. Ainsi, sur l'ensemble Ω_N, elles sont définies par

$$\mathbb{P}_N[(u,v) \in B] := \frac{|\Omega_N \cap B|}{|\Omega_N|},$$

et

$$\mathbb{E}_N[X] := \frac{\sum_{(u,v)\in\Omega_N} X(u,v)}{|\Omega_N|},$$

si X est une variable aléatoire définie sur Ω. On obtient de la même manière les probabilités $\mathbb{P}_N^+$, $\widetilde{\mathbb{P}}_N$, $\widetilde{\mathbb{P}}_N^+$ et les espérances $\mathbb{E}_N^+$, $\widetilde{\mathbb{E}}_N$, $\widetilde{\mathbb{E}}_N^+$.

1.6.2 Paramètres d'intérêt

Les différents paramètres étudiés sont de trois types, selon les grandeurs qu'ils font intervenir. En effet, l'exécution d'un algorithme euclidien génère deux sortes d'observables : celles liées aux quotients et celles liées aux restes, continuants ou convergents. Les paramètres étudiés sont soit liés à une de ces deux familles d'observables, soit aux deux.

Coûts additifs définis sur les quotients

La principale variable aléatoire liée aux quotients est celle qui permet d'exprimer le nombre d'itérations. Si on pose $i(d) = 1$, on définit la variable I par

$$I(u, v) := \sum_{i=1}^{p} i(d_i) = p.$$

Nous sommes également intéressés par les tailles $\ell(d)$ des quotients successifs. On définit la taille $\ell(d)$ d'un quotient d de différentes manières, selon la nature même de ce quotient. Ainsi, pour l'algorithme d'Euclide, pour lequel on a $d = (q)$, alors $\ell(d) = \ell_2(q)$. Si d est une paire (q, ε), alors $\ell(d) = \ell_2(q) + 1$, si c'est une paire (a, k) alors $\ell(d) = \ell_2(a) + k$. Finalement la taille d'un triplet (q, ε, k) (algorithmes pseudo-euclidiens) est $\ell_2(q) + 1 + k$.

D'une manière plus générale, nous considérons par la suite toute une famille de coûts liés aux chiffres, les coûts à croissance modérée. Nous donnerons une définition précise des coûts à croissance modérée dans le chapitre 5, mais d'une manière générale un coût est à croissance modérée si il est de la forme $O(\ell)$. Cette famille de coûts recouvre en particulier de nombreux paramètres. Outre ceux présentés ci-dessus, on peut par exemple citer le coût $c = \mathbf{1}_d$ qui permet d'exprimer le nombre d'occurrences d'un chiffre d donné.

A partir d'un tel coût c, on définit la variable aléatoire C par

$$C(u, v) := \sum_{i=1}^{p} c(d_i).$$

Les différentes variables relatives aux algorithmes interrompus sont généralement indicées par un t, désignant le paramètre d'interruption de l'algorithme. Ainsi, $I_t(u, v)$ est le nombre d'itérations de l'algorithme interrompu de paramètre t, et étant donné un coût c, la variable de coût total C_t est donnée par

$$C_t(u, v) := \sum_{i=1}^{I_t(u,v)} c(d_i).$$

Coûts définis sur les restes et continuants

Les coûts définis à partir des suites des restes et des continuants apparaissent sous diverses formes dans nos analyses. Elles interviennent dans la complexité en bits (*cf* paragraphe suivant) mais aussi dans l'étude de la variable I_t du nombre d'itérations de l'algorithme interrompu $\mathcal{E}_t$. En effet, remarquons que les évènements $[I_t > \delta p]$ et $[I_t \leq \delta p]$ où $\delta \in [0, 1]$ vérifient

$$[I_t > \delta p] = [I_t > \lfloor \delta p \rfloor] = \left[\frac{u_{\lfloor \delta p \rfloor}}{u_0^t} > 1 \right],$$

$$[I_t \leq \delta p] = [I_t \leq \lfloor \delta p \rfloor] = \left[\frac{u_{\lfloor \delta p \rfloor}}{u_0^t} \leq 1 \right],$$

par définition de l'algorithme. En appliquant l'inégalité de Markov, on obtient

$$\mathbb{P}_n[I_t > \delta p] \leq \mathbb{E}_n \left[\left(\frac{u_{\lfloor \delta p \rfloor}}{u_0^t} \right)^{\gamma} \right], \quad \text{pour tout } \gamma > 0,$$

$$\mathbb{P}_n[I_t \leq \delta p] \leq \mathbb{E}_n\left[\left(\frac{u_{\lfloor \delta p \rfloor}}{u_0^t}\right)^\gamma\right], \quad \text{pour tout } \gamma < 0,$$

et on est donc amené à étudier la variable aléatoire

$$M_{t,\delta,\gamma}(u,v) = \left(\frac{u_{\lfloor \delta p \rfloor}}{u_0^t}\right)^\gamma. \tag{1.19}$$

Enfin, avec l'algorithme interrompu $\underline{\mathcal{E}}_\delta$ nous étudions l'évolution des positions des rationnels formés par deux restes successifs. Plus précisément, nous étudions les rationnels apparaissant à une fraction du nombre total d'itérations. Si on note $x_\delta(u,v)$ le rationnel

$$x_\delta(u,v) := \frac{u_{\lfloor \delta p \rfloor + 1}}{u_{\lfloor \delta p \rfloor}},$$

et si B est un intervalle inclus dans $[0,1]$, on étudie donc le coût $U_{\delta,B}(u,v)$ défini par

$$U_{\delta,B}(u,v) := \mathbf{1}_B(x_\delta(u,v)). \tag{1.20}$$

Complexité en bits

Finalement, les coûts relatifs à la complexité en bits sont liés aux quotients et ainsi qu'aux restes. Ainsi, l'i-ème itération d'un algorithme a un coût en bit de la forme $\ell(d_i) \times \ell(u_i)$[1], le coût supplémentaire dû au calcul d'un continuant étant $\ell(d_i) \times \ell(a_i)$. Les variables aléatoires correspondantes sont donc

$$B(u,v) := \sum_{i=1}^{p} \ell(d_i) \times \ell(u_i) \quad \text{et} \quad X(u,v) := \sum_{i=1}^{p} \ell(d_i) \times \ell(a_i) \tag{1.21}$$

La variable BX de la complexité en bits d'un algorithme étendu est donnée par

$$BX(u,v) = B(u,v) + 2X(u,v).$$

Lorsque la division nécessite des opérations comme des soustractions éventuelles (comme dans le calcul des quotients des algorithmes α-euclidiens (1.6)) nous en tenons compte en modifiant légèrement la notion de taille des quotients.

Enfin, les coûts associés à la complexité en bits de l'algorithme d'Euclide interrompu de paramètre t, $\mathcal{E}_t$ sont donnés par

$$B_t(u,v) := \sum_{i=1}^{I_t(u,v)} \ell(d_i) \times \ell(u_i), \quad X_t(u,v) := \sum_{i=1}^{I_t(u,v)} \ell(d_i) \times \ell(a_i), \tag{1.22}$$

$$\text{et} \quad BX_t(u,v) = B_t(u,v) + 2X_t(u,v),$$

où $I_t(u,v)$ est toujours le nombre d'itérations effectué par l'algorithme. L'étude de ces coûts se fera à l'aide de l'autre algorithme interrompu, $\underline{\mathcal{E}}_\delta$, qui s'arrête à la $\lfloor \delta p \rfloor$-ème itération. Les coûts relatifs à cet algorithme sont notés $\underline{B}_\delta$, $\underline{X}_\delta$ et $\underline{BX}_\delta$, et sont donc donnés par

$$\underline{B}_\delta(u,v) := \sum_{i=1}^{\lfloor \delta I(u,v) \rfloor} \ell(d_i) \times \ell(u_i), \quad \underline{X}_\delta(u,v) := \sum_{i=1}^{\lfloor \delta I(u,v) \rfloor} \ell(d_i) \times \ell(a_i), \tag{1.23}$$

[1] Les coûts que nous attribuons à une division sont ceux d'une division naïve. Les algorithmes de division plus sophistiqués, utilisant la multiplication de Karatsuba par exemple, ont un coût en bits très différent...

$$\text{et} \quad \underline{BX}_\delta(u,v) = \underline{B}_\delta(u,v) + 2\underline{X}_\delta(u,v).$$

Remarquons enfin que nous serons amenés à étudier d'autres coûts en vue de la complexité en bits. Plus exactement, les coûts B et X sont relativement "difficiles" à générer, et nous travaillerons sur des approximations, dont nous montrerons qu'elles sont suffisantes pour finalement obtenir la complexité en bits moyenne (la situation est différente pour une analyse en distribution). Ces approximations sont les coûts $\widehat{B}$ et $\widehat{X}$, définis par

$$\widehat{B}(u,v) := \sum_{i=1}^{p} \ell(d_i) \times \log_2(u_i) \tag{1.24}$$

$$\widehat{X}(u,v) := \sum_{i=1}^{p} \ell(d_i) \times \log_2\left(a_i + a_{i-1}\frac{u_i}{u_{i-1}}\right). \tag{1.25}$$

Remarquons dès maintenant que ces approximations sont "relativement" bonnes, puisqu'on vérifie toujours l'inégalité

$$|\log_2(u_i) - \ell(u_i)| \leq 1. \tag{1.26}$$

Dans certains cas, on a également ce type d'inégalité pour le coût défini en (1.25) (voir chapitre 5).

Remarquons également que pour l'étude de l'algorithme LSB, nous utiliserons une autre approximations, toujours notée $\widehat{B}$, définie par

$$\widehat{B}(u,v) := \frac{1}{2}\sum_{i=1}^{p} \ell(d_i) \times \log_2(u_i^2 + u_{i+1}^2). \tag{1.27}$$

Cette approximation est naturelle si on considère la taille utilisée pour l'étude de cet algorithme (voir (1.16)). Elle est suffisante pour notre analyse puisqu'on vérifie

$$\widehat{B} - B = O(Q) \tag{1.28}$$

où Q est

$$Q(u,v) = \sum_{i=1}^{p} \ell(d_i).$$

Enfin, tous ces coûts sont également définis pour les algorithmes interrompus. Dans ce cas, on obtient les coûts $\widehat{B}_t$, $\widehat{X}_t$, $\widehat{BX}_t$, $\underline{\widehat{B}}_\delta$, $\underline{\widehat{X}}_\delta$ et finalement $\underline{\widehat{BX}}_\delta$.

1.7 Résultats connus sur les algorithmes euclidiens

Les premiers travaux dans ce domaine remontent à 1969 quand Heilbronn [Hei69] et Dixon [Dix70] ont indépendamment étudié le nombre moyen d'itérations de l'algorithme d'Euclide. Par la suite, ce paramètre a également été étudié pour d'autres algorithmes par Knuth et Yao [YK75] pour l'algorithme Soustractif, Rieger [Rie78] pour l'algorithme Centré, Vardi [Var] pour l'algorithme pseudo-Euclide, et Brent [Bre76] qui a sous certaines hypothèses montré que le nombre moyen d'itérations de l'algorithme Binaire sur l'ensemble des entrées de taille inférieures ou égales à N était asymptotiquement linéaire en N. Finalement, Hensley [Hen94] a, en 1994, effectué la première analyse en distribution du nombre d'itérations de l'algorithme d'Euclide. Toutes ces analyses reposent sur des techniques diverses, de natures purement combinatoires ou probabilistes. A partir du milieu des années 1990 s'est développée autour de Brigitte Vallée

$I(u,v) := \sum_{i=0}^{p} 1$	Nombre d'itérations
$C(u,v) := \sum_{i=1}^{p} c(d_i)$	Coût additif défini sur les quotients
$M_{t,\delta,\gamma}(u,v) = \left(\frac{u_{\lfloor \delta p \rfloor}}{u_0^t}\right)^{\gamma}$	Nombre d'itérations de l'algorithme interrompu
$U_{\delta,B}(u,v) := \mathbf{1}_B(x_\delta(u,v))$	Évolution de la distribution des restes
$B(u,v) := \sum_{i=1}^{p} \ell(d_i) \times \ell(u_i)$	Complexité en bits
$X(u,v) := \sum_{i=1}^{p} \ell(d_i) \times \ell(a_i)$	Complexité en bits : calcul d'un coefficient de Bezout
$B_t, X_t, \underline{B}_\delta, \underline{X}_\delta$	Complexité en bits : algorithmes interrompus
$\widehat{B}, \widehat{B}_t, \widehat{\underline{B}_\delta}, \widehat{X}, \widehat{X}_t, \widehat{\underline{X}_\delta}$	Complexité en bits : coûts approximants

FIG. 1.9 – Les principaux coûts étudiés

une méthode générique d'analyse d'algorithmes euclidiens, l'analyse dynamique, en partie basée sur la modélisation en système dynamique des algorithmes. Cette approche permet d'exprimer le comportement de nombreux paramètres en fonction de certaines caractéristiques du système sous-jacent à l'algorithme étudié, et ainsi permet d'énoncer de manière unifiée les différents résultats. D'une manière générale, les résultats obtenus s'expriment en fonction de l'entropie du système, et de la valeur moyenne (moyenne au sens du système dynamique) du paramètre considéré, ces deux grandeurs étant explicites dès que l'est la densité invariante du système dynamique.

Ce paragraphe est consacré aux résultats connus avant la thèse. Nous en distinguons ici trois types. Le premier traite du comportement moyen des coûts à croissance modérée, parmi lesquels le coût correspondant au nombre d'itérations. Le second traite du comportement moyen de la complexité en bits, et le dernier type de résultats concerne le comportement en distribution des coûts à croissance modérée. Cette distinction respecte l'ordre chronologique puisque l'essentiel des résultats du premier type a été obtenu entre 1998 et 2000 par Brigitte Vallée [Val98a, Val03], les premiers résultats sur la complexité en bits datent des travaux de Akhavi et Vallée en 2000 [AV00, Val00] et enfin les premières analyses en distribution remontent à l'article de Baladi-Vallée en 2004 [BV04]. Nous présentons tous ces résultats de manière "informelle", l'énoncé précis des différents théorèmes se trouvant dans le chapitre 5.

Coûts additifs

L'étude des coûts à croissance modérée a permis dans un premier temps de classifier les différents algorithmes, en fonction de leur efficacité. Ainsi, les algorithmes présentés ici peuvent être séparés en deux catégories, les algorithmes rapides et les algorithmes lents. Les algorithmes de la première famille, qui contient les algorithmes $\mathcal{E}$, $\mathcal{I}$, $\mathcal{C}$, $\widetilde{\mathcal{E}}$, $\widetilde{\mathcal{C}}$, effectuent un nombre moyen

d'itérations linéaire en la taille des entrées. Cette linéarité s'étend à tous les coûts à croissance modérée. Plus précisément, si c est un tel coût, C la variable aléatoire associée, alors le nombre moyen d'itérations d'un algorithme rapide H sur l'ensemble Ω_N (ou $\widetilde{\Omega}_N$) de ses entrées valides de taille N est asymptotiquement de la forme

$$\mathbb{E}_N[C] \sim A_H \cdot \mu_H(c) \cdot N \sim \widetilde{\mathbb{E}}_N[C],$$

où les constantes A_H et μ_H sont intrinsèquement reliées au système dynamique sous-jacent à l'algorithme. La première constante s'exprime en fonction de l'entropie du système $h(H)$ via la relation

$$A_H := \frac{2\log 2}{h(H)}. \tag{1.29}$$

La seconde est la valeur moyenne du coût c, cette moyenne devant être prise au sens du système dynamique. Lorsque la densité invariante du système dynamique est connue, on peut expliciter les constantes A_H, $h(H)$ et $\mu_H(c)$. Par exemple, les entropies des systèmes relatifs aux algorithmes d'Euclide, d'Euclide Centré et d'Euclide Impaire sont données par

$$h(\mathcal{E}) = \frac{\pi^2}{6\log 2}, \quad h(\mathcal{C}) = \frac{\pi^2}{6\log \phi}, \quad h(I) = \frac{\pi^2}{9\log \phi},$$

où ϕ est le nombre d'or $\phi = (\sqrt{5}+1)/2$. On peut directement en déduire que le nombre moyen d'itérations de ces algorithmes est

$$\mathbb{E}_N[I] \sim \begin{cases} \dfrac{12\log^2 2}{\pi^2} \cdot N \sim 0.58 \cdot N & \text{pour l'algorithme d'Euclide,} \\ \dfrac{12\log 2 \log \phi}{\pi^2} \cdot N \sim 0.41 \cdot N & \text{pour l'algorithme Centré,} \\ \dfrac{18\log 2 \log \phi}{\pi^2} \cdot N \sim 0.61 \cdot N & \text{pour l'algorithme Impair.} \end{cases}$$

On peut déjà conclure que l'algorithme Centré effectue en moyenne moins d'itérations que les deux autres.

Les algorithmes lents, dont font partie les algorithmes $\mathcal{X}$, $\mathcal{P}$ et $\widetilde{\mathcal{X}}$ n'ont plus un comportement linéaire, mais quadratique. En effet, si H est un algorithme de cette classe, alors l'espérance $\mathbb{E}_N[C]$ d'une variable aléatoire C relative à un coût à croissance modérée est

$$\mathbb{E}_N[C] \sim B_H \cdot \mu_H(c) \cdot N^2,$$

où ici aussi la constante $\mu_H(c)$ est la même que précédemment, mais la constante B_H dépend maintenant de la fonction $\zeta_H(s)$ associée aux entrées valides de l'algorithme, $\zeta_H(s) = \sum_{(u,v)\in\Omega} \ell(u,v)^{-s}$, par

$$B_H := \frac{\log^2 2}{\zeta_H(2)}.$$

Complexité en bits

Nous verrons par la suite que cette dichotomie s'observe également sur les systèmes dynamiques sous-jacents aux algorithmes des deux classes, et qu'ainsi l'analyse dynamique fournit

une explication intuitive des différences de comportement entre ces familles d'algorithmes. Elle permet également d'étendre la dichotomie à une famille plus large et peut-être plus significative de coûts, les coûts liés à la complexité en bits. Ainsi, la complexité en bits moyenne des algorithmes rapides est asymptotiquement quadratique en N

$$\mathbb{E}_N[B] \sim \frac{A_H}{2} \cdot \mu_H(\ell) \cdot N^2, \tag{1.30}$$

où A_H est la même constante que précédemment et $\mu_H(\ell)$ est la valeur moyenne (toujours au sens du système dynamique) de la longueur des quotients. On dispose pour ce coût particulier d'expressions précises, puisque par exemple la constante $\mu_{\mathcal{E}}(\ell)$ relative à l'algorithme d'Euclide est

$$\mu_{\mathcal{E}}(\ell) = 1 + \frac{1}{\log 2} \log \prod_{k=0}^{\infty} (1 + \frac{1}{2^k}).$$

La complexité en bits moyenne des algorithmes lents est asymptotiquement cubique,

$$\mathbb{E}[B] \sim \Theta(N^3).$$

Analyse en distribution

Enfin, le développement récent de l'analyse dynamique en distribution a permis d'affiner ces résultats, puisqu'on peut montrer le comportement Gaussien des coûts à croissance modérée pour les algorithmes rapides. Plus précisément, si H est un de ces algorithmes, c un coût à croissance modérée et C le coût total associé, alors Baladi et Vallée ont montré un Théorème Central Limite : pour tout N et pour tout $Y \in \mathbb{R}$,

$$\mathbb{P}_N \left[(u,v) \;\mid\; \frac{C(u,v) - A_H \mu_H(c) N}{\delta_H(c)\sqrt{N}} \leq Y \right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-x^2/2}\, dx + O\left(\frac{1}{\sqrt{N}}\right).$$

Les constantes A_H et $\mu_H(c)$ sont les mêmes que précédemment, et la constante $\delta_H > 0$ est liée à la variance. Il existe en effet $\kappa < 1$ et des constantes a, b tels que la moyenne et la variance sont données par

$$\mathbb{E}_N[C] = A_H \cdot \mu_H(c) \cdot N + a + O(\kappa^n),$$

$$\mathrm{Var}_N[C] = \delta_H^2(c) \cdot N + b + O(\kappa^N).$$

Remarquons de plus qu'il est possible d'obtenir un théorème local limite pour les algorithmes de la classe A lorsque le coût considéré appartient à une famille que nous ne préciserons pas. Enfin, ces résultats en distributions s'obtiennent également pour des quantités plus précises. Lhote a en effet montré récemment [Lho05] le comportement asymptotiquement Gaussien de la complexité en bits des algorithmes rapides étendus.

1.8 Résultats de la thèse

Nous avons étudié dans ce mémoire à un représentant de chaque classe d'algorithme.

Algorithmes α-euclidiens

Comme nous venons de le voir, le comportement des algorithmes MSB est maintenant bien connu et compris. Cependant, la dichotomie observée dans le premier théorème nous amène à nous poser quelques questions sur les algorithmes α-euclidiens. En effet, pour deux valeurs du paramètre α, 1 et 1/2, l'algorithme appartient à la classe des algorithmes rapides, alors que pour $\alpha = 0$, l'algorithme correspondant (l'algorithme Par-Excès) appartient à la classe des algorithmes lents. Qu'en est-il pour d'autres valeurs de α ? Pour quelle valeur du paramètre l'algorithme est-il le plus rapide ? Nous avons, en collaboration avec Brigitte Vallée et Jérémie Bourdon, répondu à la première de ces questions : il apparaît que pour toutes les valeurs de α différentes de 0, l'algorithme correspondant appartient à la classe des algorithmes rapides. Nous avons en effet obtenu un résultat analogue au premier résultat du paragraphe précédent, qui le généralise. Étant donné $\alpha \in]0, 1]$, un coût c à croissance modérée et la variable C correspondante, alors l'espérance de cette dernière sur les ensembles Ω_N ou $\widetilde{\Omega}_N$ est asymptotiquement linéaire,

$$\mathbb{E}_N[C] \sim \frac{2\log 2}{h(\alpha)}\mu_\alpha(c)\cdot N := A_\alpha \mu_\alpha(c)\cdot N.$$

De la même manière que pour les autres algorithmes, les constantes A_α et $\mu_\alpha(c)$ dépendent du système dynamique associé à l'algorithme, puisqu'on vérifie

$$A_\alpha := \frac{2\log 2}{h(\alpha)},$$

$h(\alpha)$ étant l'entropie du système dynamique, et que $\mu_\alpha(c)$ est toujours la valeur moyenne du coût c. Les algorithmes α-euclidiens appartiennent donc à la classe des algorithmes rapides, ce qui est confirmé par le comportement de la complexité en bits dont la valeur moyenne satisfait asymptotiquement

$$\mathbb{E}_N[B] \sim \frac{\log^2 2}{h(\alpha)}\cdot \mu_\alpha(\ell)\cdot N^2 = \frac{A_\alpha}{2}\cdot \mu_\alpha(\ell)\cdot N^2,$$

$\mu_\alpha(\ell)$ étant encore une fois la valeur moyenne de la taille des quotients.

Les valeurs exactes des constantes $h(\alpha)$ et $\mu_\alpha(\ell)$ sont connues quand α appartient aux intervalles $[\sqrt{2}-1, \phi-1]$ et $[\phi-1, 1]$ et ne le sont pas sinon, puisqu'on ne connaît pas la densité invariante du système dynamique correspondant. On observe un comportement surprenant de ces constantes sur ces intervalles. L'entropie, par exemple, est constante sur le premier d'entre eux,

$$h(\alpha) = \begin{cases} \dfrac{\pi^2}{6\log\phi}, & \text{pour } \alpha \in [\sqrt{2}-1, \phi-1]\ , \\ \dfrac{\pi^2}{6\log(\alpha+1)}, & \text{pour } \alpha \in [\phi-1, 1]\ . \end{cases} \tag{1.31}$$

Cette particularité s'étend au paramètre $\mu_\alpha(\ell)$. Notons qu'une division α-euclidienne n'a pas nécessairement le même coût qu'une division euclidienne standard. En particulier, il faut tenir compte d'une éventuelle soustraction supplémentaire dans le calcul du reste. On adapte alors l'analyse en modifiant la taille du quotient, et donc le paramètre $\mu_\alpha(\ell)$. Nous donnerons quelques valeurs de ces différents paramètres dans le chapitre 5.

Algorithmes d'Euclide interrompus

Une autre approche pour avoir une meilleure compréhension des algorithmes euclidiens consiste à étudier l'évolution des principaux paramètres au cours de l'exécution de l'algorithme. Ceci se fait par l'étude des algorithmes interrompus. En collaboration avec Brigitte Vallée, nous avons analysé les algorithmes $\mathcal{E}_t$ et $\underline{\mathcal{E}}_\delta$, présentés dans le paragraphe 1.3.2 de ce mémoire, et avons étudié le comportement moyen du nombre d'itérations et de la complexité en bits. Cette étude met en évidence la décroissance linéaire des tailles des restes aux cours de l'exécution de l'algorithme d'Euclide, puisque le nombre moyen d'itérations $\mathbb{E}_N[I_t]$ de l'algorithme d'Euclide interrompu de paramètre t sur les ensembles Ω_N et $\widetilde{\Omega}_N$ est asymptotiquement

$$\widetilde{\mathbb{E}}_N[I_t] \sim \mathbb{E}_N[I_t] \sim (1-t)\,\mathbb{E}_N[I].$$

Cette relation signifie donc qu'à une fraction t du nombre d'itérations, la taille des restes aura diminué d'autant. Plus précisément, nous avons démontré la relation suivante entre les deux variables I et I_t : pour tout $\varepsilon > 0$, il existe $K < 1$ tel que quand N tend vers l'infini,

$$\mathbb{P}_N\left[\left|\frac{I_t}{I} - (1-t)\right| > \varepsilon\right] = O(K^N).$$

Ce résultat sur le nombre d'itérations s'accompagne de résultats plus précis sur la complexité en bits de l'algorithme interrompu, B_t et de sa version standard BX_t. Ces résultats explicitent précisément les comportements des principales variables – restes, quotients et continuants – au cours de l'algorithme. Les variables B_t, X_t et BX_t sont asymptotiquement reliées à B, X et BX par

$$\begin{array}{lccc} (i) & \mathbb{E}_n[B_t] & \sim & (1-t^2)\,\mathbb{E}_n[B] \\ (ii) & \mathbb{E}_n[X_t] & \sim & (1-t)^2\,\mathbb{E}_n[X] \\ (iii) & \mathbb{E}_n[BX_t] & \sim & \frac{1}{3}(1-t)(3-t)\,\mathbb{E}_n[BX]. \end{array}$$

Enfin, nous avons étudié l'évolution de la répartition des rationnels formés par deux restes consécutifs u_{i+1}/u_i dans l'intervalle $[0,1]$ au cours de l'algorithme, et donc au coût $U_{\delta,B}$ décrit plus haut dans ce chapitre (1.20). Nous avons ainsi montré que les trajectoires de ces rationnels suivent les mêmes lois que les trajectoires réelles du système dynamique correspondant à l'algorithme d'Euclide : si δ est un paramètre réel compris entre 0 et 1, si on note x_δ le rationnel $u_{\lfloor \delta p\rfloor+1}/u_{\lfloor \delta p\rfloor}$ obtenu à la fraction $\lfloor \delta p\rfloor$ du nombre p d'itérations, alors pour tout intervalle $B \subset [0,1]$,

$$\lim_{N\to\infty} \mathbb{P}_N[x_\delta \in B] = \int_B \psi(t)dt.$$

Ici, ψ n'est rien d'autre que la densité invariante du système dynamique sous-jacent à l'algorithme d'Euclide,

$$\psi(x) = \frac{1}{\log 2}\frac{1}{1+x}.$$

Notons que Brigitte Vallée a par la suite obtenu des résultats plus précis sur les algorithmes interrompus. Elle a notamment étudié la taille des continuants, pour laquelle elle a montré un théorème Central Limite, et ce pour tous les algorithmes de la classe A.

Algorithme LSB

Un autre axe de recherche a concerné l'étude de l'algorithme LSB, $\mathcal{L}$ pour lequel il n'existait jusqu'à présent aucune analyse. Nous avons en collaboration avec Véronique Maume-Deschamps et Brigitte Vallée explicité le nombre moyen d'itérations de cet algorithme. De par son utilisation "exclusive" de la valuation 2-adique, cet algorithme est un peu "à part" dans la famille des algorithmes euclidiens. Cette spécificité se retrouve dans son comportement : les constantes impliquées ne sont plus de la même nature. En effet, le nombre moyen d'itérations de cet algorithme sur les ensembles Ω_N, $\widetilde{\Omega}_N$ de ses entrées de taille N (remarquons que la taille utilisée ici est différente des précédentes, voir 1.16) est asymptotiquement

$$\mathbb{E}_N[I] \sim \frac{1}{2-\gamma_0} \cdot N := A_{\mathcal{L}} \cdot N, \tag{1.32}$$

où γ_0 est maintenant un exposant de Lyapunov (ici relatif à un produit des matrices aléatoires) qui vaut environ 0.0497 [Fla]. Ce résultat sera publié sous le titre *The Lyapunov Tortoise and the Dyadic Hare* : ce titre mérite quelques explications. Considérons une exécution de l'algorithme $\mathcal{L}$, dans laquelle les zéros à droite des entiers sont conservés, comme dans la figure 1.10. Cette exécution peut être vue comme une course entre un lièvre et une tortue : le lièvre correspond à l'avancée des zéros, et avance d'une vitesse moyenne de 2 bits par itération, alors que la tortue représente la taille totale des entiers (plus exactement la norme $||(u_i, u_{i+1})||$ de deux restes consécutifs) et a une vitesse moyenne de γ_0 bits par étape. La tortue a donc au départ de la course une avance de N bits sur le lièvre. La course termine lorsque le lièvre a rattrapé la tortue (l'entier n'est donc composé que de zéros), c'est-à-dire au bout de $(2-\gamma_0) \cdot N$ itérations.

Dans ce cadre, nous avons en réalité obtenu des résultats plus précis, puisqu'on peut étendre (1.32) à tout coût à croissance modérée c, dont l'espérance devient

$$\mathbb{E}_N[C] \sim \frac{\mu_{\mathcal{L}}(c)}{2-\gamma_0} \cdot N,$$

où $\mu_{\mathcal{L}}(c)$ est la valeur moyenne du coût c. Enfin, nous avons montré que la complexité en bits est asymptotiquement quadratique en N, puisqu'on a

$$\mathbb{E}_N[B] = \frac{1}{2}\frac{\mu_{\mathcal{L}}(k) + \mu_{\mathcal{L}}(s)}{2-\gamma_0} \cdot N^2$$

où k et s sont les coûts liés au nombre de décalages et au nombre de soustractions, et dont les valeurs moyennes sont

$$\mu_{\mathcal{L}}(k) = 2, \qquad \mu_{\mathcal{L}}(s) = \frac{5}{2}.$$

L'apparition de ces deux constantes provient de la nature de la division LSB, qui fonctionne par décalages et soustractions successifs.

Enfin le dernier résultat obtenu sur cet algorithme traite du comportement des coûts à croissance modérée, quand la taille considérée est la taille "exotique" associée au nombre de décalages effectués par l'algorithme. Dans ce cadre, nous avons prouvé un comportement gaussien des coûts totaux C et plus précisément obtenu un théorème limite central : il existe $\mu_{\mathcal{L}}(c)$ et $\delta_{\mathcal{L}}(c)$ tels que pour tout N, et tout $Y \in \mathbb{R}$:

$$\mathbb{P}_N\left[(u,v)|\frac{C(u,v) - \mu_{\mathcal{L}}(c)N}{\delta_{\mathcal{L}}(c)\sqrt{N}} \leq Y\right] = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{Y} e^{-y^2/2}dy + O\left(\frac{1}{\sqrt{N}}\right).$$

i	u_i [base 2]	u_i[base 10]	$a_i/2^{k_i}$
0	**10001100101000001**	72001	
1	**111101011000000101** 000	2011176	-3 / 8
2	**1100100110110101** 0000	826192	1 / 2
3	**11000011000101** 0000000	1598080	1 / 8
4	**100110001111** 00000000	626432	-1 / 2
5	**111010010101** 000000000	1911296	-1 / 2
6	**11000001001** 0000000000	1582080	1 / 2
7	**1000100011** 00000000000	1120256	-1 / 2
8	**1000001011** 000000000000	2142208	1 / 2
9	**11** 00000000000000	49152	1 / 4
10	**1000001** 000000000000000	2129920	-1 / 2
11	**10001** 0000000000000000	1114112	1 / 2
12	**11** 00000000000000000	1572864	-5 / 8
13	**1** 000000000000000000	2097152	3 / 4

FIG. 1.10 – Une exécution de l'algorithme LSB

De plus, il existe $\kappa < 1$, a et b tels que

$$\mathbb{E}_N[C] = \mu_{\mathcal{L}}(c)N + a + O(\kappa^N), \quad \mathrm{Var}_N[C] = \delta^2_{\mathcal{L}}(c)N + b + O(\kappa^N).$$

Algorithmes Mixtes

Enfin nous avons étudié les algorithmes Mixtes, en particulier l'algorithme Plus-Moins. L'étude de cet algorithme se révèle être pour le moment inaccessible via les techniques d'analyse dynamique, en dépit de la grande ressemblance entre l'algorithme Plus-Moins et l'algorithme Binaire. Nous décrivons dans les chapitres suivants les premières étapes de l'analyse de cet algorithme, puis expliquons au chapitre 5 pourquoi les méthodes usuelles ne s'appliquent plus dans ce cas.

1.9 Conclusion

Nous venons dans ce chapitre de faire un survol des principaux algorithmes euclidiens et de leurs comportements. Remarquons que nous n'avons pas décrit ici l'ensemble des algorithmes euclidiens. Nous avons mentionné les travaux de Sorenson, Jebelean et Collins qui généralisent l'algorithme Binaire, mais on pourrait également citer les différents algorithmes de Sedjelmaci [SL97, Sed01] par exemple. Pour des comparaisons expérimentales des ces algorithmes, on peut mentionner Cesari [Ces98], Jebelean [Jeb93a], ou encore Lercier [Ler97], qui font apparaître que les algorithmes les plus efficaces en pratique sont l'algorithme Binaire et sa généralisation par Jebelean, ainsi que l'algorithme de Lehmer-Euclide étudié au chapitre 6. Ces observations sont notamment reprises dans la bibliothèque de calcul multi-précision GNU-MP [Ga02] dans laquelle est implémentée l'amélioration de Weber et Jebelean de l'algorithme de Sorenson.

Les chapitres suivants sont consacrés aux différentes étapes de l'analyse qui permettent d'aboutir aux résultats énoncés plus haut. En particulier, nous décrivons dans le chapitre 2 les analogues continus des algorithmes étudiés, à savoir les systèmes dynamiques.

Chapitre 2

Modélisation en Systèmes Dynamiques

Sommaire

Nous abordons maintenant la première étape de l'analyse des algorithmes, celle de la modélisation en système dynamique. Comme nous l'avons précisé dans l'introduction, cette modélisation est basée sur le développement en fraction continue des rationnels naturellement associé à l'algorithme. On étend ensuite ce développement à un ensemble continu, et on obtient alors un système dynamique. La première partie de ce chapitre est consacrée aux différentes modélisations étudiées ici : on observe en effet que chaque famille d'algorithme conduit à un système de nature différente.

Les algorithmes MSB conduisent aux systèmes les plus classiques, les systèmes dynamiques de l'intervalle. L'exemple le plus connu est bien sûr le système dynamique associé à l'algorithme d'Euclide, puisqu'il correspond au développement classique en fraction continue, qui a depuis Gauss été très largement étudié. Nous présentons tout d'abord les systèmes associés aux algorithmes d'Euclide, Centré et Par-Excès, puis décrivons les systèmes α-euclidiens, qui à ce niveau aussi peuvent être vus comme une généralisation des systèmes plus classiques.

Le système LSB est différent : la principale quantité qui guide la dynamique est ici la valuation dyadique, qui n'est pas définie sur $\mathbb{R}$. Il est plus naturel et judicieux de se placer sur l'ensemble

$\mathbb{Q}_2$ des nombres 2-adiques, sur lequel cette notion est bien définie. Nous verrons cependant par la suite que cette modélisation n'est pas suffisante pour nos besoins, que le détour par un monde 2-adique induit une perte d'information qui empêche un retour vers le monde discret des algorithmes. C'est pourquoi nous introduisons un second système dynamique, dont la dynamique est guidée par une variable 2-adique et agit également sur une variable réelle.

Enfin, la modélisation idéale d'un système Mixte devrait être à la fois réelle et 2-adique : on choisit ici de probabiliser l'aspect 2-adique, ce qui conduit en réalité à des familles de systèmes dynamiques, chacun d'entre eux étant choisi selon une certaine probabilité. Nous présentons les modélisations des algorithmes Binaires et Plus-Moins, ainsi que brièvement celles des algorithmes pseudo-euclidiens.

Au cours de ces modélisations, nous décrivons certaines des principales caractéristiques d'un système dynamique, comme la densité invariante ou l'entropie. Celles-ci peuvent intervenir dans cette thèse de deux manières différentes : tout d'abord, elles apparaissent dans le comportement moyen des algorithmes, comme nous l'avons vu dans le chapitre précédent. Mais elles interviennent également quand on étudie les propriétés métriques du système. Cette dernière étude ne fait pas à proprement parler partie de l'analyse dynamique d'algorithmes, puisqu'on s'intéresse aux propriétés des trajectoires génériques du système, alors que les trajectoires liées à l'algorithme sont les trajectoires rationnelles. Cependant, les outils utilisés ici sont utiles dans la suite de l'analyse de l'algorithme, et une telle étude permet généralement de faire apparaître des similitudes – ou des différences – entre les comportements continus et discrets. Nous définissons dans le paragraphe 2.6 quel est le cadre général de l'étude des trajectoires génériques, et énonçons aux paragraphes 2.7 et 2.8 les principaux résultats que nous avons obtenus dans ce domaine.

2.1 Les Systèmes Dynamiques

2.1.1 Définition et notations

Les systèmes considérés ici sont des systèmes dynamiques inversibles par morceaux. En toute généralité, un tel système S est une paire formée d'un ensemble compact X et d'une application $T : X \to X$, pour laquelle il existe un ensemble fini ou dénombrable $\mathcal{D}$ et une partition $\{X_d\}_{d\in\mathcal{D}}$ telle que la restriction T_d de T à chaque sous ensemble X_d est une application inversible et C^2. Chaque élément X_d de la partition est appelé intervalle fondamental. Nous noterons Y_d l'image par T (et donc T_d) de X_d, et par h_d l'application inverse de T_d,

$$T_d(X_d) = Y_d, \quad h_d(Y_d) = X_d.$$

L'ensemble des branches inverses est noté $\mathcal{H}$, et l'ensemble des compositions de k branches inverses $\mathcal{H}^k$. Nous noterons $\mathcal{H}^* := \cup_k \mathcal{H}^k$ le demi-groupe engendré par $\mathcal{H}$. Finalement, un intervalle fondamental de profondeur k désigné par $X_{d_1 d_2 \cdots d_k}$ est un raffinement de la partition initiale donné par

$$X_{d_1 d_2 \cdots d_k} = h_{d_1} \circ h_{d_2} \circ \cdots \circ h_{d_k}(Y_{d_k}).$$

Notations Du fait de la bijection entre les ensembles de branches inverses $\mathcal{H}$ et de chiffres $\mathcal{D}$, nous indicerons les éléments de la partition indifféremment par $h_d \in \mathcal{H}$ ou par $d \in \mathcal{D}$:

$$X_{h_d} = X_d \qquad \text{et} \qquad Y_{h_d} = Y_d.$$

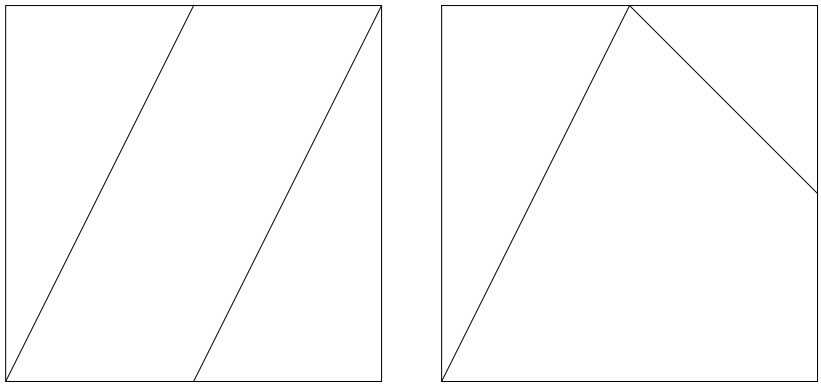

FIG. 2.1 – Les systèmes dynamiques S_1 et S_2

Exemples Nous prendrons comme exemples dans cette introduction deux systèmes dynamiques très simples, les systèmes S_1 et S_2, qui nous serviront à illustrer les différentes définitions et propriétés décrites ici. Le premier est formé de l'application définie sur $I = [0, 1]$ par

$$T(x) = 2x \bmod 1,$$

et modélise le développement en base 2. On a ici une partition de I en deux intervalles fondamentaux $I_0 = [0, 1/2]$ et $I_1 =]1/2, 1]$, et donc deux applications $T_0(x) = 2x$ et $T_1(x) = 2x - 1$. Les quatre intervalles fondamentaux de profondeur 2 sont $[0, 1/4[$, $[1/4, 1/2[$, $[1/2, 3/4[$ et $[3/4, 1]$. De plus, les intervalles images Y_1 et Y_2 sont tous deux $[0, 1]$.

Considérons maintenant le système dynamique formé toujours de l'intervalle $I = [0, 1]$ et de l'application

$$T(x) = \begin{cases} T_0(x) = 2x & \text{si } 0 \leq x < 1/2 \\ T_1(x) = \frac{3}{2} - x & \text{si } 1/2 \leq x < 1 \end{cases}$$

Les intervalles fondamentaux sont toujours $I_0 = [0, 1/2]$ et $I_1 =]1/2, 1]$. Par contre les intervalles images sont maintenant $Y_0 = [0, 1]$ et $Y_1 = [1/2, 1]$. Nous verrons par la suite que cela influe notablement sur l'analyse du système.

2.1.2 Évolution des densités, mesure invariante et entropie

Les systèmes dynamiques ont depuis longtemps été étudiés "en tant que tels". Un premier sujet d'étude naturel concerne le comportement des orbites $x, T(x), T^2(x), \ldots$. Cependant, la nature chaotique des systèmes usuellement étudiés (les orbites de deux points arbitrairement proches peuvent être très différentes) rend cette approche peu fructueuse. Il est plus judicieux de s'intéresser aux propriétés statistiques du système. Si f est une densité de probabilité sur X, on s'intéresse alors à la suite $f_0 = f, f_1, f_2, \ldots$ des densités successives du système.
Plus rigoureusement, si X est muni d'une mesure de Haar μ (ce qui est possible dès que X est un groupe topologique compact, et quand X est un intervalle la mesure μ n'est rien d'autre que la mesure de Lebesgue), on s'intéresse aux densités invariantes ψ du système, définies par

$$\int_A T \circ \psi d\mu = \int_A \psi d\mu, \quad \mu - pp, \forall A \subset X \text{ ouvert.}$$

De nombreux travaux existent pour d'une part déterminer l'existence d'une telle densité invariante, et d'autre part déterminer son unicité. Nous développerons ces points par la suite, lorsque nous présenterons les opérateurs de Perron-Frobenius, ou opérateurs transformateurs de densité, qui permettent d'expliciter la suite $f_0, f_1, f_2, \ldots$ et dont nous exploiterons les propriétés spectrales. Notons au passage que ces opérateurs ont été introduits pour l'étude du système dynamique sous-jacent à l'algorithme d'Euclide, qui correspond au développement en fraction continue classique. En effet, le problème de la densité invariante pour ce système a été soulevé dès 1800 par Gauss, qui l'a résolu sans preuve deux ans plus tard. C'est Kuzmin [Kuz28] qui 100 ans plus tard a montré que la densité décrite par Gauss était bien invariante, grâce à l'opérateur de Perron-Frobenius. Les travaux de Lévy [Lév29], puis de Wirsing [Wir74] en 1974 ont finalement répondu aux questions posées initialement par Gauss.

Mais dans un premier temps, cette densité invariante nous permet de définir une des grandeurs caractéristiques d'un système dynamique, son *entropie*. Dans les cas où cette densité invariante ψ existe, et est unique et intégrable, l'entropie topologique $h(S)$ du système est donnée par la formule de Rohlin par

$$h(S) := -\int_X \log|T'(t)|\psi(t)d\mu(t). \tag{2.1}$$

De même, si on défini un coût c sur l'ensemble $\mathcal{D}$ des chiffres du système, alors la valeur moyenne de ce coût est donnée par

$$\mu(c) := \sum_{d\in\mathcal{D}} \int_{X_d} c(d)\psi(t)d\mu(t). \tag{2.2}$$

2.2 Modélisation d'algorithmes euclidiens

Nous expliquons ici comment il est possible d'associer à un algorithme euclidien un système dynamique. Considérons un algorithme euclidien H dont la division est de la forme

$$u = vq + \varepsilon 2^k r \tag{2.3}$$

et dont une exécution est donc

$$u_0 = u, \quad u_1 = v, \quad u_0 = u_1q_1 + \varepsilon_1 2^{k_1}u_2, \quad u_1 = u_2q_2 + \varepsilon_2 2^{k_2}u_3, \ldots u_{p-1} = u_pq_p + 0, \tag{2.4}$$

Cette suite de divisions implique que le développement en fractions continues du rationnel v/u est

$$\frac{v}{u} = \cfrac{1}{q_1 + \cfrac{\varepsilon_1 2^{k_1}}{q_2 + \cfrac{\varepsilon_2 2^{k_2}}{q_3 + \cfrac{\varepsilon_3 2^{k_3}}{\ddots + \cfrac{\varepsilon_{p-1}2^{k_{p-1}}}{q_p}}}}}, \tag{2.5}$$

qui se traduit également par

$$\frac{v}{u} = h_{d_1} \circ h_{d_2} \circ \cdots \circ h_{d_p}(0) \tag{2.6}$$

si l'application h_d relative au chiffre $d = (q, \varepsilon, k)$ est définie par

$$h_d(x) = \frac{1}{q + \varepsilon 2^k x}.$$

Si X est l'ensemble dans lequel nous plongeons les rationnels v/u (en général $X = [0,1]$ ou $X = \mathcal{B}$, la boule ouverte unité de $\mathbb{Q}_2$), le système dynamique sous-jacent à l'algorithme sera la donnée de X et du prolongement à X de l'application qui au rationnel v/u associe le rationnel r/v. Plus précisément, on obtient un système dynamique dont la partition est donnée par l'ensemble $\{X_d := h_d(X), d \in \mathcal{D}\}$, $\mathcal{D}$ n'étant rien d'autre que l'ensemble des chiffres produits par l'algorithme. Les branches inverses de ce système dynamique sont les applications h_d et donc la dynamique T sera donnée par

$$T_d(x) = h_d^{-1}(x).$$

Dans le système dynamique obtenu, les trajectoires finies (qui stoppent en 0) correspondent donc aux exécutions de l'algorithme. Nous distinguerons donc par la suite les trajectoires rationnelles des trajectoires génériques.

Bien entendu, cette construction n'est valable que si le prolongement à X de l'homographie liée à la division est bien défini. Nous verrons que c'est bien le cas pour les algorithmes MSB et LSB, alors que nous serons obligés de définir ces prolongements de manière probabiliste pour les algorithmes Mixtes.

2.3 Modélisations des algorithmes MSB

Les systèmes dynamiques liés aux algorithmes MSB sont des systèmes dynamiques réels, de l'intervalle. Pour chaque algorithme, nous présentons tout d'abord le développement en fraction continue associé, puis décrivons le système dynamique correspondant : application T, partition, branches inverses. Enfin, lorsqu'elle existe et est connue nous donnons la densité invariante du système ainsi que son entropie.

2.3.1 Systèmes $S_\mathcal{E}$, $S_\mathcal{C}$ et $S_\mathcal{X}$

Tout d'abord, l'algorithme d'Euclide est relié au développement en fraction continue usuel. En effet, une division euclidienne standard $u = vq + r$ se traduit par

$$\frac{r}{v} = \frac{u}{v} - q = \frac{u}{v} - \left\lfloor \frac{u}{v} \right\rfloor = T_\mathcal{E}\left(\frac{v}{u}\right),$$

si $T_\mathcal{E}$ est l'application de Gauss,

$$T_\mathcal{E}(x) = \frac{1}{x} - \left\lfloor \frac{1}{x} \right\rfloor.$$

Une exécution de l'algorithme d'Euclide générant la suite de chiffres $(d_i)_{i=1..p} = (q_i)_{i=1..p}$ implique pour le rationnel v/u le développement en fraction continue

$$\frac{v}{u} = \cfrac{1}{q_1 + \cfrac{1}{q_2 + \cfrac{1}{q_3 + \cfrac{1}{\ddots + \cfrac{1}{q_p}}}}}. \tag{2.7}$$

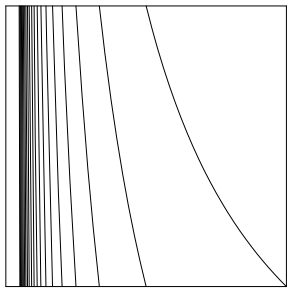

FIG. 2.2 – Le système dynamique $S_{\mathcal{E}}$

Le système dynamique $S_{\mathcal{E}}$ de l'algorithme d'Euclide est donc formé de l'intervalle $I = [0, 1]$ et de l'application $T_{\mathcal{E}}$ (nous posons $T_{\mathcal{E}}(0) = 0$). Les intervalles fondamentaux formant la partition sont les intervalles I_d donnés par

$$I_d = \left]\frac{1}{d+1}, \frac{1}{d}\right],$$

et l'ensemble des quotients d est naturellement

$$\mathcal{D}_{\mathcal{E}} = \{d \in \mathbb{N}, d \geq 1\}.$$

La branche inverse associée à chaque intervalle est une application de la forme

$$h_d(x) = \frac{1}{d+x}.$$

Le système dynamique $S_{\mathcal{E}}$ est représenté dans la figure 2.2.

Le développement en fraction continue associé à l'algorithme Centré sur une entrée valide (u, v) est de la forme

$$\frac{v}{u} = \cfrac{1}{q_1 + \cfrac{\varepsilon_1}{q_2 + \cfrac{\varepsilon_2}{q_3 + \cfrac{\varepsilon_3}{\ddots + \cfrac{\varepsilon_p}{q_p}}}}}. \tag{2.8}$$

Le système dynamique correspondant, $S_{\mathcal{C}}$ est formé de la paire $I_{\mathcal{C}} = [0, 1/2]$ et de l'application

$$T_{\mathcal{C}}(x) = \frac{1}{x} - \left\lfloor \frac{1}{x} \right\rfloor.$$

Les intervalles fondamentaux sont donnés par

$$I_d := \begin{cases} \left]\frac{1}{q+1}, \frac{2}{q+2}\right] & \text{si } d = (q, +1) \\ \left]\frac{2}{q+2}, \frac{1}{q}\right] & \text{si } d = (q, -1) \end{cases}$$

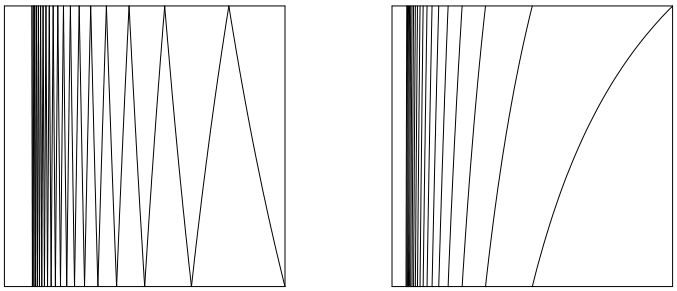

FIG. 2.3 – Les systèmes dynamiques $S_{\mathcal{C}}$ et $S_{\mathcal{X}}$

et l'ensemble des quotients possibles est $\mathcal{D}_{\mathcal{C}}$,

$$\mathcal{D}_{\mathcal{C}} = \{d = (q, \varepsilon), \quad q \geq 2, \quad \varepsilon = \pm 1\}.$$

L'algorithme Par-Excès génère lui un développement en fraction continue de la forme

$$\frac{v}{u} = \cfrac{1}{q_1 + \cfrac{-1}{q_2 + \cfrac{-1}{q_3 + \cfrac{-1}{\ddots + \cfrac{-1}{q_p}}}}}. \tag{2.9}$$

l'intervalle étant à nouveau $I = [0, 1]$ et l'application

$$T_{\mathcal{X}}(x) = \frac{1}{x} - \left\lceil \frac{1}{x} \right\rceil.$$

La partition de l'intervalle I est ici la même que celle correspondant au système dynamique $S_{\mathcal{E}}$. Les deux systèmes $S_{\mathcal{C}}$ et $S_{\mathcal{X}}$ sont représentés dans la figure 2.3.

Remarquons que les densités invariantes de ces systèmes sont connues, et sont données par

$$\psi_{\mathcal{E}}(x) = \frac{1}{\log 2}\frac{1}{1+x}, \quad \psi_{\mathcal{C}}(x) = \frac{1}{\log \phi}\left[\frac{1}{1+\phi} + \frac{1}{\phi^2 - x}\right], \text{ et } \psi_{\mathcal{X}}(x) = \frac{1}{1-x}.$$

Une fois connues les densités invariantes, on peut expliciter les entropies des systèmes $S_{\mathcal{E}}$ et S_C, qui sont

$$h(S_{\mathcal{E}}) = \frac{\pi^2}{6 \log 2}, \quad h(S_{\mathcal{C}}) = \frac{\pi^2}{6 \log \phi}.$$

L'entropie du système $S_{\mathcal{X}}$ n'est pas définie, de par la nature de la densité invariante, qui n'est pas intégrable sur l'intervalle $[0, 1]$. La densité invariante permet de plus de calculer les valeurs

moyennes de différents coûts. Par exemple, la valeur moyenne de la longueur des quotients $\ell(d)$ pour l'algorithme d'Euclide est donnée par

$$\mu_{\mathcal{E}}(\ell) = \sum_{d\in\mathcal{D}} \int_{I_d} \ell(d)\psi(t)d\mu(t) = 1 + \frac{1}{\log 2}\log\prod_{k=0}^{\infty}(1+\frac{1}{2^k}).$$

Enfin, une différence notable entre le système $S_{\mathcal{X}}$ et les systèmes $S_{\mathcal{E}}$ et $S_{\mathcal{C}}$ est qu'il existe un point fixe indifférent, c'est-à-dire un point x_0 vérifiant $|T'(x_0)| \leq 1, T(x_0) = x_0$. Pour le système $S_{\mathcal{X}}$, on a $x_0 = 1$. Cette différence a de nombreuses conséquences sur le comportement du système dynamique et explique en partie pourquoi l'algorithme appartient à la classe B des algorithmes lents. Elle amène de plus des problèmes lors de l'étude du système. Une manière de contourner ce problème est de considérer ce qu'on appelle le système dynamique induit : l'application définissant ce système est $\widetilde{T}_{\mathcal{X}}$, définie par

$$\widetilde{T}_{\mathcal{X}}(x) = \begin{cases} T(x) & \text{si } x \notin I_1 \\ T\circ T^n(x) & \text{sinon} \end{cases}$$

où n est le plus petit entier tel que $T^n(x)$ n'appartient pas à l'intervalle I_1 et donc n'emprunte pas la branche non-dilatante.

2.3.2 Systèmes $S_{\mathcal{E}_\alpha}$

La construction des systèmes dynamiques α-euclidiens $S_{\mathcal{E}_\alpha}$ se fait selon le même procédé. Nous commençons par décrire les systèmes sous-jacents aux algorithmes non-pliés $\bar{\mathcal{E}}_\alpha$, puis ceux relatifs aux algorithmes $\widehat{\mathcal{E}}_\alpha$. Remarquons que dans ce dernier cas, nous devons vérifier $S_{\widehat{\mathcal{E}}_{1/2}} = S_{\widehat{\mathcal{E}}_C}$ et $S_{\widehat{\mathcal{E}}_0} = S_{\widehat{\mathcal{E}}_P}$.

Considérons donc une exécution de l'algorithme $\bar{\mathcal{E}}_\alpha$ sur une entrée (u, v) vérifiant $v/u \in I_\alpha = [\alpha - 1, \alpha]$. Si $(d_i)_{i=1..p}$ désigne la suite de paires $d_i = (\bar{q}_i, \bar{\varepsilon}_i)$ engendrées par l'algorithme, alors on déduit pour le rationnel v/u le développement en fraction continue

$$\frac{v_1}{v_0} = \cfrac{1}{\bar{q}_1 + \cfrac{\bar{\epsilon}_1}{\bar{q}_2 + \cfrac{\bar{\epsilon}_2}{\bar{q}_3 + \cfrac{\bar{\epsilon}_3}{\ddots + \cfrac{\bar{\epsilon}_{p-1}}{\bar{q}_p}}}}}. \tag{2.10}$$

Le système $S_{\bar{\mathcal{E}}_\alpha}$ est donné par l'intervalle I_α et l'application T_α,

$$\bar{T}_\alpha(x) = \left|\frac{1}{x}\right| - \left\lfloor \frac{1}{x} + 1 - \alpha \right\rfloor \qquad \text{pour } x \neq 0, \bar{T}_\alpha(0) = 0.$$

L'ensemble des quotients générés par l'algorithme, qui permet donc de former la partition de I_α, est $\mathcal{D}_\alpha$,

$$\mathcal{D}_{\mathcal{E}_\alpha} = \left\{d = (q, -1), \quad q \geq r^-(\alpha), \right\} \cup \left\{d = (q, +1), \quad q \geq r^+(\alpha)\right\},$$

avec

$$r^+(\alpha) := \lfloor 1 + \frac{1-\alpha^2}{\alpha} \rfloor \qquad \text{et} \quad r^-(\alpha) := \lfloor 2 + \frac{\alpha^2}{1-\alpha} \rfloor,$$

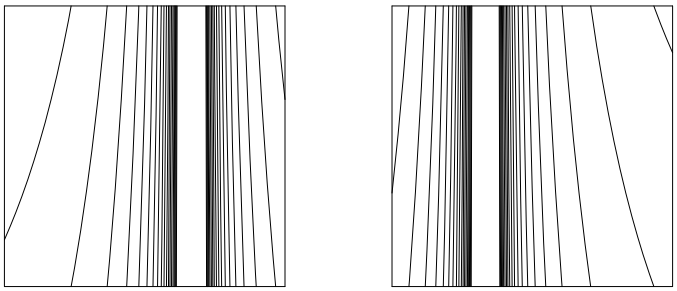

FIG. 2.4 – Les systèmes dynamiques $S_{\bar{\mathcal{E}}_{1/3}}$ et $S_{\bar{\mathcal{E}}_{2/3}}$

et les intervalles fondamentaux I_d sont donnés par

$$\overline{I}_d = \begin{cases} \left[\dfrac{1}{q+\alpha}, \dfrac{1}{q+\alpha-1}\right] & \text{pour } d=(q,+1), q \neq r^+(\alpha), \\ \left[\dfrac{1}{r^+(\alpha)+\alpha}, \alpha\right] & \text{pour } d=(r^+(\alpha),+1), \\ \left[\dfrac{-1}{q+\alpha-1}, \dfrac{-1}{q+\alpha}\right] & \text{pour } d=(q,-1), q \neq r^-(\alpha), \\ \left[1-\alpha, \dfrac{-1}{r^-(\alpha)+\alpha}\right] & \text{pour } d=(r^-(\alpha),-1). \end{cases}$$

On observe donc deux intervalles "tronqués " à chaque extrémité de l'intervalle I_α.

Remarquons dès maintenant une des caractéristiques essentielles de ces systèmes, les intervalles images $\bar{J}_d$ ne recouvrent pas tous l'intervalle I_α, comme c'était le cas pour les systèmes dynamiques rencontrés jusqu'à présent. En particulier, les deux intervalles $\overline{J}_{(r^-(\alpha),-1)}$ et $\overline{J}_{(r^+(\alpha),+1)}$ se trouvant aux extrémités sont donnés par

$$\overline{J}_{(r^-(\alpha),-1)} = [\frac{1}{\alpha-1} + r^-(\alpha), \alpha], \qquad \overline{J}_{(r^+(\alpha),+1)} = [\frac{1}{\alpha} - r^+(\alpha), \alpha]. \tag{2.11}$$

Nous verrons que cette particularité complique l'analyse du système.

Une manière plus intuitive de construire ce système dynamique est la suivante : tout d'abord tracer l'ensemble des applications $x \mapsto |1/x| - i$ pour $i \geq 1$ dans la fenêtre $[-1,1] \times [-1,1]$, et ensuite n'en garder que la fenêtre $I_\alpha \times I_\alpha$. Pour construire les systèmes $S_{\widehat{\mathcal{E}}_\alpha}$, il suffit alors de "replier" le système précédent (voir figures 2.5 et 2.6).

Enfin, les densités invariantes de ces systèmes sont connues pour certaines valeurs de α depuis les travaux de Nakada [Nak81] et Moussa, Marmi et Cassa [MCM99]. Elles sont données pour $\alpha \in [\sqrt{2}-1, \phi-1]$ et $\alpha \in [\phi-1, 1]$ par le théorème suivant :

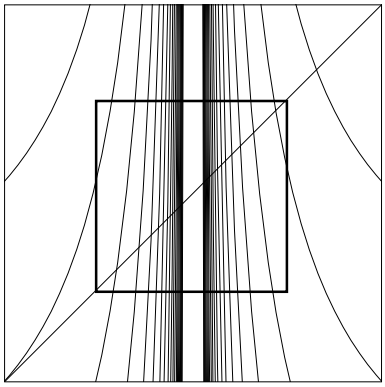

FIG. 2.5 – L'ensemble des systèmes dynamiques $S_{\bar{\mathcal{E}_\alpha}}$

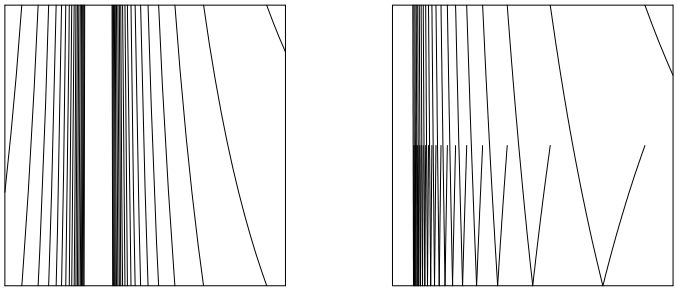

FIG. 2.6 – Les systèmes dynamiques $S_{\mathcal{E}_{1/3}}$ en version pliée et non-pliée

Théorème. [Nakada ; Moussa, Cassa, Marmi] *Pour $\alpha \geq \sqrt{2}-1$, le système dynamique $\overline{\mathcal{S}}_\alpha$ admet une unique densité invariante $\overline{\psi}^{[\alpha]}$ explicite. Cette densité est différente pour $\alpha \in [\phi-1, 1]$ et pour $\alpha \in [\sqrt{2}-1, \phi-1]$:*
pour $\phi - 1 \leq \alpha \leq 1$,

$$\overline{\psi}^{[\alpha]}(t) = \frac{1}{\log(1+\alpha)} \begin{cases} \frac{1}{2+t} & si\ t \in [\alpha-1, \frac{1-\alpha}{\alpha}[, \\ \frac{1}{1+t} & si\ t \in [\frac{1-\alpha}{\alpha}, \alpha[, \end{cases}$$

pour $1/2 \leq \alpha \leq \phi - 1$,

$$\overline{\psi}^{[\alpha]}(t) = \frac{1}{\log\phi} \begin{cases} \frac{1}{\phi^2+t} & si\ t \in [\alpha-1, \frac{1-2\alpha}{\alpha}[, \\ \frac{1}{2+t} & si\ t \in [\frac{1-2\alpha}{\alpha}, \frac{2\alpha-1}{1-\alpha}[, \\ \frac{1}{\phi+t} & si\ t \in [\frac{2\alpha-1}{1-\alpha}, \alpha[, \end{cases}$$

pour $\sqrt{2}-1 \leq \alpha \leq 1/2$,

$$\overline{\psi}^{[\alpha]}(t) = \frac{1}{\log\phi}\begin{cases} \dfrac{1}{\phi^2+t} & si\ t \in [\alpha-1, \dfrac{2\alpha-1}{1-\alpha}[, \\ \dfrac{1}{\phi^2+t}+\dfrac{1}{\phi+t}-\dfrac{1}{2+t} & si\ t \in [\dfrac{2\alpha-1}{1-\alpha}, \dfrac{1-2\alpha}{\alpha}[, \\ \dfrac{1}{\phi+t} & si\ t \in [\dfrac{1-2\alpha}{\alpha}, \alpha[. \end{cases}$$

La densité invariante $\widehat{\psi}^{[\alpha]}$ *est déduite en repliant la densité* $\overline{\psi}^{[\alpha]}$.

Quand α appartient à un des deux intervalles $[\sqrt{2}-1, \phi-1]$ ou $[\phi-1, 1]$ l'entropie est donc connue. Elle est donnée par

$$h(\alpha) = \begin{cases} \dfrac{\pi^2}{6\log\phi}, & \text{pour } \alpha \in [\sqrt{2}-1, \phi-1]\ , \\ \dfrac{\pi^2}{6\log(\alpha+1)}, & \text{pour } \alpha \in [\phi-1, 1]\ . \end{cases} \tag{2.12}$$

Finalement, la densité invariante permet de donner des expressions des valeurs moyennes de certains coûts. Nous donnons ici celle de la taille ℓ des quotients pour certaines valeurs de α. Ces valeurs, notées $\alpha := \phi_p$ correspondent à des systèmes pour lesquels l'image de la branche située à gauche recouvre tout l'intervalle I_α, et pour lesquels le point $T_\alpha(\alpha) = 1/\alpha - 1$ est l'extrémité de l'intervalle fondamental correspondant au chiffre $r = 2^p - 1$. Dans ces cas particuliers, ϕ_p est un irrationnel quadratique de la forme

$$\phi_p = \frac{1}{2}\left(-r + \sqrt{r^2+4r}\right).$$

Nous considérons ici que la taille $\ell(d)$ d'un chiffre $d(q, \varepsilon)$ est donnée par $\ell(d) = \ell_2(q) + \frac{1-\varepsilon}{2}$ (quand $\varepsilon = 1$ il n'y a pas d'opération supplémentaire pour le calcul du quotient), et les valeurs moyennes des deux termes sont données par

$$\mu_\alpha(\ell_q) = \log_{(1+\alpha)}\left[(2+\alpha)\prod_{k=2}^{p}\frac{2^k+\alpha}{2^k-1+\alpha}\prod_{k=p+1}^{\infty}\frac{2^k+2\alpha-1}{2^k+2\alpha-3}\right], \tag{2.13}$$

$$\mu_\alpha(\frac{1-\epsilon}{2}) := \Pr_\alpha[\epsilon = -1] = \begin{cases} \frac{\log 2}{\log\phi} - 1, & \text{pour } \alpha \in [\sqrt{2}-1, \phi-1], \\ \frac{\log 2}{\log(\alpha+1)} - 1, & \text{pour } \alpha \in [\phi-1, 1]. \end{cases} \tag{2.14}$$

2.4 Modélisations des algorithmes LSB

On ne peut plonger l'algorithme LSB dans un système dynamique réel. En effet, il est essentiellement basé sur la notion de valuation 2-adique, qui n'a pas de prolongement naturel dans $\mathbb{R}$. La solution est ici de plonger l'algorithme dans l'ensemble des nombres 2-adiques $\mathbb{Q}_2$, et on obtiendra le développement en fraction continue décrit par Browkin dans [Bro78, Bro01] (Ruban décrit dans [Rub70] un développement en fraction continue 2-adique très proche, qui correspond en réalité à celui obtenu si on ne prend pas la partie entière centrée).

Tout d'abord, rappelons brièvement ce qu'est cet ensemble. Étant donné un nombre premier p, la valuation p-adique de l'entier a est définie par $\nu_p(a) := \max\left\{k; p^k | a\right\}$ et sa valeur absolue p-adique par $|a|_p := p^{-\nu_p(a)}$. On étend ces notions aux rationnels en posant $\nu_p(a/b) = \nu_p(a) - \nu_p(b)$ et donc $|a/b|_p = p^{-\nu(a/b)} = |a|_p/|b|_p$. C'est une valeur absolue non-Archimédienne, l'inégalité

$$|x+y|_p \leq \max\left(|x|_p, |y|_p\right)$$

est toujours vérifiée, avec égalité si $|x|_p \neq |y|_p$. On construit alors l'ensemble $\mathbb{Q}_p$ des nombres p-adiques par complétion de $\mathbb{Q}$ en utilisant la norme $|.|_p$. L'espace $\mathbb{Q}_p$ est un espace ultramétrique localement compact, dans lequel les rationnels forment un ensemble dense.

On dispose pour étudier cet ensemble d'un outil dune grande utilité, analogue p-adique du développement en base p usuel, le développement de Hensel. Tout nombre $x \in \mathbb{Q}_p$ se décompose uniquement en

$$x = \sum_{n=\nu(x)}^{\infty} x_i p^i, \qquad \text{avec } 0 \leq x_i < p.$$

C'est en quelque sorte le développement symétrique du développement usuel des réels en base p puisqu'il est infini vers les puissances croissantes de p, et non pas vers les puissances décroissantes. Le développement de Hensel d'un entier naturel correspond à son développement en base p. Par contre, le développement d'un entier négatif est infini : par exemple -1 s'écrit

$$-1 = (p-1)(p-1)(p-1)\cdots$$

et on vérifie bien l'égalité $1 + (-1) = 0$,

$$1 + (p-1)(p-1)(p-1)(p-1)\cdots = 0000\cdots.$$

Toujours par analogie avec le développement usuel, on définit la partie entière p-adique $\lfloor x \rfloor_p$ de x en tronquant son développement,

$$\lfloor x \rfloor_p = \sum_{n=\nu(x)}^{0} x_i p^i,$$

ce qui fait qu'un nombre p-adique se décompose uniquement en

$$x = \lfloor x \rfloor_p + \{x\}_p, \quad \text{avec} \quad |\lfloor x \rfloor_p|_p \geq 1 \quad \text{et} \quad |\{x\}_p|_p < 1,$$

$\{x\}_p$ désignant la partie fractionnaire de x. La partie entière est donc un rationnel de la forme a/p^k, a étant un entier premier avec p strictement inférieur à p^{k+1}.

Nous voyons apparaître le lien avec la division LSB. Considérons en effet la division non-centrée $u = vq + r$ de u par v, $\nu(u) < \nu(v)$, dans laquelle le quotient q est

$$q = \frac{a}{2^k}, \quad k = \nu(v) - \nu(u), \quad a \text{ impair }, \quad 0 < a < 2^{k+1}.$$

Cette division se traduit sur le rationnel u/v par

$$\frac{u}{v} = q + \frac{r}{v}, \qquad \text{avec} \quad |q|_2 \geq 1 \quad \text{et} \quad \left|\frac{r}{v}\right|_2 < 1$$

et donc $q = \lfloor u/v \rfloor_2$. Par exemple, on déduit de la division de 29 par 12,

$$11101_2 = 1100_2 \times \frac{1_2 + 10_2 + 100_2}{100_2} + 1000_2, \quad i.e., 29 = \frac{7}{4} \times 12 + 8,$$

la relation

$$\left\lfloor \frac{1100_2}{11101_2} \right\rfloor_2 = \frac{111_2}{100_2} = \frac{7}{4}.$$

Cette relation se lit directement sur le développement de Hensel du rationnel 29/12, donné par

$$111.11001011000\ldots$$

Pour obtenir l'algorithme centré qui nous intéresse, nous définissons les parties entières et fractionnaires centrées en posant

$$(\lfloor x \rceil_2, \{\{x\}\}_2) := \begin{cases} (\lfloor x \rfloor_2 - 2, \{x\}_2 + 2) & \text{si } \lfloor x \rfloor_2 > 1, \\ (\lfloor x \rfloor_2 + \{x\}_2) & \text{sinon.} \end{cases}$$

Ainsi une division LSB se prolonge aux nombres 2-adiques via l'application

$$T_{\mathcal{L}}(x) := \frac{1}{x} - \left\lfloor \frac{1}{x} \right\rceil_2, \quad T_{\mathcal{L}}(0) = 0,$$

qui appliquée successivement à un nombre 2-adique x produit le développement en fraction continues 2-adique

$$x = \cfrac{1}{q_1 + \cfrac{1}{q_2 + \cfrac{1}{q_3 + \cfrac{1}{\ddots}}}} \tag{2.15}$$

tel qu'il est décrit dans [Bro78]. Si l'algorithme LSB appliqué à (u, v) génère la suite de quotients $(q_i)_{i=1..p}$ alors le développement en fraction continue 2-adique du rationnel v/u est

$$\frac{v}{u} = \cfrac{1}{q_1 + \cfrac{1}{q_2 + \cfrac{1}{q_3 + \cfrac{1}{\ddots + \cfrac{1}{q_p}}}}}. \tag{2.16}$$

Le système dynamique $S_{\mathcal{L}}$ sous-jacent à l'algorithme LSB est donc formé de l'application $T_{\mathcal{L}}$ et de la boule ouverte unité de $\mathbb{Q}_2$, que nous noterons $\mathcal{B}$:

$$\mathcal{B} := \{x \in \mathbb{Q}_2, |x|_2 < 1\} = \left\{x \in \mathbb{Q}_2, |x|_2 \leq \frac{1}{2}\right\}.$$

L'ensemble des différents chiffres apparaissant dans l'algorithme est l'ensemble $\mathcal{D}_{\mathcal{L}}$,

$$\mathcal{D}_{\mathcal{L}} = \left\{d = (a, k), \quad k \geq 1, \quad a \text{ impair }, \quad |a| < 2^k\right\},$$

et la partition de la boule $\mathcal{B}$ est formée des boules $\mathcal{B}_d$, de centre $1/q$ et de rayon $|1/q|_2^2$ pour $d=(q)$, $q=a/2^k$,

$$\mathcal{B}_d := \left\{ x \in \mathcal{B},\ \left| x - \frac{1}{q} \right|_2 < \left| \frac{1}{q} \right|_2^2 \right\} = \left\{ x \in \mathcal{B},\ \left| x - \frac{1}{q} \right|_2 \leq \frac{1}{2} \left| \frac{1}{q} \right|_2^2 \right\}.$$

La restriction T_d de T à $\mathcal{B}_d$ est

$$T_d(x) = \frac{1}{x} - q,$$

et la branche inverse correspondante est

$$h_d(x) = \frac{1}{q+x}.$$

Ce système satisfait une propriété bien particulière : la dérivée de ses branches inverses est constante. En effet, la dérivée 2-adique h'_d est donnée par

$$h'_d(x) = \frac{-1}{(q+x)^2} = \frac{-2^{2k}}{(a+2^k x)^2} \text{ si } d = q = \frac{a}{2^k}$$

et la norme 2-adique de cette dérivée est égale à

$$|h'_d(x)|_2 = 2^{-2k} \tag{2.17}$$

si x appartient à la boule unité $\mathcal{B}$. Cette propriété, qui assimile le système $S_{\mathcal{L}}$ à un système de l'intervalle à branches affines a de nombreuses conséquences comme nous le verrons par la suite. L'une d'elle est que la densité invariante n'est rien d'autre que la densité correspondant à la distribution uniforme sur $\mathcal{B}$, en d'autre termes $\psi_{\mathcal{L}} = 1$.
On déduit de cette propriété l'entropie du système dynamique $S_{\mathcal{L}}$:

$$h(S_{\mathcal{L}}) = -\int_{\mathcal{B}} \log \left| \frac{1}{t^2} \right|_2 d\mu(t) = 4 \log 2$$

(pour des exemples de calcul d'intégrales de fonctions $\mathbb{Q}_2 \to \mathbb{C}$ on peut consulter [VVZ94] par exemple). De même, on peut déduire la valeur moyenne des principaux coûts, qui d'une manière générale s'écrivent

$$\mu_{\mathcal{L}}[c] = \sum_{d \in \mathcal{D}} \int_{\mathcal{B}_d} c(d) d\mu(t) = \sum_{k \geq 1} \sum_{\substack{a \text{ impairs} \\ |a| < 2^k}} c(a/2^k) \cdot 2^{-2k}.$$

Nous étudierons par la suite un autre système dynamique associé à cet algorithme. En effet, le plongement dans $\mathbb{Q}_2$ fait disparaître au profit de la taille 2-adique toute notion de taille usuelle, sur laquelle est basée toute étude de complexité d'un algorithme. Il nous faut donc garder une trace réelle des orbites du système dynamique. La solution consiste à définir un système dynamique agissant à la fois sur $\mathbb{Q}_2$ et sur $\mathbb{R}$. En effet, étant donnée une orbite à laquelle correspond

une suite d'applications $T_{d_1}, T_{d_2}, \ldots$ il est toujours possible de considérer l'action de ces applications sur un réel x. Cependant, contrairement à ce qui se passe pour les algorithmes MSB, l'application T ne définit pas une application d'un ensemble X dans lui même, sauf à choisir $X = \mathbb{R}$ (par exemple la taille des restes générés par l'algorithme peut croître...), ce qui complique l'étude du système dynamique.

La solution est de suivre la démarche décrite dans [BL85] dans le cadre des systèmes de fonctions itérées, ou des produits de matrices aléatoires. Nous allons observer l'action de la dynamique sur la droite projective réelle : munie de la topologie projective usuelle, la droite projective est homéomorphe (par l'application tangente) au tore $J := \mathbb{R}/\pi\mathbb{Z}$, qu'on peut identifier à l'intervalle $]-\pi/2, +\pi/2[$, les deux points $-\pi/2$ et $\pi/2$ étant égaux. Étant donné un rationnel v/u, on lui associe un "angle" $w \in J$, en posant $w = \arctan v/u$. On associe de la même manière à chaque branche T_d une application $\underline{T}_d$ par conjugaison, en posant

$$\underline{T}_d(y) = \arctan\left(\frac{1}{\tan y} - d\right) \quad \text{et} \quad \underline{h}_d(y) = \arctan\left(\frac{1}{d + \tan y}\right).$$

Le système dynamique $\mathcal{S}_{\underline{\mathcal{L}}}$ est formé de la paire $(\mathcal{B} \times J, T)$ où $T_{\underline{\mathcal{L}}}$ est maintenant défini par

$$T_{\underline{\mathcal{L}}}(x, y) = (T_d(x), \underline{T}_d(y)) \quad \text{si} \quad x \in \mathcal{B}_d. \tag{2.18}$$

.

2.5 Modélisations des algorithmes Mixtes

Les algorithmes Mixtes sont basés sur deux notions à priori incompatibles : la valuation utilisée dans les algorithmes LSB n'est pas défini dans $\mathbb{R}$, et la taille guidant les divisions MSB ne l'est pas non plus dans $\mathbb{Q}_2$. Il n'existe pas d'espace dans lequel ces deux notions coexistent, et la solution à adopter est alors de considérer un des deux espaces et d'y définir l'autre notion de manière probabiliste. Nous commençons par détailler ce procédé pour les algorithmes Binaire et Plus-Moins, puis présentons une approche légèrement différentes, utilisée dans [Val03] pour traiter les algorithmes pseudo-euclidiens.

2.5.1 Systèmes dynamiques $S_{\mathcal{B}}$ et $S_{\mathcal{PM}}$

Considérons tout d'abord une itération de l'algorithme Binaire :

$$r = \frac{u - v}{2^k}, \quad k = \nu(u - v).$$

Celle-ci est suivie d'un échange si $r < v$. On en déduit sur les rationnels une application de $[0, 1]$ dans $[0, 1]$ qui à v/u associe v/r si $v < r$ et r/v sinon. Cette application est bien définie puisque dans les deux cas l'exposant k est défini à partir des numérateurs et dénominateurs du rationnel. Étant donnée une valeur de k fixée, nous étendons cette application aux réels en considérant les applications

$$T_k(x) = \begin{cases} \dfrac{2^k x}{1 - x} & \text{si } x \in \left[0, \dfrac{1}{2^k + 1}\right], \\ \dfrac{1 - x}{2^k x} & \text{si } x \in \left[\dfrac{1}{2^k + 1}, 1\right], \end{cases}$$

et en posant qu'un réel x "choisit" l'application T_k avec la probabilité $1/2^k$. Le choix de ces probabilités apparaîtra plus clairement au chapitre 4 quand nous introduirons les opérateurs de transfert. On obtient de la sorte non pas un mais une suite de systèmes dynamiques formés des paires (I, T_k). Les applications T_1 et T_2 sont représentées dans la figure 2.7. Pour chacun de ces systèmes, la partition comporte donc deux éléments, les intervalles $\left[0, \frac{1}{2^k+1}\right[$ et $\left[\frac{1}{2^k+1}, 1\right[$.

Ces systèmes présentent un inconvénient majeur : la dérivée de l'application T est parfois inférieure à 1, comme c'était le cas pour le système $S_{\mathcal{X}}$ relatif à l'algorithme Par-Excès. Ceci nous amène donc à considérer le système dynamique induit. La "mauvaise branche" est ici celle où l'algorithme ne fait pas d'échange. Nous considérons donc l'application obtenue entre deux échanges, c'est-à-dire celle définie par la suite de divisions

$$u_1 = \frac{u - v}{2^{k_1}}, \quad u_2 = \frac{u_1 - v}{2^{k_2}}, \quad u_3 = \frac{u_2 - v}{2^{k_3}}, \ldots, \quad u_r = \frac{u_{r-1} - v}{2^{k_r}},$$

qui se résume (voir (1.13)) par

$$u = va + 2^k r$$

où $a = 1 + 2^{k_1} + 2^{k_1+k_2} + \cdots + 2^{k_1+\cdots+k_{r-1}}$ et $k = k_1 + \cdots + k_r$. Nous considérons donc les

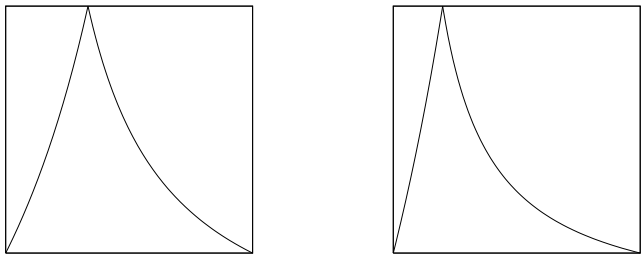

FIG. 2.7 – Les applications T_1 et T_2.

applications

$$T_{(a,k)}(x) = \frac{1 - ax}{2^k x} \tag{2.19}$$

dont les branches inverses sont

$$h_{(a,k)}(x) = \frac{1}{a + 2^k x}. \tag{2.20}$$

Les branches $T_{(a,k)}$ et $h_{(a,k)}$ sont prises avec probabilité 2^{-k}. D'une manière générale, la probabilité relative à une branche inverse h est notée p_h.

De cette manière nous venons également de définir un développement en fraction continue probabiliste pour les réels. Ce développement est de la forme

$$x = \cfrac{1}{a_1 + \cfrac{2^{k_1}}{a_2 + \cfrac{2^{k_2}}{a_3 + \cfrac{2^{k_3}}{\ddots}}}}. \tag{2.21}$$

La modélisation suit les mêmes étapes pour l'algorithme Plus-Moins. Nous obtenons non plus deux mais quatre applications définies sur les rationnels, puisqu'en plus des échanges pouvant intervenir, se rajoute le signe ε de l'opération,

$$T_{(\varepsilon,k)}(x) = \begin{cases} \dfrac{2^k x}{1-x} & \text{si } \varepsilon = -1 \text{ et } x \in \left[0, \dfrac{1}{2^k+1}\right], \\ \dfrac{1-x}{2^k x} & \text{si } \varepsilon = -1 \text{ et } x \in \left[\dfrac{1}{2^k+1}, 1\right], \\ \dfrac{2^k x}{1+x} & \text{si } \varepsilon = +1 \text{ et } x \in \left[0, \dfrac{1}{2^k-1}\right], \\ \dfrac{1+x}{2^k x} & \text{si } \varepsilon = +1 \text{ et } x \in \left[\dfrac{1}{2^k-1}, 1\right]. \end{cases}$$

En estimant qu'un réel x choisit le signe de l'opération avec probabilités $1/2, 1/2$, on obtient donc un ensemble de systèmes dynamiques formés des paires $(I, T_{(\varepsilon,k)})$, chacun d'entre eux étant choisi avec probabilité $1/2^{k+1}$ (avec $k \geq 2$ maintenant). Les partitions liées à ces systèmes sont maintenant de deux types : les intervalles fondamentaux sont $[0, 1/2^k + 1]$ et $[1/2^k + 1, 1]$ si $\varepsilon = -1$, $[0, 1/2^k - 1]$ et $[1/2^k - 1, 1]$ sinon. Remarquons que contrairement aux systèmes relatifs à l'algorithme Binaire, ceux-ci ne sont plus complets. Les branches qui correspondent à une addition suivie d'un échange ne recouvrent pas tout l'intervalle $[0, 1]$. Les applications $T_{+1,2}$ et $T_{+1,3}$ sont présentées dans la figure 2.8.

De la même manière que pour l'algorithme Binaire, nous considérons la division induite, qui se résume maintenant par

$$u = va + 2^k r$$

où $a = 1 \pm 2^{k_1} \pm 2^{k_1+k_2} \pm \cdots \pm 2^{k_1+\cdots+k_{r-1}}$ et $k = k_1 + \cdots + k_r$. Les applications sont les mêmes que pour l'algorithme Binaire (2.19, 2.20), seules les valeurs de a et k étant différentes. L'ensemble des valeurs possibles de ces deux paramètres est $\mathcal{D}_{\mathcal{PM}}$, donné par

$$\mathcal{D}_{\mathcal{PM}} = \{d = (a, k),\ k \geq 2 \text{ et } a \text{ impair }, |a| \leq m(k)\},$$

où $m(k)$ vaut $\frac{2^k-1}{3}$ si k est pair et $\frac{2^k-5}{3}$ sinon. Remarquons en effet qu'un exposant $k \geq 2$ ayant été fixé, l'ensemble des valeurs possible de $|a|$ correspond à l'ensemble des entiers positifs impairs dont l'écriture signée nécessite au plus $k-1$ bits. Rappelons que l'écriture signée d'un entier est composée de $0, 1$ et de -1, chaque 1 ou -1 étant séparé d'au moins un 0. On montre aisément par récurrence que a est inférieur ou égal à $m(k)$.

Le système obtenu ressemble à première vue énormément au système $S_{\mathcal{B}}$ de l'algorithme Binaire. Il y a cependant une différence majeure, qui jusqu'à présent rend l'analyse impossible : les branches inverses ne sont pas définie sur tout l'intervalle $[0, 1]$. Ce problème provient bien sûr du système initial dans lequel les images des branches correspondant aux échanges et aux "+" ne recouvrent pas tout [0,1].

2.5.2 Systèmes dynamiques pseudo-euclidiens

Finalement nous utilisons une approche différente pour modéliser les algorithmes pseudo-euclidiens. Soit $\widetilde{H}$ un algorithme de cette classe. L'exécution de cet algorithme est maintenant vue comme un processus markovien, en ce sens qu'on peut se trouver dans deux états différents :

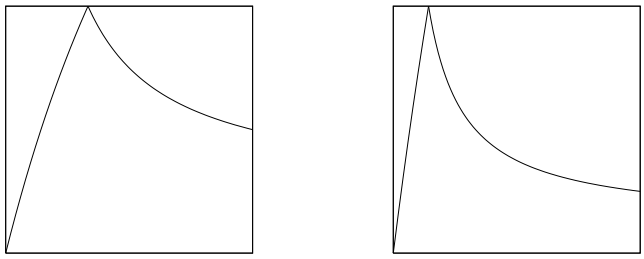

FIG. 2.8 – Les applications T_1 et T_2.

l'état 0 correspond au cas où le reste de la division effectuée est impair (le quotient est donc pair aussi puisque chaque entrée de l'algorithme est impaire) et l'état 1 dans lequel le reste est pair. On obtient donc une famille de systèmes dynamiques modélisant l'algorithme et permettant de passer d'un état à un autre. Les systèmes relatifs à l'état 0 sont déterministes, alors que ceux relatifs à l'état 1 sont probabilisés, puisque dans ce cas l'algorithme effectue un shift défini grâce à la valuation 2-adique. Ces systèmes sont notés $S_{[i,j],k}$ lorsqu'ils font passer de l'état i à l'état j et que la valuation considérée est égale à k (pour chaque valuation on a donc quatre systèmes différents). Les systèmes pseudo-euclidiens et pseudo-euclidiens centré sont présentés dans la figure 2.9 pour $k = 1$ et $k = 2$.

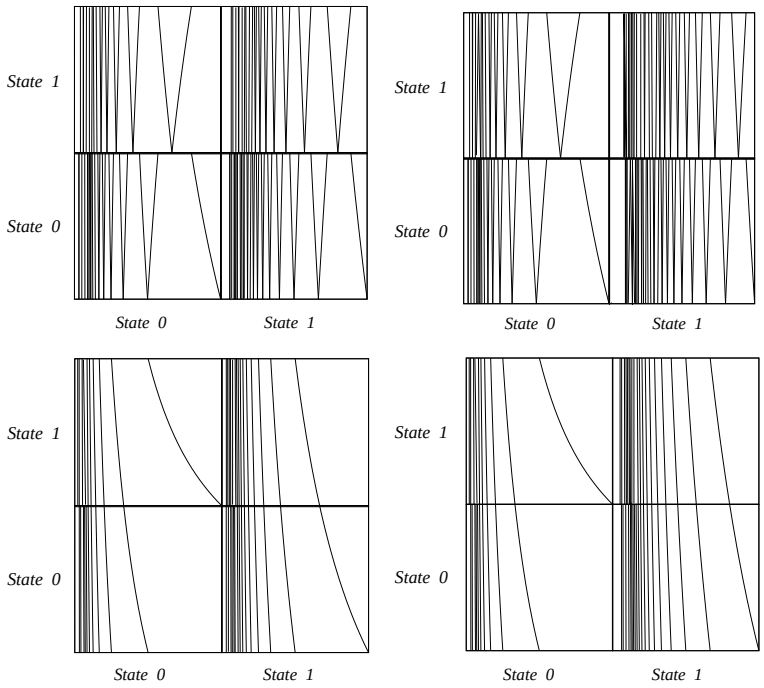

FIG. 2.9 – Les systèmes pseudo-euclidiens.

2.6 Étude probabiliste de systèmes dynamiques

Comme nous l'avons dit précédemment, étant donné un système dynamique une des questions naturelles que l'on se pose concerne l'existence et l'unicité d'une densité invariante pour cet algorithme. On arrive en raffinant cette approche à décrire de manière très précise le comportement probabiliste d'un système dynamique. Une première manière de faire est d'attacher un poids à chaque branche du système, et d'étudier la valeur moyenne du poids total sur l'ensemble des orbites de longueur n fixée.

Plus rigoureusement, étant donné un système dynamique $S = (T, X)$ dont la partition est indicée par un ensemble $\mathcal{D}$, nous considérons des fonctions de coût $c : \mathcal{D} \to \mathbb{N}$, et définissons sur un point $x \in X$ la variable aléatoire $C_n(x)$, relative aux trajectoires tronquées de longueur n :

$$C_n(x) = \sum_{i=0}^{n-1} c(d_i)$$

si x appartient à l'intervalle fondamental de profondeur n $X_{d_0 d_1 \dots d_{n-1}}$, et donc s'il emprunte successivement les branches $T_{d_0}, T_{d_1}, \dots T_{d_{n-1}}$. Nous étudions la distribution asymptotique de la variable C_n, pour un coût c à croissance modérée.

Rappelons que les coûts à croissance modérée seront rigoureusement définis au chapitre 5, mais que d'une manière générale, un coût de la forme $c = O(\ell)$, où $\ell(d)$ est la taille du chiffre d est à croissance modérée. Comme précédemment, si ce chiffre est un unique quotient $d = (q)$, alors $\ell(d) = \ell_2(q)$, sa taille binaire, si c'est une paire $d = (q, \varepsilon)$ sa taille est $\ell(d) = \ell_2(q) + 1$ et si c'est une paire de la forme $d = (a, k)$ alors $\ell(d) = \ell_2(a) + k$. Ces coûts permettent d'exprimer entre autre la longueur du codage du développement en fraction continue de longueur n de x ou encore le nombre d'occurrences d'un chiffre d donné. Remarquons enfin que les coûts c peuvent être de manière équivalente définis sur l'ensemble des branches inverses $\mathcal{H}$ du système :

$$c(h_d) = c(d).$$

Nous nous intéressons également à ce qui est l'analogue continu des coûts définis sur les restes et les continuants dans le cas discret. Considérons un point $x \in X$. On associe à son orbite de longueur n un rationnel, le n-ème convergent de x, défini par

$$Q_n(x) = \begin{pmatrix} p_n(x) \\ q_n(x) \end{pmatrix}, \qquad \frac{p_n(x)}{q_n(x)} := h_{d_1} \circ h_{d_2} \circ \dots h_{d_n}(0),$$

qui est donc le rationnel obtenu en considérant les n premiers termes du développement en fraction continues de x. Remarquons que le vecteur Q_n est également donné par

$$Q_n(x) = \mathcal{M} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \mathcal{M}_{[d_1]} \cdot \mathcal{M}_{[d_2]} \cdots \mathcal{M}_{[d_n]} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \tag{2.22}$$

où les matrices $\mathcal{M}_{[d_i]}$ sont les mêmes que celles définies au premier chapitre sur les algorithmes (dans le cas du système LSB il faut remplacer les matrices $\mathcal{M}$ par les matrices $\mathcal{N}$ pour avoir la même relation (1.12)). Nous étudions le comportement de la paire $Q_n = (p_n, q_n)$ définissant le n-ème continuant, et plus particulièrement à sa taille. Comme pour l'étude des algorithmes euclidiens nous sommes amenés à considérer plusieurs tailles selon la nature du système étudié. Elles sont toutes de la forme

$$\ell(Q_n) = \ell_2(||(p_n, q_n)||),$$

la norme ||.|| étant comme dans l'étude des algorithmes définie par

$$||(p,q)|| := \begin{cases} \sup(|p|,|q|) & \text{pour les systèmes MSB et Mixtes,} \\ (p^2+q^2)^{\frac{1}{2}} & \text{pour le système LSB.} \end{cases}$$

2.7 Résultats connus sur $S_{\mathcal{E}}$, $S_{\mathcal{C}}$ et $S_{\mathcal{X}}$

L'étude des propriétés métriques des systèmes dynamiques est bien sûr bien antérieure à l'analyse dynamique d'algorithme. On dispose de nombreux résultats dans ce domaine. Il existe en particulier des techniques génériques pour montrer le comportement gaussien des différentes variables aléatoires définies sur les systèmes lorsque ceux-ci satisfont certaines conditions. En particulier, lorsque la partition du système est finie, on peut s'inspirer des travaux fondateurs de Lasota et Yorke [LY73], Rousseau-Egele [RE83] ou encore Baladi et Keller [BK90]. Ces études sont bien résumées dans la monographie de Collet [Col96]. Le cas des partitions dénombrables est traité par Broise dans [Bro96].

Nous avons choisi de nous concentrer ici sur deux types de résultats : l'analyse des coûts à croissance modérée et celle des continuants. L'analyse dynamique permet d'énoncer les résultats dans le même formalisme que celui utilisé pour décrire le comportement des algorithme euclidiens, et met ainsi en évidence les similitudes – plus rarement les différences – entre les comportements discrets et continus.

Ainsi, pour les systèmes dynamiques sous-jacents aux algorithmes rapides (donc les systèmes $S_{\mathcal{E}}$, S_I, $S_{\mathcal{C}}$, $S_{\widetilde{\mathcal{E}}}$ et $S_{\widetilde{\mathcal{C}}}$) on montre un théorème limite central pour les coûts à croissance modérée associés aux trajectoires tronquées de longueur n. Si $S = (I,T)$ est un de ces systèmes, $\mathbb{P}$ et $\mathbb{E}$ les probabilités et espérances définies sur I avec la mesure de Lebesgue, alors il existe deux constantes $\mu_S(c)$ et $\delta_S(c)$ telles que pour tout n, et pour tout $Y \in \mathbb{R}$

$$\mathbb{P}\left[x \;\middle|\; \frac{C_n(x) - \mu_S(c)n}{\delta_S(c)\sqrt{n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-y^2/2}\, dy + O\left(\frac{1}{\sqrt{n}}\right),$$

et si S est relatif à un algorithme rapide H alors les constantes $\mu_S(c)$ et $\delta_S(c)$ sont reliées aux constantes $\mu_H(c)$ et $\delta_H(c)$ des algorithmes (voir chapitre précédent, paragraphe 1.7) par

$$\mu_S(c) = \mu_H(c), \quad \delta_S(c) = \delta_H(c).$$

Les espérances $\mathbb{E}[C_n]$ et variances $\mathrm{Var}[C_n]$ sont données par

$$\mathbb{E}\left[C_n\right] = \mu_S(c)\cdot n + a + O(\kappa^n), \quad \mathrm{Var}\left[C_n\right] = \delta_S^2(c)\cdot n + b + O(\kappa^n),$$

où κ est un réel strictement inférieur à 1.

Le résultat précédent explicite la similitude entre les trajectoires réelles et rationnelles lorsque celles-ci ne sont observées que via la suite de quotient engendrée. Cette similitude reste valable pour d'autres observables. En particulier, les comportements des continuants liés aux algorithmes et aux systèmes dynamiques sont les mêmes, puisque par analogie avec les résultats énoncés dans le chapitre 1.7, la moyenne de la taille ℓ_n du n-ème continuant $\mathbb{Q}_n$ est donnée par (voir [Phi70])

$$\mathbb{E}\left[\ell_n\right] = A_S \cdot n + a + O(\kappa^n),$$

où κ est strictement inférieur à 1. Si H désigne l'algorithme correspondant au système dynamique, alors la constante A_S est reliée à la constante A_H, définie en (1.29), par

$$A_S = \frac{1}{A_H} \tag{2.23}$$

et fait donc intervenir l'entropie du système puisque A_H était définie par

$$A_H := \frac{2\log 2}{h(H)}.$$

On peut montrer un théorème limite central pour ces coûts : pour tout n, et tout $Y \in \mathbb{R}$, on a

$$\mathbb{P}\left[x \;\middle|\; \frac{\ell_n(x) - A_S n}{\sqrt{n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-y^2/2}\, dy + O\left(\frac{1}{\sqrt{n}}\right).$$

2.8 Résultats de cette thèse pour $S_{\mathcal{E}_\alpha}$ et $S_{\mathcal{L}}$

Le premier résultat énoncé ici traite des systèmes α-euclidiens, et du comportement des coûts à croissance modérée sur ces systèmes. De même que pour les autres systèmes associés aux algorithmes rapides, on montre un théorème central limite pour $\alpha \neq 0$. Si C_n est la variable aléatoire associée à un coût c et aux trajectoires de longueur n, alors il existe une constante $\delta_\alpha(c)$ telle que pour tout n, et pour tout $Y \in \mathbb{R}$

$$\mathbb{P}\left[x \;\middle|\; \frac{C_n(x) - \mu_\alpha(c) n}{\delta_\alpha(c)\sqrt{n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-y^2/2}\, dy + O\left(\frac{1}{\sqrt{n}}\right),$$

$\mu_\alpha(c)$ étant la même constante que celle intervenant dans le cas discret (voir paragraphe 1.8). De plus, il existe $\kappa < 1$, a et b tel que l'espérance et la variance satisfont

$$\mathbb{E}\left[C^n\right] = \mu_\alpha(c) \cdot n + a + O(\kappa^n), \quad \mathrm{Var}\left[C_n\right] = \delta_\alpha^2(c) \cdot n + b + O(\kappa^n)$$

Les résultats suivants concernent le système $S_{\mathcal{L}}$, relatif à l'algorithme LSB. Nous nous sommes intéressé à deux paramètres : les coûts à croissance modérée et les tailles des continuants. Pour le premier, nous avons montré leur comportement gaussien. Si c est un coût à croissance modérée et C_n la variable aléatoire correspondante, alors il existe deux constantes $\mu_{\mathcal{L}}(c)$ et $\delta_{\mathcal{L}}(c)$ telles que pour tout n, et pour tout $Y \in \mathbb{R}$,

$$\Pr\left[x|\frac{C_n(x) - \mu_{\mathcal{L}}(c) n}{\delta_{\mathcal{L}}(c)\sqrt{n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-y^2/2} dy + O\left(\frac{1}{\sqrt{n}}\right).$$

Encore une fois, les variables $\mu_{\mathcal{L}}(c)$ et $\delta_{\mathcal{L}}$ sont les mêmes que celles apparaissant dans l'étude de l'algorithme. L'espérance et la variance de C_n sont données par

$$\mathbb{E}[C_n] = \mu_{\mathcal{L}}(c) n + a + O(\kappa^n) \text{ and } \mathrm{Var}[C_n] = \delta_{\mathcal{L}}^2(c) n + b + O(\kappa^n),$$

κ étant une variable réelle $\kappa < 1$.

Le comportement des continuants est plus intriguant. En effet, en collaboration avec Brigitte Vallée et Véronique Maume, nous avons exhibé les moyennes et variance de la variable ℓ_n, taille du n-ème continuant Q_n. En particulier, l'espérance est asymptotiquement donnée par

$$\mathbb{E}[\ell_n] \sim (2+\gamma_0)\cdot n$$

et la correspondance entre les cas discrets et continus observée précédemment (2.23,1.32) n'est plus valable puisque qu'on a

$$2+\gamma_0 \neq \frac{1}{A_{\mathcal{L}}} = 2-\gamma_0.$$

Le modèle du Lièvre et de la Tortue permet dans un certaine mesure d'expliquer ce résultat. En effet, ils ne font plus maintenant une course, mais additionnent leurs vitesses, et au bout de n étapes ont donc "parcouru" $(2+\gamma_0)$ bits.

Ce résultat interpelle : si on suppose qu'un rationnel de taille N se comporte comme un nombre 2-adique générique, alors la taille du n-ème continuant d'un rationnel est égale à $(2+\gamma_0)\cdot n$. Considérons donc un rationnel v/u de taille N. L'algorithme LSB appliqué à (u,v) s'arrête au bout de $I(N) = N/(2-\gamma_0)$ itérations, et le $I(N)$-ème continuant devrait être de taille

$$N\cdot\frac{2+\gamma_0}{2-\gamma_0},$$

alors qu'il est de taille N, puisqu'on retrouve l'entrée de l'algorithme.

Nous avons en réalité obtenu un théorème central limite pour la variable ℓ_n : il existe δ tel que pour tout n, et tout $Y\in\mathbb{R}$,

$$\mathbb{P}\left[x \;\middle|\; \frac{\ell_n(x)-(2+\gamma_0)n}{\delta(c)\sqrt{n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{Y} e^{-y^2/2}\,dy + O\left(\frac{1}{\sqrt{n}}\right).$$

2.9 Conclusion

Les principaux objets étudiés ont maintenant été présentés : les objets discrets dans le chapitre 1 et les objets continus dans celui-ci. Les méthodes que nous développons par la suite s'avèrent suffisamment puissantes pour décrire les comportements probabilistes des algorithmes comme des systèmes dynamiques. Nous verrons que la nature continue des systèmes dynamiques rend l'étude des trajectoires génériques tronquées présentée ci-dessus plus simple que celle des trajectoires rationnelles (relatives aux algorithmes).

Chapitre 3

Séries Génératrices

Sommaire

Les deux chapitres à venir sont consacrés aux deux principaux outils utilisés dans cette thèse. Chacun de ces outils est relatif à un des objets étudiés : les séries génératrices présentées maintenant sont les outils privilégiés de l'étude de structures discrètes, à fortiori des algorithmes, alors que les opérateurs de transfert du chapitre suivant sont relatifs aux systèmes dynamiques. La principale étape de l'analyse dynamique consiste à relier ces deux objets.

Une des techniques les plus standard en analyse d'algorithmes et en combinatoire est, pour étudier une famille d'objets $\mathcal{A}$ munie d'une taille $\ell(a)$, d'étudier la série génératrice associée. La séries la plus simple définie sur $\mathcal{A}$ est la série entière $A(z)$ donnée par

$$A(z) = \sum_{a \in \mathcal{A}} z^{\ell(a)},$$

qui "encapsule" l'essentiel des propriétés de la classe d'objets. Lorsqu'on s'intéresse à un paramètre particulier défini sur $\mathcal{A}$, par exemple un coût $c(a)$, on étudie alors la série bivariée

$$A(z,u) = \sum_{a \in \mathcal{A}} z^{\ell(a)} u^{c(a)}.$$

Cette approche permet de relier le comportement du coût c sur le sous-ensemble de $\mathcal{A}$ des entrées de tailles n, $\mathcal{A}_n$, au comportement des coefficients des séries $A(z)$ et $A(z,u)$. La démarche à suivre est alors d'obtenir une "bonne" expression des séries (déduite des propriétés combinatoires des objets étudiés) qui permet, grâce aux méthodes analytiques appropriées (voir par exemple [FS, Fla02, Fla92]) , d'extraire les coefficients de la série.

Nous suivons dans un premier temps cette démarche, puisque nous utilisons des séries génératrices, et comme dans l'exemple précédent, nous manipulerons usuellement deux variables

complexes, dont une marquera la taille et l'autre le paramètre à étudier. La première différence avec la série présentée ci-dessus est que les séries adaptées à l'analyse d'algorithmes euclidiens ne sont plus des séries entières, mais des séries de Dirichlet, donc de la forme

$$A(s) = \sum_{a \in \mathcal{A}} \frac{a_n}{n^s},$$

où a_n est la somme cumulée $a_n = \sum_{a \in \mathcal{A}_n} \ell(a)$.

Cette différence n'est pas anodine, puisque l'étape d'extraction de coefficients est souvent plus délicate dans ce cas. La seconde et principale différence avec l'approche classique est dans l'obtention de la "bonne" forme de la série. En effet, on ne peut pas exploiter facilement une quelconque décomposition des objets qu'on étudie, qui sont ici des nombres. La décomposition "évidente" à utiliser est le développement en fraction continue, mais celui-ci, comme nous l'avons remarqué en introduction, est basé sur un processus trop corrélé pour qu'on puisse obtenir une expression "manipulable" des séries génératrices. C'est grâce au lien que nous ferons dans le chapitre suivant entre séries génératrices et opérateurs que l'on obtiendra finalement l'expression souhaitée de la série.

Le premier paragraphe du chapitre est consacré aux séries génératrices des moments de Lévy, adaptées pour faire l'étude en distribution de problèmes de nature discrète ou continue. Puis nous décrivons les différentes séries de Dirichlet et entières utilisées ici, en précisant à chaque fois quel outil d'extraction de coefficients utiliser. Enfin nous expliquons dans le dernier paragraphe du chapitre pourquoi il est possible de restreindre nos études aux ensembles Ω formés uniquement d'entrées premières entre elles.

3.1 Séries génératrices pour l'analyse en distribution

Une des finalités de l'analyse d'algorithme est d'obtenir des résultats en distribution, et dans notre cas de prouver des comportement gaussiens. Il existe de nombreuses méthodes pour prouver un tel comportement. Nous utilisons ici une méthode basée sur les séries génératrices de Lévy [Lév29], ou séries génératrices des moments, et sur le théorème des Quasi-Puissances de H.K. Hwang [Hwa98].

3.1.1 Séries génératrices des moments et théorème des quasi-puissance

L'approche présentée dans ce paragraphe s'applique aussi bien à l'étude des algorithmes euclidiens qu'à celle des systèmes dynamiques en tant que tels. Considérons tout d'abord un algorithme H, et son ensemble d'entrées valides Ω. Si C est un coût défini sur Ω, alors la série génératrice des moments $\mathbb{E}_N[\exp(wC)]$ relative à la variable complexe w est l'espérance sur le sous-ensemble de Ω constitué des entrées de l'algorithme de taille N de la variable aléatoire $\exp(wC)$,

$$\mathbb{E}_N[\exp(wC)] = \frac{\sum_{(u,v) \in \Omega_N} \exp(wC(u,v))}{|\Omega_N|}. \tag{3.1}$$

Si on se place dans le cadre de l'étude d'un système dynamique $S = (X, T)$, et qu'on considère un coût C_n attaché aux trajectoires tronquées de longueur n, alors la série est définie par

$$\mathbb{E}[\exp(wC_n)] = \int_X \exp(w\ C_n(x)) d\mu(x), \tag{3.2}$$

où μ est la mesure de Haar sur l'ensemble X (qui est la mesure de Lebesgue pour un système dynamique de l'intervalle).

Lorsque ces séries possèdent de bonnes propriétés d'analycité au voisinage de $w = 0$, et qu'elles se comportent comme des quasi-puissances (et donc que la variable C a un comportement proche de celui d'une variable iid) alors on déduit du théorème suivant que le comportement de la variable C est asymptotiquement gaussien.

Théorème A. [Hwang] *Supposons que la série génératrice des moments* $\mathbb{E}[\exp(wR_n)]$ *d'une suite de fonctions* R_n *est analytique dans un voisinage complexe* $\mathcal{W}$ *de* $w = 0$*, et satisfait*

$$\mathbb{E}[\exp(wR_n)] = \exp[\beta_n U(w) + V(w)] \left(1 + O(\kappa_n^{-1})\right) , \tag{3.3}$$

avec β_n*,* $\kappa_n \to \infty$ *avec* n*,* $U(w)$*,* $V(w)$ *analytiques sur* $\mathcal{W}$ *et un* O *uniforme sur* $\mathcal{W}$*. Alors l'espérance et la variance satisfont*

$$\mathbb{E}[R_n] = U'(0) \cdot \beta_n + V'(0) + O(\kappa_n^{-1}) , \qquad \mathrm{Var}[R_n] = U''(0) \cdot \beta_n + V''(0) + O(\kappa_n^{-1}) .$$

De plus, si $U''(0) \neq 0$*, alors la distribution de* R_n *est asymptotiquement gaussienne, avec vitesse de convergence* $O(\kappa_n^{-1} + \beta_n^{-1/2})$*,*

$$\mathbb{P}\left[x \;\middle|\; \frac{R_n(x) - U'(0)n}{\sqrt{U''(0)n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-y^2/2}\, dy + O(\kappa_n^{-1} + \beta_n^{-1/2}) .$$

3.1.2 Séries génératrices bivariées

Les études en distribution sont donc basées sur les séries génératrices de Lévy. Lorsque le problème est de nature continue, il est possible d'étudier directement ces séries. Par contre, pour l'analyse d'algorithme, obtenir une décomposition en quasi-puissance est plus délicat, et impose un détour par d'autres séries génératrices. Ce paragraphe est donc consacré à la description des séries de Dirichlet et des séries entières bivariées. Nous n'utiliserons pas les premières directement : d'une manière générale, nous ne présentons pas ici de résultats en distribution sur le comportement des algorithmes, sauf sur l'algorithme LSB, quand la taille utilisée est la taille "exotique". Mais dans ce dernier cas, les séries qui apparaissent sont des séries entières.

Séries de Dirichlet

D'une manière générale, les séries génératrices que nous utilisons sont des séries de Dirichlet, plus à même de décrire des propriétés arithmétiques, mais plus délicates à analyser que des séries entières. Celles que nous présentons ici sont bivariées, et par analogie avec la variable z de l'introduction de ce chapitre, la variable s est attachée à une taille et la variable w au coût étudié. Elles sont définies à l'aide de l'ensemble Ω associé au problème traité, ainsi que de la taille ℓ et de la norme $||.||$ définis sur cet ensemble. Rappelons que ces deux dernières quantités sont liées par

$$\ell(u, v) = \ell_2(||(u, v)||).$$

Étant données ces différentes quantités et un coût C défini sur Ω, la série $F_C(s,w)$ est définie par

$$F_C(s,w) = \sum_{(u,v)\in\Omega} \frac{\exp(w\ C(u,v))}{||(u,v)||^s} = \sum_{n\geq 1} \frac{f_n^{(C)}}{n^s}, \tag{3.4}$$

où $f_n^{(C)}$ est la somme cumulée

$$f_n^{(C)} = \sum_{\substack{(u,v)\in\Omega \\ ||(u,v)||=n}} \exp(w\ C(u,v)).$$

On peut maintenant à l'aide de ces séries donner une première expression des séries génératrices des moments définies en (3.1) :

$$\mathbb{E}_N[\exp(w\ C)] = \frac{\sum_{n=2^{N-1}}^{2^N-1} f_n^{(C)}}{\sum_{n=2^{N-1}}^{2^N-1} |\Omega_n|}, \tag{3.5}$$

avec

$$\Omega_n = \{(u,v)\in\Omega; ||(u,v)|| = n\},$$

puisque les ensembles Ω_N vérifient

$$\Omega_N = \bigcup_{n=2^{N-1}}^{2^N-1} \{(u,v)\in\Omega; ||(u,v)|| = n\}.$$

Remarquons que les quantités $|\Omega_n|$ s'expriment également en termes de séries génératrices, comme nous le verrons dans le paragraphe suivant. On définit de la même manière sur les ensembles $\widetilde{\Omega}$, $\widetilde{\Omega}_N$ la série $\widetilde{F}_C(s,w)$, reliée à la série $\widetilde{\mathbb{E}}[\exp(wC)]$.

Séries entières

On peut également définir les séries génératrices entières, que nous n'utiliserons que dans l'étude de l'algorithme LSB, lorsque la taille et la norme considérées sont "exotiques", de la forme (voir (1.17))

$$||(u,v)|| = 2^k \quad \text{si l'algorithme fait } k \text{ shifts et } \ell(u,v) = k.$$

Ces séries sont données par

$$F_C(z,w) = \sum_{(u,v)\in\Omega} z^{\ell(u,v)} \exp(wC(u,v)) = \sum_{n\geq 1} f_n^{(C)} z^n, \tag{3.6}$$

avec

$$f_n^{(C)} = \sum_{(u,v)\in\Omega_n} \exp(wC(u,v))$$

On obtient ainsi une expression des séries génératrices des moments (3.1),

$$\mathbb{E}_N[\exp(w\ C)] = \frac{[z^N]F_C(z,w)}{|\Omega_N|}, \tag{3.7}$$

où $[z^N]F(s,w)$ désigne le N-ème coefficient de la série.

3.2 Séries génératrices pour l'analyse en moyenne

Il n'est cependant pas toujours possible d'obtenir des résultats en distribution. On opte alors pour une analyse en moyenne. Les séries génératrices utilisées ici sont légèrement plus simples à manipuler, puisqu'elles ne comportent plus qu'une seule variable, la variable s (ou z dans le cas d'une série entière) relative à la taille. Qu'elles soit entières ou de Dirichlet, ces séries sont définies à partir des séries $F_C(s,w)$ ou $F_C(z,w)$ par dérivation. Les séries de Dirichlet $T_C(s)$ sont donc données par

$$T_C(s) := \frac{d}{dw}F_C(s,w)_{w=0} = \sum_{(u,v)\in\Omega} \frac{C(u,v)}{||(u,v)||^s} = \sum_{n\geq 1} \frac{t_n^{(C)}}{n^s} \tag{3.8}$$

avec

$$t_n^{(C)} = \sum_{\substack{(u,v)\in\Omega \\ ||(u,v)||=n}} C(u,v),$$

et

$$T_1(s) = \sum_{(u,v)\in\Omega} \frac{1}{||(u,v)||^s} = \sum_{n\geq 1} \frac{t_n^{(1)}}{n^s},$$

avec

$$t_n^{(1)} = \#\left\{(u,v)\in\Omega, ||(u,v)|| = n\right\}.$$

Ainsi, l'espérance de la variable C sur l'ensemble Ω_N est exactement

$$E_N[C] = \frac{\sum_{n=2^{N-1}}^{2^N-1} t_n^{(C)}}{\sum_{n=2^{N-1}}^{2^N-1} t_n^{(1)}}. \tag{3.9}$$

Il en est de même pour les séries entières, maintenant données par

$$T_C(z) = \sum_{(u,v)\in\Omega} z^{\ell(u,v)} C(u,v) = \sum_{n\geq 1} z^n \sum_{(u,v)\in\Omega_n} C(u,v) = \sum_{n\geq 1} t_n^{(C)} z^n.$$

La série associée au coût $C(u,v) = 1$ est donc

$$T_1(z) = \sum_{(u,v)\in\Omega} z^{\ell(u,v)} = \sum_{n\geq 1} t_n^{(1)} z^n.$$

On obtient ainsi l'espérance de C,

$$\mathbb{E}_N[\exp(w\log C)] = \frac{[z^N]T_C(z)}{[z^N]T_1(z)}.$$

3.3 Extraction des coefficients

Comme nous venons de le voir, tous les résultats obtenus dans l'analyse des algorithmes le sont par extraction de coefficients de séries génératrices. Cette étape est différente selon qu'on manipule des séries entières ou de Dirichlet. Ce dernier cas est plus délicat à traiter, en particulier quand on souhaite obtenir des termes de restes suffisamment précis pour appliquer par la suite le théorème des quasi-puissances. C'est pourquoi l'analyse en distribution des algorithmes est particulièrement difficile.

Séries de Dirichlet

Pour exploiter la relation (3.9), nous utiliserons un théorème taubérien, dû à Delange [Del54]. Il relie l'asymptotique des sommes de coefficients au comportement de la série sur un demi-plan complexe de la forme $\Re(s) \geq \sigma$ sur lequel elle doit être analytique, sauf en $s = \sigma$ où elle possède une singularité polaire.

Théorème B. [Delange] *Soit $F(s)$ une série de Dirichlet à coefficients entiers telles que $F(s)$ converge pour $\Re(s) > \sigma_0 > 0$. Si*
(i) $F(s)$ est analytique pour $\Re(s) = \sigma, s \neq \sigma$, et
(ii) pour $\gamma \geq 0$, $F(s)$ s'écrit

$$F(s) = A(s)(s-\sigma)^{-\gamma-1} + C(s),$$

où $A(s)$ et $C(s)$ sont analytiques en $s = \sigma$ et $A(\sigma) \neq 0$, alors quand $N \to \infty$,

$$\sum_{n \leq N} a_n = \frac{A(\sigma)}{\sigma\Gamma(\gamma+1)}\, N^\sigma \log^\gamma N \quad [1 + \epsilon(N)\,], \quad \epsilon(N) \to 0.$$

En particulier, on a

$$\sum_{n=2^{2N-1}}^{2^{2N}} t_n = \frac{A(\sigma)}{\sigma\Gamma(\gamma+1)}\,(1-2^{-\sigma})\,(2\log 2)^\gamma \cdot 2^{2N\sigma}\, N^\gamma \cdot\, [1+\epsilon(N)\,], \quad \lim_{N\to\infty} \epsilon(N) = 0.$$

Quand on souhaite exploiter la relation (3.5) pour faire une analyse en distribution, la situation est plus délicate. Il faut ici faire appel à la formule de Perron qui impose des conditions supplémentaires sur une bande verticale qui contient la singularité σ. L'analyse est donc plus difficile, et pour plus de détails concernant cette approche, on peut consulter [BV04].

Séries entières

Le cas des séries entières est généralement plus simple, puisqu'on restreint l'analyse de la série à l'intérieur d'un rayon de convergence (donc dans un ensemble compact). Cette approche est également plus fructueuse, puisqu'elle s'applique très bien aux séries bivariées pour ensuite appliquer le théorème de Hwang (relation (3.7)), et obtenir des résultats en distribution.

Nous utiliserons le résultat classique suivant (voir [FS] par exemple), basé sur la formule du binôme de Newton.

Théorème C. *Soit $Q(z)$ une série entière qui se décompose en*

$$Q(z) = \left(\frac{1}{1-z}\right)^k P(z)$$

où $P(z)$ est analytique pour $|z| \leq 1$ et k un entier positif. Alors asymptotiquement,

$$[z^n]Q(z) = \frac{n^{k-1}}{(k-1)!} P(1)\left(1 + O(\frac{1}{n})\right).$$

3.4 Ω ou $\widetilde{\Omega}$?

Nous abordons maintenant un point volontairement laissé dans le flou jusqu'à présent, le choix de l'ensemble d'entrées à étudier : en effet, une analyse "rigoureuse" d'un algorithme euclidien se fait sur l'ensemble $\widetilde{\Omega}$ de toutes les entrées valides d'un algorithme, et non sur l'ensemble Ω des entrées premières entre elles. On se rend compte qu'en réalité les deux analyses mènent aux mêmes résultats. Considérons tout d'abord un coût additif, défini sur les quotients, qui vérifie donc $C(du, dv) = C(u, v)$ pour tout entier d. Considérons maintenant la série de Dirichlet $\widetilde{F}_C(s, w)$ associée à ce coût et à l'ensemble $\widetilde{\Omega}$, si la taille utilisée est relative à la norme sup, $||(u, v)|| = \max(u, v)$. Elle vérifie

$$\begin{aligned}\widetilde{F}_C(s,w) &= \sum_{(u,v)\in\widetilde{\Omega}} \frac{\exp(w\ C(u,v))}{u^s},\\ &= \sum_{d\geq 1}\sum_{(u,v)\in\Omega} \frac{\exp(w\ C(du,dv))}{(du)^s},\\ &= \zeta(s)F_C(s,w),\end{aligned}$$

où $\zeta(s)$ est la fonction zeta de Riemann,

$$\zeta(s) = \sum_{n\geq 1} \frac{1}{n^s}.$$

On obtient le même type de relation si la norme utilisée est la norme euclidienne,

$$\begin{aligned}\widetilde{F}_C(s,w) &= \sum_{(u,v)\in\widetilde{\Omega}} \frac{\exp(w\ C(u,v))}{(u^2+v^2)^s},\\ &= \sum_{d\geq 1}\sum_{(u,v)\in\Omega} \frac{\exp(w\ C(du,dv))}{(u^2+v^2)^s d^{2s}},\\ &= \zeta(2s)F_C(s,w).\end{aligned}$$

Dans l'étude des coûts additifs, il est donc suffisant de se placer directement sur Ω. Il faut être plus attentif avec la complexité en bits. Nous n'allons traiter que les séries univariées $T_B(s)$, pour la seule norme usuelle. Soit (u, v) un élément de Ω, et (du, dv) un élément de $\widetilde{\Omega}$. Si (u_i) est la suite de restes liée à une exécution de l'algorithme étudié sur entrée (u, v), alors les complexités en bits $B(u, v)$ et $B(du, dv)$ sont reliées par

$$B(du, dv) = \log dQ(u, v) + B(u, v),$$

où Q est le coût additif défini par

$$Q(u, v) = \sum_{i=0}^{p} \ell(d_i).$$

On en déduit la relation

$$\widetilde{T}_B(s) = -\zeta'(s)T_Q(s) + \zeta(s)T_B(s)$$

quand on utilise la norme usuelle. Là encore, on en conclut qu'il est suffisant de se placer sur l'ensemble Ω.

3.5 Conclusion

Certaines des séries présentées ici seront plus utilisées que d'autres. En particulier, celles dont nous nous servirons le plus sont les séries de Dirichlet univariées, qui permettent d'obtenir des résultats en moyenne sur le comportement des algorithmes. Nous utiliserons également les séries génératrices des moments pour étudier les paramètres définis sur les systèmes dynamiques, ce qui nous permettra d'en faire une analyse en distribution. Enfin, la seule série bivariée dont nous nous servirons est une série entière, qui permet de faire l'analyse en distribution de l'algorithme LSB, lorsque l'ensemble Ω est muni de la taille liée au nombre de décalage.

Nous expliquons dans les prochains chapitres comment le détour par les systèmes dynamiques et les opérateurs de transfert permet de satisfaire les conditions des théorèmes A, B et C présentés ici.

Chapitre 4

Opérateurs de Transfert

Sommaire

Nous abordons maintenant la description de l'objet central de cette thèse, l'opérateur de transfert. Il est d'une certaine manière l'analogue continu des séries génératrices du chapitre précédent puisqu'il encapsule l'essentiel des propriétés du système. Nous allons dans ce chapitre construire les différents opérateurs utilisés, et ceci en plusieurs étapes.

Dans un premier temps, nous allons étudier l'opérateur de Perron-Frobenius associé à chaque système. Cet opérateur, également appelé opérateur transformateur de densité, est central en théorie des systèmes dynamiques car il permet de calculer de manière explicite l'évolution des densités successives au cours du temps. Il est donc particulièrement approprié lorsqu'on cherche à connaître les densités invariantes du systèmes, par exemple, et c'est dans cette optique que Kuzmin l'a introduit en 1928 [Kuz28] pour calculer la densité invariante du système $S_{\mathcal{E}}$ relatif au développement en fraction continue classique (voir chapitre 2). Il a depuis été abondamment étudié, et on peut par exemple consulter les livres de Boyarsky et Gora [BG97], de Lasota et Mackey [LM94] ou encore de Baladi [Bal00] pour les principales propriétés de cet opérateur.

Une fois défini l'opérateur de Perron-Frobenius, nous allons le modifier légèrement, plus précisément le perturber, pour finalement obtenir les opérateurs de transfert. La perturbation apportée nous est utile pour engendrer des quantités liées aux algorithmes, mais elle a initialement été introduite par David Ruelle dans les années 70, dans un formalisme thermodynamique (l'opérateur de transfert est parfois appelé opérateur de Ruelle). Il a de nombreuses applications dans divers domaines et a été intensivement étudié par de nombreux auteurs : on peut par exemple citer Mayer [May91], pour ses études de l'opérateur de transfert relatif au système $S_{\mathcal{E}}$, ou encore Baladi [Bal00]

Finalement nous explicitons les propriétés génératrices des opérateurs de transfert dans le dernier paragraphe du chapitre. En effet, nous relions dans les propositions 4.1 à 4.9 ces opérateurs aux séries génératrices du chapitre précédent, ce qui constitue le principal transfert de propriétés entre le processus discret (l'algorithme) et le processus continu (le système dynamique).

4.1 L'opérateur transformateur de densité

Comme nous l'avons dit dans le chapitre 2, étant donné un système dynamique, on cherche souvent à étudier l'évolution des densités au cours du temps. On cherche en particulier à savoir s'il existe une densité invariante, et si c'est le cas, à la décrire. Un des principaux outils utilisés dans ce cadre est l'opérateur transformateur de densité, ou opérateur de Perron-Frobenius, que nous noterons généralement $\mathbf{H}$. Cet opérateur s'est avéré être un outil particulièrement efficace, puisque l'essentiel des résultats obtenus par les dynamiciens l'ont été via son étude.

Son rôle est prépondérant puisqu'étant donnée une densité f, son image par l'opérateur $\mathbf{H}[f]$ n'est autre que la densité obtenue après une itération du système. Ainsi, si $f_0, f_1, f_2, \ldots, f_i, \ldots$ est la suite de densités successives, alors on vérifie

$$\mathbf{H}[f_i] = f_{i+1}.$$

Si ψ est une densité invariante du système, alors elle vérifie en particulier

$$\mathbf{H}[\psi] = \psi, \tag{4.1}$$

elle est fonction propre de l'opérateur. Pour donner une expression précise de l'opérateur, nous allons d'abord nous placer dans le cadre plus simple des systèmes dynamiques de l'intervalle. Puis, nous détaillerons pour chaque type de système dynamique l'opérateur associé.

Soit $S = (X, T)$ un système dynamique de l'intervalle. Rappelons que la partition de l'intervalle X est $(X_d)_{d\in\mathcal{D}}$ où $\mathcal{D}$ est un ensemble fini ou dénombrable, et que les applications T_d, h_d et les intervalles Y_d vérifient (voir paragraphe 2.1.1)

$$T_d(X_d) = Y_d \qquad \text{et} \qquad h_d(Y_d) = T_d^{-1}(Y_d) = X_d.$$

Enfin, rappelons que l'ensemble $\mathcal{H}$ désigne l'ensemble des branches inverses h. Une manière de construire l'opérateur est la suivante. Soit f une densité définie sur X, muni de la mesure de Lebesgue μ. Alors la densité $\mathbf{H}[f]$ obtenue après une itération de l'algorithme doit vérifier pour tout intervalle b inclus dans X :

$$\begin{aligned}\int_b \mathbf{H}[f](t)d\mu(t) &= \int_{T^{-1}(b)} f(t)d\mu(t), \\ &= \sum_{h\in\mathcal{H}} \int_{h(b)} f(t)d\mu(t).\end{aligned}$$

En appliquant le changement de variable $t = h(y)\mathbf{1}_{Y_h}(y)$, on obtient finalement

$$\begin{aligned}\int_b \mathbf{H}[f](t)d\mu(t) &= \sum_{h\in\mathcal{H}} \int_b |h'(y)| \cdot f\circ h(y) \cdot \mathbf{1}_{Y_h}(y) \\ &= \int_b \sum_{h\in\mathcal{H}} |h'(y)| \cdot f\circ h(y) \cdot \mathbf{1}_{Y_h}(y).\end{aligned}$$

Cette égalité étant vraie pour tout intervalle ouvert b, on en déduit que l'opérateur est donné par

$$\mathbf{H}[f](x) = \sum_{h\in\mathcal{H}} |h'(x)| \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x). \tag{4.2}$$

Ainsi, la nouvelle densité au point x est déduite de la valeur de l'ancienne densité aux antécédents de x par T, la dérivée $|h'(x)|$ provenant du changement de variable. Les propriétés multiplicatives de la dérivation font que l'expression (4.2) de l'opérateur reste valable pour ses puissances. Par exemple l'opérateur $\mathbf{H}^2 = \mathbf{H}\circ\mathbf{H}$ vérifie

$$\begin{aligned}\mathbf{H}\circ\mathbf{H}[f](x) &= \sum_{h\in\mathcal{H}} |h'(x)| \cdot \sum_{g\in\mathcal{H}} |g'(h(x))| \cdot \ f\circ g\circ h(x) \cdot \mathbf{1}_{Y_g}(h(x)) \cdot \mathbf{1}_{Y_h}(x)\\ &= \sum_{h\in\mathcal{H}^2} |h'(x)| \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x)\end{aligned}$$

et d'une manière générale on a

$$\mathbf{H}^n[f](x) = \sum_{h\in\mathcal{H}^n} |h'(x)| \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x). \tag{4.3}$$

Exemples

Considérons les deux systèmes dynamiques de l'intervalle présentés en introduction du chapitre 2. Ils sont tous deux définis sur l'intervalle $[0,1]$, par les applications

$$T(x) = \begin{cases} T_0(x) = 2x & \text{si } 0 \le x < 1/2,\\ T_1(x) = 2x-1 & \text{si } 1/2 \le x < 1,\end{cases} \quad \text{et } T(x) = \begin{cases} T_0(x) = 2x & \text{si } 0 \le x < 1/2,\\ T_1(x) = \frac{3}{2} - x & \text{si } 1/2 \le x < 1.\end{cases}$$

Les branches inverses du premier sont de la forme

$$h_0(x) = \frac{x}{2} \qquad \text{et} \qquad h_1(x) = \frac{x+1}{2},$$

et sont définies sur tout l'intervalle $[0,1]$. On en déduit que l'opérateur transformateur de densité de ce système est

$$\mathbf{H}[f](x) = \frac{1}{2}f(\frac{x}{2}) + \frac{1}{2}f(\frac{x+1}{2}).$$

L'image par cet opérateur de la fonction 1 est

$$\mathbf{H}[1](x) = \frac{1}{2} + \frac{1}{2} = 1,$$

et donc la distribution uniforme sur $[0,1]$ est préservée par le système.

L'opérateur de Perron-Frobenius relatif au second système est défini à partir des deux branches inverses

$$h_0(x) = \frac{x}{2} \qquad \text{et} \qquad h_1(x) = \frac{3}{2} - x,$$

définies sur $[0,1]$ pour la première et $[1/2,1]$ pour la seconde. L'opérateur est donc

$$\mathbf{H}[f](x) = \frac{1}{2}f(\frac{x}{2}) + f(\frac{3}{2} - x)\mathbf{1}_{[1/2,1]}(x).$$

Observons que dans ce cas, l'image de la fonction 1 est

$$\mathbf{H}[1] = \frac{1}{2} + \mathbf{1}_{[1/2,1]}(x),$$

et ce système ne préserve plus la distribution uniforme. D'une manière plus générale, le fait qu'un des intervalles images ne recouvre pas l'intervalle tout entier induit la présence de fonctions indicatrices dans l'expression de l'opérateur. Cette présence implique notamment que l'image par **H** d'une fonction continue n'est à priori plus continue.

4.1.1 Expression des différents opérateur de Perron-Frobenius

Nous décrivons maintenant les opérateurs relatifs aux systèmes $S_{\mathcal{E}}$, S_α, $S_{\mathcal{L}}$, $S_{\underline{\mathcal{L}}}$, $S_{\mathcal{B}}$ ou $S_{\mathcal{PM}}$ que nous étudierons par la suite. Même si on construit ces opérateurs sur le même modèle que l'opérateur **H** présenté en introduction, il faudra prendre en compte les spécificités de chaque système : variables 2-adiques pour les systèmes $S_{\mathcal{L}}$ et $S_{\underline{\mathcal{L}}}$, systèmes probabilisés pour les systèmes $S_{\mathcal{B}}$ et $S_{\mathcal{PM}}$... Nous donnerons donc un nom différent à chaque type d'opérateur rencontré :

- **H** désigne un opérateur de Perron-Frobenius "générique", relatif à toute sorte de système,
- **G** désigne les opérateurs relatifs aux systèmes MSB, donc entre autres aux systèmes $S_{\mathcal{E}}$, $S_{\mathcal{C}}$ et S_α,
- **K** désigne l'opérateur relatif au système LSB $S_{\mathcal{L}}$, défini sur la boule unité $\mathcal{B}$ de $\mathbb{Q}_2$,
- **L** désigne l'opérateur relatif au système LSB $S_{\underline{\mathcal{L}}}$, défini sur $\mathcal{B}\times J$, J étant le tore $]-\pi/2, \pi/2[$,
- **M** désigne les opérateurs relatifs relatifs aux systèmes Mixtes, donc aux systèmes probabilisés $S_{\mathcal{B}}$ et $S_{\mathcal{PM}}$.

Opérateurs de Perron-Frobenius pour les systèmes MSB

Les systèmes MSB étant des systèmes dynamiques de l'intervalle, la définition donnée plus haut de l'opérateur de Perron-Frobenius s'applique directement. Les opérateurs **G** sont donc donnés par

$$\mathbf{G}[f](x) = \sum_{h\in\mathcal{H}} |h'(x)| \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x). \tag{4.4}$$

Remarquons cependant quelques différences entre les opérateurs relatifs aux systèmes $S_{\mathcal{E}}$, $S_{\mathcal{C}}$ et $S_{\mathcal{X}}$ et ceux relatifs aux systèmes α-euclidiens S_α. En particulier, les fonctions indicatrices ne sont pas nécessaires pour les premiers opérateurs. En effet, elles ne le sont que quand les images des intervalles fondamentaux ne recouvrent pas l'intervalle X tout entier. Dans ce dernier cas, les fonctions indicatrices sont toutes de la forme $Y_h(x) = 1$ pour tout $x \in X$, et peuvent être omises.
La présence ou non de fonction indicatrice n'est pas anodine : elle peut en effet apporter des discontinuités à chaque étape. Ainsi, on peut d'ores et déjà affirmer que la densité invariante d'un système non-complet n'est pas une fonction continue, puisque les discontinuités apportées par l'opérateur impliquent que si f est continue, $\mathbf{G}[f]$ ne l'est pas (comme dans l'exemple précédent).

Enfin le cas de l'opérateur relatif au système $S_{\mathcal{X}}$ (algorithme Par-Excès) est légèrement différent puisqu'on étudie un système induit. Nous reviendrons brièvement sur ce cas à la fin du chapitre 5.

Opérateurs de Perron-Frobenius pour les systèmes LSB

Considérons tout d'abord le système $S_{\mathcal{L}}$, défini sur la boule unité $\mathcal{B}$ de $\mathbb{Q}_2$. On obtient l'opérateur de Perron-Frobenius en suivant les mêmes étapes que précédemment. Seule la formule de changement de variable diffère. En effet, si K_1 et K_2 sont deux ouverts de $\mathcal{B}$, et σ une application telle que $K_1 = \sigma(K_2)$, alors la formule de changement de variables devient (voir [VVZ94] par exemple)

$$\int_{K_1} f(x)d\mu(x) = \int_{K_2} |\sigma'(y)|_2 \cdot f(\sigma(y))d\mu(y),$$

où μ est la mesure de Haar définie sur la boule unité $\mathcal{B}$. La principale différence avec l'expression (4.2) est l'utilisation de la valeur absolue 2-adique à la place de la valeur absolue usuelle. De plus, le système dynamique étant complet (pour tout $d \in \mathcal{D}$, $T_d(\mathcal{B}_d) = \mathcal{B}$), on peut omettre les fonctions indicatrices et on obtient finalement l'opérateur $\mathbf{K}$, défini sur les fonctions $f : \mathcal{B} \to \mathbb{R}$ par

$$\mathbf{K}^n[f](x) = \sum_{h \in \mathcal{H}^n} |h'(x)|_2 \cdot f \circ h(x).$$

Remarquons que la propriété (4.1) relative à une densité invariante donne ici immédiatement un résultat. En effet, chaque branche inverse h_d est de la forme

$$h_d(x) = \frac{1}{q+x} = \frac{2^k}{a+2^k x}$$

où d est le chiffre associé au quotient $q = a/2^k$. Comme nous l'avons remarqué dans le chapitre 2 (relation (2.17)), la valeur absolue 2-adique de cette branche est constante sur $\mathcal{B}$,

$$|h'_d(x)|_2 = 2^{-2k},$$

et l'image par $\mathbf{K}$ de la fonction 1 est donc

$$\mathbf{K}[1](x) = \sum_{h \in \mathcal{H}} |h'(x)|_2 = \sum_{k \geq 1} \sum_{\substack{a \text{ impair} \\ |a|<2^k}} 2^{-2k} = 1.$$

Ainsi, la densité uniforme est bien préservée par le système. Cette propriété est vérifiée d'une manière générale par tous les systèmes complets dont la valeur absolue de la dérivée (usuelle ou 2-adique) est constante. Ils s'apparentent ainsi à des processus sans mémoire. On peut en effet considérer un système dynamique comme étant une source qui émet des symboles aux cours du temps. On associe à une orbite $x, T(x), T^2(x)...$ les symboles $d_0, d_1, d_2...$ correspondant aux intervalles fondamentaux successivement "traversés" par l'orbite. On observe aisément qu'un système dynamique dont les branches sont affines et complètes préserve la distribution uniforme et correspond alors à une source sans mémoire.

L'opérateur relatif au système $S_{\underline{\mathcal{L}}}$ (défini sur $\mathcal{B} \times J$) se construit toujours sur le même modèle. Cependant, l'application T définissant ce système agissant maintenant sur deux variables (une variable 2-adique $x \in \mathcal{B}$ et une variable $y \in J =]-\pi/2, \pi/2[$) il faut maintenant remplacer la dérivée par le Jacobien de l'application, qui est donné par $|h'(x)|_2 \cdot |\underline{h}'(y)|$. L'opérateur $\mathbf{L}$ de Perron-Frobenius relatif au système $S_{\underline{\mathcal{L}}}$ est donc, pour une fonction $f : \mathcal{B} \times J \to \mathbb{R}$,

$$\mathbf{L}[f](x,y) = \sum_{h \in \mathcal{H}} |h'(x)|_2 \cdot |\underline{h}'(y)| \cdot f(h(x), \underline{h}(y)).$$

Si on note δ_h la quantité $|h'(x)|_2 = 1/\det h$, on utilise plutôt l'expression

$$\mathbf{L}[f](x,y) = \sum_{h\in\mathcal{H}} \delta_h \cdot |\underline{h}'(y)| \cdot f(h(x), \underline{h}(y)). \tag{4.5}$$

Nous n'étudierons par la suite que la partie réelle du système, et n'observerons l'évolution de la densité que sur l'ensemble J. Nous définissons donc un dernier opérateur, agissant sur des fonctions $f : J \to \mathbb{R}$ à une variable réelle par

$$\mathbf{L}[f](y) = \sum_{h\in\mathcal{H}} \delta_h \cdot |\underline{h}'(y)| \cdot f \circ \underline{h}(y). \tag{4.6}$$

Remarquons que cet opérateur ne correspond plus exactement à un opérateur transformateur de densité pour un système dynamique, puisqu'en se restreignant à la seule variable réelle, on s'éloigne du formalisme usuel (partition, branches inverses etc..) Par contre, on se retrouve dans le cadre d'un système de fonctions itérées : si on considère l'ensemble de fonctions

$$\left\{ \underline{h}_q : J \to J, \quad \underline{h}_q(x) = \arctan\left(\frac{1}{q+\tan x}\right), q = \frac{a}{2^k}, k \geq 1, a \text{ impair }, |a| < 2^k \right\}$$

où chaque élément $\underline{h}$ est choisi avec probabilité δ_h, alors l'opérateur $\mathbf{L}$ est l'opérateur relatif à ce système (voir par exemple le livre de Bougerol et Lacroix [BL85]).

Opérateurs de Perron-Frobenius pour les systèmes Mixtes

Enfin, on traite le cas des systèmes Mixtes (et donc probabilistes) en "probabilisant" l'opérateur. Rappelons que les systèmes $S_{\mathcal{B}}$ et $S_{\mathcal{PM}}$ sont en réalité des familles de systèmes (voir paragraphe 2.5.1), la probabilité de choisir chacun de ces systèmes dépendant du décalage effectué (et du signe $\pm$ de l'opération choisie pour le système $S_{\mathcal{PM}}$). Ces probabilités s'étendent aux systèmes induits, lesquels sont formés d'un ensemble d'applications T_d et h_d (branches inverses) indicées par les chiffres $d = (a,k)$ (Binaire) ou $d = (a,k,\varepsilon)$ (Plus-Moins), la probabilité de choisir une de ces applications étant finalement $p_{h_d} := p_d = 2^{-k} = \frac{1}{\det h_d}$. On se retrouve dans une situation similaire à celle décrite dans le paragraphe précédent, et l'opérateur $\mathbf{M}$ est donné par

$$\mathbf{M}[f](x) = \sum_{h\in\mathcal{H}} p_h \cdot |h'(x)| \cdot f \circ h(x) \cdot \mathbf{1}_{Y_h}(x). \tag{4.7}$$

La principale différence entre les opérateurs des systèmes Binaires et Plus-Moins réside dans la présence de la fonction indicatrice. En effet, les branches inverses relatives au système Binaire sont définies sur tout l'intervalle $[0,1]$, ce qui n'est pas le cas pour celles relatives au système Plus-Moins. On trouvera donc des fonctions indicatrices dans l'opérateur associé au système Plus-Moins, ce qui rendra son étude plus difficile puisque les discontinuités crées par l'opérateur empêchent de se placer sur des ensembles de fonctions continues.

4.2 Construction des opérateurs de transfert

Comme nous l'avons dit, le rôle de l'opérateur de transfert est de relier les propriétés des séries génératrices à celles des systèmes dynamiques. C'est l'utilisation de cet opérateur qui permet le transfert du continu au discret, la dernière étape de l'analyse dynamique d'algorithmes. Cependant, cet opérateur n'est pas une "invention" de l'analyse dynamique. La perturbation

G	$\mathbf{G}[f](x) = \sum_{h\in\mathcal{H}} \lvert h'(x)\rvert \cdot f\circ h(x)\cdot \mathbf{1}_{Y_h}(x)$	Systèmes MSB
K	$\mathbf{K}[f](x) = \sum_{h\in\mathcal{H}} \lvert h'(x)\rvert_2 \cdot f\circ h(x)$	Système $S_{\mathcal{L}}$
L	$\mathbf{L}[f](x,y) = \sum_{h\in\mathcal{H}} \delta_h \cdot \lvert \underline{h}'(y)\rvert \cdot f(h(x),\underline{h}(y))$	Système $S_{\underline{\mathcal{L}}}$
L	$\mathbf{L}[f](y) = \sum_{h\in\mathcal{H}} \delta_h \cdot \lvert \underline{h}'(y)\rvert \cdot f\circ \underline{h}(y)$	Système $S_{\underline{\mathcal{L}}}$
M	$\mathbf{M}[f](x) = \sum_{h\in\mathcal{H}} p_h \lvert h'(x)\rvert \cdot f\circ h(x)\cdot \mathbf{1}_{Y_h}(x).$	Systèmes $S_{\mathcal{B}}$ et $S_{\mathcal{PM}}$

FIG. 4.1 – Les différents opérateurs de Perron-Frobenius

permettant de passer de l'opérateur de Perron-Frobenius à l'opérateur de transfert est due à David Ruelle (voir par exemple [Rue78]), qui l'a utilisée dans les années 70 dans le cadre d'études thermodynamiques. Son utilisation en analyse d'algorithmes est, du coup, plus récente. L'opérateur apparaît déjà dans les travaux de Brent [Bre76] sur l'algorithme Binaire, ainsi que dans ceux d'Hensley dans son étude en distribution de l'algorithme d'Euclide [Hen94]. Son utilisation systématique en analyse d'algorithmes est due à Brigitte Vallée.

Ce paragraphe est consacré à la description de cet opérateur. Plus précisément, nous expliquons comment perturber l'opérateur de Perron-Frobenius pour générer les différentes quantités qui nous intéressent.

En effet, on perturbe l'opérateur transformateur de densité différemment selon la quantité qu'on souhaite mettre en valeur. D'une manière générale, les variables apparaissant dans ces perturbations jouent le même rôle que celles apparaissant dans les séries génératrices : la variable s sert à marquer des tailles et la variable w sert à marquer un coût. Prenons l'exemple de l'opérateur de Perron-Frobenius décrit dans (4.2), relatif à un système dynamique de l'intervalle. Alors l'opérateur de transfert correspondant est défini par

$$\mathbf{H}_{s,w}[f](x) := \sum_{h\in\mathcal{H}} \lvert h'(x)\rvert^s \cdot \exp(wc(h)) \cdot f\circ h(x)\cdot \mathbf{1}_{Y_h}(x). \tag{4.8}$$

$$\mathbf{H}^n_{s,w}[f](x) := \sum_{h\in\mathcal{H}^n} \lvert h'(x)\rvert^s \cdot \exp(wc(h)) \cdot f\circ h(x)\cdot \mathbf{1}_{Y_h}(x).$$

Nous expliquons dans les paragraphes suivants en quoi la dérivée $|h'|$ est généralement associée à une taille, et ainsi en quoi un tel opérateur permet de générer les principales quantités étudiées. Nous donnons dans le paragraphe 4.2.3 les expressions finales de tous les opérateurs utilisés.

4.2.1 Génération des tailles

L'opérateur présenté en (4.8) est en réalité le bon opérateur de transfert pour les systèmes (et algorithmes) MSB. Nous aurons besoin de le modifier légèrement pour obtenir les opérateurs relatifs aux autres algorithmes et systèmes dynamiques.

Systèmes MSB

Considérons donc tout d'abord les systèmes dynamiques MSB. Pour tous ces systèmes, les branches inverses h sont des homographies du type

$$h(x) = \frac{ax+b}{cx+d},$$

dont les dérivées sont donc de la forme

$$h'(x) = \frac{\det h}{(D[h](x))^2},$$

où $D[h]$ désigne la fonction dénominateur de h. On remarque de plus que ces homographies sont toutes de déterminant ± 1, et qu'ainsi

$$|h'(x)| = \frac{1}{(D[h](x))^2}. \tag{4.9}$$

Soit maintenant (u, v) une entrée valide de l'algorithme correspondant au système étudié. Cette entrée correspond à une trajectoire rationnelle du système, et plus précisément il existe une homographie de profondeur p, $h \in \mathcal{H}^p$ telle que

$$\frac{v}{u} = h(0) \text{ et } D[h](0) = u \tag{4.10}$$

si l'algorithme fait p itérations sur entrée (u, v) (voir (2.6)) et que cette dernière appartient à Ω (et donc qu'on vérifie pgcd $(u, v) = 1$). On déduit de cette relation et de (4.9) l'égalité

$$|h'(0)| = \frac{1}{u^2} = \frac{1}{||(u,v)||^2} \tag{4.11}$$

et ainsi la dérivée d'une homographie $h \in \mathcal{H}^p$ permet d'exprimer la norme de l'entrée de l'algorithme correspondant. Il est donc naturel de mettre la variable s, marquant la taille des entrées d'un algorithme en exposant de cette dérivée.

Systèmes Mixtes

La génération des normes est presque identique lorsque le système est Mixte. La principale différence vient du fait que les déterminants des branches inverses ne sont plus ± 1. Prenons l'exemple des systèmes dynamiques Binaire et Plus-Moins. Les branches inverses relatives à un chiffre $d = (a, k)$ ou $d = (a, k, \varepsilon)$ sont de la forme

$$h_d(x) = \frac{1}{a + 2^k x} \quad \text{ou} \quad h_d(x) = \frac{1}{\varepsilon a + 2^k x}$$

et donc de déterminant 2^k. Cependant, chacune de ces branches est empruntée avec probabilité $p_{h_d} = 2^{-k}$, et donc en vertu de la relation (2.6) entre une entrée valide (u, v) de l'algorithme et une branche inverse $h \in \mathcal{H}^p$ de profondeur p, on vérifie

$$p_h |h'(0)| = \frac{1}{u^2} = \frac{1}{||(u,v)||^2}. \tag{4.12}$$

La variable s qui marque les normes devra donc être placée en exposant de la quantité $p_h|h'(x)|$ dans l'opérateur de Perron-Frobenius défini en (4.7). C'est pour obtenir une telle propriété qu'on a défini de la sorte le système de probabilités au paragraphe 2.5.

Systèmes LSB

La situation est ici très différente. Tout d'abord, la dérivée $\delta_h = |h'|_2$ ne permet plus de générer une norme "raisonnable". Nous avons vu en effet au paragraphe 2.4 que cette quantité est constante sur la boule unité $\mathcal{B}$ de $\mathbb{Q}_2$, et égale à 2^{-2k} si h est la branche relative au chiffre $d = (a, k)$. Elle ne génère plus maintenant que la norme "exotique" définie par $||(u, v)|| = 2^k$ (voir paragraphe 1.6.1). Nous verrons par la suite que dans ce cas très particulier, la série génératrice associée est une série entière.

Toujours est-il que le système dynamique 2-adique $S_{\mathcal{L}}$, s'il modélise bien la dynamique de l'algorithme, ne permet pas d'avoir accès aux notions de tailles usuelles ; le détour par un monde 2-adique implique une perte d'information. C'est pourquoi il est plus judicieux pour étudier l'algorithme LSB de considérer le système $S_{\underline{\mathcal{L}}}$, dans lequel la dynamique est guidée par la variable 2-adique, et ainsi correspond bien à celle de l'algorithme, la variable réelle étant ici utile pour avoir accès aux tailles (ou normes) dont nous avons besoin.

En effet, soit (b, c) un vecteur de $\mathbb{Z}^2$, w le rationnel $w = b/c$ (w correspond ici à la cotangente de l'angle formé par la droite projective passant par (b, c) et l'axe des abscisses) et $y = \arctan w \in J$ l'élément correspondant de J. Soit $d = (a, k) \in \mathcal{D}_{\mathcal{L}}$ un quotient (donc de la forme $d = q = a/2^k$), $h_d \in \mathcal{H}$ et $\underline{h}_d$ les homographies correspondantes, et $\mathcal{M}_{[d]}$, $\mathcal{N}_{[d]}$ les matrices associées (voir (1.9,1.11)),

$$\mathcal{M}_{[d]} = \frac{1}{2^k} \begin{pmatrix} 0 & 2^k \\ 2^k & a \end{pmatrix}, \quad \mathcal{N}_{[d]} := \begin{pmatrix} 0 & 2^k \\ 2^k & a \end{pmatrix}.$$

Si la norme euclidienne sur $\mathbb{Z}^2$ est donnée par $||(b, c)||^2 = b^2 + c^2$, alors on vérifie

$$\frac{||(b, c)||^2}{||\mathcal{N}_{[d]}(b, c)||^2} = \frac{1}{2^{2k}} \frac{1 + w^2}{1 + (w + q)^2} = \delta_h \cdot |\underline{h}'_d(y)|. \tag{4.13}$$

En particulier, soit (u, v) une entrée de l'algorithme, pour laquelle celui-ci fait p itérations. Si $d_1, ...d_p$ est la suite de quotients engendrée par l'algorithme et si $\mathcal{N} = \mathcal{N}_{[d_1]} \cdots \mathcal{N}_{[d_p]}$ est la matrice associée, alors (voir (1.12))

$$\begin{pmatrix} v \\ u \end{pmatrix} = \mathcal{N} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

et donc on déduit de (4.13) la relation

$$||(u, v)||^2 = ||\mathcal{N}(1, 0)||^2 = \frac{1}{\delta_h \cdot |\underline{h}'(0)|}, \tag{4.14}$$

qui permet de générer les tailles des entrées de l'algorithme. Le paramètre s marquant les tailles devra donc être mis en exposant de l'expression $\delta_h \cdot |\underline{h}'|$.

4.2.2 Génération des coûts définis sur les quotients

La génération des coûts à l'aide des opérateurs est maintenant plus simple. La perturbation associée est la même que celle qui a mené de l'opérateur de Perron-Frobenius défini en (4.2) à l'opérateur défini en (4.8).

Plus précisément, étant donné un algorithme, le système dynamique associé, et un coût défini sur les chiffres, on peut de manière équivalente définir ce coût sur l'ensemble $\mathcal{H}$ des branches inverses du système. On étend par additivité ce coût à des trajectoires de longueur quelconque (et

donc à des trajectoires rationnelles correspondant aux algorithmes euclidiens ou à des trajectoires génériques tronquées). L'opérateur de transfert est obtenu en ajoutant l'expression $\exp(wc(h))$.

4.2.3 Forme finale

Nous donnons maintenant les formes finales des opérateurs de transfert que nous utiliserons. Ils sont tous construit sur le même modèle que l'opérateur $\mathbf{H}_{s,w}$ présenté en introduction : on définira donc les opérateurs $\mathbf{G}_{s,w}$, $\mathbf{K}_{s,w}$, $\mathbf{L}_{s,w}$ et $\mathbf{M}_{s,w}$. Nous serons de plus amenés à définir un opérateur à trois variables pour l'étude de l'algorithme LSB, en l'occurrence l'opérateur $\mathbf{L}_{s,t,w}$. D'une manière générale, les opérateurs du type $\mathbf{H}_{s,w}$ sont utiles pour l'analyse en distribution des algorithmes, les opérateurs $\mathbf{H}_{s,0}$ pour les analyses en moyenne des algorithmes et les opérateurs $\mathbf{H}_{1,w}$ pour les analyses en distribution des trajectoires tronquées des systèmes dynamiques. La variables s n'intervient donc que quand le problème traité est de nature discrète.

Opérateurs de transfert pour les systèmes MSB

Comme nous l'avons écrit au début de cette section, les opérateurs de transfert pour les systèmes MSB sont donnés par (4.8).

$$\mathbf{G}_{s,w}[f](x) := \sum_{h\in\mathcal{H}} |h'(x)|^s \cdot \exp(wc(h)) \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x). \tag{4.15}$$

Quand on s'intéresse aux systèmes $S_{\mathcal{E}}$ et $S_{\mathcal{C}}$, on peut omettre les fonctions indicatrices, ce qui simplifie l'étude des opérateurs.

Opérateurs de transfert pour les systèmes LSB

Considérons tout d'abord l'opérateur $\mathbf{L}_{s,w}$. En suivant les instructions données au chapitre précédent, on obtient

$$\mathbf{L}_{s,w}[f](x) = \sum_{h\in\mathcal{H}} \delta_h^s \cdot |\underline{h}'(x)|^s \cdot \exp(wc(h)) \cdot f(\underline{h}(x)). \tag{4.16}$$

On définit un opérateur à trois variables quand on veut dissocier l'action 2-adique associée à δ_h de l'action réelle associée à $|h'(x)|$. Pour cela, on ne place pas une seule variable en exposant du Jacobien, mais deux en exposant de chacune des composantes de ce Jacobien, et on obtient l'opérateur $\mathbf{L}_{s,t,w}$ à trois paramètres complexes :

$$\mathbf{L}_{s,t,w}[f](x) = \sum_{h\in\mathcal{H}} \delta_h^t \cdot |\underline{h}'(x)|^s \cdot \exp(wc(h)) \cdot f(\underline{h}(x)). \tag{4.17}$$

On vérifie donc $\mathbf{L}_{s,w} = \mathbf{L}_{s,s,w}$.

Enfin, l'opérateur $\mathbf{K}_{s,w}$ est relatif au système purement 2-adique $S_{\mathcal{L}}$ et est utile pour l'analyse en distribution de l'algorithme $\mathcal{L}$ lorsque la taille des entiers est la taille "exotique", ainsi que pour l'étude des trajectoire génériques tronquées. Il est défini par

$$\mathbf{K}_{s,w}[f](x) = \sum_{h\in\mathcal{H}} |h'(x)|_2^s \cdot \exp(wc(h)) \cdot f\circ h(x). \tag{4.18}$$

Opérateurs de transfert pour les systèmes Mixtes

Finalement, les opérateurs de transfert pour les systèmes mixtes sont déduits de leurs opérateurs de Perron-Frobenius en y ajoutant la variable s en exposant de $p(h)|h'(x)|$ et la quantité $\exp(wc(h))$:

$$\begin{aligned}\mathbf{M}_{s,w}[f](x) &:= \sum_{h\in\mathcal{H}} p_h^s \cdot |h'(x)|^s \cdot \exp(wc(h)) \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x) \\ &= \sum_{h\in\mathcal{H}} \frac{1}{(D[h](x))^{2s}} \cdot \exp(wc(h)) \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x).\end{aligned}$$

$\mathbf{G}_{s,w}$	$\mathbf{G}_{s,w}[f](x) = \sum_{h\in\mathcal{H}} \lvert h'(x)\rvert^s \cdot \exp(wc(h)) \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x)$	Systèmes MSB
$\mathbf{K}_{s,w}$	$\mathbf{K}_{s,w}[f](x) = \sum_{h\in\mathcal{H}} \lvert h'(x)\rvert_2^s \cdot \exp(wc(h)) \cdot f\circ h(x)$	Système $S_{\mathcal{L}}$
$\mathbf{L}_{s,w,t}$	$\mathbf{L}_{s,w,t}[f](y) = \sum_{h\in\mathcal{H}} \delta_h^t \cdot \lvert\underline{h}'(y)\rvert^s \cdot \exp(wc(h)) \cdot f\circ \underline{h}(y)$	Système $S_{\underline{\mathcal{L}}}$
$\mathbf{M}_{s,w}$	$\mathbf{M}_{s,w}[f](x) = \sum_{h\in\mathcal{H}} p_h^s \lvert h'(x)\rvert^s \cdot \exp(wc(h)) \cdot f\circ h(x) \cdot \mathbf{1}_{Y_h}(x)$	Systèmes $S_{\mathcal{B}}$ et $S_{\mathcal{PM}}$

FIG. 4.2 – Les différents opérateurs de transfert

4.3 Propriétés génératrices des opérateurs de transfert

Nous abordons ici ce qui constitue la troisième et dernière étape d'une analyse dynamique, le transfert des propriétés du modèle continu, le système dynamique, à l'algorithme étudié. Ce transfert s'effectue via une relation entre les séries génératrices décrites au chapitre 3 et les opérateurs de transfert. Lors de leur définition, nous avons vu comment les opérateurs de transfert sont naturellement reliés aux principales quantités étudiées ici, qui sont d'une part les tailles des entrées et d'autre part les coûts étudiés. Nous explicitons ces liens maintenant, en leur donnant une forme "définitive" : nous associons à chaque type de coût -et donc à chaque série génératrice- l'opérateur qui lui est associé. Ce paragraphe est séparé en deux sous-paragraphes : nous traitons dans le premier les coûts associés aux algorithmes, donc de nature discrète. Les séries génératrices sont des séries de Dirichlet. Nous y distinguons les coûts additifs (nombre d'itérations par exemple), les coûts non-additifs (complexité en bits) et les coûts associés à l'étude de l'algorithme interrompu. Le second paragraphe concerne les coûts associés aux trajectoires génériques tronquées des systèmes dynamiques. Les séries sont donc maintenant des séries génératrices des moments et, comme pour l'étude des algorithmes, on distingue les coûts additifs (définis sur les quotients) des coûts non-additifs, tels les tailles des continuants.

Certaines des propositions énoncées dans ce paragraphe sont valides pour tous les algorithmes et systèmes dynamiques, d'autres non. Dans le premier cas, l'opérateur de transfert est désigné par $\mathbf{H}_{s,w}$, qui est la dénomination générique des opérateurs. Dans le second cas, on spécifie l'opérateur utilisé, qui peut donc être $\mathbf{G}_{s,w}$, $\mathbf{K}_{s,w}$, $\mathbf{L}_{s,w}$ ou $\mathbf{M}_{s,w}$.

4.3.1 Séries génératrices de Dirichlet et opérateurs de transfert

Ce paragraphe contient trois propositions, chacune d'entre elle traitant un type de coût particulier. Pour les coûts additifs, la construction au paragraphe 4.2 des opérateurs de transfert apporte presque immédiatement la relation entre séries génératrices bivariées et opérateurs $\mathbf{H}_{s,w}$. Plus précisément, cette relation fait intervenir le quasi-inverse $(I-\mathbf{H}_{s,w})^{-1}[1](0)$, qui est un objet central des différentes analyses. On déduit de cette relation le lien entre les séries génératrices univariées, utilisées pour l'analyse en moyenne, et les quasi-inverses $(I-\mathbf{H}_{s,0})^{-1}[1](0)$. La situation est plus délicate pour la complexité en bits, puisqu'en plus de coûts associés aux quotients, il faut générer les tailles des restes et des coefficients apparaissant au cours d'une exécution de l'algorithme. Notons de plus que dans ce dernier cas, nous ne traitons que la série génératrice $T_B(s)$ utilisée pour l'analyse en moyenne. Finalement, nous étudions dans les propositions 4.4, 4.5 et 4.6 les opérateurs liés à l'étude des algorithmes d'Euclide interrompus $\mathcal{E}_t$ et $\underline{\mathcal{E}}_\delta$.

Enfin, en vertu des observations faîtes à la fin du chapitre 3 sur les ensembles Ω et $\widetilde{\Omega}$, nous ne considérons ici que les séries génératrices définies sur l'ensemble Ω des entrées valides de l'algorithme premières entre elles.

Coûts additifs

La proposition suivante ainsi que celle qui en découle sont valides pour tous les algorithmes euclidiens.

Proposition 4.1 *Soit H un algorithme euclidien. Soit c un coût additif défini sur les quotients et C le coût total associé. Soit $F_C(s,w)$ la série génératrice bivariée associée à ce coût, telle qu'elle est définie en (3.4). Soit $\mathbf{H}_{s,w}$ l'opérateur de transfert associé au système dynamique sous-jacent à l'algorithme H et au coût c. Alors $F_C(s,w)$ et $\mathbf{H}_{s,w}$ sont reliés par*

$$F_C(2s,w) = (I-\mathbf{H}_{s,w})^{-1}[1](0). \tag{4.19}$$

Preuve : L'essentiel de la preuve repose sur le fait que le quasi-inverse parcourt tout l'ensemble Ω des entrées valides et premières entre elles. Notons en effet $\Omega^{[p]}$ le sous-ensemble de Ω constitué des entrées pour lesquelles l'algorithme fait exactement p itérations. On peut associer à chaque élément (u,v) de $\Omega^{[p]}$ un élément h de $\mathcal{H}^p$ via la relation (4.10)

$$\frac{v}{u} = h(0),$$

ce qui définit une bijection entre les ensembles $\Omega^{[p]}$ et $\mathcal{H}^p$. Cette relation permet d'une part de générer la quantité

$$\frac{1}{||(u,v)||}$$

grâce à (4.11), (4.12) et (4.14), et d'autre part le coût total $C(u,v)$. En effet, ce coût est de la forme

$$C(u,v) = \sum_{i=1}^{p} c(d_i)$$

où (d_i) est la suite de quotients générées par l'algorithme. Ce coût s'étend (voir paragraphe 4.2.2) à l'homographie $h = h_{d_1} \circ \cdots \circ h_{d_p}$ et on a donc

$$C(u,v) = c(h).$$

On déduit donc dans un premier temps l'égalité

$$\mathbf{H}^p_{s,w}[1](0) = \sum_{(u,v)\in\Omega^{[p]}} \frac{\exp(wC(u,v))}{||(u,v)||^{2s}}$$

puis par sommation sur p,

$$\begin{aligned}(I-\mathbf{H}_{s,w})^{-1}[1](0) &= \sum_{p\geq 1} \mathbf{H}^p_{s,w}[1](0) = \sum_{p\geq 1}\sum_{(u,v)\in\Omega^{[p]}} \frac{\exp(wC(u,v))}{||(u,v)||^{2s}} \\ &= \sum_{(u,v)\in\Omega} \frac{\exp(wC(u,v))}{||(u,v)||^{2s}} \\ &= F_C(2s,w).\end{aligned}$$

■

La proposition suivante, qui découle de la précédente, nous sera en réalité d'une plus grande utilité, puisque c'est d'elle dont nous nous servons pour les analyses en moyenne.

Proposition 4.2 *Soit H un algorithme euclidien. Soit c un coût un coût additif défini sur les quotients et C le coût total associé. Soient $T_C(s)$, $T_1(s)$ les séries génératrices univariées associées à ce coût, telles qu'elles sont définies en (3.8). Soit $\mathbf{H}_{s,0}$ l'opérateur de transfert associé au système dynamique sous-jacent à l'algorithme H. Alors $T_C(s)$, et $\mathbf{H}_{s,0}$ sont reliés par*

$$T_C(2s) = (I-\mathbf{H}_{s,0})^{-1} \circ \mathbf{H}^{[c]}_{s,0} \circ (I-\mathbf{H}_{s,0})^{-1}[1](0) \tag{4.20}$$

où $\mathbf{H}^{[c]}_{s,0}$ est déduit de $\mathbf{H}_{s,w}$ par dérivation en $w=0$,

$$\mathbf{H}^{[c]}_{s,0} = \frac{d}{dw}\mathbf{H}_{s,w\ |w=0}.$$

La série $T_1(s)$ est donnée par

$$T_1(2s) = (I-\mathbf{H}_{s,0})^{-1}[1](0).$$

Preuve : On déduit ce corrolaire en dérivant la relation (4.19) et en utilisant la relation (3.8) entre les séries $F_C(s,w)$ et $T_C(s)$. Il faut juste remarquer que l'opérateur $\mathbf{H}^p_{s,w}$ se dérive de la manière suivante,

$$\frac{d}{dw}\mathbf{H}^p_{s,w} = \sum_{i=1}^{p} \mathbf{H}^{i-1}_{s,w} \circ \mathbf{H}^{[c]}_{s,0} \circ \mathbf{H}^{p-i}_{s,0}$$

dont on déduit le double quasi-inverse dans (4.20). ■

Complexité en bits

Comme nous l'avons fait remarquer en introduction du paragraphe, la situation est ici plus délicate. Il faut en effet maintenant générer les tailles $\ell(u_i)$ des restes apparaissant au cours d'une exécution de l'algorithme. Plus exactement, nous allons générer les quantités $\log_2(u_i)$, de manière à obtenir les coûts $\widehat{B}$ et $\widehat{X}$, approximations des coûts B et X (1.24,1.25,1.27). Ceci se fait en remarquant que ces différentes quantités s'observent sur le développement en fraction continue associé à une entrée (u, v) de l'algorithme.

La technique utilisée ici est celle introduite par Akhavi et Vallée dans [AV00, Val98a]. Soit $(u, v) \in \Omega$ une entrée valide d'un algorithme euclidien (on a $\mathrm{pgcd}(u, v) = 1$), et $h \in \mathcal{H}^p$ l'homographie associée, $v/u = h(0)$. Si cette homographie s'écrit $h = h_1 \circ h_2 \circ \cdot \circ h_p$, alors on définit les homographies $e_i(h)$ et $b_i(h)$ (pour ending et begining) par

$$b_i(h) = h_1 \circ h_2 \circ \ldots \circ h_{i-1}, \qquad e_i(h) = h_i \circ h_{i+1} \circ \ldots \circ h_p. \tag{4.21}$$

Ces deux homographies sont reliées aux restes u_i et aux coefficients a_i, b_i calculés par la version étendue de l'algorithme. En effet, elles définissent deux rationnels

$$e_i(h)(0) = \frac{v_i}{w_i}, \qquad b_i(h)(0) = \frac{p_i}{q_i}, \tag{4.22}$$

où p_i, q_i et v_i, w_i sont des entiers premiers entre eux, reliés avec les restes (u_i) et les coefficients (a_i, b_i) par

$$u_i = w_i, \qquad p_i = |a_{i+1}|, \qquad q_i = |b_{i+1}|. \tag{4.23}$$

La première relation n'est valide que si $(u, v) \in \Omega$, donc si u et v sont premiers entre eux. Dans le cas général, on a

$$u_i = \mathrm{pgcd}(u, v) w_i.$$

Finalement, les relations (4.22,4.23) conduisent aux égalités

$$|e_i'(h)(0)|^s = \frac{1}{u_i^{2s}} \qquad \text{et} \qquad \frac{-1}{\log 2}\frac{d}{ds}|e_i'(h)(0)|^s = \log_2 u_i \;\; |e_i'(h)(0)|^s, \tag{4.24}$$

utilisées dans la preuve de la proposition 4.3.

Remarquons que la même approche appliquée à l'algorithme LSB permet d'obtenir le coût $\widehat{B}$ défini en (1.27), via l'égalité

$$|e_i'(h)(0)|^s = \frac{1}{(u_i^2 + u_{i-1}^2)^s}, \qquad \frac{-1}{\log 2}\frac{d}{ds}|e_i'(h)(0)|^s = \log_2(u_i^2 + u_{i-1}^2) \;\; |e_i'(h)(0)|^s. \tag{4.25}$$

Proposition 4.3 *Soit H un algorithme euclidien. Soit $\widehat{B}$ le coût associé à la complexité en bits de cet algorithme et $\ell(d)$ la taille d'un quotient d. Soit $T_{\widehat{B}}(s)$ la série génératrice associée, et $\mathbf{G}_{s,0}$, $\mathbf{M}_{s,0}$ les opérateurs associés aux algorithmes. Les séries et opérateurs sont reliés par*

$$T_{\widehat{B}}(2s) = (I - \mathbf{H}_{s,0})^{-1} \circ \Delta\mathbf{H}_{s,0} \circ (I - \mathbf{H}_{s,0})^{-1} \circ \mathbf{H}_{s,0}^{[\ell]} \circ (I - \mathbf{H}_{s,0})^{-1}[1](0), \tag{4.26}$$

où Δ est l'opérateur de dérivation par rapport à s défini par

$$\Delta\mathbf{L}_s := -\frac{1}{\log 2}\frac{d}{ds}\mathbf{L}_s. \tag{4.27}$$

Si HX est la version étendue de l'algorithme H, et $\widehat{X}$ le coût associé au calcul d'un coefficient, alors la série $T_{\widehat{X}}$ et les opérateurs $\mathbf{G}_{s,0}$, $\mathbf{M}_{s,0}$ sont reliés par

$$T_{\widehat{X}}(2s) = (I - \mathbf{H}_{s,0})^{-1} \circ \mathbf{H}_{s,0}^{[\ell]} \circ (I - \mathbf{H}_{s,0})^{-1} \circ \Delta\mathbf{H}_{s,0} \circ (I - \mathbf{H}_{s,0})^{-1}[1](0).$$

Preuve : Considérons tout d'abord les algorithmes MSB et Mixtes. Rappelons que si $(u, v) \in \Omega^{[p]}$ est une entrée pour laquelle l'algorithme génère les suites de restes $u_0, \ldots, u_p$ et de quotients $d_1, \ldots d_p$ alors la complexité en bits d'une telle exécution est (voir (1.21))

$$B(u, v) = \sum_{i=1}^{p} \ell(d_i)\ell(u_i),$$

et l'approximation $\widehat{B}$ que nous en faisons est donnée par (1.24),

$$\widehat{B}(u, v) = \sum_{i=1}^{p} \ell(d_i) \log_2(u_i).$$

De même, si a_i est une des deux suites calculées dans la version étendue de l'algorithme, alors le coût supplémentaire est (voir (1.21))

$$X(u, v) := \sum_{i=1}^{p} \ell(d_i) \times \ell(a_i).$$

L'approximation $\widehat{X}$ de ce coût est (1.25) :

$$\widehat{X}(u, v) := \sum_{i=1}^{p} \ell(d_i) \times \log_2 \left(a_i + a_{i-1} \frac{u_i}{u_{i-1}} \right).$$

La taille des quotients $\ell(d_i)$ est engendrée de la même manière que précédemment dans l'opérateur $\mathbf{H}_{s,w}$. Les quantités $\log_2(u_i)$ et $\log_2 \left(a_i + a_{i-1} \frac{u_i}{u_{i-1}} \right)$ sont engendrées grâce à la relation (4.24). Considérons en effet l'opérateur $\mathbf{H}_{s,0}^{i-1} \circ \mathbf{H}_{s,0}^{[\ell]} \circ \Delta\mathbf{H}_{s,0}^{p-i}$. En vertu de (4.24), cet opérateur vérifie

$$\mathbf{H}_{s,0}^{i-1} \circ \mathbf{H}_{s,0}^{[\ell]} \circ \Delta\mathbf{H}_{s,0}^{p-i}[1](0) = \sum_{(u,v)\in\Omega^{[p]}} \frac{\log_2(u_i)}{u^{2s}} \ell(d_i).$$

On a donc de la sorte engendré la complexité en bits de la i-ème itération. On obtient la complexité en bits totale en parcourant toutes les itérations :

$$\sum_{i=1}^{p} \mathbf{H}_{s,0}^{i-1} \circ \mathbf{H}_{s,0}^{[\ell]} \circ \Delta\mathbf{H}_{s,0}^{p-i}[1](0) = \sum_{(u,v)\in\Omega^{[p]}} \widehat{B}(u, v). \tag{4.28}$$

Finalement, on remarque que l'opérateur $\Delta\mathbf{H}_{s,0}^{q}$ vérifie

$$\Delta\mathbf{H}_{s,0}^{q} = \sum_{j=1}^{q} \mathbf{H}_{s,0}^{j-1} \circ \Delta\mathbf{H}_{s,0} \circ \mathbf{H}_{s,0}^{q-j},$$

ce qui entraîne

$$\begin{aligned} T_B(2s) &= \sum_{p\geq 1}\sum_{i=1}^{p} \mathbf{H}_{s,0}^{i-1}\circ\mathbf{H}_{s,0}^{[\ell]}\circ\Delta\mathbf{H}_{s,0}^{p-i}[1](0) \\ &= (I-\mathbf{H}_{s,0})^{-1}\circ\Delta\mathbf{H}_{s,0}\circ(I-\mathbf{H}_{s,0})^{-1}\circ\mathbf{H}_{s,0}^{[\ell]}\circ(I-\mathbf{H}_{s,0})^{-1}[1](0) \end{aligned}$$

Considérons maintenant la complexité supplémentaire liée au calcul des coefficients de Bezout. L'opérateur $\Delta\mathbf{H}_{s,0}^{i-1}\circ\mathbf{H}_{s,0}^{[\ell]}\circ\mathbf{H}_{s,0}^{p-i}$ vérifie

$$\Delta\mathbf{H}_{s,0}^{i-1}\circ\mathbf{H}_{s,0}^{[\ell]}\circ\mathbf{H}_{s,0}^{p-i}[1](0) = \sum_{(u,v)\in\Omega^{[p]}} \frac{1}{u^{2s}}\log_2\left(a_i + a_{i-1}\frac{u_i}{u_{i-1}}\right)\ell(d_i). \tag{4.29}$$

On applique alors la même démarche que précédemment, et on obtient le résultat souhaité.

Pour l'algorithme LSB, la démarche est la même : le coût

$$\widehat{B}(u,v) = \frac{1}{2}\sum_{i=1}^{p}\ell(d_i)\times\log_2(u_i^2+u_{i+1}^2)$$

s'obtient exactement de la même manière grâce à (4.25).

■

Coûts relatifs aux algorithmes interrompus

Nous nous concentrons maintenant sur les coûts relatifs aux algorithmes interrompus $\mathcal{E}_t$, et $\underline{\mathcal{E}}_\delta$.

Le premier, $M_{t,\delta,\gamma}$ défini en (1.19) par

$$M_{t,\delta,\gamma}(u,v) = \left(\frac{u_{\lfloor\delta p\rfloor}}{u_0^t}\right)^\gamma,$$

est utilisé pour déterminer le nombre d'itérations de l'algorithme interrompu de paramètre t, $\mathcal{E}_t$. Les coûts $\underline{\widehat{B}}_\delta$ et $\underline{\widehat{X}}_\delta$ sont définis en (1.241.25), et sont les coûts "approximatifs" servant à l'analyse de la complexité en bits de l'algorithme interrompu $\underline{\mathcal{E}}_\delta$. Ils nous permettront également d'obtenir la complexité en bits de l'algorithme interrompu $\mathcal{E}_t$, comme nous le montrerons dans le chapitre 5. Enfin, le coût U_δ, défini en (1.20) par

$$U_{\delta,B}(u,v) := \mathbf{1}_B(\frac{u_{\lfloor\delta p\rfloor+1}}{u_{\lfloor\delta p\rfloor}}),$$

est utilisé pour étudier l'évolution de la distribution des restes au cours d'une exécution de l'algorithme.

Proposition 4.4 *Soit $M_{t,\delta,\gamma}$ le coût défini en (1.19) relatif au nombre d'itérations de l'algorithme interrompu $\mathcal{E}_t$. Soit $T_M(s)$ la série génératrice associée, et $\mathbf{G}_{s,0}$ l'opérateur de transfert relatif au système $S_{\mathcal{E}}$. Soient s^+ et s^- définis par*

$$s^+ = s + t\gamma, \qquad s^- = s - \beta\gamma$$

avec $\beta = 1 - t$. Alors $T_M(s)$ et $\mathbf{G}_{s,0}$ sont reliés par

$$T_M(2s) = \sum_{p\geq 0} \mathbf{G}_{s^-,0}^{p-\lfloor \delta p\rfloor} \circ \mathbf{G}_{s^+,0}^{\lfloor \delta p\rfloor}[1](0).$$

Preuve : L'approche utilisée ici est très proche de celle adoptée dans la preuve de la proposition 4.3, puisqu'ici aussi le coût à générer fait intervenir les restes apparaissant au cours d'une exécution de l'algorithme.

Soit $(u, v) \in \Omega$ une entrée de l'algorithme, associée à une homographie $h \in \mathcal{H}^p$. Découpons cette homographie de la même manière que précédemment, c'est-à-dire en deux partie $e_i(h)$ et $b_i(h)$. Soient s et t deux paramètres complexes. On observe alors la relation

$$|e_i'(h)(0)|^t \cdot |b_i'(u)(e_i(h)(0))|^s = |e_i'(h)(0)|^{t-s} \cdot |h'(0)|^s = \frac{u_i^{s-t}}{u^{2s}}$$

En posant $s = s^-$, $t = s^+$ et $i = \lfloor \delta p\rfloor$ et en étendant cette relation à toutes les branches inverses $h \in \mathcal{H}^p$, puis à toutes les profondeurs p on déduit le résultat. ∎

Les séries génératrices relatives à la complexité en bits des algorithmes interrompus ont une forme légèrement différente de celles obtenues dans la proposition 4.3.

Proposition 4.5 *Soient $\underline{\widehat{B}}_\delta$ et $\underline{\widehat{X}}_\delta$ les coûts relatifs à la complexité en bits de l'algorithme interrompu de paramètre δ, $\underline{\mathcal{E}}_\delta$. Les séries génératrices associées $T_{\underline{\widehat{B}}_\delta}(s)$ et $T_{\underline{\widehat{X}}_\delta}(s)$ sont reliées aux opérateurs de transfert $\mathbf{G}_{s,0}$ par*

$$T_{\underline{\widehat{B}}_\delta}(2s) = \sum_{p\geq 0} \sum_{i=1}^{\lfloor (1-\delta)p\rfloor} \Delta\mathbf{G}_{s,0}^{p-i} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \mathbf{G}_{s,0}^{i-1}[f](0),$$

$$T_{\underline{\widehat{X}}_\delta}(2s) = \sum_{p\geq 0} \sum_{i=1}^{\lfloor (1-\delta)p\rfloor} \mathbf{G}_{s,0}^{p-i} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \Delta\mathbf{G}_{s,0}^{i-1}[f](0).$$

Preuve : La démarche est ici exactement la même que dans la proposition 4.3, la seule différence étant que l'algorithme n'effectue pas toutes les itérations. Plus exactement, l'équation (4.28) devient dans notre cas

$$\sum_{i=1}^{\lfloor \delta p\rfloor} \mathbf{G}_{s,0}^{i-1} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \Delta\mathbf{G}_{s,0}^{p-i}[1](0) = \sum_{(u,v)\in\Omega^{[p]}} \underline{\widehat{B}}(u, v). \tag{4.30}$$

On en déduit la proposition en sommant sur p. ■

Proposition 4.6 *Soit $T_U(s)$ la série génératrice associée au coût $U_{\delta,B}$ défini en (1.20) et $\mathbf{H}_{s,0}$ l'opérateur de transfert associé au système $S_{\mathcal{E}}$. Alors*

$$T_U(2s) = \sum_{p\leq 1} \mathbf{H}_{s,0}^{p-\lfloor \delta p\rfloor}[\mathbf{1}_B \mathbf{H}_{s,0}^{\lfloor \delta p\rfloor}[1]](0).$$

Preuve : Il s'agit maintenant de générer la quantité $\mathbf{1}_B(x_\delta(u,v))/||(u,v)||^s$. Les mêmes raisonnements que ceux utilisés précédemment s'appliquent. Si (u,v) appartient à l'ensemble $\Omega^{[p]}$, si $h \in \mathcal{H}^p$ est l'homographie correspondante, et si g et r sont les applications $g := e_{\lfloor \delta p\rfloor}(h)$ et $r := b_{\lfloor \delta p\rfloor}(h)$, alors on vérifie

$$\mathbf{1}_B(x_\delta(u,v))/||(u,v)||^s = \mathbf{1}_B(g(0)) \cdot [(r\circ g)'(0)]^s.$$

On en déduit le résultat. ■

Enfin, quand on étudie algorithme LSB, et que l'ensemble Ω des entrées de l'algorithme est muni de la taille associée au nombre de décalages effectués par l'algorithme, on manipule des séries entières de la forme (3.6)

$$F_C(z,w) = \sum_{(u,v)\in\Omega} \exp(wC(u,v)) z^{||(u,v)||},$$

ainsi que l'opérateur $\mathbf{K}_{s,w}$ associé au système uniquement 2-adique $S_{\mathcal{L}}$.

Proposition 4.7 *Soit c un coût à croissance modérée et C la variable de coût total associée. Soit $F_C(z,w)$ la série génératrice entière associée à ce coût et à la taille*

$$\ell(u,v) = k$$

si l'algorithme $\mathcal{L}$ effectue k décalages sur entrée (u,v). Alors cette série est reliée à l'opérateur $\mathbf{K}_{s,w}$ par

$$(\mathrm{I} - \mathbf{K}_{s,w})^{-1}[f](0) \quad = \quad F(2^{-2s}, w). \tag{4.31}$$

En particulier, la série associée au coût $C = 1$ vérifie

$$(\mathrm{I} - \mathbf{K}_{s,0})^{-1}[f](0) \quad = \quad T_1(2^{-2s}) \tag{4.32}$$

Preuve : Considérons l'opérateur $\mathbf{K}_{s,w}^p$. Rappelons qu'il est donné par

$$\mathbf{K}_{s,w}^p[f](x) = \sum_{h\in\mathcal{H}^p} |h'(x)|_2 \cdot \exp(wc(h)) \cdot f\circ h(x).$$

Si (u, v) est une entrée de l'algorithme, associée à l'homographie $h \in \mathcal{H}^p$, alors on a

$$h(0) = \frac{u}{v}, \qquad |h'(0)|_2 = 2^{-2k}$$

où k n'est rien d'autre que le nombre de décalages effectués par l'algorithme, et donc

$$|h'(0)|_2 = 2^{-2||(u,v)||}.$$

On en déduit donc

$$\begin{aligned} \mathbf{K}^p_{s,0}[1](0) &= \sum_{h \in \mathcal{H}^p} |h'(0)|_2^s \exp(wc(h)) \\ &= \sum_{(u,v)\in\Omega^{[p]}} 2^{-2||(u,v)||s} \exp(wC(u,v)) \\ &= F(2^{-2s}, w). \end{aligned}$$

■

4.3.2 Séries génératrices des moments et opérateurs de transfert

Nous étudions maintenant l'expression des séries génératrices des moments liées aux trajectoires tronquées d'un système dynamique. Étant donné un système $S = (X, T)$ et un coût C_n défini sur les trajectoires de longueur n, rappelons que la série associée est

$$\mathbb{E}[\exp(wC_n)] = \int_X \exp(wC_n(x)) d\mu(x),$$

μ désignant la mesure de Haar définie sur l'ensemble X. Nous distinguons deux cas, selon la nature du coût : dans la première proposition, nous traitons le cas des coûts définis sur l'ensemble des quotients $\mathcal{D}$, et dans la seconde le coût lié à la taille des continuants. Le premier cas est le plus simple : aucune taille n'intervient, et l'opérateur associé ne dépend donc plus du paramètre s. Il en va autrement pour le second cas : la taille est confondue avec le coût à étudier, et la variable w se place donc dans l'opérateur en exposant de la dérivée.

La première relation est valide pour tous les systèmes étudiés ici, que nous appelons systèmes dynamiques euclidiens, et nous employons donc l'opérateur "générique" $\mathbf{H}_{1,w}$. Les formes des relations relatives aux tailles des continuants diffèrent selon le système. Nous traitons le système LSB, pour lequel nous utilisons l'opérateur à trois variables $\mathbf{L}_{s,t,w}$.

Proposition 4.8 *Soit $S = (X, T)$ un système dynamique euclidien. Soit c un coût défini sur l'ensemble $\mathcal{D}$ des chiffres du système et C_n la variable aléatoire correspondante. Si X est muni de la mesure de Haar μ, alors la série génératrice des moments $\mathbb{E}[\exp(wC_n)]$ est reliée à l'opérateur $\mathbf{H}_{1,w}$ par*

$$\mathbb{E}[\exp(wC_n)] = \int_X \mathbf{H}^n_{1,w}[1](t) d\mu(t).$$

Preuve : Par définition, la série $\mathbb{E}[\exp(wC_n)]$ est donnée par

$$\mathbb{E}[\exp(wC_n)] = \int_X \exp(wC_n(t))d\mu(t) = \sum_{h\in\mathcal{H}^n} \int_{h(X)} \exp(wc(h))d\mu(t)$$

puisque l'ensemble $\{h(X), h \in \mathcal{H}^n\}$ forme une partition de X. En appliquant le changement de variables $y = h(t)\mathbf{1}_{Y_h}(t)$, on obtient

$$\mathbb{E}[\exp(wC_n)] = \sum_{h\in\mathcal{H}^n} \int_X |h'(y)| \cdot \exp(wc(h))| \cdot \mathbf{1}_{Y_h}(t)d\mu(y) = \int_X \mathbf{H}^n_{1,w}[1](y)d\mu(y).$$

■

Proposition 4.9 *Soit $S_{\underline{\mathcal{L}}} = (\mathcal{B}\times J, \underline{T})$ le système dynamique lié à l'algorithme LSB. Soit $Q_n(x)$ le n-ème continuant $Q_n(x) = (p_n(x), q_n(x))$ du nombre 2-adique $x \in \mathcal{B}$. Alors si $\mathcal{B}$ est muni de la mesure de Haar μ, la série génératrice des moments de la norme $||Q_n||$ du continuant est reliée à l'opérateur $\mathbf{L}_{s,t,w}$ par*

$$\mathbb{E}[\exp(2w\log||Q_n||)] = \mathbf{L}^n_{1-w,-w,0}[1](0)$$

Preuve : Rappelons tout d'abord que l'opérateur $\mathbf{L}_{s,t,w}$ est défini par (voir (4.17))

$$\mathbf{L}_{s,t,w}[f](x) = \sum_{h\in\mathcal{H}} \delta_h^t \cdot |\underline{h}'(x)|^s \cdot \exp(wc(h)) \cdot f(\underline{h}(x)).$$

Soit un nombre 2-adique $x \in \mathcal{B}$ dont la trajectoire de longueur n est associée à la suite de chiffres $d_1, \dots d_n$ et ainsi à la matrice $\mathcal{N}_{[d]}$, définie par

$$\mathcal{N}_{[d]} = \mathcal{N}_{[d_1]} \cdot \mathcal{N}_{[d_2]} \cdots \mathcal{N}_{[d_n]}.$$

Le n-ème continuant Q_n vérifie donc (voir (2.22))

$$Q_n = \mathcal{N}_{[d]} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Nous réutilisons maintenant la relation (4.14) entre la taille de la matrice $\mathcal{N}_{[d]}$ et la quantité $\delta_h \cdot |\underline{h}'(y)|$ pour en déduire

$$||Q_n||^2 = ||\mathcal{N}_{[d]}(1,0)||^2 = \frac{1}{\delta_h|\underline{h}'(0)|}.$$

On en déduit donc que la série $\mathbb{E}[\exp(2w \log ||Q_n||)]$ vérifie

$$\begin{aligned}
\mathbb{E}[\exp(2w \log ||Q_n||)] &= \sum_{d \in \mathcal{D}^n} \int_{\mathcal{B}_d} ||\mathcal{N}_{[d]}(1,0)||^{2w} d\mu(t) \\
&= \sum_{d \in \mathcal{D}^n} ||\mathcal{N}_{[d]}(1,0)||^{2w} \int_{\mathcal{B}_d} d\mu(t) \\
&= \sum_{h \in \mathcal{H}^n} \delta_h^{-w} |\underline{h}'(0)|^{-w} \delta_h \\
&= \mathbf{L}^n_{1-w,w,0}[1](0).
\end{aligned}$$

L'avant dernière égalité vient de

$$\int_{\mathcal{B}_d} d\mu(t) = \delta_{h_d}.$$

■

4.4 Conclusion

Les correspondances établies dans les propositions 4.1 à 4.9 sont les bases de l'analyse dynamique : c'est grâce à elles que le détour par les systèmes dynamiques se justifie pleinement. Il présente en outre l'intérêt de faire intervenir des objets abondamment étudiés dans la littérature, à savoir les opérateurs transformateurs de densité, et dans une moindre mesure les opérateurs de transfert. Nous allons donc par la suite pouvoir nous baser sur de nombreux travaux existant : cependant, les systèmes manipulés (à l'exception du système $S_{\mathcal{E}}$, pour lequel on dispose déjà d'une étude poussée) présentent des caractéristiques plus rarement abordées dans la littérature : partition dénombrable, discontinuités, etc... Il a donc fallu adapter les travaux antérieurs à notre cadre. Nous précisons ces points dans le prochain chapitre.

Chapitre 5

Résultats et Preuves

Sommaire

Arrivés à ce stade, nous avons introduit tous les outils de l'analyse. L'étude des algorithmes décrits dans le chapitre 1 se fait à l'aide des séries génératrices du chapitre 3. L'étude des systèmes dynamiques du chapitre 2 se fait à l'aide des opérateurs de Perron-Frobenius du chapitre 4. Nous avons établi des liens entre algorithmes et systèmes dynamiques à deux niveaux : au niveau de la modélisation, puisque les systèmes dynamiques $S_{\mathcal{E}}$, $S_{\mathcal{E}_\alpha}$, $S_{\mathcal{L}}$ et $S_{\underline{\mathcal{L}}}$ ont été présentés comme extensions continues des algorithmes $\mathcal{E}$, $\mathcal{E}_\alpha$ et $\mathcal{L}$, puis au niveau des outils, puisque nous avons au paragraphe 4.3 relié les séries génératrices aux opérateurs de transfert. Il reste maintenant une étape, une dernière correspondance, qui est celle du transfert des propriétés. Plus précisément, afin d'obtenir le comportement probabiliste des algorithmes, nous devons appliquer aux séries génératrices divers théorèmes (théorème Taubérien, théorème des Quasi-Puissances...), qui tous reposent sur les propriétés analytiques des séries. Les propositions du chapitre précédent sous-entendent qu'elles s'énoncent en termes d'opérateurs. Plus précisément, ce sont les propriétés

spectrales des opérateurs qui vont intervenir.

Nous allons dans ce chapitre procéder en deux temps. La première partie du chapitre est consacrée à la description des propriétés spectrales requises, cette description étant présentée sous la forme d'une preuve générique. Nous décrivons dans cette partie quelques unes des techniques utilisées par la suite. Puis nous montrons comment appliquer cette démarche générale à chaque type de système rencontré : nous traiterons d'abord les systèmes α-euclidiens, ce qui nous permettra de montrer les théorèmes 5.22,5.23,5.24, puis les deux systèmes LSB, pour lesquels nous montrerons les théorèmes 5.31,5.32,5.40,5.42. Enfin, nous énoncerons et prouverons les théorèmes 5.43,5.48,5.49 obtenus sur l'algorithme interrompu. Dans cette dernière partie, nous réutiliserons certains des résultats obtenus sur les algorithmes α-euclidiens. La conclusion du chapitre est essentiellement consacrée aux algorithmes Mixtes, en particulier à l'algorithme Plus-Moins, pour lequel nous n'avons pas obtenu de résultats. Nous expliquons en quoi les méthodes utilisées ici ne s'appliquent (pour le moment) pas.

5.1 Propriétés spectrales des opérateurs : généralités

Nous énonçons ici les différentes conditions que doivent satisfaire les opérateurs pour finalement conclure les analyses. Cette présentation suit la démarche générale d'une "preuve d'analyse dynamique", ce qui nous permet d'introduire au fur et à mesure les différents objets étudiés.
Ce paragraphe étant "générique" nous utiliserons l'opérateur $\mathbf{H}$. Rappelons tout d'abord ce qu'on souhaite faire avec cet opérateur :

- appliquer le théorème Taubérien (théorème B) aux séries des propositions 4.2,4.3,4.4,4.5,4.6, reliées aux opérateurs $\mathbf{H}_{s,0}$, $\mathbf{H}_{s,0}^{[c]}$ ainsi qu'au quasi-inverse $(I-\mathbf{H}_{s,0})^{-1}$,
- appliquer le théorème des Quasi-Puissances de Hwang (théorème A) aux séries des propositions 4.8,4.9, reliées à l'opérateur $\mathbf{H}_{1,w}$ (et $\mathbf{L}_{1-w,w,0}$ pour l'algorithme LSB) ,
- enfin appliquer le théorème C d'extraction de coefficient à la série entière $F_C(z,w)$ de la proposition 4.7, reliée au quasi-inverse $(I-\mathbf{H}_{s,w})^{-1}$ (en fait $(I-\mathbf{K}_{s,w})^{-1}$ puisqu'on ne s'intéresse ici qu'au système $S_{\mathcal{L}}$).

On cherche donc à exhiber des comportements différents selon les cas. Si on se concentre sur la variable s, on cherche une singularité polaire et des propriétés d'analycité sur un demi-plan complexe, et si on étudie la variable w, on se contente essentiellement d'analycité dans un voisinage de 0. D'une manière générale, nous montrerons toutes ces propriétés en trois temps :

- on étudie tout d'abord l'opérateur de Perron-Frobenius, et principalement ses propriétés spectrales lorsqu'il agit sur un espace adéquat,
- on étend ces propriétés aux opérateurs de transfert par perturbation, ce qui fait apparaître la singularité souhaitée pour le théorème Taubérien et la décomposition en quasi-puissance pour le théorème de Hwang,
- puis on fait une étude plus fine des propriétés spectrales dominantes des opérateurs de transfert de manière à satisfaire toutes les conditions d'analycité.

Les deux premières étapes sont communes aux analyses en moyenne ou en distribution, alors que la dernière est propre à chaque résultat qu'on cherche à obtenir. Nous énonçons toutes les propriétés requises pour ces trois étapes dans les trois paragraphes suivants.

5.1.1 Propriétés de l'opérateur de Perron-Frobenius

Considérons donc un opérateur transformateur de densité $\mathbf{H}$, ainsi qu'un espace fonctionnel $\mathcal{F}$, muni d'une norme $||.||_{\mathcal{F}}$. On impose tout d'abord à l'opérateur d'agir sur cet espace, et donc de vérifier la propriété (**A1**),

(**A1**) $$\sup_{f\in\mathcal{F}} \frac{||\mathbf{H}[f]||_{\mathcal{F}}}{||f||_{\mathcal{F}}} < \infty.$$

Dans toute la suite, nous supposerons que l'opérateur agit sur l'espace $\mathcal{F}$. La première propriété recherchée dans ce cas concerne le spectre de l'opérateur. Nous noterons $\mathrm{Sp}(\mathbf{H})$ ce spectre, défini par

$$\mathrm{Sp}(\mathbf{H}) = \{\lambda \in \mathbb{C}, \quad (\mathbf{H} - \lambda I) \text{ n'est pas inversible}\}.$$

Nous appellerons valeur spectrale un élément du spectre. Une valeur spectrale λ peut donc être de deux types :

- soit $(\mathbf{H} - \lambda I)$ n'est pas injective, auquel cas λ est une valeur propre, associée à (au moins) une fonction propre $f_\lambda \neq 0$ de sorte que
$$\mathbf{H}[f_\lambda] = \lambda f_\lambda,$$
- soit $(\mathbf{H} - \lambda I)$ n'est pas une valeur propre, auquel cas nous dirons que c'est une valeur spectrale de type 2.

Nous noterons $r_0(\mathbf{H})$, ou plus simplement r_0, le rayon spectral de l'opérateur,

$$r_0 := \sup\{|\lambda|, \quad \lambda \in \mathrm{Sp}(\mathbf{H})\}.$$

En particulier, le rayon spectral est relié à la norme $||\mathbf{H}||_{\mathcal{F}}$ de l'opérateur par la relation

$$r_0(\mathbf{H}) = \lim_{n\to\infty} ||\mathbf{H}^n||_{\mathcal{F}}^{\frac{1}{n}},$$

donnée par le théorème du rayon spectral (voir [Kat80] III.6.2). Rappelons qu'ici la norme de l'opérateur $||\mathbf{H}||_{\mathcal{F}}$ n'est rien d'autre que

$$||\mathbf{H}|| = \sup_{\substack{f\in\mathcal{F},\\ ||f||_{\mathcal{F}}<1}} ||\mathbf{H}[f]||_{\mathcal{F}},$$

et qu'on déduit directement du théorème du rayon spectral l'inégalité

$$r_0(\mathbf{H}) \leq ||\mathbf{H}||_{\mathcal{F}}. \tag{5.1}$$

Nous demanderons à l'opérateur d'avoir des propriétés spectrales dominantes très fortes, résumées par la condition (**A2**) ci-dessous.

(**A2**) L'opérateur $\mathbf{H}$ possède une unique valeur propre dominante, simple, isolée du reste du spectre.

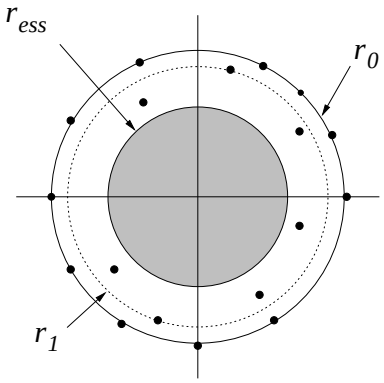

FIG. 5.1 – Le spectre d'un opérateur quasi-compact.

Cette propriété en contient en réalité deux distinctes. On demande tout d'abord que sur le cercle de rayon $|r_0(\mathbf{H})|$, le spectre soit réduit à un unique élément λ qui doit être une valeur propre simple. De plus, on demande que λ soit isolée du reste du spectre. Ceci signifie (voir [Kat80], III.6.4, p. 178) qu'il existe un disque Γ du plan complexe, de frontière γ, tel que le seul point du spectre contenu dans Γ est la valeur propre λ. La courbe γ sépare donc le spectre en deux parties.

Cette dernière condition est notamment vérifiée quand l'opérateur est quasi-compact. On trouve dans la littérature différentes définitions (souvent équivalentes) de la quasi-compacité. Nous adoptons ici le formalisme de Nussbaum [Nus70]. La quasi-compacité d'un opérateur dépend de son spectre essentiel. Un élément λ du spectre essentiel $\mathrm{Sp}_{ess}(\mathbf{H})$ de l'opérateur $\mathbf{H}$ peut être de trois types :

- (A) λ est une valeur propre de multiplicité infinie,
- (B) λ n'est pas isolée dans le spectre,
- (C) l'image de $(\mathbf{H} - \lambda\mathbf{I})$ n'est pas fermée,

les trois types n'étant exclusifs les uns des autres. Une valeur propre qui appartient au spectre essentiel est nécessairement de type (A) ou (B) (voir [Nus70]). Nous noterons $r_{ess}(\mathbf{H})$ le rayon spectral essentiel, c'est-à-dire

$$r_{ess}(\mathbf{H}) := \sup\{|\lambda|, \quad \lambda \in \mathrm{Sp}_{ess}(\mathbf{H})\}.$$

Nous pouvons donc maintenant définir la quasi-compacité : un opérateur est quasi-compact si l'inégalité stricte

$$r_{ess}(\mathbf{H}) < r_0(\mathbf{H})$$

est vérifiée. L'ensemble $\mathrm{Sp}(\mathbf{H}) \setminus \mathrm{Sp}_{ess}(\mathbf{H})$ est alors formé de valeurs propres isolées de multiplicité finie, ce qui est un premier pas vers le trou spectral.

Nous illustrons dans la figure 5.1 le spectre d'un opérateur quasi-compact. Le comportement de la partie dominante du spectre est très importante, puisque sa nature discrète induit une décomposition spectrale de l'opérateur $\mathbf{H}$ de la forme

$$\mathbf{H}[f] = \sum_{\substack{\lambda \in \mathrm{Sp}(\mathbf{H}) \\ |\lambda| = r_0}} \lambda \mathbf{P}_\lambda[f] + \mathbf{N}[f], \tag{5.2}$$

où la somme parcourt l'ensemble des valeurs propres de module maximum, $\mathbf{P}_\lambda$ est le projecteur sur le sous-espace propre engendré par λ, et $\mathbf{N}$ est relatif au reste du spectre. Ces opérateurs

vérifient pour toutes valeurs propres λ, λ' de module maximal, $\lambda \neq \lambda'$,

$$\mathbf{P}_\lambda \circ \mathbf{P}_\lambda = \mathbf{P}_\lambda, \qquad \mathbf{P}_\lambda \circ \mathbf{P}_{\lambda'} = \mathbf{P}_{\lambda'} \circ \mathbf{P}_\lambda = 0, \qquad \mathbf{P}_\lambda \circ \mathbf{N} = \mathbf{N} \circ \mathbf{P}_\lambda = 0. \tag{5.3}$$

Plus exactement, le projecteur $\mathbf{P}$ est obtenu par intégration sur la courbe qui isole la valeur propre dominante. Si γ est la courbe qui sépare le spectre en deux parties, et à l'intérieur de laquelle se trouve λ, alors $\mathbf{P}_\lambda$ est donné par (voir par exemple [Kat80], IV.3 ou [DS58])

$$\mathbf{P}_\lambda := \frac{1}{2i\pi} \int_\gamma (z\mathbf{1} - \mathbf{H})^{-1} dz.$$

Si on note r_1 le rayon spectral de $\mathbf{N}$ (donné par $\mathbf{N} := \mathbf{H} - \mathbf{P}$), donc défini par

$$r_1 = \sup \{|\lambda|, \quad \lambda \in \mathrm{Sp}, |\lambda| \neq r_0\},$$

alors le théorème du rayon spectral implique

$$\lim_{n\to\infty} ||\mathbf{N}^n||_{\mathcal{F}}^{\frac{1}{n}} = r_1 < r_0.$$

Obtention de la quasi-compacité

Nous ouvrons ici une parenthèse pour expliquer comment montrer la quasi-compacité d'un opérateur sur un espace de Banach $\mathcal{F}$. La technique que nous décrivons est assez classique, basée sur le théorème de Ionescu-Tulcea et Marinescu [ITM50] et sa généralisation par Hennion [Hen93]. L'utilisation de ce théorème dans le cadre d'étude d'opérateur de Perron-Frobenius remonte aux travaux de Lasota et Yorke [LY73]. Ces auteurs ont traité le cas des systèmes dynamiques de l'intervalle dont le nombre de branches est fini, celles-ci pouvant éventuellement être incomplètes (leur image ne recouvre pas tout l'intervalle de définition du système). L'espace utilisé était l'espace des fonction à variation bornées (comme nous le ferons au paragraphe 5.2).

Cette technique s'applique aux opérateurs définis sur un espace de Banach $\mathcal{F}$ qui comme l'espace des fonctions à variation bornée est muni d'une "double" norme, $||.||_{\mathcal{F}} = ||.||_1 + ||.||_2$ (on peut par exemple penser à $||f||_{\mathcal{F}} = \sup|f| + \sup|f'|$), où la norme $||.||_1$ définit donc un espace contenant $\mathcal{F}$, et où $||.||_2$ est généralement une semi-norme ($\sup|f'| = 0$ si f est constante...). Dans ce cas, la quasi-compacité de l'opérateur est fortement liée à son comportement par rapport à la (semi)-norme $||.||_2$. Plus précisément on cherche à obtenir une inégalité de Lasota-Yorke, c'est-à-dire une relation du type

$$||\mathbf{H}||_{\mathcal{F}} \leq \alpha||f||_{\mathcal{F}} + \beta||f||_1,$$

où α est un réel strictement inférieur à 1 et β est un réel positif quelconque. Une fois obtenue une telle inégalité, et si les espaces choisis satisfont de bonnes conditions de compacité, on peut appliquer le théorème de Ionescu-Tulcea et Marinescu [ITM50], généralisé par Hennion [Hen93], dont nous donnons une version maintenant. Par la suite, nous appellerons ce théorème le théorème d'Hennion.

Théorème D.[Ionescu-Tulcea et Marinescu, Hennion] *Soient $(B_1, ||.||)$ et $(B_2, |||.|||)$ deux espaces de Banach avec $B_2 \subset B_1$ et B_2 dense dans B_1 (pour la norme de B_1). Supposons que la boule unité fermée de B_2 soit compacte dans la boule unité de B_1. Soit P un opérateur borné de B_1 qui envoie B_2 sur lui-même. Supposons qu'il existe deux suites $\{\alpha_n\}$ et $\{\Gamma_n\}$ de nombres positifs telles que pour tout $n \geq 1$ et pour tout $f \in B_2$,*

$$|||P^n f||| \leq \alpha_n |||f||| + \Gamma_n ||f||,$$
$$\lim_{n\to\infty} \inf(\alpha_n)^{\frac{1}{n}} < r_0(P),$$

où $r_0(P)$ est le rayon spectral de P. Alors le rayon spectral essentiel $r_{ess}(P)$ vérifie

$$r_{ess}(P) \leq \lim_{n\to\infty} \inf(\alpha_n)^{\frac{1}{n}} < r_0(P).$$

En particulier, l'opérateur est quasi-compact.

Notons que nous aurons souvent besoin d'obtenir la quasi-compacité d'un même opérateur sur plusieurs espaces. Pour cela, nous utiliserons le lemme suivant, souvent utilisé par la suite.

Lemme 5.1 *Soit $\mathbf{Q}$ un opérateur linéaire agissant sur deux espaces de Banach $\mathcal{F}_1$ et $\mathcal{F}_2$. Notons Sp_i et $\mathrm{Sp}_{ess,i}$ les spectres et spectres essentiels de $\mathbf{Q}$ sur $\mathcal{F}_i$, $i = 1, 2$. Supposons que $\mathcal{F}_1$ et $\mathcal{F}_2$ satisfont les trois propriétés suivantes*
(i) $\mathcal{F}_1 \subset \mathcal{F}_2$ et $\mathcal{F}_1$ est dense dans $\mathcal{F}_2$,
(ii) l'injection $\mathcal{F}_1 \to \mathcal{F}_2$ est continue,
(iii) la boule unité de $\mathcal{F}_1$ est $\mathcal{F}_2$-compacte dans $\mathcal{F}_2$.

Alors on a

$$\mathrm{Sp}_1 \subset \mathrm{Sp}_2.$$

Supposons de plus que
(iv) la boule unité de $\mathcal{F}_1$ est $\mathcal{F}_2$-compacte dans $\mathcal{F}_1$
alors l'inclusion

$$\mathrm{Sp}_{ess,1} \subset \mathrm{Sp}_{ess,2}$$

est vérifiée.

Preuve : Nous noterons $\mathbf{R}$ l'opérateur $\mathbf{R} := \mathbf{Q} - \lambda I$, et $||.||_i$ la norme de l'espace $\mathcal{F}_i$. Dans cette preuve, nous utiliserons deux fois le lemme suivant.

Lemme 5.2 *Supposons que les espaces $\mathcal{F}_1$ et $\mathcal{F}_2$ satisfont l'hypothèse (ii) du lemme. Soit $\mathbf{T}$ un opérateur linéaire agissant sur les deux espaces. Supposons qu'il existe $g \notin \mathbf{T}[\mathcal{F}_1]$ et une suite (f_n) de fonctions de $\mathcal{F}_2$ telles que la limite dans $\mathcal{F}_2$ de la suite $\mathbf{T}[f_n]$ est g. Alors il existe une suite (ψ_n) dans $\mathcal{F}_1$ avec $\|\psi_n\|_1 = 1$ telle que $\|\mathbf{T}[\psi_n]\|_2$ tend vers 0.*

Preuve : Puisque g n'appartient pas à $\mathbf{T}[\mathcal{F}_1]$, la suite (f_n) n'a pas de limite pour la topologie de $\mathcal{F}_1$: puisque la convergence dans $\mathcal{F}_1$ implique la convergence dans $\mathcal{F}_2$, toute limite h (pour la topologie de $\mathcal{F}_1$) doit satisfaire $\mathbf{T}[h] = g$. En particulier, la suite (f_n) n'est pas une suite de Cauchy. Il existe donc $\varepsilon > 0$ et une sous-suite n_k telle que pour tout $k \in \mathbb{N}$, on vérifie $||f_{n_k} - f_{n_{k+1}}||_1 > \varepsilon$. Posons $\phi_k := f_{n_k} - f_{n_{k+1}}$ et $\psi_k = \frac{\phi_k}{\|\phi_k\|_1}$. Alors $\|\psi_k\|_1 = 1$ et l'inégalité $\|\phi_k\|_1 > \varepsilon$ montrent que la suite $\mathbf{T}[\psi_k]$ tend vers 0 dans $\mathcal{F}_2$. ∎

Commençons par montrer l'inclusion des spectres. Tout d'abord, si λ est une valeur propre de $\mathbf{Q}$ sur $\mathcal{F}_1$, alors c'est également une valeur propre sur $\mathcal{F}_2$.

Soit donc un élément $\lambda \in \mathrm{Sp}_1$ de type 2, qui n'appartient pas à Sp_2. Alors $\mathbf{R}$ n'est pas surjectif sur $\mathcal{F}_1$ mais l'est sur $\mathcal{F}_2$: il existe donc $g \in \mathcal{F}_1$ qui n'appartient pas à $\mathbf{R}[\mathcal{F}_1]$. Mais g appartient à $\mathbf{R}[\mathcal{F}_2]$, et il existe $f \in \mathcal{F}_2$ tel que $g = \mathbf{R}[f]$. Puisque $\mathcal{F}_1$ est dense dans $\mathcal{F}_2$, il existe une suite de fonctions de $\mathcal{F}_1$ qui converge vers f dans $\mathcal{F}_2$. Donc, la suite $\mathbf{R}[f_n]$ converge vers $\mathbf{R}[f] = g$ dans $\mathcal{F}_2$. En utilisant le lemme 5.2 ainsi que l'hypothèse (iii), on montre que la suite ψ_k a une limite non nulle ψ (pour la topologie de $\mathcal{F}_2$) qui appartient à $\mathcal{F}_2$ et satisfait $\mathbf{R}[\psi] = 0$. Ceci signifie que λ est une valeur propre de $\mathbf{Q}$ dans $\mathcal{F}_2$, ce qui constitue une contradiction. Nous avons donc prouvé que chaque élément de Sp_1 est également un élément de Sp_2.
Montrons maintenant l'inclusion des spectres essentiels. Si λ est une valeur spectrale de type (A) ou (B) sur $\mathcal{F}_1$, alors c'en est également une sur $\mathcal{F}_2$.
Supposons donc que λ est une valeur spectrale de type (C) pour $\mathcal{F}_1$. L'ensemble $\mathbf{R}[\mathcal{F}_1]$ n'est pas fermé (pour la topologie de $\mathcal{F}_1$), et il existe $g \notin \mathbf{R}[\mathcal{F}_1]$ qui est la limite (toujours pour la topologie de $\mathcal{F}_1$) d'une suite $\mathbf{R}[f_n]$, avec $f_n \in \mathcal{F}_1$. Donc g est également la limite (pour la topologie de $\mathcal{F}_2$) de la suite $\mathbf{R}[f_n]$. Le lemme 5.2 et l'hypothèse (iv) impliquent maintenant qu'il existe une limite non-nulle ψ d'une suite ψ_k (pour la topologie de $\mathcal{F}_2$) qui appartient à $\mathcal{F}_1$ et satisfait $\mathbf{R}[\psi] = 0$. Ceci signifie donc que λ est une valeur propre de $\mathbf{Q}$ dans $\mathcal{F}_1$. Puisque λ est un élément de $\mathrm{Sp}_{ess,1}$, nous savons donc que c'est une valeur spectrale de type (A) ou (B). C'est donc également une valeur spectrale de type (A) ou (B) pour $\mathcal{F}_2$. Nous avons donc montré l'inclusion $\mathrm{Sp}_{ess,1} \subset \mathrm{Sp}_{ess,2}$.
■

Une fois qu'on a obtenu la quasi-compacité (avec le théorème d'Hennion, le lemme précédent ou la conjonction des deux) on a donc montré la moitié de la condition (**A2**). Il reste à étudier les valeurs propres de module maximal. D'une manière générale, les arguments que nous emploierons ici sont relativement classiques, décrit dans le livre de Viviane Baladi [Bal00] par exemple, et font essentiellement appel aux propriétés de mélange du système dynamique considéré. Lorsque l'espace fonctionnel $\mathcal{F}$ satisfait de bonnes propriétés de compacité (similaires à celle requises pour le théorème d'Hennion), alors la partie dominante du spectre est réduite à une unique valeur propre simple si le système est topologiquement mélangeant, c'est-à-dire si pour tous ouverts (V, W) de X, il existe $n_0 \geq 1$ tel que pour tout $n \geq n_0$, on vérifie

$$T^{-n}V \cap W \neq \emptyset.$$

Nous détaillerons ce point lors de l'analyse des systèmes α-euclidiens et des systèmes LSB.

Supposons maintenant que quand il agit sur $\mathcal{F}$, l'opérateur satisfait la condition (**A2**). Nous allons imposer une condition supplémentaire, la condition (**A3**).

(**A3**) La valeur propre dominante de l'opérateur est $\lambda = 1$.

Quand cette condition est vérifiée, alors la décomposition spectrale (5.2) devient

$$\mathbf{H}[f] = \mathbf{P}[f] + \mathbf{N}[f],$$

ce qui, en vertu de (5.3), s'étend aux puissances de l'opérateur,

$$\mathbf{H}^n[f] = \mathbf{P}[f] + \mathbf{N}^n[f]. \tag{5.4}$$

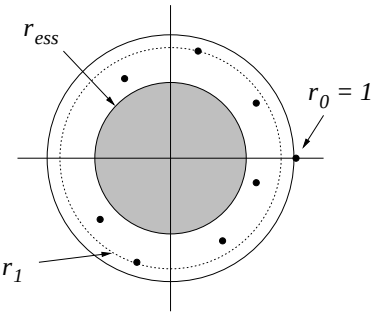

FIG. 5.2 – Le spectre d'un opérateur quasi-compact avec "trou spectral"

Remarquons que si ψ est la densité invariante du système, donc la fonction propre relative à la valeur propre 1, alors le projecteur $\mathbf{P}$ s'écrit

$$\mathbf{P}[f] = \psi \int_X f d\mu. \tag{5.5}$$

5.1.2 Perturbation

On arrive maintenant à l'étude des opérateurs de transfert $\mathbf{H}_{s,w}$. La première condition à vérifier est bien-sûr que ceux-ci sont bien définis, et donc qu'ils vérifient la propriété (**B0**).

(**B0**) il existe $\beta < 1$ et un voisinage $\mathcal{W}$ de 0 tel que quand $(s, w) \in \{\Re(s) \geq \beta\} \times \mathcal{W}$ l'opérateur $\mathbf{H}_{s,w}$ est bien défini.

Nous cherchons à étendre la décomposition spectrale (5.4) aux opérateurs de transfert. Ceci est possible grâce à la théorie de la perturbation.

Celle-ci stipule (voir [Kat80], VII.1.3 p 368) que si T_z est une famille d'opérateurs analytiques en z dans un voisinage de $z = z_0$, et que si le spectre de T_{z_0} est séparé en deux parties par une courbe fermée, alors cette séparation reste valable pour T_z si z est suffisamment proche de z_0. Cet opérateur se décompose donc en somme de deux opérateurs, $T_z = T_z^{[1]} + T_z^{[2]}$, et $T_z^{[1]}$ et $T_z^{[2]}$ sont analytiques en z.

Nous voulons appliquer ce résultat avec $T_z = \mathbf{H}_{s,w}$, $z_0 = (1, 0)$, et $T_{z_0}^{[1]} = \lambda(1,0)\mathbf{P}_{1,0}$ ($\lambda(s, w)$ désigne la valeur propre de $\mathbf{H}_{s,w}$, on a donc $\lambda(1, 0) = 1$), $T_{z_0}^{[2]} = \mathbf{N}_{1,0}$. Il faut donc vérifier la dépendance analytique des opérateurs en (s, w).

(**B1**) l'application $(s, w) \mapsto \mathbf{H}_{s,w}$ est analytique dans un voisinage de $(s, w) = (1, 0)$,
(**B1′**) l'application $s \mapsto \mathbf{H}_{s,0}$ est analytique dans un voisinage de $s = 1$,
(**B1″**) l'application $w \mapsto \mathbf{H}_{1,w}$ est analytique dans un voisinage de $w = 0$.

Bien sûr (**B1**) entraîne (**B1′**) et (**B1″**).

C'est usuellement pour montrer ces propriétés qu'intervient la restriction que nous avons imposée sur le coût c. C'est pourquoi nous donnons une définition plus précise d'un coût à croissance modérée : un coût c défini sur les branches inverses $h \in \mathcal{H}$ d'un système dynamique

(et donc de manière équivalente sur l'ensemble des chiffres indiçant la partition) est à croissance modérée s'il existe $\beta < 1$ et un voisinage $\mathcal{W}$ de 0 tel que l'application

$$(s, w) \mapsto \sum_{h \in \mathcal{H}} \delta_h^s \exp(wc(h)),$$

où $\delta_h := \sup_{x \in X} |h(x)|$, est analytique pour $(s, w) \in \{\Re(s) \geq \beta\} \times \mathcal{W}$.

Prenons l'exemple d'un opérateur de transfert $\mathbf{H}_{s,w}$ satisfaisant la condition (**B1**). Alors il se décompose autour de $(s, w) = (1, 0)$ de la manière suivante,

$$\mathbf{H}_{s,w}^n[f] = \lambda^n(s, w)\mathbf{P}_{sw}[f] + \mathbf{N}_{s,w}^n[f], \tag{5.6}$$

où $\lambda(s, w)$ est l'unique valeur propre dominante (et simple) de l'opérateur, $\mathbf{P}_{s,w}$ le projecteur sur le sous-espace engendré par $\lambda(s, w)$ et $\mathbf{N}_{s,w}$ l'opérateur relatif au reste du spectre. Dans un voisinage de $(1, 0)$ ces objets sont tous analytiques en (s, w). Toujours autour de $(1, 0)$, le rayon spectral $r_0(\mathbf{N}_{s,w})$ est inférieur à $\rho|\lambda(s, w)|$, avec $\rho < 1$, de sorte qu'on a pour n suffisamment grand l'inégalité suivante,

$$||\mathbf{N}_{s,w}^n||_{\mathcal{F}} \leq \tau^n |\lambda^n(s, w)|, \tag{5.7}$$

avec $\rho < \tau < 1$. Cette décomposition s'étend maintenant au quasi-inverse $(I - \mathbf{H}_{s,w})^{-1}$, qui s'écrit donc

$$(I - \mathbf{H}_{s,w})^{-1}[f] = \frac{\lambda(s, w)}{1 - \lambda(s, w)}\mathbf{P}_{s,w}[f] + (I - \mathbf{N}_{s,w})^{-1}[f]. \tag{5.8}$$

L'égalité $\lambda(1, 0) = 1$ fait donc apparaître une singularité polaire dans l'opérateur, alors que le second terme est analytique.

Plaçons nous maintenant dans le cadre plus particulier où on étudie le quasi-inverse $(I - \mathbf{H}_{s,0})^{-1}$. Si la dérivée par rapport à s, $\lambda'_s(s, 0)$, est différente de 1, la décomposition précédente s'écrit autour de $s = 1$

$$(I - \mathbf{H}_{s,0})^{-1}[f] \sim \frac{1}{(s - 1)} \frac{-1}{\lambda'_s(1, 0)}\mathbf{P}_{1,0}[f].$$

Avec l'expression (5.5) de $\mathbf{P}_{1,0}$ on peut finalement écrire

$$(I - \mathbf{H}_{s,0})^{-1}[f] \sim \frac{1}{s - 1} \frac{-1}{\lambda'_s(1, 0)} \psi \int_X f d\mu, \tag{5.9}$$

ce qui est la décomposition souhaitée pour appliquer le théorème taubérien. Les autres conditions pour appliquer ce théorème sont discutées dans le paragraphe suivant.

Enfin, la décomposition spectrale (5.6) nous sera utile pour appliquer le théorème des quasi-puissances. En effet, pour $\mathbf{P}_{s,w}[f] \neq 0$, elle peut également s'écrire, avec l'inégalité (5.7)

$$\mathbf{H}_{s,w}^n[f] = \exp\left[n \log \lambda(s, w) + \log \mathbf{P}_{s,w}[f]\right] (1 + O(\tau^n)) \tag{5.10}$$

ce qui est la décomposition en quasi-puissance recherchée pour appliquer le théorème de Hwang aux séries génératrices des moments (voir propositions 4.8,4.9).
Il reste maintenant à préciser le comportement des principaux objets spectraux en fonction de s w, ce qui est fait dans la partie suivante.

5.1.3 Dernière étape

Propriétés relatives à la variable s

Nous venons d'exhiber une singularité en $(s,w)=(1,0)$ pour les quasi-inverses $(I-\mathbf{H}_{s,w})^{-1}$, et donc en $s=1$ pour les quasi-inverses $(I-\mathbf{H}_{s,0})^{-1}$. Nous souhaitons appliquer le théorème Taubérien à ces derniers. Il faut donc satisfaire sur le demi-plan $\{\Re(s)\geq 1, s\neq 1\}$ des propriétés supplémentaires d'analycité. Ces conditions se transforment en conditions spectrales sur les opérateurs, plus précisément nous chercherons à vérifier la propriété suivante.

(**C1**) l'opérateur $\mathbf{H}_{s,0}$ a un rayon spectral strictement inférieur à 1 pour $\Re(s)=1, s\neq 1$. On vérifie de plus $r_{ess}(\mathbf{H}_{s,0})<1$.

La seconde partie de cette condition doit être vérifiée pour ne pas avoir de point d'accumulation sur la droite $\Re(s)=1$.

Si $\mathbf{H}_{s,0}$ est un opérateur qui vérifie la condition (**C1**), alors le quasi-inverse associé est analytique sur la demi-droite $\{\Re(s)=1, s\neq 1\}$ et possède une singularité en $s=1$, et donc, avec la décomposition (5.8), les conditions du théorème taubérien sont vérifiées.

L'étude du résidu apparaissant dans le théorème taubérien fait maintenant intervenir l'entropie $h(S)$ du système. En effet, étant donné un système $S=(X,T)$, l'entropie $h(S)$ du système est donnée par (2.1)

$$h(S) := -\int_X \log|T'(t)|\psi(t)d\mu(t).$$

On en déduit notamment la relation

$$h(S) = -\int_X \Delta\mathbf{H}_{1,0}[\psi](t)d\mu(t),$$

où Δ est l'opérateur de dérivation par rapport à s. En dérivant la relation $\mathbf{H}_{s,0}[\psi_{s,0}]=\lambda(s,0)\psi_{s,0}$, qui conduit à

$$\Delta\mathbf{H}_{s,0}[\psi_{s,0}]+\mathbf{H}_{s,0}[\Delta\psi_{s,0}]=\lambda'(s,0)\psi_{s,0}+\lambda(s,0)\Delta\psi_{s,0}.$$

En posant $s=1$, et en intégrant sur X on obtient finalement

$$\lambda'_s(1,0)=\int_X \Delta\mathbf{H}[\psi]d\mu=-h(S). \tag{5.11}$$

Remarquons de plus qu'étant donné un coût c, et l'opérateur $\mathbf{H}^{[c]}_{s,0}$, défini dans la proposition 4.2 par

$$\mathbf{H}^{[c]}_{s,0}=\frac{d}{dw}\mathbf{H}_{s,w}\ _{|w=0},$$

on a également la relation

$$\int_X \mathbf{H}^{[c]}_{1,0}[\psi](t)d\mu(t)=\sum_{h\in\mathcal{H}} c(h)\int_{X_h}\psi(t)d\mu(t)=\mu_S(c), \tag{5.12}$$

qui permet d'exprimer la valeur moyenne du coût c, $\mu_S(c)$, telle qu'elle est définie en (2.2).

Propriétés relatives à l'application $(s,w) \mapsto \Lambda(s,w)$

Nous aurons à plusieurs reprises besoins des propriétés des applications

$$(s,w) \mapsto \lambda(s,w), \qquad (s,w) \mapsto \Lambda(s,w) := \log \lambda(s,w).$$

Il faudra en particulier vérifier les deux conditions suivantes :

(**C2**) $\Lambda''_s(1,0) > 0$,
(**C3**) $\Lambda''_w(1,0) \neq 0$.

Les indices dans les dérivées indiquent la variable utilisée : λ'_s correspond à la dérivée par rapport à s, λ''_s à la dérivée seconde par rapport à s, et $\lambda''_{s,w}$ à la dérivée seconde par rapport à s et w. Il en va de même pour Λ.

La première condition nous servira lors de l'étude de l'algorithme interrompu, et la seconde lorsqu'on voudra appliquer le théorème des quasi-puissances. En effet, la décomposition (5.9) implique que ce théorème s'applique avec $U(w) = \Lambda(1,w)$. Pour pouvoir montrer un comportement gaussien, il faut donc vérifier la condition (**C3**).

Nous avons déjà évoqué les valeurs spéciales de $\lambda'_s(1,0)$, qui font intervenir l'entropie. On dispose d'une relation analogue pour la dérivée $\lambda'_w(1,0)$, qui est reliée à la valeur moyenne du coût étudié par

$$\Lambda'_w(1,0) = \lambda'_w(1,0) = \mu_S(c). \tag{5.13}$$

En effet, si on part de l'égalité $\mathbf{H}_{1,w}[\psi_{1,w}] = \lambda(1,w)\psi_{1,w}$, et qu'on dérive par rapport à w, on en déduit que pour tout $x \in X$,

$$\sum_{h\in\mathcal{H}} c(h)\cdot \exp(wc(h))\cdot |h'(x)|\cdot \psi_{1,w}\circ h(x) + \mathbf{H}_{1,w}[\psi'_{1,w}](x) = \lambda'_w(1,w)\psi_{1,w}(x) + \lambda(1,w)\psi'_{1,w}(x),$$

ce qui en $w=0$, devient

$$\sum_{h\in\mathcal{H}} c(h)\cdot |h'(x)|\cdot \psi\circ h(x) + \mathbf{H}[\psi'](x) = \lambda'_w(1,0)\psi(x) + \psi'(x).$$

En intégrant sur X, on obtient l'égalité (5.13).

Enfin, lorsqu'on souhaite utiliser ensemble les deux variables s et w, il faut vérifier une condition supplémentaire sur les valeurs propres $\lambda(s,w)$, ceci afin de pouvoir appliquer le théorème Taubérien.

(**C4**) $\lambda'_s(1,0) \neq 0$.

Cette propriété permet d'appliquer le théorème des fonctions implicites, duquel on déduit qu'il existe un voisinage complexe W de $w=0$ et une fonction $\sigma : W \to \mathbb{C}$ telle que $\lambda(\sigma(w),w) = 1$ pour $w \in W$. Cette fonction est analytique et on a $\sigma(0) = 1$.

Obtention de la propriété (C3)

Nous ouvrons une dernière parenthèse, dédiée à la propriété (**C3**). En effet, nous utiliserons à plusieurs reprises les mêmes arguments, que nous introduisons dès maintenant. La technique usuelle pour obtenir cette propriété consiste à montrer l'équivalence

$$\Lambda''_w(1,0) = 0 \Leftrightarrow \text{le coût } c \text{ est constant.}$$

Ce type de preuve se trouve sous diverses formes dans la littérature, on peut citer entre autres les travaux de Broise [Bro01], Rousseau-Egele [RE83], résumés par Collet [Col96]. Cependant tous ces auteurs traitent du cas ou la fonction de coût c appartient à l'espace fonctionnel sur lequel ils travaillent. Ce n'est pas le cas pour les coûts que nous étudions, qui correspondent à des fonctions "en escalier" puisque les fonctions $c : X \to \mathbb{N}$ sont constantes sur les éléments X_d de la partition. Il nous a donc fallu adapter les précédents travaux à notre cadre. La preuve présentée ici est basée sur celle de Broise ainsi que sur des modifications de Brigitte Vallée et Eda Cesarato [], et consiste en la proposition suivante.

Proposition 5.3 *Soit $\mathbf{H}_{s,w}$ un opérateur quasi-compact (dans un voisinage de $w = 0$) sur un espace de Banach $\mathcal{F}$. Supposons que 1 est la seule valeur propre de l'opérateur $\mathbf{H}$ sur le cercle unité (c'est également la valeur propre dominante) et qu'elle est simple. Soit $\Lambda(s,w)$ la fonction définie par $\Lambda(s,w) := \log \lambda(s,w)$ où $\lambda(s,w)$ est la valeur propre dominante de $\mathbf{H}_{s,w}$ sur $\mathcal{F}$. Si $\mathcal{F}$ est dense dans l'espace $L^2(X)$ des fonctions dont la norme $||.||_2$ définie par*

$$||f||_2 = \int_X f^2 d\mu$$

est finie, alors les deux propositions suivantes sont équivalentes.
(i) $\Lambda''_w(1,0) = 0$
(ii) le coût c est constant sur X.

Preuve : Cette preuve se décompose en une suite de lemmes. Mais tout d'abord, nous allons introduire quelques notations ainsi que quelques objets.
Nous allons devoir travailler avec un opérateur normalisé. Si $\mathbf{H}_{s,w}$ est l'opérateur de transfert étudié, et $\psi_{1,w}$ sa fonction invariante, alors on définit l'opérateur $\mathbf{L}_{1,w}$ par

$$\mathbf{L}_{1,w}[g] := \frac{1}{\lambda(1,w)\psi_{1,w}} \mathbf{H}[g\psi_{1,w}].$$

L'intérêt de cet opérateur est que la fonction propre relative à la valeur propre 1 est la fonction constante égale à 1 :

$$\mathbf{L}_{1,w}[1] := \frac{1}{\lambda(1,w)\psi_{1,w}} \mathbf{H}[\psi_{1,w}] = 1.$$

Ainsi, si $\bar{\mu}$ est la mesure invariante relative à l'opérateur $\mathbf{L}_{1,w}$, donnée par

$$d\bar{\mu} = \psi_{1,w} d\mu,$$

alors la décomposition spectrale de $\mathbf{L}_{1,w}$ est de la forme

$$\mathbf{L}_{1,w}[g] = \int_X g d\bar{\mu} + \mathbf{M}_{1,w}[g] \tag{5.14}$$

où $\mathbf{M}_{1,w}$ est un opérateur au rayon spectral strictement inférieur à 1. L'intérêt de cet opérateur est que le terme dominant de sa décomposition spectrale définit une fonction constante. Nous

utiliserons également les deux observations suivantes, obtenues directement par un changement de variables :

$$\int_X g \cdot (f \circ T)\, d\mu = \int_X (\mathbf{H}[g] \cdot f)\, d\mu, \tag{5.15}$$

$$\int_X g \cdot (f \circ T)\, d\bar{\mu} = \int_X (\mathbf{L}[g] \cdot f)\, d\bar{\mu}. \tag{5.16}$$

Enfin, étant donné un coût c, nous définissons le coût centré $\bar{c}$ par

$$\bar{c} = c - \int_X c d\bar{\mu},$$

de sorte qu'on vérifie

$$\int_X \bar{c} d\bar{\mu} = 0. \tag{5.17}$$

Finalement, nous noterons $\bar{C}_n$ le coût défini par

$$\bar{C}_n = \sum_{i=0}^{n-1} \bar{c} \circ T^i.$$

C'est en particulier pour le lemme suivant qu'il faut adapter les précédents travaux dans lesquels on utilise essentiellement la décomposition spectrale de $\mathbf{H}[c]$ ou $\mathbf{L}[c]$, qui n'est pas valide dans notre cas puisque en général c n'appartient pas à l'espace sur lequel l'opérateur est quasi-compact.

Lemme 5.4 *Si $\Lambda''(0) = 0$, alors $\bar{C}_n$ est uniformément borné dans $L^2(X)$.*

Preuve : Considérons la série génératrice des moments du coût $\bar{C}_n$, $\mathbb{E}[\exp(u\bar{C}_n)]$. Elle s'écrit

$$\mathbb{E}[\exp(u\bar{C}_n)] = \int_X \exp(uC_n(t)) d\mu(t) = \sum_{h \in \mathcal{H}} \exp(uC_n(t)) \int_{X_h} d\mu(t) = \int_X \mathbf{L}_{1,w+u}.$$

On en déduit

$$\mathbb{E}[\exp(u\bar{C}_n)] = \lambda^{-n}(1,w) \int_X \mathbf{H}_{1,w+u}[\psi_{1,w}](t) d\mu(t).$$

Cette égalité entraîne une décomposition en quasi-puissance de la série $\mathbb{E}[\exp(u\bar{C}_n)]$, de la forme

$$\mathbb{E}[\exp(u\bar{C}_n)] = \exp\left[nU(u) + V(u)\right](1 + O(\kappa^n)),$$

où $U(u)$ est donné par

$$U(u) = \log \lambda(1, w+u) - \log \lambda(1,w),$$

et $V(u)$ est

$$V(u) = \log \int_X \mathbf{P}_{1,w+u}[\psi_{1,w}][1](t) d\mu(t)$$

et κ est strictement inférieur à 1. Les fonctions U et V sont analytiques en u quand u est proche de 0. La première partie du théorème de Hwang s'applique, et la variance est donnée par

$$\mathrm{Var}[\bar{C}_n] = n\Lambda''(1,w) + V''(0) + O(\kappa^n).$$

Or par construction $\mathbb{E}[\bar{C}_n] = O(1)$ (grâce à la relation (5.17)), donc si $\Lambda''(1,0) = 0$, alors $\mathrm{Var}[\bar{C}_n]$ est $O(1)$, et donc la suite $\bar{C}_n$ est uniformément bornée dans $L^2(X)$. ■

Lemme 5.5 *Si $\Lambda_w''(0) = 0$, alors pour tout $g \in \mathbb{F}$, la suite $\bar{C}_n$ converge dans $L^2(X)$, et elle admet une limite $\bar{C}$ appartenant à L^2. En particulier, pour toute fonction $g \in L^2(X)$, on a*

$$\lim_{n\to\infty} \int_X \bar{C}_n g d\bar{\mu} = \int_X \bar{C} g d\bar{\mu}.$$

Preuve : Ce lemme découle directement du précédent. Puisque $\bar{C}_n$ est uniformément bornée dans L^2, alors cette suite converge dans L^2 vers un élément $\bar{C} \in L^2$. En particulier, pour toute fonction $g \in L^2(X)$, on vérifie

$$\lim_{n\to\infty} \int_X \bar{C}_n g d\bar{\mu} = \int_X \bar{C} g d\bar{\mu}.$$

■

Lemme 5.6 *Si $\Lambda''(0) = 0$, alors $\bar{c} = \bar{C} - \bar{C} \circ T$ dans L^2 et $\bar{C}$ appartient à $\mathcal{F}$. Finalement, l'égalité est vérifiée à l'intérieur de chaque élément de la partition du système dynamique.*

Preuve : On observe tout d'abord la relation

$$\bar{c} - \bar{c} \circ T^n = \bar{C}_n - \bar{C}_n \circ T.$$

Donc, pour tout $g \in L^2$, on vérifie

$$\begin{aligned} \int_X (\bar{c} - \bar{c} \circ T^n)\, g d\bar{\mu} &= \int_X \left(\bar{C}_n - \bar{C}_n \circ T\right) g d\bar{\mu} \\ &= \int_X \bar{C}_n g d\bar{\mu} - \int_X \mathbf{L}[g] \bar{C}_n d\bar{\mu}. \end{aligned}$$

en vertu de la relation (5.16). Puisque $L^2(X)$ est stable par $\mathbf{L}$, le lemme précédent s'applique et après passage à la limite on obtient la relation suivante pour tout $g \in L^2(X)$

$$\lim_{n\to\infty} \int_X (\bar{c} \circ T^n)\, g d\bar{\mu} = \int_X \bar{C} g d\bar{\mu} - \int_X \bar{C} \mathbf{L}[g] d\bar{\mu} - \int_X \bar{c} g d\bar{\mu}.$$

Considérons le membre gauche de cette relation. Si g est un élément de $\mathcal{F}$, alors la décomposition spectrale de $\mathbf{H}[g]$ existe et s'étend à $\mathbf{L}[g]$ et on obtient

$$\int_X (c \circ T^n)\, g d\bar{\mu} = \int_X \bar{c} \mathbf{N}^n[g] d\bar{\mu}, \tag{5.18}$$

puisqu'en effet

$$\int_X (c \circ T^n)\, g d\bar{\mu} = \int_X \bar{c} \mathbf{L}^n[g] d\bar{\mu}$$

le terme dominant étant donc avec (5.17)

$$\int_X \bar{c} \left(\int_X g d\bar{\mu} \right) d\bar{\mu} = 0.$$

On déduit de (5.18) que

$$\lim_{n\to\infty}\int_X (c\circ T^n)\, g d\bar{\mu} = 0, \tag{5.19}$$

puisque l'opérateur $\mathbf{N}$ a un rayon spectral strictement inférieur à 1. Si g n'appartient pas à $\mathcal{F}$, l'équation (5.19) reste valide, par densité de $\mathcal{F}$ dans $L^2(X)$. On en déduit donc que pour tout g dans $L^2(X)$,

$$\int_X \bar{c} g d\bar{\mu} = \int_X \bar{C}\mathbf{L}[g] d\bar{\mu} - \int_X g\bar{C} d\bar{\mu}.$$

Donc l'égalité

$$\bar{c} = \bar{C} - \bar{C}\circ T$$

est vérifiée dans $L^2(X)$.
Pour tout n, on a $\bar{C}_n - \bar{C}_{n-1}\circ T = \bar{c}$, de sorte que

$$\mathbf{L}[\bar{C}_{n-1}\circ T] = \mathbf{L}[\bar{C}_n] - \mathbf{L}[c].$$

D'un autre côté, on vérifie également

$$\mathbf{L}[\bar{C}_{n-1}\circ T] = \bar{C}_{n-1}.$$

Ceci entraîne $\bar{C} = \mathbf{L}[\bar{C}] - \mathbf{L}[\bar{c}]$ et donc

$$\bar{C} = -(I-\mathbf{L})^{-1}\circ\mathbf{L}[\bar{c}] = -(I-\mathbf{N})^{-1}\circ\mathbf{N}[\bar{c}],$$

de sorte que $\bar{C}$ appartient à $\mathcal{F}$.
Finalement l'égalité $\bar{c} = \bar{C} - \bar{C}\circ T$ est vérifiée là où $\bar{c}$ et $\bar{C}\circ T$ sont bien définis, c'est-à-dire à l'intérieur de chaque élément de la partition. ■

Il existe donc une fonction $u \in L^2(X)$ telle que l'égalité $\bar{c} = u - u\circ T$ est vérifiée à l'intérieur de chaque intervalle fondamental X_d. Si on considère maintenant l'ensemble des points fixes $h^\star$ de chaque branche, on a alors $\bar{c}(h^\star) = 0$. Puisque l'application c est constante sur chaque intervalle fondamental, on en déduit qu'elle est constante sur tout l'ensemble X. ■

Conclusion et exemples

Nous venons dans cette première partie de chapitre de faire une preuve générique d'analyse dynamique. Nous avons exhibé les différentes propriétés demandées aux quasi-inverses (pour le théorème taubérien) où à l'opérateur $\mathbf{H}^n_{s,w}$ (pour le théorème de Hwang), et expliqué quand c'était possible comment obtenir ces propriétés. Ces propriétés sont résumées dans le tableau 5.3.

Pour conclure cette première moitié de chapitre, nous résumons ce qui vient d'être dit dans les propositions suivantes.

La première est consacrée au théorème Taubérien : considérant une série de Dirichlet reliée à des quasi-inverses, nous précisons le comportement asymptotique de ses coefficients en fonction du nombre de quasi-inverses apparaissant et exhibons les constantes résiduelles. Nous appliquons

(**A1**)	L'opérateur $\mathbf{H}$ agit sur $\mathcal{F}$.
(**A2**)	L'opérateur $\mathbf{H}$ admet une unique valeur propre sur le cercle unité. Elle est simple et isolée du reste du spectre.
(**A3**)	$\mathbf{H}$ admet $\lambda = 1$ comme valeur propre dominante.
(**B0**)	Il existe un voisinage complexe W de $w = 0$ et $\beta < 1$ tel que $\mathbf{H}_{s,w}$ agit sur $\mathcal{F}$ pour $(s, w) \in \{\Re(s) \geq \beta\} \times W$
(**B1**)	L'application $(s, w) \mapsto \mathbf{H}_{s,w}$ est analytique dans un voisinage de $(s, w) = (1, 0)$.
(**C1**)	L'opérateur $\mathbf{H}_{s,0}$ a un rayon spectral strictement inférieur à 1 pour $\Re(s) \geq 1, s \neq 1$.
(**C2**)	$\Lambda''_s(1,0) > 0, \quad \Lambda(s,w) := \log \lambda(s,w)$.
(**C3**)	$\Lambda''_w(1,0) \neq 0 \quad \Lambda(s,w) := \log \lambda(s,w)$.
(**C4**)	$\lambda'_s(1,0) \neq 0$

FIG. 5.3 – Les différentes propriétés de l'opérateur

ensuite cette proposition à des exemples "jouets", qui nous serons utiles par la suite : le premier concerne les séries $T_1(s)$, permettant de générer les cardinaux des ensembles Ω_n, le second traite de l'approximation des coûts B, X liés à la complexité en bits par les coûts $\widehat{B}$, $\widehat{X}$.

Proposition 5.7 *Soit une série de Dirichlet $T(s) = \sum_{n\geq 1} t_n n^{-s}$, reliée à un opérateur de transfert $\mathbf{H}_{s,0}$, et à des opérateurs $\mathbf{Q}_s^{[i]}$ dépendant de s, de la manière suivante :*

$$T(s) = (I - \mathbf{H}_{s,0})^{-1} \circ \mathbf{Q}_s^{[1])} \circ (I - \mathbf{H}_{s,0})^{-1} \circ \mathbf{Q}_s^{[2]} \circ \cdots \circ (I - \mathbf{H}_{s,0})^{-1} \circ \mathbf{Q}_s^{[k]}[1](0)$$

Soit $\mathcal{F}$ un espace fonctionnel tel que :
(1) *$\mathbf{Q}_s^{[k]}$ est un opérateur borné sur $\mathcal{F}$, pour tout $1 \leq i \leq k$, analytique en s pour $\Re(s) > \beta, \beta < 1$,*
(2) *quand il agit sur $\mathcal{F}$, l'opérateur $\mathbf{H}_{s,0}$ satisfait les propriétés* (**A1**, **A2**, **A3**, **B0**, **B1′**, **C1**).

Alors les sommes $\sum_{n\leq N} t_n$ et $\sum_{\ell(n)=N} t_n$ sont asymptotiquement

$$\sum_{n\leq N} t_n \sim \frac{1}{(k-1)!} \cdot N \cdot \log^{k-1} N \cdot \left(\frac{1}{h(H)}\right)^k \cdot \psi(0) \cdot \prod_{i=1}^{k} \int_X \mathbf{Q}_1^{[i]}[\psi] d\mu$$

$$\sum_{\ell(n)=N} t_n \sim \frac{1}{(k-1)!} \cdot (2\log 2)^{k-1} \cdot 2^{2N-1} \cdot N^{k-1} \cdot \left(\frac{1}{h(H)}\right)^k \cdot \psi(0) \cdot \prod_{i=1}^{k} \int_X \mathbf{Q}_1^{[i]}[\psi] d\mu$$

où μ est la mesure de Haar définie sur X et ψ, $h(H)$ sont la densité invariante et l'entropie du système dynamique considéré.

Preuve : Les propriétés (**A1**, **A2**, **B1′**) de l'opérateur $\mathbf{H}_{s,0}$ sur $\mathcal{F}$ entraînent la décomposition spectrale de l'opérateur $\mathbf{H}$, qui s'étend dans un voisinage de $s = 1$ à $\mathbf{H}_{s,0}$, ainsi qu'à la série $T(s)$ qui finalement s'écrit autour de $s = 1$

$$T(s) = \left(\frac{\lambda(s,0)}{1-\lambda(s,0)}\right)^k \mathbf{P}_{s,0} \circ \mathbf{Q}_s^{[1]} \circ \cdots \circ \mathbf{P}_{s,0} \circ \mathbf{Q}_s^{[k]}[1](0) + \mathbf{R}_s[1](0), \tag{5.20}$$

où $\mathbf{R}_s[1](0)$ est un opérateur qui s'exprime à l'aide des quasi-inverses $(I - \mathbf{N}_{s,0})^{-1}$, et qui est donc analytique en s (grâce à la relation (5.7)). La condition (ii) du théorème taubérien est donc vérifiée, puisque de (5.20) et (**A3**) on déduit

$$T(s) \sim \frac{1}{(s-1)^k} \left(\frac{-1}{\lambda'_s(1,0)} \right)^k \mathbf{P}_{1,0} \circ \mathbf{Q}_1^{[1]} \circ \cdots \circ \mathbf{P}_{1,0} \circ \mathbf{Q}_1^{[k]}[1](0).$$

La condition (i) du théorème taubérien est également vérifiée grâce à la propriété (**C1**), qui garanti le comportement analytique sur la droite $\Re(s) = 1, s \neq 1$, et on peut donc appliquer le théorème à $T(s)$. Le résultat est finalement obtenu des valeurs spéciales de $\lambda'_s(s,0)$ et $\mathbf{P}_{s,0}$ en $s = 1$ (relations (5.11,5.5)). ■

Nous allons maintenant donner quelques exemples d'application de cette proposition, sur des exemples types. Le premier exemple traite de la série de Dirichlet $T_1(s)$, qui permet de générer les cardinaux $|\Omega_N|$. Cette série intervient dans tous les calculs de moyennes. Le second exemple traite de la complexité en bits. Il est intéressant dans la mesure où il fait intervenir des coûts de différentes nature : nous commençons dans la proposition 5.9 par étudier la valeur moyenne du coût à croissance modérée ℓ, correspondant à la taille des quotients. Cette étude nous permet de montrer que les coûts approximatifs pour la complexité en bits sont suffisants pour une analyse en moyenne. L'analyse de la complexité en bits est finalement faite dans la proposition 5.10. Ces trois propositions sont de "complexité" croissante, puisque dans la première on ne traite qu'un quasi-inverse, dans la seconde on en traite deux, et trois dans la dernière.

Proposition 5.8 *Soit H un algorithme euclidien, et Ω l'ensemble de ses entrées valides premières entre elles. Supposons que l'opérateur $\mathbf{H}_{s,0}$ associé au système dynamique sous-jacent à l'algorithme H satisfait les conditions* (**A1**, **A2**, **A3**, **B0**, **B1′**, **C1**) *quand il agit sur un espace de Banach $\mathcal{F}$. Alors asymptotiquement, $|\Omega_N|$ est donné par*

$$|\Omega_N| = 2^{2N-1} \cdot \left(\frac{1}{h(H)} \right) \cdot \psi(0),$$

où ψ est la densité invariante du système dynamique.

Preuve : Tout d'abord, rappelons que $|\Omega_N|$ est relié à la série $T_1(s)$ (associée au coût $C = 1$) par

$$T_1(2s) = \sum_{n \geq 1} \frac{t_n^{(1)}}{n^s} \qquad |\Omega_N| = \sum_{\ell(n)=N} t_n^{(1)}.$$

La série $T_1(s)$ est elle-même reliée à l'opérateur $\mathbf{H}_{s,0}$ par

$$T_1(2s) = (I - \mathbf{H}_{s,0})^{-1}[1](0).$$

Il suffit maintenant d'appliquer la proposition précédente. ■

Notons que le précédent résultat fait apparaître une relation simple entre $|\Omega_N|$, qu'on peut généralement calculer par d'autres méthodes, et les grandeurs caractéristiques du système dynamique, à savoir l'entropie et la valeur en 0 de la densité invariante. Remarquons également que les relations décrites au paragraphe 3.4 du chapitre 3 permettent d'obtenir les quantités $|\widetilde{\Omega}_N|$

Proposition 5.9 *Soit H un algorithme euclidien. Soit $\ell(d)$ la taille du quotient d, et $L(u,v)$ le coût total d'une entrée $(u,v) \in \Omega$ de l'algorithme, donné par*

$$L(u,v) = \sum_{i=0}^{p} \ell(d_i).$$

Si l'opérateur associé $\mathbf{H}_{s,0}$ vérifie les conditions $(\mathbf{A1}, \mathbf{A2}, \mathbf{A3}, \mathbf{B0}, \mathbf{B1}', \mathbf{C1})$ sur un espace de Banach $\mathcal{F}$, alors l'espérance $\mathbb{E}_N[L]$ est asymptotiquement linéaire en N,

$$\mathbb{E}_N[L] \sim \frac{2\log 2}{h(H)} \cdot \mu_H(\ell) \cdot N.$$

Preuve : Cette espérance est définie comme le rapport des sommes de coefficients de deux séries de Dirichlet, la première étant la série $T_L(s)$ donnée (voir proposition 4.2) par

$$T_L(2s) = (I - \mathbf{H}_{s,0})^{-1} \circ \mathbf{H}_{s,0}^{[\ell]} \circ (I - \mathbf{H}_{s,0})^{-1}[1](0),$$

et la seconde étant

$$T_1(2s) = (I - \mathbf{H}_{s,0})^{-1}[1](0),$$

étudiée dans la proposition précédente. En appliquant le théorème taubérien à ces deux séries (suivant la technique développée dans la proposition 5.7), on en déduit le résultat, en observant que (voir (5.12))

$$\int_X \mathbf{H}_{1,0}^{[\ell]}[\psi] d\mu = \mu_H(\ell).$$

■

Proposition 5.10 *Soit H un algorithme euclidien. Soit $B(u,v)$ la complexité en bits de l'algorithme sur entrée (u,v). Si l'opérateur associé $\mathbf{H}_{s,0}$ vérifie les conditions $(\mathbf{A1}, \mathbf{A2}, \mathbf{A3}, \mathbf{B0}, \mathbf{B1}', \mathbf{C1})$ sur un espace de Banach $\mathcal{F}$, alors l'espérance $\mathbb{E}_N[B]$ est asymptotiquement quadratique en N,*

$$\mathbb{E}_N[B] \sim \frac{\log^2 2}{h(H)} \cdot \mu_H(\ell) \cdot N^2. \tag{5.21}$$

Preuve : Étudions tout d'abord le coût approximatif $\widehat{B}$, défini en (1.24) par

$$\widehat{B}(u,v) := \sum_{i=1}^{p} \ell(d_i) \times \log_2(u_i).$$

La série génératrice $T_{\widehat{B}}(s)$ est reliée aux opérateurs par la relation de la proposition 4.3, qui est

$$T_{\widehat{B}}(2s) = (I - \mathbf{H}_{s,0})^{-1} \circ \Delta\mathbf{H}_{s,0} \circ (I - \mathbf{H}_{s,0})^{-1} \circ \mathbf{H}_{s,0}^{[\ell]} \circ (I - \mathbf{H}_{s,0})^{-1}[1](0).$$

La proposition 5.7 s'applique dans ce cas, et avec l'expression de la série $T_1(s)$ donnée par la proposition 5.8 on obtient finalement l'expression (5.21) pour l'espérance $\mathbb{E}_N[\widehat{B}]$.

Si on utilise l'inégalité (1.26), qui relie B, $\widehat{B}$ et ℓ on en déduit

$$\mathbb{E}_N[B] - \mathbb{E}_N[\widehat{B}] = O(N),$$

puisque d'après la proposition 5.9 la taille moyenne des quotients est linéaire en N. On en déduit donc qu'asymptotiquement on vérifie

$$\mathbb{E}_N[B] \sim \mathbb{E}_N[\widehat{B}].$$

■

Notons que cette dernière proposition n'est valable que lorsqu'on dispose d'une bonne relation entre le coût réel et le coût approximatif. C'est généralement le cas pour la complexité en bits d'un algorithme "simple", mais ce n'est pas toujours vrai pour les versions étendues. Nous développerons ce point lors de l'étude des algorithmes interrompus.

Nous suivons la même démarche pour les séries génératrices des moments dans la proposition 5.11.

Proposition 5.11 *Soit un système dynamique $S = (X, T)$. Soit la série génératrice des moments $\mathbb{E}(\exp(wC_n))$ associée à un coût à croissance modérée c. Supposons que cette série est reliée à un opérateur $\mathbf{H}_{1,w}$ par*

$$\mathbb{E}(\exp(wC_n)) = \int_X \mathbf{H}_{1,w} d\mu.$$

Alors si quand il agit sur l'espace fonctionnel $\mathcal{F}$, l'opérateur vérifie les conditions $(\mathbf{A1}, \mathbf{A2}, \mathbf{A3}, \mathbf{B1''}, \mathbf{C3})$, la variable C_n suit alors asymptotiquement une loi gaussienne, et ses moyennes et variances sont données par

$$\mathbb{E}[C_n] = n\mu_S(c) + \nu_1 + O(\kappa_n^{-1})$$
$$\mathbb{V}[C_n] = n\Lambda''_w(1,0) + \nu_2 + O(\kappa_n^{-1})$$

où $\mu_S(c)$ est la valeur moyenne du coût c, Λ est l'application $\Lambda(s,w) = \log\lambda(s,w)$, et ν_1, ν_2 sont les valeurs en 0 des dérivées premières et secondes de l'application $w \mapsto \int_X \log \mathbf{P}_{1,w}[1]d\mu$.

Preuve : Ici encore, les propriétés $(\mathbf{A1}, \mathbf{A2}, \mathbf{B1'})$ entraînent la décomposition suivante de l'opérateur $\mathbf{H}^n_{1,w}$ autour de $w = 0$:

$$\mathbf{H}^n_{1,w}[1](0) = \lambda^n(1,w)\mathbf{P}_{1,w}[1](0) + \mathbf{N}^n_{1,w}[1](0).$$

On peut donc écrire $\mathbb{E}[\exp(wC_n)]$ de la même manière qu'en (5.10), c'est-à-dire

$$\mathbb{E}[\exp(wC_n)] = \exp\left[n\log\lambda(1,w) + \log\int_X \mathbf{P}_{1,w}[1]d\mu\right](1 + O(\tau^n))$$

où τ est strictement inférieur à 1. La condition $(\mathbf{C3})$ entraîne qu'on peut appliquer le théorème des quasi-puissances de Hwang avec

$$U(w) = \log\lambda(1,w), \qquad V(w) = \log\int_X \mathbf{P}_{1,w}[1]d\mu, \qquad \kappa = \tau.$$

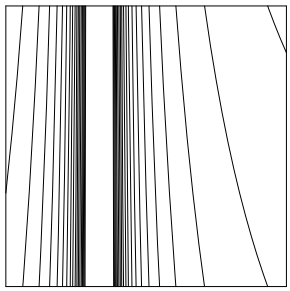

FIG. 5.4 – Un système α-euclidien

La valeur de $\Lambda(w) = \log \lambda(1, w)$ en $w = 0$ (5.13) implique le résultat final. ■

Nous venons d'énoncer les principales propriétés qui doivent être satisfaites par les opérateurs pour qu'on puisse leur appliquer le théorème taubérien et la théorème des quasi-puissances. Lorsque c'était possible, nous avons également décrit les techniques pouvant être utilisées pour obtenir de telles propriétés. La fin de l'analyse diffère maintenant selon le type de système qu'on étudie, et donc selon l'espace fonctionnel choisi. Nous expliquons dans les paragraphes suivants quels espaces associer à chaque système, pour pouvoir conclure les analyses.

5.2 Analyse dynamique des algorithmes α-euclidiens

Nous concluons dans ce paragraphe l'analyse des algorithmes α-euclidiens. Nous procédons en deux temps. Tout d'abord nous montrons que sous certaines conditions, l'opérateur de transfert associé à un système dynamique de l'intervalle à branches incomplètes satisfait l'ensemble des conditions décrites dans le paragraphe précédent. Nous énonçons ce résultat dans le théorème 5.12. Puis nous montrons que quand $\alpha > 0$, les systèmes α-euclidiens vérifient toutes les propriétés requises, et nous en déduisons les principaux résultats. Les opérateurs utilisés dans ce paragraphe sont relatifs à des systèmes dynamiques de l'intervalle simples, ils sont donc notés $\mathbf{G}_{s,w}$.

5.2.1 Analyse fonctionnelle pour l'opérateur $\mathbf{G}_{s,w}$

Rappelons tout d'abord brièvement le cadre général. Un système α-euclidien est défini sur un intervalle $I_\alpha = [\alpha - 1, \alpha]$ et possède usuellement deux branches incomplètes (voir figure 5.4). L'opérateur correspondant est noté $\mathbf{G}_{s,w}$, et est donné par (voir (4.15))

$$\mathbf{G}_{s,w}[f](x) := \sum_{h \in \mathcal{H}} |h'(x)|^s \cdot \exp(wc(h)) \cdot f \circ h(x) \cdot \mathbf{1}_{J_h}(x),$$

où les intervalles Y_h sont donnés dans (2.11). Ainsi que nous l'avons déjà observé, il n'est pas possible ici d'utiliser des espaces de fonctions continues. Il existe cependant des espaces fonctionnels particulièrement adaptés à la gestion des discontinuités apportées par le système, lorsqu'on dispose d'un contrôle suffisant sur ces discontinuités. Traditionnellement, on considère l'espace

des fonctions à variation bornée. Ce type d'analyse est essentiellement basé sur les travaux de Lasota et Yorke [LY73], et est bien résumé par exemple dans la monographie de Collet [Col96] ou dans [BG97]. Ces auteurs ont en particulier développé la technique générique présentée en introduction de ce paragraphe, qui consiste à utiliser des inégalités de Lasota-Yorke pour ensuite appliquer les théorèmes de Ionescu-Tulcea et Marinescu ou d'Hennion. Ces analyses sont toutes relatives à des systèmes possédant une partition finie, et il a donc fallu les adapter au cas particulier des systèmes S_α.

Nous commençons par définir et décrire l'espace BV des fonctions à variation bornée. Puis nous montrons que si un système dynamique de l'intervalle satisfait l'ensemble Σ_1 de conditions énoncées plus bas, alors l'opérateur de transfert associé $\mathbf{G}_{s,0}$ vérifie les conditions $(\mathbf{A1}, \mathbf{A2}, \mathbf{A3}, \mathbf{B0}, \mathbf{B1}', \mathbf{C1})$, conditions suffisantes pour l'application du théorème taubérien. Ce résultat est résumé dans le théorème 5.12.

Soit I un intervalle réel. Rappelons que l'espace $L^1(I)$ désigne l'espace des fonctions intégrables sur l'intervalle I et que muni de la norme $||.||_{L_1}$ donnée par

$$||f||_{L_1} = \int_I |f| d\mu,$$

c'est un espace de Banach.
On définit la variation $\bigvee_a^b f$ d'une fonction f sur l'intervalle $[a,b]$ par

$$\bigvee_a^b f = \sup_{\pi \in \mathcal{Q}} \sum_{i=1}^{n} |f(x_i) - f(x_{i-1})|,$$

la somme étant faite sur l'ensemble $\mathcal{Q}$ des partitions π de $[a,b]$ de la forme $a = x_0 < x_1 < \ldots < x_n = b$. L'espace des fonctions dont la variation sur un intervalle I est bornée est noté $BV(I)$,

$$BV(I) := \left\{ f, \quad \bigvee_I f < \infty \right\}.$$

Muni de la norme $||.||_{BV}$ définie par

$$||f||_{BV} = \bigvee_I f + ||f||_{L_1},$$

alors $BV(I)$ est un espace de Banach, dense dans $L^1(I)$. De plus, la boule unité de $BV(I)$ est précompacte dans la boule unité de L^1. On se retrouve donc dans les conditions d'application du théorème d'Hennion. Nous allons ensuite utiliser plusieurs propriétés importantes de la variation, que nous rappelons dès maintenant.

$$(i) \bigvee_a^b (f+g) \leq \bigvee_a^b f + \bigvee_a^b g, \qquad (ii) \bigvee_a^b f + \bigvee_b^c f = \bigvee_a^c f, \qquad (iii) \bigvee_a^b (f \circ g) = \bigvee_c^d f, \quad \text{si } g([a,b]) = [c,d],$$

$$(iv) \bigvee_a^b |fg| \leq \sup_{[a,b]} |g| \bigvee_a^b |f| + \sup_{[a,b]} |f| \bigvee_a^b |g|$$

$$(v) \bigvee_a^b |f| \mathbf{1}_{[c,d]} \leq 2 \bigvee_c^d |f| + 2 \sup_{[c,d]} |f| \quad \text{pour} \quad [c,d] \subset [a,b]$$

$$(vi)||f||_\infty \leq \bigvee_a^b |f| + \frac{1}{b-a}||f||_{L_1}, \qquad (vii)\bigvee_a^b |f| = \int_a^b |f'(x)|dx \text{ pour } f \in C^1.$$

Pour adapter les travaux déjà réalisés dans le cadre de systèmes à partition finie, nous avons dû imposer certaines conditions de "contrôle" sur les discontinuités. Considérons un système dynamique de l'intervalle S, défini sur un intervalle I, $S = (I, T)$. Si $\mathcal{H}$ désigne l'ensemble des branches inverses, nous désignerons indifféremment par $I_d, d \in \mathcal{D}$ ou $I_h, h \in \mathcal{H}$ les éléments de la partition et par J_d ou J_h les intervalles images. Les conditions supplémentaires pour traiter les partitions infinies forment l'ensemble Σ_1 ci-dessous.

Conditions Σ_1

– $(p0)$ *Il existe une constante réelle $b > 0$, et un exposant positif $\beta > 0$ tels que*

$$|h'(x)| \geq b|h(x)|^\beta, \quad \forall h \in \mathcal{H}, \forall x \in J_h.$$

– $(p1)$[Faible dilatation] *Le système dynamique est faiblement dilatant : Δ_1, défini par*

$$\delta_h := \sup\{|h'(x)|; x \in J_h\}, \qquad \Delta_n := \sup\{\delta_h; h \in \mathcal{H}^n\},$$

satisfait $\Delta_1 \leq 1$.

– $(p2)$ [Forte dilatation] *Le système dynamique est fortement dilatant : il existe un entier n_0 et une constante réelle $\gamma < 1$ tels que $\Delta_{n_0} \leq \gamma$.*

– $(p3)$ [Distorsion bornée] *Le système dynamique est à distortion bornée : il existe une constante réelle $c > 0$ telle que*

$$|h''(x)| \leq c|h'(x)|, \quad \forall h \in \mathcal{H}, \ \forall x \in J_h.$$

– $(p4)$ [Quasi-Markov] *Le système dynamique est quasi-markovien : toutes les quantités ℓ_n définies par*

$$\ell_n := \inf\{|J_h|; h \in \mathcal{H}^n, |J_h| > 0\},$$

sont strictement positives.

– $(p5)$ [Mélange topologique] *Le système dynamique est topologiquement mélangeant : pour tous ouverts (V, W) de X, il existe $n_2 \geq 1$ tel que pour tout $n \geq n_2$, $T^{-n}V \cap W \neq \emptyset$.*

Remarques Faisons tout d'abord quelques remarques sur ces différentes conditions. Un système possédant un nombre fini de branches satisfait automatiquement les propriétés $(p3)$ et $(p4)$. La propriété $(p4)$ généralise la notion de système markovien : un système dynamique de l'intervalle est markovien d'ordre 1 si pour tout $h \in \mathcal{H}$, l'intervalle J_h est une réunion d'intervalles $I_{h'}$ de la partition (cette notion se généralise naturellement à des ordres supérieurs en raffinant la partition, ainsi qu'à des systèmes non définis sur des intervalles). En particulier, un système markovien à partition finie satisfait donc la propriété $(p4)$. Remarquons de plus que si un système satisfait les propriétés $(p1)$ et $(p3)$, alors il en va de même de ses itérées. Plus précisément, la constante c_n alors obtenue avec la propriété $(p3)$ satisfait $c_n \leq cn$. Enfin, si un système satisfait à la fois les conditions $(p1)$, $(p2)$ et $(p3)$, alors ses itérées aussi, et les constantes obtenues vérifient pour tout n (voir [LM94] ou [BG97])

$$\Delta_n \leq \gamma^{\lfloor \frac{n}{n_0} \rfloor}, \qquad c_n = c\sum_{i=0}^{n-1} \Delta_i \leq \frac{cn_0}{1-\gamma}. \tag{5.22}$$

Soit un réel ρ. Alors les propriétés $(p1)$ (pour $\rho \geq 1$) et $(p0)$ (pour $\rho < 1$) entraînent

$$|h'(x)|^\rho \leq |h'(x)| \qquad \text{pour } \rho \geq 1, \qquad |h'(x)|^\rho \leq b^{\rho-1}|h'(x)|\,|h(x)|^{\beta(\rho-1)} \qquad \text{pour } \rho < 1. \quad (5.23)$$

Le changement de variables $u := h(x)$ nous amènent à définir

$$I(\rho) := 1 \qquad \text{pour } \rho \geq 1, \qquad I(\rho) := b^{\rho-1} \int_I |u^{\beta(\rho-1)}| du \qquad \text{pour } \rho < 1, \quad (5.24)$$

et puisque l'intégrale $I(\rho)$ est convergente pour $\rho = \Re(s) > 1 - (1/\beta)$, on obtient, pour $\rho = \Re(s) > 1 - (1/\beta)$

$$\sum_{h \in \mathcal{H}} \int_{J_h} |h'(x)|^\rho \, dx \leq I(\rho). \quad (5.25)$$

Enfin, nous définissons la propriété $(p6)$ (voir [LM94] ou [BG97]), automatiquement vérifiée dès que l'est $(p3)$,

$(p6)$ *Il existe $d > 0$ tel que, pour tout $h \in \mathcal{H}$, on a $\sup_{J_h} |h'(x)| \leq d \inf_{J_h} |h'(x)|$.*

La condition $(p6)$ implique, avec $(p4)$, que pour tout $h \in \mathcal{H}$ et pour tout réel positif ρ,

$$\sup_{J_h} |h'(x)|^\rho \leq d^\rho \inf_{J_h} |h'(x)|^\rho \leq \frac{d^\rho}{|J_h|} \int_{J_h} |h'(x)|^\rho dx \leq \frac{d^\rho}{\ell_1} \int_{J_h} |h'(x)|^\rho dx, \quad (5.26)$$

et on obtient donc avec (5.26) et (5.25)

$$\sum_{h \in \mathcal{H}} \delta_h^\rho \leq \frac{d^\rho}{\ell_1} I(\rho). \quad (5.27)$$

Les résultats que nous obtiendrons dans ce paragraphe sont résumés dans le théorème suivant.

Théorème 5.12 *Soit $S = (I, T)$ un système dynamique de l'intervalle. S'il satisfait l'ensemble Σ_1 de conditions, alors quand il agit sur l'espace $BV(I)$ l'opérateur de transfert $\mathbf{G}_{s,w}$ vérifie les conditions* **(A1)**, **(A2)**, **(A3)**, **(B0)**, **(B1)**, **(C1)** *et* **(C3)**.

Preuve du théorème 5.12

La preuve du théorème 5.12 est une succession de propositions.

Proposition 5.13 *Soit $S = (I, T)$ un système dynamique de l'intervalle satisfaisant les propriétés $(p0)$–$(p4)$. Alors pour $\Re(s) > 1 - (1/\beta)$ l'opérateur $\mathbf{G}_{s,0}$ correspondant agit sur $BV(I)$ et est analytique en s. Il satisfait donc les propriétés* $(\mathbf{A1}, \mathbf{B1}')$.

Preuve : Soit une fonction f à variation bornée, $f \in BV(I)$. Soit $s \in \mathbb{C}$, $\Re(s) = \rho > 1 - (1/\beta)$. Alors on vérifie

$$||\mathbf{G}_{s,0}[f]||_{L_1} \leq \sum_{h \in \mathcal{H}} \int_I |h'^s(x)\, f \circ h(x) \;\; \mathbf{1}_{J_h}(x)| dx \leq \sum_{h \in \mathcal{H}} \int_{J_h} |h'(x)|^\rho \, |f \circ h(x)| \, dx.$$

En utilisant la relation (5.25) et la propriété (vi) des fonctions à variation bornée, on en déduit dans un premier temps

$$||\mathbf{G}_{s,0}[f]||_{L_1} \leq I(\rho)||f||_\infty \leq I(\rho)\; ||f||_{BV}. \quad (5.28)$$

D'autre part la variation de $\mathbf{G}_{s,0}[f]$ vérifie

$$\bigvee_I \mathbf{G}_{s,0}[f] = \bigvee_I \sum_{h\in\mathcal{H}} h'^s\, f\circ h\, \mathbf{1}_{J_h} \le \sum_{h\in\mathcal{H}} \bigvee_I |h'^\rho\, f\circ h\ \mathbf{1}_{J_h}|.$$

En appliquant (iv) et (v) à chaque terme de la somme on obtient finalement

$$\bigvee_I \mathbf{G}_{s,0}[f] \le A+B+C \qquad \text{avec} \qquad A=\sum_{h\in\mathcal{H}} A_h,\ \ B=\sum_{h\in\mathcal{H}} B_h,\ \ C=\sum_{h\in\mathcal{H}} C_h,$$

$$\text{et} \qquad A_h=\delta_h^\rho \bigvee_{J_h} |f\circ h|, \qquad B_h=4\delta_h^\rho \sup_{I_h}|f|, \qquad C_h=\sup_{I_h}|f| \bigvee_{J_h} |h'^\rho|.$$

Les propriétés (ii) et (iii) des fonctions à variation bornée et la propriété $(p1)$ entraînent $A_h \le \bigvee_{I_h}|f|$ et donc $A\le \bigvee_I |f|$. Pour la somme C_h, on utilise la propriété (5.25), en remarquant que $(p3)$ implique

$$|h'^\rho|' = \rho|h''||h'|^{\rho-1} \le c\rho\,|h'|^\rho.$$

On en déduit finalement

$$\bigvee_I \mathbf{G}_{s,0}[f] \le [1+(4\frac{d^\rho}{\ell_1}+c\rho)I(\rho)]\ ||f||_{BV},$$

dont on déduit la première partie de la proposition.
La dérivée de l'opérateur par rapport à s est l'opérateur $\mathbf{R}_s$, donné par

$$\mathbf{R}_s[f](x)=\sum_{h\in\mathcal{H}} |h'(x)|^s\,\log|h'(x)|\ f\circ h(x)\,\mathbf{1}_{J_h}(x).$$

Comme précédemment, on utilise les inégalités suivantes :

$$(|h'^\rho|\,\log|h'|)' \le |h'^\rho|\,\frac{|h''|}{|h'|}+|s||h''||h'|^{\rho-1}\ |\log|h'|| \le c\,(|\rho|+|\log|h'||)\,|h'|^\rho,$$

ce qui amène à considérer l'intégrale

$$J(\rho)=\int_I |u^{\beta(\rho-1)}|\log|u|du.$$

Cette intégrale est convergente pour $\Re(s)\ge 1-(1/\beta)$, et donc l'application $s\to \mathbf{G}_{s,0}$ est analytique dans ce demi-plan. ■

On montre maintenant que l'opérateur de Perron-Frobenius est quasi-compact, en exhibant une inégalité de Lasota-Yorke.

Proposition 5.14 *Soit S un système dynamique satisfaisant les propriétés $(p1)-(p4)$. Alors il existe n_0 tel que pour tout $n\ge n_0$*

$$||\mathbf{G}^n[f]||_{BV} \le 2\Delta_n\,||f||_{BV}+\left(2\,c_n+\frac{2}{\ell_n}+1\right)||f||_{L_1}, \qquad \textit{avec} \qquad \Delta_n\le\gamma^{\lfloor\frac{n}{n_0}\rfloor},\ c_n\le\frac{cn_0}{1-\gamma}. \tag{5.29}$$

Donc le rayon spectral essentiel r_{ess} satisfait $r_{ess}\le\gamma^{1/n_0}$.

Preuve : C'est en particulier pour cette proposition qu'il faut adapter les travaux antérieurs aux systèmes à partition dénombrable. Nous utiliserons quelques variantes des propriétés (iv) et (v) des fonctions à variation bornée, qui relient la variation d'une fonction à sa norme $||.||_{L_1}$:

$$(iv') \bigvee_a^b |fg| \leq \sup_{[a,b]} |g| \bigvee_a^b |f| + \int_a^b |f(x)g'(x)|dx$$

$$(v') \bigvee_a^b |f|\mathbf{1}_{[c,d]} \leq 2 \bigvee_c^d f + \frac{2}{d-c} \int_c^d |f(x)|dx \quad \text{pour} \quad [c,d] \subset [a,b]$$

On vérifie tout d'abord

$$||\mathbf{G}[f]||_{L_1} \leq \sum_{h\in\mathcal{H}} \int_I |h'(x)\, f\circ h(x)\ \mathbf{1}_{J_h}(x)|dx.$$

Le changement de variable $u := h(x)$ entraîne

$$||\mathbf{G}[f]||_{L_1} \leq \sum_{h\in\mathcal{H}} \int_{I_h} |f(u)|du = ||f||_{L_1}. \tag{5.30}$$

La variation de la fonction $\mathbf{G}^n[f]$, $n \geq 1$ est donnée par

$$\bigvee_I \mathbf{G}^n[f] = \sum_{h\in\mathcal{H}^n} \bigvee_I |h'\, f\circ h\ \mathbf{1}_{J_h}|.$$

En appliquant (iv') et (v') à chaque terme de la somme, on obtient

$$\bigvee_I \mathbf{G}^n[f] \leq A + B + C \qquad \text{avec} \qquad A = \sum_{h\in\mathcal{H}^n} A_h, \quad B = \sum_{h\in\mathcal{H}^n} B_h, \quad C = \sum_{h\in\mathcal{H}^n} C_h,$$

$$\text{et} \qquad A_h = 2\ \sup_{J_h} |h'| \bigvee_{J_h} |f\circ h|, \qquad B_h = 2\int_{J_h} |h''(x)|\, |f\circ h(x)|dx,$$

$$C_h = \frac{2}{|J_h|} \int_{J_h} |h'(x)|\, |f\circ h(x)|dx.$$

La quantité A_h vérifie, grâce à (ii) et (iii)

$$A_h \leq 2\Delta_n \bigvee_{I_h} |f| \tag{5.31}$$

et donc A est bornée,

$$A \leq 2\Delta_n \bigvee_I |f|.$$

La propriété $(p3)$ implique l'inégalité $|h''| \leq c_n|h'|$ et

$$B_h + C_h \leq 2(c_n + \frac{1}{\ell_n}) \int_{J_h} |h'(x)|\, |f\circ h(x)|dx.$$

En utilisant le changement de variables $u := h(x)$ et en sommant sur l'ensemble $\mathcal{H}$, on obtient avec (ii)

$$B + C \leq 2\ (c_n + \frac{1}{\ell_n})\ ||f||_{L_1}. \tag{5.32}$$

Avec (5.22), (5.30), (5.31) et (5.32), l'inégalité (5.29) est prouvée. ■

Il s'agit maintenant de préciser les propriétés spectrales dominantes de l'opérateur $\mathbf{G}$, à savoir la nature de son spectre sur le cercle unité. Remarquons déjà que l'on sait que $r_0(\mathbf{G}) \leq 1$: en effet, l'inégalité $||\mathbf{G}||_{L_1} \leq 1$ implique (5.1) que le rayon spectral de $\mathbf{G}$ sur $L^1(I)$ est inférieur ou égal à 1. Il en est donc de même sur $BV(I)$. Nous précisons la nature du spectre sur le cercle unité dans la proposition suivante.

Proposition 5.15 *Soit S un système dynamique satisfaisant les propriétés $(p1)--(p4)$. Alors l'opérateur $\mathbf{G}$ associé admet une densité invariante à variation bornée, et son rayon spectral est donc égal à 1. La condition* (**A3**) *est donc vérifiée.*

Preuve : Les propriétés $(p1)$, $(p2)$ ainsi que la relation (5.22) impliquent que la suite (Δ_n) de la proposition précédente tend vers 0. Pour tout $\delta < 1$ fixé, il existe donc un entier n_1 pour lequel $2\Delta_{n_1} \leq \delta$. Donc, la norme de $\mathbf{G}^{n_1}[f]$ satisfait

$$||\mathbf{G}^{n_1}[f]||_{BV} \leq \delta||f||_{BV} + L||f||_{L_1},$$

où L est une quantité finie. Donc, pour tout $n \geq 1$, la norme de $\mathbf{G}^{nn_1}[f]$ vérifie

$$||\mathbf{G}^{nn_1}[f]||_{BV} \leq \delta^n||f||_{BV} + \frac{L}{1-\delta}||f||_{L_1}.$$

L'ensemble $\{\mathbf{G}^{nn_1}[\mathbf{1}_I], n \geq 0\}$ est donc un ensemble borné de $BV(I)$. Il en est de même de l'ensemble

$$\mathcal{F} := \{\ f_n = \frac{1}{n}\sum_{j=0}^{n-1} \mathbf{G}^{jn_1}[\mathbf{1}_I]\ , n \geq 1\}.$$

Le théorème d'Helly s'applique alors, et il existe donc une sous-suite de $\mathcal{F}$ qui converge dans $L^1(I)$ vers une fonction $f^\star$ de $BV(I)$. Chaque élément de $\mathcal{F}$ étant une densité, $f^\star$ en est une également. De plus il est clair que $\mathbf{G}^{n_1}[f^\star] = f^\star$, de sorte que $f^\star$ est une densité invariante à variation bornée pour l'opérateur $\mathbf{G}^{n_1}$. Finalement, la densité

$$g^* := \frac{1}{n_1}\sum_{j=0}^{n_1-1} \mathbf{G}^j[f^*]$$

est une densité invariante pour $\mathbf{G}$, à variation bornée. ■

Nous avons obtenu la quasi-compacité, il faut donc maintenant étudier le spectre sur le cercle unité pour finalement obtenir la condition (**A2**). Nous utilisons ici le résultat classique suivant, qu'on trouve par exemple dans le livre de Baladi [Bal00].

Proposition *Soit un système dynamique dont le transformateur de densité est quasi-compact et possède une densité invariante sur $BV(I)$. Si le système est topologiquement mélangeant, alors la valeur propre $\lambda = 1$ est simple et est la seule valeur propre de module égal à 1.*

Nous avons montré que cette proposition s'applique dans notre cadre. On a donc la proposition suivante.

Proposition 5.16 *La condition* (**A2**) *est vérifiée.*

On obtient donc une décomposition spectrale de $\mathbf{G}$ qui s'étend par perturbation à l'opérateur de transfert $\mathbf{G}_{s,0}$ puisque la propriété d'analycité est vérifiée. Il ne reste plus maintenant qu'à étudier le comportement de l'application $s \mapsto \lambda(s,0)$ quand s vit dans un demi-plan complexe.

Proposition 5.17 *Dans un voisinage de $s = 1$, la fonction $s \mapsto \lambda(s,0)$ est strictement décroissante sur l'axe réel.*

Preuve : La valeur propre dominante $\lambda(s,0)$ de $\mathbf{G}_{s,0}$ est donnée par

$$\lambda(s,0) = \lim_{k\to\infty} [\mathbf{G}_{s,0}^k[1](0)]^{1/k}.$$

D'un autre coté,

$$\mathbf{G}_{s,0}^k[1](0) \leq \sup_{h\in\mathcal{H}^k} \delta_h^{(s-1)}\ \mathbf{G}^k[1](0) \leq \gamma^{k(s-1)}\mathbf{G}^k[1](0).$$

On en déduit donc l'inégalité (pour $s > 1$)

$$\lambda(s,0) \leq \gamma^{(s-1)/n_0}\lambda(1,0).$$

■

Proposition 5.18 *Sur la droite $\Re(s) = 1, s \neq 1$, les valeurs propres de l'opérateur $\mathbf{G}_{s,0}$ sont toutes de module strictement inférieur à 1. De plus, on vérifie également $r_{ess}(\mathbf{G}_{s,0}) < 1$. La condition* (**C1**) *est donc vérifiée.*

Preuve : Supposons que le rayon spectral de l'opérateur $\mathbf{G}_{1+it,0}^n$ n'est pas strictement inférieur à 1. De l'inégalité

$$||\mathbf{G}_{1+it,0}^n||_{BV} \leq ||\mathbf{G}_{1,0}||_{BV}$$

et de l'inégalité de la proposition 5.14, on déduit que l'opérateur est quasi-compact, avec un rayon spectral essentiel strictement inférieur à 1, et donc que la partie dominante du spectre est constituée de valeurs propres.

Soit λ une valeur propre de l'opérateur $\mathbf{G}_{1+it,0}$, et soit f une fonction propre relative à λ. Soit f_0 une fonction propre relative à la valeur propre 1. Il a été prouvé dans [Bal00] et [BG97] qu'une telle fonction peut-être choisie semi-continue inférieure, de sorte que $f_0(x) \geq a > 0$, pour un certain a et pour tout $x \in I$. On peut de plus supposer que la fonction $f(x)/f_0(x)$ est de module au plus 1 sur I, et est de module exactement 1 en x_0. On a alors

$$|\lambda\, f(x_0)| = |\mathbf{G}_{1+it,0}[f](x_0)| = |\sum_{h\in\mathcal{H}} h'(x_0)^{1+it}\ f\circ h(x_0)|\, \mathbf{1}_{J_h}(x_0) \tag{5.33}$$

$$\leq |\sum_{h\in\mathcal{H}} |h'(x_0)|\ |f\circ h(x_0)|\, \mathbf{1}_{J_h}(x_0) \leq \sum_{h\in\mathcal{H}} |h'(x_0)|\ f_0\circ h(x_0)\mathbf{1}_{J_h}(x_0) = f_0(x_0), \tag{5.34}$$

et la définition de x_0 prouve l'inégalité $|\lambda| \leq 1$. Supposons maintenant que $|\lambda| = 1$. Alors, la suite d'inégalités (5.33,5.34) devient une suite d'égalités, et pour tout $h \in \mathcal{H}$ de sorte que x_0 appartienne à J_h on a

$$|f\circ h(x_0)| \ = \ f_0\circ h(x_0). \tag{5.35}$$

D'un autre côté, la suite $a_h := |h'(x_0)| f \circ h(x_0) \mathbf{1}_{J_h}(x_0)$ vérifie l'égalité $|\sum a_h| = \sum |a_h|$. Il existe donc θ (de module 1) tel que $a_h = \theta |a_h|$ pour tout h, et pour tout h tel que x_0 appartienne à Y_h on a

$$f \circ h(x_0) \ |h'(x_0)|^{it} = \ \theta \ |f \circ h(x_0)|. \tag{5.36}$$

Pour tout $x_0 \in I$, l'ensemble $\{h(x_0), h \in \mathcal{H}\}$ contient la suite $\{1/(q + x_0), q \geq q_0\}$. Cette suite a pour limite 0, de sorte que l'égalité (5.35) prouve que $\lim_{x\to 0} |f(x)| \ = \lim_{x\to 0} f_0(x) \neq 0$. Finalement, la relation (5.36) montre que la suite

$$\left(\frac{1}{q + x_0}\right)^{it}$$

a une limite égale à θ quand $q \to \infty$, ce qui ne peut être vrai que pour $t = 0$. On a donc montré la première partie de la proposition.

■

Nous avons étudié l'opérateur de transfert $\mathbf{G}_{s,0}$. Quand on s'intéresse aux trajectoires tronquées du système dynamique, on étudie le comportement de l'opérateur $\mathbf{G}_{1,w}$. Il y a deux choses à montrer ici, c'est-à-dire les propriétés (**B2′**) et (**C2**). Tout d'abord, il faut vérifier l'analycité en w, ce qui est presque immédiat si le coût est à croissance modérée.

Proposition 5.19 *Il existe un voisinage complexe de $w = 0$ tel que l'opérateur $\mathbf{G}_{1,w}$ dépende analytiquement de w. Les conditions* (**B0**, **B1″**) *sont donc satisfaites.*

Preuve : Rappelons que le coût c apparaissant dans l'expression de $\mathbf{G}_{1,w}$ étant à croissance modérée, la série

$$\sum_{h\in\mathcal{H}} \delta_h \exp(wc(h)),$$

où $\delta_h = \sup_{x\in I} |h'(x)|$, est analytique en w. Notons $\mathbf{R}_w[f]$ l'opérateur dérivé de $\mathbf{G}_{1,w}$:

$$\mathbf{R}_w[f](x) := \sum_{h\in\mathcal{H}} c(h)|h'(x)| \exp(wc(h)) f \circ h(x) \mathbf{1}_h(x).$$

Alors clairement

$$||\mathbf{R}_w||_{L_1} \leq ||f||_{L_1} \sum_{h\in\mathcal{H}} \delta_h c(h) \exp(wc(h)),$$

et

$$\bigvee_I \mathbf{R}_w[f] \leq \sum_{h\in\mathcal{H}} c(h) \exp(wc(h)) \bigvee_I |h'(x)| f \circ h(x) \mathbf{1}_h(x).$$

Clairement, ces deux séries convergent pour w suffisamment proche de 0 (pour la seconde série, il suffit de s'inspirer de la preuve de la proposition 5.13). ■

Enfin, on doit s'assurer que la dérivée seconde de $w \mapsto \log \lambda(1, w)$ est non-nulle pour pouvoir appliquer le théorème des quasi-puissances. D'après la proposition 5.3 de la première partie du chapitre, cette propriété est vérifiée dès que le coût étudié est non-nul. Remarquons que le dernier

argument utilisé dans la preuve de cette proposition n'est à priori valide que pour un système à branches complètes : on montre que pour tout point fixe $h^\star$ d'une branche inverse, on a l'égalité $\bar{c}(h^\star) = 0$. Quand toutes les branches possèdent un point fixe, puisque $\bar{c}$ représente l'écart entre le coût c et sa valeur moyenne et que c est constant sur chaque intervalle fondamental, on en déduit le résultat. Le résultat reste cependant valable lorsqu'une seule branche ne possède pas de point fixe, ce qui est le cas pour les algorithmes α-euclidiens (il s'agit de la branche située à l'extrémité gauche de l'intervalle I_α), et donc la proposition suivante est montrée.

Proposition 5.20 *On a $\Lambda''_w(1,0) \neq 0$. La condition* (**C3**) *est donc vérifiée.*

5.2.2 Théorèmes 5.22, 5.23 et 5.24

L'analyse présentée plus haut s'applique aux systèmes α-euclidiens, pour $\alpha \neq 0$. En effet, l'ensemble de conditions Σ_1 est vérifié.

Proposition 5.21 *Soit S_α un système α-euclidien, $\alpha > 0$. Alors S_α satisfait l'ensemble Σ_1 de conditions.*

Preuve : Pour tout $\alpha \in [0,1]$, les branches inverses $h \in \mathcal{H}$ vérifient

$$|h'(x)| = |h(x)|^2, \qquad |h'(x)| \leq 1, \qquad |h''(x)| = 2|h'(x)|^{3/2} \leq 2|h'(x)|,$$

et les propriétés $(p0)$, $(p1)$, $(p3)$ sont vérifiées. La propriété $(p2)$ est vérifiée avec $n_0 = 1$ dès que α est strictement compris entre 0 et 1. Pour $\alpha = 1$, elle l'est avec $n_0 = 2$. La situation est différente pour $\alpha = 0$, puisque dans ce cas, $x = 1$ est un point fixe indifférent, c'est-à-dire $T(x) = x$ et $|T'(x)| = 1$. Il n'existe pas d'entier n_0 tel que la propriété $(p2)$ soit vérifiée. Considérons maintenant la propriété $(p4)$ de contrôle des discontinuités. Soit $\mathcal{T}$ l'ensemble des extrémités des intervalles images J_h, $h \in \mathcal{H}$. On a $\mathcal{T} = \{\alpha, 1-\alpha, T_\alpha(\alpha), T_\alpha(1-\alpha)\}$. L'ensemble $\mathcal{T}^n$ des extrémités des intervalles J_h, avec $h \in \mathcal{H}^n$ est toujours un ensemble fini. On en déduit que la quantité ℓ_n est un minimum pris dans un ensemble fini, et est donc strictement positive. La condition $(p4)$ est donc satisfaite. Il reste maintenant à satisfaire la condition de mélange topologique $(p5)$.
Soient donc U et V deux intervalles ouverts non nuls de I_α. Soit x un élément rationnel de U. Alors il existe $n_0 > 0$ tel que $T^{n_0}(x) = 0$. Ainsi, il existe $\varepsilon > 0$ tel que $[0,\varepsilon] \subseteq T_\alpha^{n_0}(U)$. Finalement, il existe $d \in \mathcal{D}$ tel que $I_d \subset [0,\varepsilon]$, et donc

$$I_\alpha = T_d(I_d) \subset T_d[0,\varepsilon] \subseteq T^{n_0+1}(U)$$

et ainsi $T^{n_0+1}(U) \cap V \neq \emptyset$. ∎

Du théorème 5.12 et de la proposition 5.21, on déduit qu'on peut appliquer le théorème taubérien aux séries génératrices de Dirichlet associées aux algorithmes α-euclidiens (voir proposition 5.7). Nous avons donc montré le théorème suivant.

Théorème 5.22 *Soit $\alpha \in]0,1]$ et $\mathcal{E}_\alpha$ l'algorithme correspondant. Soient $\mathbb{E}_N, \widetilde{\mathbb{E}}_N$ les espérances définies sur les ensembles $\Omega_N, \widetilde{\Omega}_N$. Alors le nombre moyen d'itérations de l'algorithme est asymptotiquement linéaire en N*

$$\widetilde{\mathbb{E}}_N[I] \sim \mathbb{E}_N[I] \sim \frac{2\log 2}{h(\alpha)} \cdot N := A_\alpha \cdot N,$$

où $h(\alpha)$ est l'entropie du système dynamique sous-jacent à l'algorithme. Plus généralement, si c est un coût à croissance modérée, et C la variable aléatoire associée, alors l'espérance $\mathbb{E}_N[C]$ est asymptotiquement linéaire en N,

$$\widetilde{\mathbb{E}}_N[C] \sim \mathbb{E}_N[C] \sim \frac{2\log 2}{h(\alpha)} \cdot \mu_\alpha(c) \cdot N := A_\alpha \cdot \mu_\alpha(c) \cdot N,$$

où $\mu_\alpha(c)$ est la valeur moyenne du coût c.

De même, la proposition 5.10 sur la complexité en bits s'applique, et on en déduit le théorème suivant.

Théorème 5.23 *Soit b le coût associé à une division, et soit B la variable aléatoire relative à la complexité en bits. Soit ℓ la taille définie sur l'ensemble des quotients. Alors l'espérance de cette variable est asymptotiquement quadratique en N,*

$$\widetilde{\mathbb{E}}_N[B] \sim \mathbb{E}_N[B] \sim \frac{\log^2 2}{h(\alpha)} \cdot \mu_\alpha(\ell) \cdot N^2 = \frac{A_\alpha}{2} \cdot \mu_\alpha(\ell) \cdot N^2,$$

où $\mu_\alpha(\ell)$ est la valeur moyenne du paramètre ℓ.

On peut choisir d'attribuer une valeur particulière au paramètre ℓ, pour tenir compte de la nature des divisions α-euclidiennes. Un chiffre d est de la forme $d = (q, \varepsilon)$, où $\varepsilon = \pm 1$ correspond à une éventuelle soustraction. On peut donc poser $\ell(d) = \ell_2(q) + \frac{1-\varepsilon}{2}$, auquel cas, pour certaines valeurs de α, $\mu_\alpha(\ell)$ est donné par (2.13,2.14).

De la même manière, on déduit de la proposition 5.21 et du théorème 5.12 qu'on peut appliquer le théorème de Hwang aux séries génératrices des moments liées aux coûts à croissance modérée, définis sur les trajectoires tronquées des systèmes dynamiques α-euclidiens (voir proposition 4.8). Nous avons donc prouvé le théorème suivant.

Théorème 5.24 *Soit $S_{\mathcal{E}_\alpha} = (I_\alpha, T_\alpha)$, $\alpha > 0$ un système dynamique α-euclidien. Soit c un coût à croissance modérée et C_n la variable aléatoire correspondante. Soient $\mathbb{P}$ et $\mathbb{E}$ la probabilité et l'espérance définies sur I_α muni de la mesure de Lebesgue. Alors il existe $\bar{A}_\alpha$, $\mu_\alpha(c)$ et $\delta_\alpha(c)$ tels que pour tout n, et pour tout $Y \in \mathbb{R}$*

$$\mathbb{P}\left[x \;\middle|\; \frac{C_n(x) - \bar{A}_\alpha \mu_\alpha(c) n}{\delta_\alpha(c)\sqrt{n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{Y} e^{-y^2/2}\, dy + O\left(\frac{1}{\sqrt{n}}\right),$$

la constante $\bar{A}_\alpha$ étant reliée à la constante A_α du théorème 5.22 par

$$\bar{A}_\alpha = \frac{1}{A_\alpha}$$

De plus, pour tout il existe a, b et $\kappa < 1$ tels que l'espérance et la variance satisfont

$$\mathbb{E}[C^n] = \bar{A}_\alpha \mu_\alpha(c) \cdot n + a + O(\kappa^n), \quad \mathrm{Var}[C_n] = \delta_\alpha^2(c) \cdot n + b + O(\kappa^n)$$

5.2.3 Le cas $\alpha = 0$

Nous avons déjà fait remarquer que dans le cas $\alpha = 0$, l'algorithme correspondant, l'algorithme Par-Excès $\mathcal{X}$ appartient à la classe des algorithmes lents. Son comportement moyen est asymptotiquement quadratique pour les coûts additifs, et asymptotiquement cubique pour la complexité en bits. Nous allons brièvement décrire l'analyse de cet algorithme, faite initialement par Brigitte Vallée dans [Val03].

Nous avons observé au chapitre 2 que le système $S_{\mathcal{X}}$ ne satisfait par de bonnes propriétés d'expansion, car il possède un point fixe indifférent en $x = 1$, c'est-à-dire qu'on vérifie $T(x) = 1, |T'(x)| = 1$. On remarque donc qu'une des propriétés requises pour l'analyse des autres systèmes α-euclidiens n'est plus vérifiée. Dans cette situation, on a recours au système induit : si on note a la "mauvaise" branche du système, alors l'ensemble $\widetilde{\mathcal{H}}$ de branches inverses du système induit est

$$\widetilde{\mathcal{H}} := (a^\star B)^\star \qquad \text{avec} \qquad B := \mathcal{H} \setminus \{a\}.$$

On peut donc construire l'opérateur de transfert associé $\mathbf{C}_s$, défini par

$$\mathbf{C}_s := \sum_{k \geq 0} \mathbf{B}_s \, \mathbf{A}_s^k = \mathbf{B}_s (I - \mathbf{A}_s)^{-1}$$

où les opérateurs sont

$$\mathbf{A}_s[f](x) = |a'(x)|^s f \circ a(x), \qquad \mathbf{B}_s[f](x) = \sum_{h \neq a} |h'(x)|^s f \circ h(x).$$

On montre maintenant facilement qu'étant donné un coût c, la série génératrice associée à ce coût s'exprime à l'aide des opérateurs $\mathbf{C}_s$ et $\mathbf{A}_s$ par

$$(I - \mathbf{C}_s)^{-1} \circ \mathbf{C}_s^{[c]} \circ (I - \mathbf{A}_s)^{-1} \circ (I - \mathbf{C}_s)^{-1}[1](0).$$

On voit maintenant apparaître trois quasi-inverses, ce qui implique un pôle d'ordre 3 (on vérifie que le quasi-inverse $(I - \mathbf{A}_s)^{-1}$ amène un pôle supplémentaire), et donc un comportement général quadratique. Le même phénomène se produit quand on étudie la complexité en bits, et explique le comportement asymptotiquement cubique.

5.2.4 Discussion des résultats

Ces résultats sont relativement étonnants si on observe que, quand elle sont connues, la densité invariante et l'entropie des systèmes α-euclidiens ont un comportement inattendu. Nous avons décrit au paragraphe 2.3.2 la densité invariante, dont le comportement est différent selon la valeur de α, ainsi que l'entropie, qui est donnée par (2.12)

$$h(\alpha) = \begin{cases} \dfrac{\pi^2}{6 \log \phi}, & \text{pour } \ \alpha \in [\sqrt{2} - 1, \phi - 1] \ , \\ \dfrac{\pi^2}{6 \log(\alpha + 1)}, & \text{pour } \ \alpha \in [\phi - 1, 1] \ . \end{cases}$$

On remarque donc que l'entropie est indépendante de α pour $\alpha \in [\sqrt{2}-1, \phi-1]$. On en déduit que les algorithmes correspondants ont le même comportement, puisque le même type de phénomène s'observe pour les valeurs moyennes des différents coûts. On peut en conclure que les algorithmes

les plus efficaces appartiennent à cette famille (pour $\alpha \in]\phi - 1, 1]$ la fonction $\alpha \mapsto h(\alpha)$ est décroissante, pour $\alpha \in]0, \sqrt{2}-1[$ on ne sait pas, mais il est naturel de conjecturer que l'application est croissante). C'est en particulier le cas de l'algorithme Centré (qui correspond à $\alpha = 1/2$), qui est donc le plus rapide des algorithmes "usuels".

Reste la question d'une transition de phase entre $\alpha = 0$ et $\alpha > 0$. En effet, pour $\alpha = 0$, le comportement est radicalement différent. Il est naturel de conjecturer

$$\lim_{\alpha \to 0+} h(\alpha) = 0.$$

5.3 Analyse dynamique de l'algorithme LSB

Ce paragraphe conclut l'analyse de l'algorithme LSB. Nous distinguons ici deux études distinctes, chacune faisant intervenir un système dynamique différent.

Dans la première, nous étudions le système $S_{\mathcal{L}}$. L'opérateur de transfert associé à ce système, $\mathbf{K}_{s,w}$, est adapté à deux types d'analyses : l'analyse des coûts à croissance modérée définis sur les trajectoires tronquées du système, et l'analyse de ces mêmes coûts définis sur l'algorithme lorsque la taille des entrées est liée au nombre de décalages.
Dans la seconde étude, nous étudions le système $S_{\underline{\mathcal{L}}}$. Ceci nous permet, via l'étude des opérateurs $\mathbf{L}_{s,w}$ et $\mathbf{L}_{s,t,w}$, d'obtenir le comportement asymptotique des principaux coûts définis sur l'algorithme LSB, ainsi que celui des continuants liés au développement en fraction continue 2-adique. Pour chacune des deux analyses, nous utilisons deux espaces fonctionnels différents.

5.3.1 Analyse fonctionnelle pour l'opérateur $\mathbf{K}_{s,w}$

Nous sommes maintenant amenés à étudier l'opérateur $\mathbf{K}_{s,w}$, relatif à un système dynamique défini sur le corps $\mathbb{Q}_2$ des nombres 2-adiques. Nous allons voir que sous certaines conditions (vérifiées par le système $S_{\mathcal{L}}$), une telle étude est plus simple que dans le cas usuel. En particulier, les propriétés de la valeur absolue $|.|_2$ simplifient souvent les calculs.

D'une manière générique, nous allons considérer un système dynamique $S = (\mathcal{B}, T)$, vérifiant l'ensemble Σ_2 de conditions énoncées ci-dessous.

Conditions Σ_2

- $(r1)$ *Le système dynamique est complet, on a donc* $\forall h \in \mathcal{H}$

$$T(\mathcal{B}_h) = \mathcal{B},$$

- $(r2)$ *Les dérivées* $|h'(x)|_2$ *sont constantes sur* $\mathcal{B}$ *et il existe* $\beta < 1$ *tel que*

$$\sum_{h \in \mathcal{H}} |h'|_2^2 = \beta$$

- $(r3)$ *Le système dynamique est topologiquement mélangeant : pour tous ouverts non nuls* U, V *de* $\mathcal{B}$, *il existe* $n_0 \geq 1$ *tel que pour tout* $n \geq n_0$, $T^{-n}V \cap W \neq \emptyset$.

Nous nous plaçons pour l'étude de tels systèmes sur l'espace $C^1(\mathcal{B})$ des fonctions C^1 sur $\mathcal{B}$. Cet espace est muni de la norme $||.||_1$ définie par

$$||f||_1 = ||f||_0 + ||f'||_0,$$

où $||.||_0$ est la norme sup,

$$||f||_0 = \sup_{x\in\mathcal{B}} |f|.$$

Avec la norme $||.||_1$, $C^1(\mathcal{B})$ est un espace de Banach. Enfin, nous noterons $C^0(\mathcal{B})$ l'ensemble des fonctions continues à valeurs complexe définies sur $\mathcal{B}$, muni de la norme $||.||_0$. L'espace $C^1(\mathcal{B})$ est donc un sous-espace de $C^0(\mathcal{B})$.

Nous allons montrer le résultat suivant.

Théorème 5.25 *Soit un système dynamique S défini sur $\mathbb{Q}_2$. S'il satisfait l'ensemble Σ_2 de conditions, alors l'opérateur de transfert associé $\mathbf{K}_{s,w}$ vérifie les conditions* (**A1**), (**A2**),(**A3**),(**B0**), (**B1**), (**C1**), (**C2**) *et* (**C3**) *quand il agit sur l'espace $C^1(\mathcal{B})$.*

Preuve du théorème 5.25

Nous commençons par montrer que l'opérateur $\mathbf{K}$ agit sur cet espace et qu'il y est quasi-compact. Rappelons que cet opérateur est ici de la forme

$$\mathbf{K}[f](x) = \sum_{h\in\mathcal{H}} |h'(x)|_2 \cdot f\circ h(x).$$

Nous supposerons dans la suite de cette preuve que l'opérateur $\mathbf{K}$ (de même que l'opérateur $\mathbf{K}_{s,w}$) est relatif à un système dynamique satisfaisant l'ensemble Σ_2 de conditions.

Proposition 5.26 *L'opérateur $\mathbf{K}$ agit sur l'espace $C^1(\mathcal{B})$ et donc la condition* (**A1**) *est vérifiée. De plus, il y satisfait une inégalité de Lasota-Yorke :*

$$||\mathbf{K}^n[f]||_1 \leq r_n||f||_1 + t_n||f||_0, \text{ avec } r_n := \beta^n \text{ et } t_n := (1-\beta)\left(1+\beta+\beta^2+\ldots+\beta^{n-1}\right). \quad (5.37)$$

Preuve : Tout d'abord, vérifions que l'opérateur agit bien sur $C^0(\mathcal{B})$:

$$||\mathbf{K}[f]||_0 = \sup_{x\in\mathcal{B}} \sum_{h\in\mathcal{H}} |h'(x)|_2 f\circ h(x) \leq ||f||_0 \sum_{h\in\mathcal{H}} |h'(x)|_2 = ||f||_0. \quad (5.38)$$

Il agit également sur $C^1(\mathcal{B})$: soient $x, y \in \mathcal{B}$, avec $x \neq y$ et $f \in C^1(\mathcal{B})$, alors

$$|\mathbf{K}[f](x) - \mathbf{K}[f](y)| = \left| \sum_{h\in\mathcal{H}} |h'(x)|_2 f\circ h(x) - \sum_{h\in\mathcal{H}} |h'(y)|_2 f\circ h(y) \right|.$$

Les dérivées $|h'(x)|_2$ étant constantes sur $\mathcal{B}$, on peut écrire

$$\begin{aligned} |\mathbf{K}[f](x) - \mathbf{K}[f](y)| &= \left| \sum_{h\in\mathcal{H}} |h'(x)|_2 \left(f\circ h(x) - f\circ h(y)\right) \right| \\ &\leq \left| \sum_{h\in\mathcal{H}} |h'(x)|_2 ||f'||_0 |h(x) - h(y)|_2 \right| \\ &\leq ||f'||_0 |x-y|_2 \sum_{h\in\mathcal{M}} |h'(x)|_2^2 = \beta||f'||_0|x-y|_2, \end{aligned}$$

en vertu de la propriété $(r2)$. On en déduit donc l'inégalité

$$||\mathbf{K}[f]||_1 \leq \beta||f||_1 + (1-\beta)||f||_0.$$

Celle-ci s'étend aux puissances de l'opérateur en

$$||\mathbf{K}^n[f]||_L \leq r_n||f||_1 + t_n||f||_0, \tag{5.39}$$

avec

$$r_n := \beta^n \text{ et } t_n := (1-\beta)\left(1+\beta+\beta^2+\ldots+\beta^{n-1}\right).$$

■

Pour pouvoir appliquer le théorème d'Hennion, il faut de plus vérifier que l'injection de $C^1(\mathcal{B})$ dans $C^0(\mathcal{B})$ est compacte, ce que nous montrons dans le lemme suivant.

Lemme 5.27 *Soit (f_p) une suite de fonctions bornées de $C^1(\mathcal{B})$. Alors il existe une sous-suite convergente dans $C^0(\mathcal{B})$ vers un élément de $C^1(\mathcal{B})$.*

Preuve : Nous utilisons le théorème d'Ascoli-Arzola, dont nous rappelons une version ici (cette version se trouve dans [Rud87], par exemple).

Théorème [Ascoli-Arzola]*Soit X un ensemble compact, et $C^0(X)$ l'espace de Banach des fonctions continues à valeurs complexe muni de la norme sup. Soit $\Psi \subset C^0(X)$ un sous-ensemble de fonctions tel que :*
(i) $\sup\{|f(x)|;\ f \in \Psi\} < \infty$ pour tout $x \in X$ et
(ii) pour tout $\varepsilon > 0$ et pour tout $x \in X$ il existe un voisinage V de x tel que $|f(y) - f(x)| < \varepsilon$ pour tout $y \in V$ et pour tout $f \in \Psi$.
Alors toute suite d'éléments de Ψ admet une sous-suite uniformément convergente.

Ce théorème implique donc qu'il existe une sous-suite de (f_p) qui converge dans $C^0(\mathcal{B})$ vers une fonction f. Soit M tel que $||f_p'|| < M$, pour tout $p \geq 0$. Alors, quand p tend vers l'infini, la relation

$$|f_p(x) - f_p(y)| \leq M|x-y|_2,\ \forall x \neq y \in \mathcal{B},$$

implique que $f \in C^1(\mathcal{B})$. ■

Nous avons donc obtenu la quasi-compacité de l'opérateur sur $C^1(\mathcal{B})$. Nous savons déjà que 1 est valeur propre (puisque $\mathbf{K}[1] = 1$) et que c'est une valeur propre dominante (en effet, on déduit de la preuve de la proposition précédente l'inégalité $||\mathbf{K}||_1 \leq 1$ et le théorème du rayon spectral implique $r_0(\mathbf{K}) \leq ||K||_1$ (voir 5.1)). On montre que 1 est valeur propre simple, et est la seule valeur propre sur le cercle unité de la même manière que dans le chapitre précédent sur les systèmes α-euclidiens, c'est-à-dire en utilisant les propriétés de mélange topologique . L'argument qui avait été utilisé pour la proposition 5.16 est en réalité une adaptation aux fonctions à variation bornée d'un résultat classique portant initialement sur les fonctions C^1. Ce résultat, qu'on trouve par exemple dans le livre [Bal00] (théorème 1.5 p. 35), s'applique donc dans notre cas. On a donc montré la proposition suivante.

Proposition 5.28 *L'opérateur $\mathbf{K}$ vérifie les propriétés* $(\mathbf{A2}, \mathbf{A3})$.

Il faut maintenant étudier la dépendance de l'opérateur $\mathbf{K}_{s,w}$ en s et w, pour en déduire par perturbation la décomposition spectrale de l'opérateur. Nous étudions d'abord la dépendance en w.

Proposition 5.29 *Il existe un voisinage complexe $\mathcal{W}$ de $w = 0$ tel que pour $(s, w) \in \{\Re(s) > \beta\} \times \mathcal{W}$, l'opérateur $\mathbf{K}_{s,w}$ agisse sur $C^1(\mathcal{B})$. De plus, l'application $(s, w) \mapsto \mathbf{K}_{s,w}$ y est analytique. Les conditions* $(\mathbf{B0}), (\mathbf{B1})$ *sont donc satisfaites.*

Preuve : En reprenant les étapes de la preuve de la proposition 5.26, on obtient les inégalités

$$||\mathbf{K}_{s,w}[f]||_0 \leq ||f||_0 \sum_{h\in\mathcal{H}} |h'|_2^s \cdot \exp(wc(h)),$$

$$||\mathbf{K}_{s,w}[f]||_1 \leq ||f||_1 \sum_{h\in\mathcal{H}} |h'|_2^{s+1} \cdot \exp(wc(h)).$$

Ces deux quantités sont finies si c est un coût à croissance modérée.
Considérons maintenant les deux opérateurs obtenus par dérivation par rapport à s et par rapport à w :

$$\mathbf{R}_w[f](x) := \sum_{h\in\mathcal{H}} c(h)|h'(x)|_2^s \exp(wc(h)) f \circ h(x),$$

$$\mathbf{R}_s[f](x) := \sum_{h\in\mathcal{H}} \log|h'(x)|_2 |h'(x)|_2^s \exp(wc(h)) f \circ h(x).$$

On obtient les inégalités

$$||\mathbf{R}_w[f]||_0 \leq ||f||_0 \sum_{h\in\mathcal{H}} c(h) \cdot |h'|_2^s \cdot \exp(wc(h)),$$

$$||\mathbf{R}_w[f]||_1 \leq ||f||_1 \sum_{h\in\mathcal{H}} c(h) \cdot |h'|_2^{s+1} \cdot \exp(wc(h)),$$

$$||\mathbf{R}_s[f]||_0 \leq ||f||_0 \sum_{h\in\mathcal{H}} |h'|_2^s \cdot \log|h'(x)|_2 \cdot \exp(wc(h)),$$

$$||\mathbf{R}_s[f]||_1 \leq ||f||_1 \sum_{h\in\mathcal{H}} |h'|_2^{s+1} \cdot \log|h'(x)|_2 \cdot \exp(wc(h)).$$

Les différentes séries convergent pour $(s, w) \in \{\Re(s) > \beta\} \times \mathcal{W}$ si c est un coût à croissance modérée. ■

Enfin, remarquons que le fait que la distribution uniforme soit préservée simplifie les calculs des valeurs spectrales dominantes.

Proposition 5.30 *La dérivée $\lambda'_s(1, 0)$ est différente de 0. La condition* $(\mathbf{C4})$ *est donc vérifiée, et il existe donc un voisinage W de $w = 0$ et une fonction $\sigma : W \to \mathbb{C}$ telle que $\lambda(\sigma(w), w) = 1$. Cette fonction est analytique et satisfait $\sigma(0) = 1$. De plus, les dérivées secondes des applications $w \mapsto \Lambda(1, w)$ et $w \mapsto \sigma(w)$ sont strictement positives. Les conditions* $(\mathbf{C2}, \mathbf{C3})$ *sont donc vérifiées.*

Preuve : Observons tout d'abord les applications $w \mapsto \lambda(1, w)$ et $w \mapsto \Lambda(1, w)$. Soit $\psi_{1,w}$ la fonction propre associée à la valeur propre $\lambda(1, w)$ de $\mathbf{K}_{1,w}$. Alors on vérifie

$$(\mathbf{K}_{1,w}[\psi_w](x))' = (\lambda(1, w)\psi_{1,w}(x))',$$

$$\sum_{h\in\mathcal{H}} c(h)\cdot|h'(x)|_2\cdot\exp(wc(h))\cdot\psi_{1,w}\circ h(x)+\sum_{h\in\mathcal{H}}\cdot|h'(x)|_2\cdot\exp(wc(h))\cdot\psi'_{1,w}\circ h(x)$$
$$=\lambda'_w(1,w)\psi_{1,w}(x)+\lambda(1,w)\psi'_{1,w}(x),$$

la dérivation se faisant par rapport à w. Puisque $\psi_{1,0}(x)=1$ pour tout $x\in\mathcal{B}$ et $\lambda(1,0)=1$, on en déduit

$$\sum_{h\in\mathcal{H}} c(h)\cdot|h'(x)|_2+\sum_{h\in\mathcal{H}}|h'(x)|_2\cdot\psi'_{1,0}\circ h(x)=\lambda'(1,0)+\psi'_{1,0}(x).$$

En intégrant sur $\mathcal{B}$, on obtient finalement l'égalité

$$\sum_{h\in\mathcal{H}} c(h)\cdot|h'(x)|_2=\lambda'_w(1,0).$$

Le même procédé pour la dérivée seconde mène à

$$\lambda''_w(1,0)=\sum_{h\in\mathcal{H}} c^2(h)\cdot|h'(x)|_2,$$

et donc à

$$\Lambda''_w(1,0)=\sum_{h\in\mathcal{H}} c^2(h)\cdot|h'(x)|_2-\left(\sum_{h\in\mathcal{H}} c(h)\cdot|h'(x)|_2\right)^2.$$

Étudions maintenant l'application $w\mapsto\sigma(w)$. La relation $\Lambda(\sigma(w),w)=0$ entraîne dans un premier temps

$$\sigma'(w)\Lambda'_s(\sigma(w),w)+\Lambda'_w(\sigma(w),w)=0,$$

dont on déduit

$$\sigma'(0)=\frac{-\Lambda'_w(1,0)}{\Lambda'_s(1,0)}. \tag{5.40}$$

Il nous reste donc à étudier l'application $s\mapsto\Lambda(s,0)$. Comme précédemment on part de la relation $\mathbf{K}_{s,0}[\psi_{s,0}](x)=\lambda(s,0)\psi_{s,0}(x)$ pour aboutir finalement à

$$\lambda'_s(1,0)=\sum_{h\in\mathcal{H}}\log|h'|_2\cdot|h'|_2,$$

et

$$\lambda''_s(1,0)=\sum_{h\in\mathcal{H}}\log^2|h'|_2\cdot|h'|_2.$$

Enfin, toujours selon le même procédé, la dérivée $\lambda''_{s,w}(1,0)$ est donnée par

$$\lambda''_{s,w}(1,0)=\sum_{h\in\mathcal{H}} c(h)\cdot\log|h'|_2\cdot|h'|_2.$$

On en déduit, à partir de la relation

$$\sigma''(w)\Lambda'_s(\sigma(w),w)+(\sigma')^2(w)\Lambda''_s(\sigma(w),w)+2\sigma'(w)\Lambda''_{s,w}(\sigma(w),w)+\Lambda''_w(\sigma(w),w)=0,$$

qui entraîne que $\sigma''(0)$ vérifie

$$\sigma''(0)=\frac{-1}{\Lambda'_s(1,0)}\left[(\sigma'(0))^2\Lambda''_s(1,0)+2\sigma'(0)\Lambda''_{s,w}(1,0)+\Lambda''_w(1,0)\right].$$

Avec la valeur de $\sigma'(0)$ donnée en (5.40), on en conclut que $\sigma''(0)>0$. ■

5.3.2 Théorèmes 5.31 et 5.32

Pour obtenir les résultats de ce paragraphe, il faut tout d'abord vérifier que le système $S_{\mathcal{L}}$ vérifie l'ensemble Σ_2 de conditions. Ceci est immédiat : le système est complet, les normes des dérivées des branches inverses sont toutes constantes sur $\mathcal{B}$, et elles vérifient de plus

$$\sum_{h\in\mathcal{H}} |h'(x)|_2^2 = \sum_{k\geq 1} \sum_{\substack{|a|<2^k \\ a \text{ impair}}} 2^{-4k} = \sum_{k\geq 1} 2^{-3k} = \frac{1}{7}.$$

On peut donc choisir ici $\beta = 1/7$. Il reste à vérifier la propriété de mélange : la preuve est exactement la même que pour les systèmes α-euclidiens. Si U et V sont deux ouverts non-nuls de $\mathcal{B}$, alors il existe un rationnel $x \in U$ (par densité de $\mathbb{Q}$ dans $\mathbb{Q}_2$) et un entier n_0 tel que $T^{n_0}(x) = 0$. On en déduit qu'il existe $\varepsilon > 0$ tel que la boule B_ε de centre 0 et de rayon ε soit incluse dans l'image par T^{n_0} de U. Maintenant, pour q suffisamment petit, la boule fondamentale $\mathcal{B}_q$ est incluse dans B_ε. Comme le système est complet, on a $\mathcal{B} = T(\mathcal{B}_q) \subset T^{n_0+1}(U)$, et donc $T^{n_0+1}(U)\cap V \neq \emptyset$.

On obtient finalement les résultats souhaités dans ce paragraphe, à savoir un comportement gaussien des coûts à croissance modérée définis sur les trajectoires tronquées du système dynamique ou directement sur l'algorithme. Rappelons que dans ce dernier cas, l'ensemble Ω_N d'entrées de l'algorithme est muni de la taille liée au nombre de décalages. Les valeurs spectrales dominantes s'expriment très facilement ici, puisqu'on vérifie d'après la proposition 5.30

$$\mu_{\mathcal{L}}(c) = \Lambda'_w(1,0) = \sum_{k\geq 1} \sum_{\substack{|a|<2^k \\ a \text{ impair}}} c\left(\frac{a}{2^k}\right)\cdot 2^{-2k}$$

$$-h(S_{\mathcal{L}}) = \Lambda'_s(1,0) = \sum_{\substack{|a|<2^k \\ a \text{ impair}}} \log 2^{-2k}\cdot 2^{-2k} = -4\log 2.$$

Ces deux grandeurs apparaissent dans chacun des théorèmes suivants.

Théorème 5.31 *Soit $S_{\mathcal{L}} = (\mathcal{B}, T_{\mathcal{L}})$ le système dynamique 2-adique LSB. Soit c un coût à croissance modérée et C_n la variable aléatoire correspondante. Soient $\mathbb{P}$ et $\mathbb{E}$ la probabilité et l'espérance définies sur $\mathcal{B}$ muni de la mesure de Haar. Alors il existe $\mu_{\mathcal{L}}(c)$ et $\delta_{\mathcal{L}}(c)$ avec*

$$\delta^2_{\mathcal{L}}(c) = \mu_{\mathcal{L}}(c^2) - \mu_{\mathcal{L}}(c)^2$$

tels que pour tout n, et pour tout $Y \in \mathbb{R}$

$$\mathbb{P}\left[x \Big| \frac{C_n(x) - \mu_{\mathcal{L}}(c)n}{\delta_{\mathcal{L}}(c)\sqrt{n}} \leq Y\right] = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{Y} e^{-y^2/2}dy + O\left(\frac{1}{\sqrt{n}}\right).$$

De plus, la moyenne et la variance sont :

$$\mathbb{E}[C_n] = \mu_{\mathcal{L}}(c)n + a + O(\kappa^n) \text{ et } \mathrm{Var}[C_n] = \delta^2_{\mathcal{L}}(c)n + b + O(\kappa^n).$$

où a, b, κ sont des constantes avec $\kappa < 1$.

Pour le dernier théorème, la démarche basée sur les séries entières n'ayant pas encore été utilisée, nous allons la décrire brièvement dans la preuve.

Théorème 5.32 *Soit $\mathcal{L}$ l'algorithme LSB. Soit $\mathbb{P}_N$ la probabilité définie sur l'ensemble Ω_N des entrées valides de taille N, quand la taille est liée au nombre de décalages faits par l'algorithme. Soit c un coût à croissance modérée et C la variable correspondante. Alors il existe $\mu_{\mathcal{L}}(c)$ and $\delta_{\mathcal{L}}(c)$ tels que pour tout N, et tout $Y \in \mathbb{R}$:*

$$\mathbb{P}_N\left[(u,v)|\frac{C(u,v)-\frac{1}{2}\mu_{\mathcal{L}}(c)N}{\widehat{\delta}_{\mathcal{L}}(c)\sqrt{N}} \leq Y\right] = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{Y} e^{-y^2/2}dy + O\left(\frac{1}{\sqrt{N}}\right).$$

De plus, la moyenne et la variance sont

$$\mathbb{E}_N[C] = \frac{1}{2}\mu_{\mathcal{L}}(c)N + a + O(\kappa^N), \quad \mathrm{Var}_N[C] = \widehat{\delta}^2_{\mathcal{L}}(c)N + b + O(\kappa^N),$$

où a, b, κ sont des constantes avec $\kappa < 1$, et $\mu_{\mathcal{L}}$ est la même constante qu'au théorème précédent.

Preuve : Considérons tout d'abord les séries entières $F_C(z,w)$ et $T_1(z)$, reliées à l'opérateur $\mathbf{K}_{s,w}$ par (voir proposition 4.7)

$$(\mathrm{I}-\mathbf{K}_{s,w})^{-1}[1](0) = F(2^{-2s}, w),$$
$$(\mathrm{I}-\mathbf{K}_{s,0})^{-1}[1](0) = T_1(2^{-2s}).$$

Soit w proche de zéro fixé. Alors les quasi-inverses se décomposent autour de $(\sigma(w), w)$ en

$$(\mathrm{I}-\mathbf{K}_{s,w})^{-1}[1](0) = \frac{1}{1-\lambda(s,w)}\frac{-1}{\lambda'_s(s,w)}\mathbf{P}_{s,w}[1](0) + (\mathrm{I}-\mathbf{N}_{s,w})^{-1}[1](0),$$

et donc les séries ont un pôle en $(s,w) = (\sigma(w), w)$. Le résidu en ce point est

$$\frac{-1}{\lambda'_s(\sigma(w),w)}\mathbf{P}_{\sigma(w),w}[1](0),$$

et comme le théorème C s'applique, on en déduit le comportement asymptotique des coefficients $[z^n]F_C(z,w)$ et $[z^n]T_1(z)$, donné par

$$[z^n]F_C(z,w) \sim \left(\frac{1}{2^{-2\sigma(w)}}\right)^n \frac{-1}{\lambda'_s(\sigma(w),w)}\mathbf{P}_{\sigma(w),w}[f](0)(1+O(\frac{1}{n})),$$
$$[z^n]T_1(z) \sim \left(\frac{1}{2^{-2}}\right)^n \frac{-1}{\lambda'_s(1,0)}\mathbf{P}_{1,0}[f](0)(1+O(\frac{1}{n})).$$

On en déduit une décomposition en quasi-puissance pour la série $\mathbb{E}_N[\exp(wC)]$,

$$\begin{aligned}\mathbb{E}_n[\exp(wC)] &= \left(2^{2(\sigma(w)-1)}\right)^n \frac{\lambda'_s(1,0)}{\lambda'_s(\sigma(w),w)}\frac{\mathbf{P}_{\sigma(w),w}[f](0)}{\mathrm{P}_{1,0}[f](0)}(1+O(\frac{1}{n})) \\ &= \exp\left[nU(w)+V(w)\right](1+O(\frac{1}{n}))\end{aligned}$$

avec

$$U(w) = 2(\sigma(w)-1)\cdot\log 2 \text{ et } V(w) = \log\left(\frac{\lambda'_s(1,0)}{\lambda'_s(\sigma(w),w)}\right) + \log\left(\frac{\mathbf{P}_{\sigma(w),w}[f](0)}{\mathbf{P}_{1,0}[f](0)}\right)$$

D'après la proposition 5.30, les hypothèses du théorème de Hwang sont vérifiées, et on en déduit en particulier

$$U'(0) = -\frac{\mu_{\mathcal{L}}(c)}{2}.$$

■

5.3.3 Analyse fonctionnelle pour l'opérateur $\mathbf{L}_{s,t}$

Pour l'étude de l'algorithme LSB, ainsi que pour celle des continuants du développement en fraction continue 2-adiques, le système étudié est $S_{\underline{\mathcal{L}}}$ et l'opérateur est donc différent. Comme nous l'avons remarqué précédemment, cet opérateur est plus un opérateur relatif à un système de fonctions itérées qu'à un système dynamique. On peut également le considérer comme un opérateur issu d'un système de produits de matrices aléatoires. Avant de commencer l'analyse proprement dite, nous commençons donc par détailler le cadre dans lequel nous nous trouvons.

L'opérateur que nous avons introduit au chapitre précédent est un opérateur à trois variables s, t et w. Nous n'utiliserons pas la variable w, et afin de ne pas alourdir les notations nous adopterons jusqu'à la fin du chapitre la convention suivante

$$\mathbf{L}_{s,t} := \mathbf{L}_{s,t,0}$$

et donc l'opérateur $\mathbf{L}_{s,t}$ s'écrit

$$\mathbf{L}_{s,t}[f](x) = \sum_{h\in\mathcal{H}} \delta_h^t |\underline{h}'(x)|^s f\circ \underline{h}(x),$$

où l'ensemble $\mathcal{H}$ est

$$\mathcal{H} = \left\{ h(x) = \frac{1}{q+x}, q = \frac{a}{2^k}, k \geq 1, a \text{ impair }, |a| < 2^k \right\}$$

et la branche $\underline{h}$ est définie sur le tore $J =]-\pi/2, \pi/2[$ par

$$\underline{h}(x) = \arctan(h(\tan x)).$$

La quantité δ_h est donnée par

$$\delta_h = \frac{1}{2^{2k}} \text{ si } h(x) = \frac{1}{q+x} \text{ et } q = \frac{a}{2^k},$$

et vérifie bien sûr $\delta_h = |h'(x)|_2$, pour $x \in \mathcal{B}$. Nous allons maintenant poser $\ell_h = \underline{h}$, $\delta_{\ell_h} = \delta_h$ et

$$\mathcal{L} = \{\ell_h, h \in \mathcal{H}\}.$$

L'opérateur $\mathbf{L}_{s,t}$, qui s'écrit dorénavant

$$\mathbf{L}_{s,t}[f](x) = \sum_{\ell\in\mathcal{L}} \delta_\ell^t |\ell'(x)|^s f\circ \ell(x)$$

correspond parfaitement à l'opérateur relatif au système de fonction itérées pour lequel chaque fonction $\ell \in \mathcal{L}$ est choisie avec probabilité δ_ℓ.

Un point de vue -équivalent- est celui qui consiste à voir cet opérateur comme étant relatif à un produit de matrice aléatoire. Cette correspondance est basée sur la relation (4.14) entre les matrices $\mathcal{N}_\ell$, $\mathcal{M}_\ell$ relatives aux divisions effectuées par l'algorithme et les quantités δ_ℓ et $|\ell'|$. Rappelons que cette relation est

$$||\mathcal{N}_\ell(1,0)||^2 = \frac{1}{\delta_\ell |\ell'(0)|}, \quad ||\mathcal{M}_\ell(1,0)||^2 = \frac{1}{|\ell'(0)|} \tag{5.41}$$

et que les matrices $\mathcal{N}$ et $\mathcal{M}$ sont données par (voir (1.9,1.11))

$$\mathcal{N}_\ell = \begin{pmatrix} 0 & 2^k \\ 2^k & a \end{pmatrix}, \qquad \mathcal{M}_\ell = \frac{1}{2^k} \cdot \mathcal{N}_\ell$$

pour une fonction ℓ relative à un quotient $q = a/2^k$. Si on considère l'ensemble $\mathcal{M}$ de matrices, la matrice $\mathcal{M}_\ell$ étant tirée avec probabilité δ_ℓ, l'opérateur $\mathbf{L}_{s,t}$ correspond à celui introduit dans les travaux de Furstenberg [Fur63], Guivarc'h et Raugi [GR85] ou Le Page [LP82], qui sont résumés dans le livre de Bougerol et Lacroix [BL85]. L'opérateur sert dans ce cadre à montrer l'existence, et la positivité de l'exposant de Lyapunov défini pour un ensemble $\mathcal{S}$ de matrices par

$$\gamma := \lim_{n\to\infty} \frac{1}{n} \mathbb{E}[\log ||S_1\, S_2\, \dots\, S_n||].$$

En particulier, le lien entre l'opérateur et l'exposant de Lyapunov (lorsqu'il existe et est positif) est

$$2\gamma = -\lambda'_t(1,0),$$

$\lambda(s,t)$ étant la valeur propre dominante de l'opérateur $\mathbf{L}_{s,t}$ et λ'_t désignant la dérivée par rapport à t. Cette relation met en évidence une différence entre l'opérateur de Bougerol et le nôtre. L'étude du produit de matrice se fait autour de $(s,t) = (1,0)$, ce qui s'explique naturellement si on considère la quantité $|\ell'|^s$ comme une perturbation de la forme $|\ell'|^s = \exp(s \log |\ell'|)$, ce qui implique une étude autour de $s = 0$ pour appliquer des théorèmes comme le théorème des quasi-puissances (c'est d'ailleurs ce que nous faisons pour l'étude des continuants du développement en fraction continue, voir proposition 4.9). Lorsqu'on étudie l'algorithme LSB, on étudie plutôt l'opérateur en $(s,t) = (1,1)$ (voir proposition 4.2), ce qui complique la situation.

Cependant, beaucoup des arguments décrits dans [BL85] par exemple nous seront utiles. En particulier, étant donné un ensemble de matrices $\mathcal{S}$, si on note $\bar{\mathcal{S}}$ le semi-groupe engendré par $\mathcal{S}$ et $L^+(S)$ la quantité

$$L^+(S) := \sup\{\log^+ ||S||, \log^+ ||S^{-1}||\} \qquad \text{avec} \qquad \log^+ x := \sup(0, \log x),$$

alors nous définissons l'ensemble Σ_3 de propriétés par

Conditions Σ_3

- ($q1$) *Il existe une suite (S_n) d'éléments de $\bar{\mathcal{S}}$ pour laquelle $||S_n||^{-1} \cdot S_n$ converge vers une matrice de rang 1.*
- ($q2$) *Il n'existe pas d'union finie W de lignes $V_1, V_2, \dots V_k$ invariante par toutes les matrices $S \in \mathcal{S}$.*
- ($q3$) $\mathbb{E}[\exp(wL^+(S))] < \infty$ *pour un réel positif w suffisamment petit.*

Cet ensemble de conditions est suffisant pour étudier l'opérateur autour de $(1,0)$ sur l'espace $\mathbf{H}_\alpha(J)$ des fonctions α-Hölder sur J. L'espace $\mathbf{H}_\alpha(J)$ est l'ensemble des fonctions continues sur J pour lesquelles il existe une constante M telle que

$$|f(x) - f(y)| < M \cdot |x - y|^\alpha, \quad \forall x, y \in J, \quad x \neq y.$$

Cet espace est dense dans $C^1(J)$, et sa boule unité est compacte dans celle de $C^1(J)$. Nous utiliserons le théorème suivant.

Théorème E. [Furstenberg, Guivarc'h et Raugi, Le Page] *Soit $\mathcal{S}$ un ensemble de matrices aléatoires qui satisfait l'ensemble Σ_3 de conditions. Alors*
(i) l'exposant de Lyapunov défini par

$$\gamma := \lim_n \frac{1}{n}\mathbb{E}[\log ||S_1\, S_2\, \dots S_n||]$$

est strictement positif.
(ii) Pour α, t suffisamment petits, l'opérateur de transfert $\mathbf{L}_{1,t} : \mathbf{H}_\alpha(J) \to \mathbf{H}_\alpha(J)$ est quasi-compact sur $\mathbf{H}_\alpha(J)$, et admet une unique valeur propre dominante $\lambda(1,t)$. L'exposant γ est donné par $2\gamma = -\lambda'_t(1,0)$.

Notons que le théorème précédent a tout d'abord été introduit pour l'étude d'un opérateur noté T_z défini par

$$T_z[f](x) = \sum_{\mathcal{N}_{[q]} \in \mathcal{N}} \delta_{\mathcal{N}_{[q]}} \left(\frac{||\mathcal{N}_{[q]}(u,v)||}{||(u,v)||} \right)^z \cdot f \circ h(x)$$

où h est l'application relative à la matrice $\mathcal{N}_{[q]}$ et (u,v) un vecteur de $\mathbb{R}^2$ associé au point x de la droite projective $\mathbf{P}(\mathbb{R})$. Cet opérateur n'est rien d'autre que le conjugué (par l'application tangente) de notre opérateur.

Le théorème précédent, quoique très utile, n'est pas suffisant pour notre analyse puisqu'il étudie les variables s, t dans un voisinage de $(1,0)$. Nous avons besoin d'imposer des conditions supplémentaires sur notre système.

Conditions Σ_4
- $(q4)$ Il existe $\ell \in \mathcal{L}$ et $x \in J$ tel que $\ell(x) = x$
- $(q5)$ Pour tout $\ell \in \mathcal{L}$, et tout $x \in J$, on a $\phi^{-2} \leq |\ell'(x)| \leq \phi^2$.
- $(q6)$ Pour tout $x \in J$ et tout $\ell \in \mathcal{L}$, on a $|\ell''(x)| \leq \sqrt{5}\,|\ell'(x)|$.

Nous allons donc montrer le théorème suivant.

Théorème 5.33 *Soit $\mathcal{S}$ un ensemble de matrices aléatoires qui satisfait les ensembles Σ_3 et Σ_4 de conditions. Alors l'opérateur $\mathbf{L}_{s,w}$ associé vérifie les conditions* (**A1**), (**A2**),(**B1**), (**C1**) *et* (**C3**).

Preuve du théorème 5.33

D'une manière générale, cette preuve se fait en exhibant des relations entre les spectres de différents opérateurs, agissant sur différents espaces. En particulier, nous utiliserons l'opérateur $\widehat{\mathbf{L}}$: alors que l'opérateur $\mathbf{L}$ est relatif à l'ensemble $\mathcal{M}$ de matrices (ou à l'ensemble $\mathcal{L}$ d'applications), l'opérateur $\widehat{\mathbf{L}}$ est défini à partir de l'ensemble $\mathcal{M}^{-1}$,

$$\mathcal{M}^{-1} = \{\mathcal{M}_{[q]}^{-1}, \mathcal{M}_{[q]} \in \mathcal{M}\}.$$

La probabilité de choisir la matrice $\mathcal{M}_{[q]}^{-1}$ étant toujours donnée par δ_q. De la même manière qu'on associe l'application ℓ_q à la matrice $\mathcal{M}_{[q]}$, on associe l'application ℓ_q^{-1} à la matrice $\mathcal{M}_{[q]}^{-1}$.

L'opérateur $\widehat{\mathbf{L}}_{s,t}$ est finalement défini par

$$\widehat{\mathbf{L}}_{s,t} = \sum_{\ell \in \mathcal{L}^{-1}} \delta_\ell^t \cdot |\ell'(x)|^s \cdot f \circ \ell(x).$$

Considérons l'involution $(x,1) \mapsto (-1,x)$ de la droite projective. Elle s'exprime exactement à l'aide de l'application Tilde de J définie par

$$\widetilde{y} : y \mapsto y + \pi/2. \tag{5.42}$$

Les relations

$$\mathcal{M}_{[q]}^{-1}(-1,x) = (x+q,-1), \quad \mathcal{M}_{[q]}(x,1) = (1,x+q)$$

entraînent finalement les relations suivantes entre ℓ et ℓ^{-1} :

$$\ell^{-1}(\widetilde{x}) = \widetilde{\ell(x)} \qquad |\ell^{-1}(\widetilde{x})|' = \ell'(x), \tag{5.43}$$

qui nous serons utiles pour relier les spectres de $\widehat{\mathbf{L}}$ et de $\mathbf{L}$.
Notons enfin que si des ensembles $\mathcal{L}$ d'applications ou $\mathcal{M}$ de matrices satisfont les conditions Σ_3 et Σ_4, il en va de même pour $\mathcal{M}^{-1}$ et $\mathcal{L}^{-1}$.

Rappelons qu'on cherche ici à étudier le spectre de l'opérateur $\mathbf{L}_{s,t}$ autour de $(1,1)$. L'espace fonctionnel sur lequel nous allons obtenir les propriétés souhaitées de l'opérateur est l'espace de fonction continues $C^0(J)$, dont la norme est donnée par

$$||f||_0 := \sup_{x \in J} |f(x)|.$$

Nous allons procéder en plusieurs étapes. Nous allons tout d'abord utiliser les résultats de Bougerol, Le Page, etc... pour montrer la quasi-compacité de $\mathbf{L}_{0,1}$ sur $C^1(J)$, espace dont la norme est donnée par

$$||f||_1 = ||f||_0 + ||f'||_0.$$

Puis nous allons "remonter" cette propriété à l'opérateur $\mathbf{L}_{1,1}$ sur l'espace $C^0(J)$.
Les relations plus précises entre valeurs propres dominantes se feront entre d'une part la valeur propre dominante de l'opérateur $\mathbf{L}_{s,t}$ et $\widehat{\mathbf{L}}_{1-s,t}$ et d'autre part entre celle de $\mathbf{L}_{s,t}$ et celle de $\widehat{\mathbf{L}}_{s,t}$. Ces relations permettront au final d'obtenir l'exposant de Lyapunov γ en fonction du comportement de la valeur propre dominante de $\mathbf{L}_{s,t}$ au voisinage de $(1,1)$.

La première partie de la preuve fera donc un usage "intensif" du lemme 5.1, du paragraphe 5.1.1, qui relie les spectres d'un même opérateur agissant sur deux espaces différents.

Nous allons maintenant montrer les trois lemmes suivants.

Lemme 5.34 *Soit $\mathcal{A} := \{(s,t) \in \mathbb{C}^2; \Re t > 1/2\}$. Pour $(s,t) \in \mathcal{A}$, les opérateurs $\mathbf{L}_{s,t}$ et $\widehat{\mathbf{L}}_{s,t}$ agissent sur $C^0(J)$ et $C^1(J)$. De plus, les applications $(s,t) \mapsto \mathbf{L}_{s,t}$, $(s,t) \mapsto \widehat{\mathbf{L}}_{s,t}$ sont analytiques. Les conditions* (**A1**), (**B1**) *sont donc vérifiées.*

Preuve : Soit $(s,t) \in \mathcal{A}$, et $\sigma := \Re s, \tau := \Re t$. Remarquons tout d'abord que si $(s,t) \in \mathcal{A}$, la série $S(\tau) := \sum_{\ell \in \mathcal{L}} \delta_\ell^\tau$ est convergente et satisfait

$$S(\tau) := \sum_{\ell \in \mathcal{L}} \delta_\ell^\tau = \sum_{k \geq 1} [2^{1-2\tau}]^k = \frac{2^{1-2\tau}}{1-2^{1-2\tau}}.$$

Si on définit les composants $\mathbf{L}_{s,t}^{(\ell)}$ de $\mathbf{L}_{s,t}$ par

$$\mathbf{L}_{s,t}^{(\ell)}[f] := |\ell'|^s \cdot f \circ \ell,$$

alors avec $(q5)$ on a

$$||\mathbf{L}_{s,t}^{(\ell)}[f]||_0 \leq \phi^{2|\sigma|} \cdot ||f||_0,$$

$$\left(\mathbf{L}_{s,t}^{(\ell)}[f]\right)'(x) = s \cdot \ell''(x) \cdot \ell'(x)^{s-1} \cdot f \circ \ell(x) + \ell'(x)^{s+1} \cdot f' \circ \ell(x),$$

de sorte que

$$|| \left(\mathbf{L}_{s,t}^{(\ell)}[f]\right)' ||_0 \leq \sqrt{5} \cdot \phi^{2|\sigma|} \cdot |s| \cdot \|f\|_0 + \phi^{2|\sigma+1|} \|f'\|_0,$$

et finalement

$$||\mathbf{L}_{s,t}||_1 \leq |s| \cdot \phi^{2|\sigma|+2} \cdot S(\tau). \tag{5.44}$$

Ceci prouve que la somme définissant $\mathbf{L}_{s,t}$ converge normalement sur tous les sous-ensembles compacts de $\mathcal{A}$ dans $C^0(J)$ et dans $C^1(J)$. Donc $\mathbf{L}_{s,t}$ est un opérateur borné de $C^0(J)$ et $C^1(J)$, et les applications $(s,t) \mapsto \mathbf{L}_{s,t}$ sont analytiques.
■

Lemme 5.35 *Les opérateurs* $\mathbf{L}_{1,0}$ *et* $\widehat{\mathbf{L}}_{1,0}$ *sont quasi-compacts sur* $C^1(J)$ *. La condition* (**A2**) *est donc vérifiée pour* $\mathbf{L}_{1,0}$ *.*

Preuve : Le théorème de Bougerol entraîne que les opérateurs sont quasi-compacts sur $\mathbf{H}_\alpha(J)$. Puisque les opérateurs $\mathbf{L}_{1,0}$ et $\widehat{\mathbf{L}}_{1,0}$ agissent sur $C^1(J)$, et puisque la décomposition spectrale implique les relations

$$\mathbf{R}_{1,0}[f] = \mathbf{L}_{1,0}[f] - \int f d\nu, \qquad \widehat{\mathbf{R}}_{1,0}[f] = \widehat{\mathbf{L}}_{1,0}[f] - \int f d\widehat{\nu}$$

on en déduit que les opérateurs $\mathbf{R}_{1,0}$ et $\widehat{\mathbf{R}}_{1,0}$ agissent également sur $C^1(J)$. Le lemme 5.1 implique maintenant que $\mathbf{R}_{1,0}$, agissant sur $C^1(J)$, a un rayon spectral strictement inférieur à 1. ■

Lemme 5.36 *L'opérateur* $\mathbf{L}_{1,1}$ *est quasi-compact sur* $C^0(J)$ *et* $C^1(J)$*. Sa valeur propre dominante est 1. Les conditions* (**A2**, **A3**) *sont donc vérifiées.*

Preuve : Nous étudions donc l'opérateur $\mathbf{L}_{1,1}$, et montrons qu'il satisfait une inégalité de Lasota-Yorke. L'inégalité se fera avec les deux normes $||.||_0$ et $||.||_{L^1}$.
Tout d'abord, observons que la boule unité $B = \{g \in C^0(J); \quad \|g\|_0 \leq 1\}$ est compacte dans $L^1(J)$: puisque les fonctions de B sont uniformément bornées, elles sont bornées dans $L^1(J)$ et uniformément equi-integrables.
Soit maintenant une fonction de $C^0(J)$ telle que $I(f) := \int_J f(u)du = 0$. Alors toute primitive F

de f est une fonction appartenant à $C^1(J)$ (elle satisfait $F(\frac{\pi}{2}) = F(-\frac{\pi}{2})$). Soit a un point de J, et soit l'opérateur $\mathbf{F}_n$ défini par

$$\mathbf{F}_n[f](x) := \int_a^x \mathbf{L}_{1,1}^n[f](u)du = \sum_{\ell\in\mathcal{L}^n} \delta_\ell \int_a^x |\ell'(u)| f\circ\ell(u)du = \sum_{\ell\in\mathcal{L}^n} \delta_\ell \int_{\ell(a)}^{\ell(x)} f(u)du$$

$$= \sum_{\ell\in\mathcal{L}^n} \delta_\ell [F\circ\ell(x) - F\circ\ell(a)] = \mathbf{L}_{1,0}^n[F](x) - \mathbf{L}_{1,0}^n[F](a) = \mathbf{R}_{1,0}^n[F](x) - \mathbf{R}_{1,0}^n[F](a).$$

Puisque $\mathbf{R}_{1,0} : C^1(J) \to C^1(J)$ a un rayon spectral ρ strictement inférieur à 1, on a pour tout $\rho < \kappa < 1$, et pour une constante K_1,

$$||\mathbf{F}_n[f]||_2 \le 2\|\mathbf{R}_{1,0}^n[F]\|_2 \le K_1 \cdot \kappa^n \|F\|_2.$$

F étant une primitive de f, on a $\|F\|_2 \le \pi\|f\|_1$. Puisque $\mathbf{F}_n[f]$ est une primitive de $\mathbf{L}_{1,1}^n[f]$, on a $||\mathbf{F}_n[f]||_2 \ge ||\mathbf{L}_{1,1}^n[f]||_1$, et finalement

$$\text{si} \quad I(f) = \int_J f(u)du = 0, \qquad \text{alors} \quad \|\mathbf{L}_{1,1}^n[f]\|_1 \le K_2 \cdot \kappa^n \|f\|_1. \tag{5.45}$$

Pour une fonction $f \in C^0(J)$, on peut écrire $f = g + I(f)$. Donc, g vérifie $I(g) = 0$ et $\|g\|_1 \le (\pi+1)\|f\|_1$. Ainsi, il existe une constante K_3,

$$\|\mathbf{L}_{1,1}^n[f]\|_1 \le \|\mathbf{L}_{1,1}^n[g]\|_1 + |I(f)| \cdot \|\mathbf{L}_{1,1}^n[\mathbf{1}]\|_1 \le K_3 \left[\kappa^n\|f\|_1 + t_n \cdot \|f\|_0\right] \tag{5.46}$$

avec $t_n = \|\mathbf{L}_{1,1}^n[\mathbf{1}]\|_1$. Cette relation montre que l'opérateur $\mathbf{L}_{1,1}$, quand il agit sur $C^0(J)$ vérifie une inégalité de Lasota-Yorke. On peut donc appliquer le théorème d'Hennion, et son spectre essentiel est donc strictement inférieur à 1.
Enfin, la relation

$$\int_J \mathbf{L}_{1,1}[f](u)du = \int_J f(u)du$$

montre que le rayon spectral de $\mathbf{L}_{1,1}$ vaut 1.
Montrons maintenant que 1 est une valeur propre simple de $\mathbf{L}_{1,1}$ sur $C^1(J)$. Soient φ et ψ deux fonctions propres associées à la valeur propre 1, avec $I(\varphi) = I(\psi) = 1$. Donc $I(\varphi - \psi) = 0$. Si on applique (5.45) à la fonction $\varphi - \psi$:

$$\|\varphi - \psi\|_1 = \|\mathbf{L}_{1,1}^n[\varphi - \psi]\|_1 \le K_2 \cdot \kappa^n \|\varphi - \psi\|_1 \tag{5.47}$$

on obtient l'égalité $\varphi = \psi$, de sorte que 1 est valeur propre simple.
Montrons enfin qu'il n'y a pas d'autre valeur propre de module 1 : soit λ une valeur propre de module 1, associée à une fonction propre f. Alors la relation $\int \mathbf{L}_{1,1}[|f|](u)du = \int |f(u)|du$ et l'inégalité triangulaire impliquent $\mathbf{L}_{1,1}[|f|] = |f|$. Donc, f est égale à $\alpha\varphi$ où α est de module 1. La relation $\mathbf{L}_{1,1}[f] = \lambda f$ implique pour tout $\ell \in \mathcal{L}$, et $x \in J$, l'égalité $\alpha\circ\ell(x)\varphi\circ\ell(x) = \lambda\alpha(x)\varphi(x)$. Enfin, en prenant comme x un point fixe pour une fonction ℓ (qui existe grâce à la propriété $(q4)$) on conclut que $\lambda = 1$. ∎

A ce stade de la preuve, on a obtenu les quasi-compacité des opérateurs $\mathbf{L}_{1,0}$, $\widehat{\mathbf{L}}_{1,0}$ sur $C^2(J)$ et de $\mathbf{L}_{1,1}$ sur $C^1(J)$, de même que l'analycité des applications $(s,t) \mapsto \mathbf{L}_{s,t}$, $(s,t) \mapsto \widehat{\mathbf{L}}_{s,t}$ sur $\mathcal{A}$. On en déduit en particulier un décomposition spectrale des opérateurs $\mathbf{L}_{s,t}$ et $\widehat{\mathbf{L}}_{s,t}$ pour (s,t) dans des voisinages de $(1,1)$ et $(1,0)$. Nous avons maintenant besoin de relier les spectres de ces opérateurs, ce que nous faisons dans la proposition suivante.

Proposition 5.37 *On a*
(i) Pour $(s,t) \in \mathcal{D}_0 \cup \mathcal{D}_1$, $\lambda(s,t) = \widehat{\lambda}(s,t)$.
(ii) Soit une paire réelle $(s,t) \in \mathcal{D}_0$, *alors on a la relation suivante entre* $\widehat{\mathbf{L}}_{s,t}$ *et* $\mathbf{L}_{1-s,t}$ *: pour tout* $f \in C^1(J)$, $g \in L^1$, *et tout* $n \in \mathbb{N}$,

$$\int \widehat{\mathbf{L}}^n_{s,t}[f] \cdot g dx = \int \mathbf{L}^n_{1-s,t}[g] \cdot f dx. \tag{5.48}$$

On en déduit l'égalité $\lambda(s,t) = \widehat{\lambda}(1-s,t)$.

Preuve : Montrons d'abord (i). Si on utilise l'involution Tilde définie en (5.42), et qu'on note $\widetilde{f}$ l'application de J dans J, définie pour une fonction $f : J \to J$ par $\widetilde{f}(\theta) = f(\widetilde{\theta})$, alors on a avec (5.43),

$$\widehat{\mathbf{L}}_{s,t}[f](\widetilde{\theta}) = \sum_{\ell \in \mathcal{L}} \delta_\ell^t \cdot |(\ell^{-1}(\widetilde{\theta}))'|^s \cdot f \circ \ell^{-1}(\widetilde{\theta}) = \sum_{\ell \in \mathcal{L}} \delta_\ell^t \cdot |\ell'(\theta)|^s \cdot f(\widetilde{\ell(\theta)})$$

$$= \sum_{\ell \in \mathcal{L}} \delta_\ell^t \cdot |\ell'(\theta)|^s \cdot \widetilde{f} \circ \ell(\theta) = \mathbf{L}_{s,t}[\widetilde{f}](\theta).$$

Ceci montre, pour (s,t) proche de $(0,1)$, l'égalité entre les valeurs propres dominantes $\lambda(s,t)$ et $\widehat{\lambda}(s,t)$.
Montrons maintenant (ii). Soit $f \in C^1(J)$, $g \in C^0$,et $n \in \mathbb{N}$. On a alors la relation suivante

$$\int_J \mathbf{L}^n_{s,t}[f](u) \cdot g(u) du = \sum_{\ell \in \mathcal{L}^n} \delta_\ell^t \int_J |\ell'(u)|^s f(\ell(u)) g(u) du = \sum_{\ell \in \mathcal{L}^n} \delta_\ell^t \int_J |\ell'(\ell^{-1}(v))|^{s-1} \cdot g \circ \ell^{-1}(v) \cdot f(v) dv$$

$$= \sum_{\ell \in \mathcal{L}^n} \delta_\ell^t \int_J |(\ell^{-1})'(v))|^{1-s} \cdot g \circ \ell^{-1}(v) \cdot f(v) dv = \int_J \widehat{\mathbf{L}}^n_{1-s,t}[g](v) \cdot f(v) dv.$$

La quasi-compacité des opérateurs $\mathbf{L}_{s,t}$ et $\mathbf{L}_{1-s,t}$ [pour (s,t) proche de (0,1)] sur $C^1(J)$ entraîne la relation

$$\lambda(s,t)^n = \widehat{\lambda}(1-s,t)^n \cdot [1 + O(\kappa^n)],$$

qui prouve (ii). ∎

En particulier, nous venons de montrer (avec la partie (ii) du théorème de Bougerol), la relation

$$2\gamma = -\lambda'_s(0,1) = \lambda'_s(1,1).$$

Nous devons maintenant montrer les propriétés finales, à savoir les comportement en fonction de s sur le demi-plan $\Re(s) \geq 1, s \neq 1$. Ceci est fait dans la proposition suivante. Notons qu'on peut se concentrer maintenant sur l'opérateur $\mathbf{L}_{s,0} = \mathbf{L}_{s,s}$.

Proposition 5.38 *On a les propriétés suivantes :*
(i) Pour tout s *tel que* $\Re s > 1$, *le rayon spectral* $r_0(s)$ *de* $\mathbf{L}_{s,0}$ *est strictement inférieur à 1.*
(ii) Sur la droite $\Re s = 1, s \neq 1$, *le rayon spectral* $r_0(s)$ *de* $\mathbf{L}_{s,0}$ *est strictement inférieur à 1.*
(iii) La dérivée seconde de l'application $s \mapsto \log \lambda(-s, 1-s)$ *est non-nulle en 0.*
Les conditions (**C1**), (**C3**) *sont donc vérifiées.*

Preuve : (i) Pour $\sigma := \Re(s) > 1$, on a

$$||\mathbf{L}_{s,0}[f]||_{L^1} = \int_J |\mathbf{L}_{s,0}[f](y)|\, dy \leq \sum_{\ell\in\mathcal{L}} \delta_\ell^\sigma \int_J |\ell'(y)|^\sigma |f(\ell(y))| dy \leq \sum_{\ell\in\mathcal{L}} \delta_\ell^\sigma \int_J |\ell'(\ell^{-1}(x))|^{\sigma-1} |f(x)| dx$$

$$\leq \phi^{2(\sigma-1)} \sum_{\ell\in\mathcal{L}} \delta_\ell^\sigma \cdot ||f||_{L^1} = \phi^{2(\sigma-1)} \frac{2^{1-2\sigma}}{1-2^{1-2\sigma}} ||f||_{L^1} \leq \left(\frac{\phi}{2}\right)^{2(\sigma-1)} \cdot ||f||_{L^1}.$$

On en déduit que le rayon spectral de $\mathbf{L}_{s,0}$ sur $L^1(J)$ est au plus $(\phi/2)^{2(\sigma-1)}$, ce qui est strictement inférieur à 1 pour $\Re s > 1$. Grâce au lemme 5.1, la même propriété est vérifiée pour $\mathbf{L}_{s,0}$ agissant sur $C^0(J)$ ou $C^1(J)$.
(ii) Les arguments développés au point (i) impliquent que pour $s = 1 + it$, le rayon spectral de $\mathbf{L}_{s,0}$ sur $C^0(J)$ est au plus 1. Il reste maintenant deux étapes dans la preuve de (ii). Montrons tout d'abord que le rayon spectral essentiel de $\mathbf{L}_{s,0}$ sur $C^0(J)$ est strictement inférieur à 1.
L'inégalité $||\mathbf{L}_{1+it,0}[f]||_0 \leq ||\mathbf{L}_{1,0}[f]||_0$, la relation (5.46) appliquée à la fonction $f_1 := |f|$ et le théorème d'Hennion montrent que le spectre essentiel de $\mathbf{L}_{1+it,0}$ sur $C^0(J)$ est strictement inférieur à 1.
Montrons maintenant qu'il n'y a pas de valeur propre de module 1.
Supposons que pour $s = it + 1$, le rayon spectral de $\mathbf{L}_{s,0}$ soit égal à 1. Puisque le rayon spectral essentiel est strictement inférieur à 1, $\mathbf{L}_{s,0}$ à une valeur propre égale de module 1. Si on suit la preuve de la proposition 9 de [Val01], on obtient l'existence d'une fonction μ avec $|\mu| = 1$, telle que pour tout $n \in \mathbb{N}$ et pour tout $\ell \in \mathcal{L}^n$,

$$\delta_\ell^{it} \cdot |\ell'|^{it} \cdot \mu \circ \ell = \mu. \tag{5.49}$$

Il existe donc une fonction bornée ζ telle que, pour tout $\ell \in \mathcal{L}^n$, il existe un entier $J(\ell)$ de sorte que

$$\log \delta_\ell + \log |\ell'| = \zeta - \zeta \circ \ell + \frac{2\pi}{t} J(\ell).$$

Ceci entraîne que J est additif, c'est-à-dire $J(\ell_1 \circ \ell_2) = J(\ell_1) + J(\ell_2)$, et donc $\mathbb{E}_n[J] = n\mathbb{E}[J]$. Si on note Γ l'exposant de Lyapunov de l'ensemble $\mathcal{N}$, qui est donc égal à $2\log 2 + \gamma$. Alors la partie (ii) du théorème de Bougerol ainsi que 4.14 entraînent que

$$\frac{2\pi}{t}\mathbb{E}[J] = -2\Gamma, \quad \mathbb{E}[(\log ||\mathcal{N}_1 \cdots \mathcal{N}_n|| - n\Gamma)^2] < K$$

pour une constante K. Or, le Lemme 5.3 p123 dans [BL85], stipule que sous les conditions du théorème de Bougerol, l'espérance $\mathbb{E}[(\log ||\mathcal{N}_1 \cdots \mathcal{N}_n|| - n\Gamma)^2]$ n'est pas bornée. Cette contradiction termine la preuve du point (ii). Grâce au lemme 5.1, la même propriété est vérifiée sur $C^0(J)$.

(iii) La quantité $l(s) = \lambda(1-s, -s)$ est la valeur propre dominante de l'opérateur $\mathbf{L}_{1-s,-s,0}$, qui est exactement l'opérateur utilisé dans le livre de Bougerol et Lacroix. D'après le lemme 5.2 de [BL85], on vérifie $l''(0) = \Gamma^2$ (et donc $(\log\lambda)''(1,0) = 0$)) si et seulement si l'espérance $\mathbb{E}[(\log ||\mathcal{N}_1 \cdots \mathcal{N}_n|| - n\Gamma)^2]$ est bornée, ce qui n'est pas le cas d'après le lemme précédent. ■

5.3.4 Théorèmes 5.40, 5.42 et 5.42

L'analyse que nous venons de présenter s'applique au système $S_{\underline{\mathcal{L}}}$, comme nous le montrons dans la proposition suivante.

Proposition 5.39 *Le système $S_{\underline{\mathcal{L}}}$ satisfait les ensembles de conditions Σ_3 et Σ_4.*

Preuve : Chaque matrice $\mathcal{M}_{[q]} \in \mathcal{M}$ est symétrique, de déterminant -1. Elle a deux valeurs propres distinctes, λ_q^- et λ_q^+, vérifiant

$$|\lambda_q^+| = \frac{1}{2}(|q| + \sqrt{q^2+4}) = |\lambda_q^-|^{-1}.$$

La norme euclidienne de la matrice $\mathcal{M}_{[q]}$ est donc égale à $|\lambda_q^+|$. Puisque chaque quotient q satisfait $|q| < 1$, on a

$$||\mathcal{M}_{[q]}|| \leq \phi, \qquad ||\mathcal{M}_{[q]}^{-1}|| \leq \phi, \qquad L^+(\mathcal{M}_{[q]}) \leq \log \phi.$$

Ceci prouve ($q3$). La relation (4.14) implique ($q5$). Soit maintenant S_n la puissance $n$ième d'une matrice $\mathcal{M}_{[q]}$, dont les valeurs propres sont de module $|\lambda_q^+|^n, |\lambda_q^-|^n$. Alors le lemme III.1.4 de [BL85] implique ($q1$). Finalement, l'existence de vecteurs propres pour $\mathcal{M}_{[q]}$ entraîne l'existence d'un point fixe pour h, et donc pour ℓ : on a donc ($q4$).
Supposons maintenant qu'il existe W comme dans la condition ($q2$). Alors, pour tout $S \in \mathcal{S}$, il existe une permutation σ_S de $[1..k]$ pour laquelle $S(V_i) = V_{\sigma_S(i)}$, de sorte que chaque V_i est invariant par toutes les matrices $\mathcal{M}_{[q]}^{k!}$ relatives à $\mathcal{M}_{[q]} \in \mathcal{N}$. Ceci implique que $k = 2$ et que $\{V_1, V_2\}$ est une base pour toutes les matrices $\mathcal{M}[q]^2$. On en déduit donc que chaque paire de matrice $\mathcal{M}_1^2, \mathcal{M}_2^2$ commute, ce qui n'est pas vrai. La condition ($q2$) est donc vérifiée.
Enfin soit un quotient q. La quantité $\gamma_q(x) := \ell_q''(x)/\ell_q'(x)$ est

$$\gamma_q(x) = \frac{2q(\tan^2 x + q\tan x - 1)}{1 + (q + \tan x)^2},$$

dont les extremas sont $\pm q\sqrt{q^2+4}$. Puisque chaque quotient q vérifie $-1 < q < 1$, on en déduit

$$|\gamma_q(x)| \leq \sqrt{5}, \quad \forall q \in \mathcal{Q} \text{ et } x \in J,$$

et donc la condition ($q6$) est vérifiée. ■

De la proposition 5.39, du théorème 5.25 et des propositions 5.7 et 5.11 on déduit finalement les principaux résultats sur l'algorithme LSB.

Théorème 5.40 *Soit $\mathcal{L}$ l'algorithme LSB, et $\mathbb{E}_N$ l'espérance définie sur l'ensemble Ω_N des entrées valides de l'algorithme. Alors le nombre moyen d'itérations $\mathbb{E}_N[I]$ de l'algorithme est asymptotiquement linéaire en N,*

$$\widetilde{\mathbb{E}}_N[I] \sim \mathbb{E}_N[I] \sim \frac{1}{2-\gamma_0} \cdot N,$$

où γ_0 est l'exposant de Lyapunov de l'ensemble de matrices $\mathcal{M}$. Plus généralement, si c est un coût à croissance modérée, et C la variable aléatoire associée, alors l'espérance $\mathbb{E}_N[C]$ est asymptotiquement linéaire en N,

$$\widetilde{\mathbb{E}}_N[C] \sim \mathbb{E}_N[C] \sim \frac{1}{2-\gamma_0} \cdot \mu_{\mathcal{L}}(c) \cdot N,$$

où $\mu_{\mathcal{L}}(c)$ est la valeur moyenne du coût c.

Si par exemple on considère les coûts k, s, c_a correspondant respectivement aux nombres de décalages, de soustractions et d'occurrences d'un quotient $a/2^k$, alors on obtient

$$\mu(k) = 2, \qquad \mu(s) = \frac{5}{2}, \qquad \mu(c_a) = \frac{4}{3} \cdot 4^{-\ell(a)}.$$

Il faut noter que le calcul du nombre de soustractions s est particulier. Le calcul d'une division LSB se fait en effet en deux temps : on calcule tout d'abord le quotient non-centré par soustractions successives, le nombre de soustractions effectuées étant alors le nombre de 1 dans l'écriture en base 2 du numérateur a du quotient, puis on effectue une éventuelle soustraction supplémentaire pour centrer le quotient. Le calcul de $\mu_{\mathcal{L}}(s)$ se fait donc sur l'ensemble $\{q = a/2^k, k \geq 1, a \text{ impair }, 0 < a < 2^{k+1}\}$, sur lequel on compte le nombre moyen de 1 dans l'écriture binaire de a, auquel on ajoute 1 pour les quotients vérifiant $q > 1$.

Les coûts k et s interviennent également dans la complexité en bit de l'algorithme. En effet, on obtient celle-ci des propositions 4.3, 5.9 et 5.10. La première et la dernière de ces propositions permettent d'obtenir le comportement asymptotique du coût approché $\widehat{B}$, donné par (1.27)

$$\widehat{B}(u,v) := \frac{1}{2} \sum_{i=1}^{p} \ell(d_i) \times \log_2(u_i^2 + u_{i+1}^2).$$

L'inégalité (1.28) montre que la différence entre ces deux coûts est $O(Q)$ avec

$$Q(u,v) = \sum_{i=1}^{p} \ell(d_i),$$

qui est linéaire en N. L'approximation est donc valable, et on en déduit le théorème suivant.

Théorème 5.41 *Soit $\mathcal{L}$ l'algorithme LSB, et $\mathbb{E}_N$ l'espérance définie sur l'ensemble Ω_N des entrées valides de l'algorithme. Alors la complexité en bit moyenne $\mathbb{E}_N[B]$ est asymptotiquement quadratique en N,*

$$\mathbb{E}_N[B] = \frac{1}{2} \frac{\mu_{\mathcal{L}}(k) + \mu_{\mathcal{L}}(s)}{2-\gamma_0} \cdot N^2$$

où k et s sont les coûts liés au nombre de décalages et au nombre de soustractions.

On obtient un résultat similaire sur l'évolution des continuants, dans le modèle continu.

Théorème 5.42 *Soit $S_{\mathcal{L}}$ le système dynamique LSB. Soit $\mathcal{Q}_n(x)$ le n-ème continuant du nombre 2-adique x. Si $||.||$ est la norme euclidienne et que la boule unité $\mathcal{B}$ est munie de la distribution uniforme, alors la variable aléatoire $\log ||\mathcal{Q}_n||$ suit asymptotiquement une loi gaussienne, avec une vitesse de convergence optimale en $O(1/\sqrt{n})$. De plus, l'espérance et la variance satisfont*

$$\mathbb{E}[\log ||\mathcal{Q}_n||] = (2+\gamma_0) \cdot n + a + O(\kappa^{-n}), \mathrm{Var}[\log ||\mathcal{Q}_n||] = b \cdot n + c + O(\kappa^{-n}),$$

où a, b, c, κ sont des constantes avec $b > 0, \kappa < 1$.

5.3.5 Discussion des résultats

Nous venons de fournir une analyse complète de l'algorithme LSB ainsi que du développement en fraction continue 2-adique. La mise en parallèle des deux analyses peut d'ailleurs surprendre, comme nous l'avons fait remarquer au paragraphe 2.8. On observe en effet une différence entre le modèle continu et le modèle discret, plus précisément entre les constantes des théorèmes 5.40 et 5.42. Usuellement, ces constantes sont inverses l'une de l'autre.

En effet, si (u, v) est une entrée de taille N de l'algorithme, alors au bout de $N/(2-\gamma_0)$ itérations l'algorithme s'arrête, et les coefficients de Bezout obtenus avec la version étendue de l'algorithme sont de taille N. Par contre, la taille d'un rationnel obtenu après $N/(2-\gamma_0)$ étapes du développement en fraction continue est $N \cdot (2+\gamma_0)/(2-\gamma_0)$ et non N. Le processus discret et son extension continue n'ont donc pas le même comportement.

Même si chacun des deux comportements possède son explication propre, on ne dispose pas à l'heure actuelle d'explication satisfaisante du phénomène.

Enfin, on peut noter que l'analyse présentée ici s'applique également à l'algorithme LSB non-centré. Nous avons remarqué dans la section 1.4 que cet algorithme ne termine pas toujours. Il est cependant possible de corriger ce problème, en modifiant les tests d'arrêt de l'algorithme. Dans ce cas, on manipule un ensemble de matrices de la forme

$$\{\begin{pmatrix} 0 & 1 \\ 1 & q \end{pmatrix}, \quad q = \frac{a}{2^k}, \quad k \geq 1, \quad a \text{ impair}, \quad 0 < a < 2^{k+1}\}.$$

L'exposant de Lyapunov binaire $\bar{\gamma}_0$ relatif à cet ensemble vaut alors $\bar{\gamma}_0 \sim 0.651$ ce qui est environ 13 fois supérieur à la constante relative à l'algorithme centré. On en déduit que même quand on force l'arrêt de l'algorithme non-centré, celui-ci est moins rapide en moyenne que sa version centrée.

5.4 Analyse dynamique des algorithmes interrompus

L'étude des algorithmes interrompus est dans un sens différente des autres analyses. L'essentiel de l'analyse de l'opérateur a été fait au paragraphe 5.2 lors de l'étude des algorithmes α-euclidiens. Les coûts étudiés sont plus complexes que ceux considérés jusqu'à présent. Ceci nous amènera, comme nous le verrons plus loin, à étudier des séries dont les pôles ne sont plus exactement en $s = 1$, mais sont localisés autour de cette valeur. L'utilisation qu'on fera du théorème taubérien sera modifiée, puisqu'ici c'est la localisation précise du pôle qui sera importante.

Notons que pour l'étude de l'algorithme d'Euclide, on peut choisir entre plusieurs espaces fonctionnels. Le choix des fonctions à variation bornée ne s'impose en réalité que lorsque les branches sont incomplètes, ce qui n'est pas le cas ici. On peut par exemple se placer sur un ensemble de fonctions analytiques : c'est dans ce cadre qu'ont été faites les analyses de Vallée (voir [Val03, Val00]), lesquelles se sont inspirées des travaux de D. Mayer sur l'opérateur de transfert (voir [May91]). Quand on choisit ce cadre fonctionnel, on considère l'espace $\mathcal{A}_\infty(I)$ des fonctions holomorphes sur un domaine $\mathcal{V}$ comprenant l'intervalle $I = [0, 1]$, muni de la norme $||.||_\mathcal{A}$:

$$||f||_\mathcal{A} = \sup_{u \in V} |f(u)|.$$

Une des particularités de cet espace est que l'opérateur de Perron-Frobenius $\mathbf{G}$ y est compact : ceci signifie entre autres que tout le spectre de l'opérateur sur $\mathcal{A}$ forme un ensemble discret (sauf éventuellement en 0 qui peut être un point d'accumulation). En d'autres termes, le rayon essentiel r_{ess} est nul, ce qui implique bien évidement la quasi-compacité ainsi que le saut spectral

recherché. Enfin, les arguments employés pour obtenir la nature du spectre sur le cercle unité reposent en partie sur l'utilisation de la méthode des cônes de Krasnoselskii [Kra64]. Les résultats obtenus dans ce cadre sont résumés dans le théorème suivant.

Théorème F.[Vallée]*Soit $S = (I, T)$ un système dynamique de l'intervalle. On dit que S satisfait l'ensemble Σ_3 de conditions s'il vérifie les propriétés $(r1)$ et $(r2)$ suivantes :*

$(r1)$ L'ensemble de branches inverses $\mathcal{H}$ est constitué de Transformations Linéaires Fractionnaires à coefficients entiers, et contient, étant donné un entier $A > 0$, un sous-ensemble de la forme

$$\mathcal{A} := \left\{ h; \quad h(x) = \frac{A}{c+x}, \qquad \text{avec des entiers } c \to \infty \right\}.$$

$(r2)$ Il existe un disque ouvert $\mathcal{V}$ qui contient I, et un réel $\alpha < 2$ tels que

(i) chaque branche inverse $h \in \mathcal{H}$ a une continuation analytique sur $\mathcal{V}$, et envoie la clôture $\bar{\mathcal{V}}$ de $\mathcal{V}$ dans $\mathcal{V}$,

(ii) pour toute branche $h \in \mathcal{H}$, il existe $\delta(h) < 1$ tel que la continuation analytique de $|h'|$, notée $\widetilde{h}$, satisfait $0 < |\widetilde{h}(z)| \leq \delta(h)$, pour tout z dans $\mathcal{V}$,

(iii) la série $\displaystyle\sum_{h \in \mathcal{H}} \left| \frac{\delta(h)}{\det(h)} \right|^s$ converge sur le demi-plan $\Re(s) > \alpha$.

Si S satisfait Σ_3, alors quand il agit sur l'espace $\mathcal{A}_\infty(I)$, l'opérateur $\mathbf{H}_{s,w}$ vérifie les propriétés (**A1**), (**A2**), (**A3**), (**B0**), (**B1**), (**C1**) *et* (**C2**).

Les deux approches -espaces $BV(I)$ ou $\mathcal{A}_\infty(I)$- sont dans un sens des approches "extrêmes" : l'espace $BV(I)$ est peut-être le plus grand espace pouvant être choisi (il doit être suffisamment grand pour traiter les discontinuités) alors que l'espace $\mathcal{A}_\infty(I)$ est particulièrement petit (le spectre de l'opérateur y est très réduit). On peut choisir un compromis entre ces deux espaces en se plaçant sur un espace de fonctions continues, comme le font Vallée et Baladi dans [BV04]. Cet espace présente l'avantage d'être relativement "maniable", et est en outre l'espace adéquat lorsqu'on veut obtenir des résultats en distribution.

Afin de ne pas compliquer davantage la situation, nous nous placerons comme précédemment sur l'espace des fonctions à variations bornées. Rappelons que l'opérateur $\mathbf{G}_{s,w}$ relatif au système $S_{\mathcal{E}}$, vérifie toutes les conditions énoncées au premier paragraphe de ce chapitre. Une condition que nous n'avons pas encore utilisé est la condition (**C4**), qui porte sur l'application $s \mapsto \log \lambda(s, 0) := \Lambda(s, 0)$, et qui est

$$\Lambda''_s(s, 0) > 0.$$

Cette application est vérifiée (on peut consulter [Clé00] par exemple).

5.4.1 Nombre d'itérations

Le principal résultat de ce paragraphe est énoncé dans le théorème suivant.

Théorème 5.43 *Soit $t \in [0, 1]$ et soit $\mathcal{E}_t$ l'algorithme d'Euclide interrompu correspondant, défini dans le paragraphe 1.3.2. Soit $\mathbb{P}_N$ la probabilité définie sur l'ensemble Ω_N. Pour tout $\varepsilon > 0$, il existe $K < 1$ tel que les deux variables aléatoires I et I_t sont reliées asymptotiquement par*

$$\mathbb{P}_N \left[\left| \frac{I_t}{I} - (1-t) \right| > \varepsilon \right] = O(K^N).$$

De plus, les espérances de ces variables vérifient asymptotiquement

$$\mathbb{E}_N[I_t] \sim (1-t)\,\mathbb{E}_N[I] \sim (1-t)\frac{2\log 2}{h(\mathcal{E})}\cdot N,$$

$h(\mathcal{E})$ étant l'entropie du système dynamique sous-jacent à l'algorithme d'Euclide.

Remarque : la preuve de ce théorème ne reposant pas sur des résultats nouveaux sur les opérateurs, nous énoncerons les résultats successifs directement en termes de séries de Dirichlet, et non plus sous la forme : la condition (**A1**) (par exemple) est vérifiée.

Comme nous l'avons vu au paragraphe 4.3, la preuve de ce théorème repose sur l'étude du coût $M_{t,\delta,\gamma}$ défini en (1.19) et donné par

$$M_{t,\delta,\gamma}(u,v) = \left(\frac{u_{\lfloor \delta p\rfloor}}{u_0^t}\right)^{\gamma}.$$

La série génératrice associée à ce coût est reliée à l'opérateur $\mathbf{G}_{s,0}$ par (voir proposition 4.4)

$$T_M(2s) = \sum_{p\geq 0} \mathbf{G}_{s^-,0}^{p-\lfloor \delta p\rfloor} \circ \mathbf{G}_{s^+,0}^{\lfloor \delta p\rfloor}[1](0), \tag{5.50}$$

où s^+ et s^- sont donnés par

$$s^+ = s + t\gamma, \qquad s^- = s - \beta\gamma.$$

Rappelons qu'ici, δ dénote une fraction de la période, t est le paramètre d'interruption de l'algorithme et γ provient de l'inégalité de Markov qui a conduit au coût $M_{t,\delta,\gamma}$. Supposons maintenant que δ est un rationnel donné par

$$\delta = \frac{c}{c+d}.$$

L'indice de sommation p apparaissant dans (5.50) s'écrit

$$p = (c+d)k + j, \quad j < c+d,$$

et on pose

$$j' = \lfloor \delta p\rfloor,$$

ce qui fait que (5.50) devient finalement

$$T_M(2s) = \sum_{j=0}^{c+d-1} \mathbf{G}_{s^-,0}^{j-j'} \circ \left(\sum_{k\geq 0} \mathbf{G}_{s^-,0}^{dk} \circ \mathbf{G}_{s^+,0}^{ck}\right) \circ \mathbf{G}_{s^+,0}^{j'}[f](0), \tag{5.51}$$

Les singularités de la série sont maintenant données par les deux lemmes suivants.

Lemme 5.44 *Soit s un nombre réel $s>1$. Soit $r(s)$ le rayon spectral de l'opérateur $\mathbf{G}_{s,0}$. Soit $\phi(s)$ la fonction*

$$\phi(s) := r^d(s^-)\; r^c(s^+). \tag{5.52}$$

Alors pour tout $0<\gamma<1$, l'équation $\phi = 1$ a une unique solution réelle ρ. Cette solution appartient à l'intervalle $[1-t\gamma, 1+\beta\gamma]$, et la série $T_M(s)$ est analytique sur le demi-plan $\Re(s) > \rho$.

Preuve : La fonction ϕ est définie pour $s > 1 + \beta\gamma$. Notons que si une solution ρ existe, alors on vérifie $\rho^- < 1 < \rho^+$. Il est donc suffisant d'étudier $\phi(s)$ sur l'intervalle

$$[1 - t\gamma, 1 + \beta\gamma], \tag{5.53}$$

sur lequel ϕ est définie : l'inégalité $0 < \gamma < 1$ implique $1 - t\gamma > 1 + \beta\gamma$. Puisque $s \mapsto r(s)$ est une application strictement décroissante sur l'axe réel, il en est de même pour $s \mapsto \phi(s)$, et la suite d'inégalités

$$\phi(1 - t\gamma) = r^d(1 - \gamma) > \lambda(1, 0) = 1 > r^c(1 + \gamma) = \phi(1 + \beta\gamma)$$

montre que l'équation a une unique solution ρ.

On déduit de (5.51), en considérant les normes des opérateurs sur l'espace BV,

$$|T_M(2s)| \leq ||f||_{BV} \left(\sum_{j=0}^{c+d-1} ||\mathbf{G}_{s^-,0}^{j-j'}||_{BV} \; ||\mathbf{G}_{s^+,0}^{j'}||_{BV} \right) \left(\sum_{k \geq 0} ||\mathbf{G}_{s^-,0}^{dk}||_{BV} \, ||\mathbf{G}_{s^+,0}^{ck}||_{BV} \right).$$

La partie droite de l'inégalité définit une série de terme général équivalent à $r(s^-)^{dk} r(s^+)^{ck}$. Cette série est convergente quand $\phi(s) = r(s^-)^d r(s^+)^c$ est inférieur à 1. ∎

Lemme 5.45 *Soit γ positif et suffisamment proche de 0. Alors la série $T_M(s)$ a un pôle d'ordre 1 en $s = \rho$. De plus, $T_M(s)$ est analytique sur le demi-plan $\{\Re(s) \geq \rho, s \neq \rho\}$.*

Preuve : Si γ est suffisamment petit, les valeurs propres dominantes $\lambda(s^+, 0)$ et $\lambda(s^-, 0)$ des opérateurs $\mathbf{G}_{s^+,0}$ et $\mathbf{G}_{s^-,0}$ sont bien définies. La décomposition spectrale de ces opérateurs s'étend à la série $T_M(s)$, dont le terme dominant est obtenu en remplaçant chaque occurrence de $\mathbf{G}_{s,0}$ par $\lambda(s, 0)\mathbf{P}_{s,0}$ dans (5.51). Il est donc de la forme $T_M^+(s)\, \mathbf{P}_{s^-,0} \circ \mathbf{P}_{s^+,0}[1](0)$ avec

$$\begin{aligned} T_M^+(2s) &= \left(\sum_{j=0}^{c+d-1} \lambda^{j-j'}(s^-, 0)\lambda^{j'}(s^+, 0) \right) \left(\sum_{k \geq 0} (\lambda^d(s^-, 0)\lambda^c(s^+, 0))^k \right), \\ &= \frac{1}{1 - \phi(s)} \left(\sum_{j=0}^{c+d-1} \lambda^{j-j'}(s^-, 0)\lambda^{j'}(s^+, 0) \right). \end{aligned} \tag{5.54}$$

Les pôles dominants de la série provenant de $T_M^+(s)$, on en déduit qu'on a un pôle dominant en $s = \rho$ si ρ est solution de l'équation $\phi(s) = 1$. On a de plus, près de $s = \rho$,

$$T_M(2s) \sim \frac{1}{s - \rho} \, \frac{-1}{\phi'(\rho)} \sum_{j=0}^{c+d-1} \lambda^{j-j'}(\rho^-, 0)\lambda^{j'}(\rho^+, 0)\mathbf{P}_{\rho^-,0} \circ \mathbf{P}_{\rho^+,0}[f](0). \tag{5.55}$$

L'application $s \mapsto \phi(s)$ hérite de la fonction $s \mapsto \lambda(s, 0)$ ses propriétés le long des axes horizontaux et verticaux, de sorte que $\phi(s) < 1$ pour $\Re(s) = \rho, s \neq \rho$. ∎

En appliquant le théorème taubérien à $T_M(s)$, quand γ est positif, on obtient la proposition suivante.

Proposition 5.46 *Pour tout γ strictement positif et suffisamment proche de 0, pour tout $t, \delta \in [0,1]$, il existe ρ (qui dépend de t, γ et δ) tel que l'espérance $\mathbb{E}_N[M_{t,\delta,\gamma}]$ du coût $M_{t,\delta,\gamma}$ est donnée par*

$$\mathbb{E}_N\left[\left(\frac{a_{\lfloor \delta p \rfloor}}{a_0^t}\right)^\gamma\right] = O(2^{N(\rho-1)}).$$

Nous nous intéressons maintenant au cas où δ est proche de $(1-t)$.

Lemme 5.47 *Soit $\delta = (1-t)+\varepsilon$, avec $\varepsilon > 0$. Il existe $\gamma > 0$ tel que l'unique solution ρ de l'équation $\phi(s) = 1$ est strictement inférieure à 1. On peut de plus choisir $1-\rho = \Omega(\varepsilon^2)$.*

Preuve : Supposons que $\delta = \frac{c}{c+d}$ est de la forme

$$\delta = (1-t)+\varepsilon.$$

On a alors

$$\frac{c}{d} = \frac{1-t+\varepsilon}{t-\varepsilon} = \frac{\beta+\varepsilon}{t-\varepsilon}.$$

L'équation $\phi(s) = 1$ s'écrit alors, en posant $\Lambda(s) := \log \lambda(s,0)$, de la manière suivante

$$\Phi(s) := \frac{|\Lambda(s^-)|}{|\Lambda(s^+)|} = \frac{c}{d} = \frac{\beta+\varepsilon}{t-\varepsilon}.$$

Dans un voisinage de $s = 1$, la fonction $\Phi(s)$ est strictement décroissante. Il est donc suffisant de montrer qu'il existe $\gamma > 0$ (qui dépend donc de ε), tel que

$$\Phi(1) < \Phi(\rho) = \frac{\beta+\varepsilon}{t-\varepsilon}.$$

La fonction $s \mapsto \Phi(s)$ vérifie

$$\Lambda(1) = 0, \qquad \Lambda'(1) < 0, \qquad \Lambda''(1) > 0,$$

de sorte que pour γ suffisamment petit, on a

$$\Phi(1) = \frac{|\Lambda(1-\beta\gamma)|}{|\Lambda(1+t\gamma)|} < \frac{\beta}{t}\,\frac{1+\beta\varepsilon_1}{1-t\varepsilon_1} < \frac{\beta+\varepsilon_1}{t-\varepsilon_1}, \qquad \text{avec} \qquad \varepsilon_1 := \frac{3\gamma}{4}\frac{\Lambda''(1)}{|\Lambda'(1)|}.$$

On peut donc prendre

$$\gamma = \varepsilon\frac{|\Lambda'(1)|}{\Lambda''(1)}$$

de sorte que

$$\varepsilon_1 = \frac{3\varepsilon}{4} < \varepsilon \qquad \text{et} \quad \Phi(1) < \Phi(\rho).$$

On a dans ce cas $\rho < 1$, ce qui prouve la première partie du lemme. On souhaite maintenant évaluer $(1-\rho)$ en tant que fonction de ε. Tout d'abord,

$$1-\rho \sim \frac{|\Phi(1)-\Phi(\rho)|}{\Phi'(1)}.$$

On a donc

$$|\Phi(1)-\Phi(\rho)| \geq \frac{\beta+\varepsilon}{t-\varepsilon} - \frac{\beta+\varepsilon_1}{t-\varepsilon_1} \geq \frac{\varepsilon-\varepsilon_1}{t^2} = \frac{\varepsilon}{4t^2}.$$

D'un autre côté, en utilisant la dérivée logarithmique et le fait que près de $x = 1$ on a $|(\Lambda'/\Lambda)(1+x)| \sim (1/x)$, on obtient

$$\frac{|\Phi'(1)|}{\Phi(1)} = \left|\frac{\Lambda'}{\Lambda}(1-\beta\gamma)\right| + \left|\frac{\Lambda'}{\Lambda}(1+t\gamma)\right| \qquad \text{so that} \qquad |\Phi'(1)| \sim \frac{1}{t^2\gamma}$$

et finalement

$$1 - \rho \geq \frac{\varepsilon\gamma}{4} = \frac{\varepsilon^2}{4}\frac{|\Lambda'(1)|}{\Lambda''(1)}.$$

■

Il nous reste maintenant à traiter le cas où γ est négatif. Plus précisément, considérons le coût $M_{t,\delta,\gamma}$ avec les paramètres $(t, \delta, -\gamma)$, $\gamma > 0$. On observe que $s^-(t,\gamma) = s^+(1-t,-\gamma)$. De plus, même si les coûts $M_{t,\delta,\gamma}$ associés aux paramètres $(t,\delta,-\gamma)$ et $(1-t, 1-\delta, \gamma)$ sont à priori différents, les termes dominants de leurs séries génératrices sont les mêmes, et sont donnés par

$$\sum_{k\geq 0} \mathbf{G}^{dk}_{s^-,0} \circ \mathbf{G}^{ck}_{s^+,0},$$

et la fonction ϕ est donc la même dans les deux cas. Si on note $\rho(t,\delta,\gamma)$ l'unique solution de l'équation $\phi(s) = 1$, où ϕ est relative aux paramètres (t,δ,γ), alors l'égalité

$$\rho(t,\delta,-\gamma) = \rho(1-t, 1-\delta, \gamma)$$

et le lemme 5.47 impliquent que pour tout $\varepsilon > 0$, il existe $\gamma > 0$ tel que pour tout t,

$$\rho(t, 1-t-\varepsilon, -\gamma) = \rho(1-t, t+\varepsilon, \gamma) < 1.$$

Ceci entraîne que pour tout $\varepsilon > 0$, il existe $\gamma > 0$ tel que

$$\rho^+ = \rho(t, 1-t+\varepsilon, \gamma)$$

et

$$\rho^- = \rho(t, 1-t-\varepsilon, -\gamma)$$

sont tous les deux inférieurs à 1. Ainsi, on a

$$\mathbb{P}_N\left[\left|\frac{I_t}{T} - (1-t)\right| > \varepsilon\right] = O(K^N),$$

avec $K = 2^{\max(\rho^+,\rho^-)-1}$. Nous avons donc montré la première partie du théorème 5.43.

Enfin, appelons Q_t la variable aléatoire $Q_t := |I_t - (1-t)I|$. Considérons pour $\varepsilon >$ l'évènement $A(\varepsilon) := [Q_t \geq \varepsilon I]$. Le pire des cas de l'algorithme d'Euclide implique qu'on a sur Ω_N l'égalité $I = O(N)$. Ceci entraîne

$$\mathbb{E}_N[Q_t] \leq KN\ (\mathbb{P}_N[A(\varepsilon)] + \varepsilon)\,,$$

ce qui, avec la première partie du théorème 5.43 implique

$$\mathbb{E}_N[I_t] \sim (1-t)\mathbb{E}_N[I].$$

Le théorème 5.43 est donc entièrement montré.

5.4.2 Complexité en bits

Il est maintenant aisé d'étendre le résultat précédent à des coûts plus significatifs que le nombre d'itérations, en particulier à la complexité en bits. La preuve du théorème suivant se fait en deux temps. On exhibe tout d'abord la complexité en bits moyenne de l'algorithme interrompu $\underline{\mathcal{E}}_\delta$ de paramètre δ, puis on montre que celle de l'algorithme $\mathcal{E}_\delta$ est la même, en vertu du résultat précédent.

Théorème 5.48 *Soit $t \in [0,1]$ et soit $\mathcal{E}_t$ l'algorithme d'Euclide interrompu correspondant. Soit $\mathbb{E}_N$ l'espérance définie sur l'ensemble Ω_N. Alors les variables aléatoires B_t, X_t et BX_t sont asymptotiquement reliées aux variables B, X et BX relatives à l'algorithme d'Euclide par*

$$\begin{array}{cccc} (i) & \mathbb{E}_N[B_t] & \sim & (1-t^2)\,\mathbb{E}_N[B] \\ (ii) & \mathbb{E}_N[X_t] & \sim & (1-t)^2\,\mathbb{E}_N[X] \\ (iii) & \mathbb{E}_N[BX_t] & \sim & \frac{1}{3}(1-t)(3-t)\,\mathbb{E}_N[BX] \end{array}$$

Preuve : Le travail consiste en réalité à relier les complexités en bits des deux algorithmes interrompus $\mathcal{E}_t$ et $\underline{\mathcal{E}}_\delta$, avec $t = \delta$.
Les coûts "approximatifs" associés à ce dernier algorithme, $\underline{\widehat{B}}_\delta$, $\underline{\widehat{X}}_\delta$ et $\underline{\widehat{BX}}_\delta$, vérifient asymptotiquement

$$\begin{array}{ccc} \mathbb{E}_N[\underline{\widehat{B}}_\delta] & \sim & (1-\delta^2)\,\mathbb{E}_N[\widehat{B}], \\ \mathbb{E}_N[\underline{\widehat{X}}_\delta] & \sim & (1-\delta)^2\,\mathbb{E}_N[\widehat{X}], \\ \mathbb{E}_N[\underline{\widehat{BX}}_\delta] & \sim & \frac{1}{3}(1-\delta)(3-\delta)\,\mathbb{E}_N[\widehat{BX}]. \end{array} \tag{5.56}$$

En effet, les séries génératrices associées aux coûts $\underline{\widehat{B}}_\delta$ et $\underline{\widehat{X}}_\delta$, sont reliées aux opérateurs $\mathbf{G}_{s,0}$ par (voir proposition 4.5) :

$$T_{\underline{\widehat{B}}_\delta}(2s) = \sum_{p\geq 0} \sum_{i=1}^{\lfloor (1-\delta)p \rfloor} \Delta\mathbf{G}_{s,0}^{p-i} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \mathbf{G}_{s,0}^{i-1}[1](0),$$

$$T_{\underline{\widehat{X}}_\delta}(2s) = \sum_{p\geq 0} \sum_{i=1}^{\lfloor (1-\delta)p \rfloor} \mathbf{G}_{s,0}^{p-i} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \Delta\mathbf{G}_{s,0}^{i-1}[1](0).$$

Considérons maintenant les parties dominantes de ces séries. Elles sont au voisinage de $s = 1$ de la forme

$$\begin{aligned} T_{\underline{\widehat{B}}_\delta}(2s) &\sim \sum_{p\geq 0} \left(\sum_{i=0}^{\lfloor (1-\delta)p \rfloor} (p-i) \right) \lambda^{p-1}(s,0)\mathbf{P}_{s,0} \circ \Delta\mathbf{G}_{s,0} \circ \mathbf{P}_{s,0} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \mathbf{P}_{s,0}, \\ &\sim (1-\delta^2) \left(\frac{1}{1-\lambda(s,0)} \right)^3 \mathbf{P}_{s,0} \circ \Delta\mathbf{G}_{s,0} \circ \mathbf{P}_{s,0} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \mathbf{P}_{s,0}, \end{aligned}$$

$$\begin{aligned} T_{\underline{\widehat{X}}_\delta}(2) &\sim \sum_{p\geq 0} \left(\sum_{i=0}^{\lfloor (1-\delta)p \rfloor} i \right) \lambda^{p-1}(s,0)\mathbf{P}_{s,0} \circ \mathbf{G}_{s,0}^{[c]} \circ \mathbf{P}_{s,0} \circ \Delta\mathbf{G}_{s,0} \circ \mathbf{P}_{s,0}, \\ &\sim (1-\delta)^2 \left(\frac{1}{1-\lambda(s,0)} \right)^3 \mathbf{P}_{s,0} \circ \mathbf{G}_{s,0}^{[\ell]} \circ \mathbf{P}_{s,0} \circ \Delta\mathbf{G}_{s,0} \circ \mathbf{P}_{s,0}. \end{aligned}$$

On déduit (5.56) en appliquant le théorème taubérien à ces séries. Maintenant, la démarche de la proposition 5.10 s'applique toujours ici. Le cas des coefficients de Bezout se traite de la même manière. Lorsque le système sous-jacent à l'algorithme satisfait la propriété de distorsion bornée, c'est-à-dire qu'il existe une constante réelle $c > 0$ telle que

$$|h''(x)| \leq c|h'(x)|, \quad \forall h \in \mathcal{H},\ \forall x \in J_h,$$

alors l'inégalité suivante est vérifiée :

$$\left| \log_2 \left(a_i + a_{i-1} \frac{u_i}{u_{i-1}} \right) - \ell(a_i) \right| \leq 1.$$

On en déduit finalement les équivalences

$$\begin{array}{rcl} \mathbb{E}_N[\underline{B}_\delta] & \sim & (1-\delta^2)\,\mathbb{E}_N[B], \\ \mathbb{E}_N[\underline{X}_\delta] & \sim & (1-\delta)^2\,\mathbb{E}_N[X], \\ \mathbb{E}_N[\underline{BX}_\delta] & \sim & \frac{1}{3}(1-\delta)(3-\delta)\,\mathbb{E}_N[BX]. \end{array} \tag{5.57}$$

Nous venons dons d'exprimer la complexité en bits de l'algorithme interrompu $\underline{\mathcal{E}}_\delta$. Supposons maintenant que $\delta = t$. Soit Q_t la variable aléatoire $Q_t = |I_t - (1-t)I|$, et soit R_t une des deux variables $R_t := |B_t - \underline{B}_t$ ou $R_t := |X_t - \underline{X}_t|$. Le pire des cas de l'algorithme d'Euclide entraîne que sur Ω_N, on a toujours $R_t = O(N^2)$. De plus, si on considère l'évènement exceptionnel $A(\varepsilon) := [Q_t \geq \varepsilon I]$, on obtient la relation

$$\mathbb{E}_N[R_t] \leq K' N^2\, \mathbb{P}_N\left[A(\varepsilon)\right] + N\, \mathcal{E}_N\ [L_t] \qquad \text{avec} \qquad L_t := \sum_{i=\lfloor (1-t-\varepsilon)I \rfloor}^{\lfloor (1-t+\varepsilon)I \rfloor} \ell(q_i). \tag{5.58}$$

Le théorème 5.43 montre que la première partie de cette expression est en $o(N^2)$. Maintenant, la série de Dirichlet $T_{L_t}(s)$ associée au coût L_t est reliée à l'opérateur $\mathbf{G}_{s,0}$ par

$$T_{L_t}(2s) = \sum_{p \geq 0} \sum_{i=\lfloor (1-t-\varepsilon)p \rfloor}^{\lfloor (1-t+\varepsilon)p \rfloor} {\mathbf{G}_{s,0}}^{p-i} \circ {\mathbf{G}_{s,0}}^{[\ell]} \circ {\mathbf{G}_{s,0}}^{i-1}[1](0).$$

Le terme dominant de cette série est de la forme

$$\varepsilon \left(\sum_p p\lambda(s,0)^{p-1} \right) \mathbf{P}_{s,0} \circ {\mathbf{G}_{s,0}}^{[\ell]} \circ \mathbf{P}_{s,0}[1](0) = \varepsilon \left(\frac{1}{1-\lambda(s,0)} \right)^2 \mathbf{P}_{s,0} \circ {\mathbf{G}_{s,0}}^{[\ell]} \circ \mathbf{P}_{s,0}[1](0),$$

ce qui entraîne que le second terme est également de la forme $o(N^2)$. On en conclut $\mathbb{E}_N[R_t] = o(N^2)$, et donc finalement

$$\mathbb{E}_N[\underline{B}_t] \sim \mathbb{E}_N[B_t],$$

$$\mathbb{E}_N[\underline{X}_t] \sim \mathbb{E}_N[X_t],$$

ce qui conclut la preuve du théorème. ∎

5.4.3 Évolution des distributions

Le dernier résultat obtenu sur les algorithmes interrompus traite de l'évolution de la distribution des restes au cours de l'exécution de l'algorithme. Il s'agit ici de l'étude de l'algorithme $\underline{\mathcal{E}_\delta}$.

Théorème 5.49 *Soit $\mathbb{P}_N$ la probabilité définie sur l'ensemble Ω_N. Soit (u,v) une entrée valide pour laquelle l'algorithme fait p itérations. Nous notons x_δ le rationnel $x_{\lfloor \delta p\rfloor+1}/x_{\lfloor \delta p\rfloor}$. Soit B un intervalle inclus dans $[0,1]$, alors*

$$\lim_{N\to\infty} \mathbb{P}_N[x_\delta \in B] = \int_B \psi(t)dt \qquad où \qquad \psi(x) = \frac{1}{\log 2}\frac{1}{1+x}.$$

Preuve : Le coût à étudier ici est le coût $U_{\delta,B}$ défini en 1.20 par

$$U_{\delta,B}(u,v) := \mathbf{1}_B(\frac{u_{\lfloor \delta p\rfloor+1}}{u_{\lfloor \delta p\rfloor}}).$$

La série génératrice associée $T_U(s)$ est reliée à l'opérateur $\mathbf{G}_{s,0}$ par (voir proposition 4.6)

$$T_U(2s) = \sum_{p\leq 1} \mathbf{G}_{s,0}^{p-\lfloor \delta p\rfloor}[\mathbf{1}_B \mathbf{G}_{s,0}^{\lfloor \delta p\rfloor}[1]](0).$$

La partie dominante de cet opérateur est donnée par

$$\sum_p \lambda(s,0)^p \mathbf{P}_{s,0}\left[\mathbf{1}_B \mathbf{P}_{s,0}[1]\right](0).$$

Le théorème taubérien appliqué à cette série fait apparaître l'intégrale $\int_I \mathbf{1}_B(t)\mathbf{P}[1](t)dt$, ce dont on déduit le résultat. ■

5.4.4 Discussion des résultats

Les résultats présentés ici ne sont pas surprenants. Nous montrons un comportement attendu, à savoir une décroissance logarithmique des restes au cours de l'exécution l'algorithme, ainsi qu'une croissance logarithmique des coefficients de l'algorithme étendu. D'une manière générale, les résultats de ce paragraphe fournissent une "photographie" de l'état des principaux paramètres au cours de l'exécution de l'algorithme. Cette approche est particulièrement instructive quand on veut étudier des algorithmes plus sophistiqués comme l'algorithme Lehmer-Euclide, qui sont une succession d'algorithmes interrompus. Les algorithmes récursifs comme l'algorithme de Schönhage requièrent une analyse plus fine, qui n'a pas été faite ici. Cependant, les travaux récents de Vallée et Baladi laissent penser qu'une analyse en distribution des paramètres étudiés dans ce paragraphe est possible grâces aux techniques qu'elles ont développées.

Enfin, remarquons que nous n'avons présenté cette analyse que pour l'algorithme d'Euclide interrompu. Il est cependant possible de procéder à la même étude pour nombre d'autres algorithmes, en particulier pour les algorithmes rapides. Les comportements observés sont tous similaires à ce que nous venons de décrire.

5.5 Conclusion

Nous venons de conclure l'analyse des algorithmes "simples", qui procèdent par divisions successives. Nous venons de décrire pour chaque type de système dynamique rencontré dans quel espace fonctionnel se placer, et comment déduire des propriétés spectrales de l'opérateur les propriétés analytiques des séries génératrices. Cette démarche présente l'avantage de présenter de manière unifiée le comportement des différents algorithmes, puisque toutes les constantes exhibées s'expriment en termes de systèmes dynamiques. On peut de la sorte procéder à une rapide comparaison des algorithmes. On observe par exemple le caractère spécifique de l'algorithme LSB puisque c'est le seul qui fait intervenir un exposant de Lyapunov.

Cependant, nous n'avons pas conclu les analyses pour les algorithmes Mixtes. Les opérateurs relatifs aux systèmes mixtes nécessitent également un traitement particulier. Considérons par exemple l'opérateur de Perron-Frobenius relatif à l'algorithme Binaire, qui est donné (voir 4.7) par

$$\mathbf{M}[f](x) = \sum_{k\geq 1} \sum_{\substack{a \text{ impair} \\ a<2^k}} \frac{1}{(a+2^k x)^2} f\left(\frac{1}{a+2^k x}\right).$$

On ne peut utiliser comme pour les opérateurs relatifs aux systèmes $\mathcal{E}$ ou $\mathcal{E}_{\mathcal{C}}$ l'espace $\mathcal{A}_\infty(V)$ de fonctions analytiques : il est en effet impossible de trouver un domaine V adéquat comprenant l'intervalle $[0,1]$ et sur lequel toutes les branches inverses sont analytiques, puisque la suite des pôles de ces branches est $-a/2^k$ et a un point d'accumulation en 0. Il n'est pas possible non plus d'utiliser l'espace des fonctions à variation bornée, puisqu'on observe facilement que la fonction $\mathbf{H}[1]$ n'étant pas bornée (la série $\mathbf{H}[1](0)$ diverge), sa variation ne l'est pas plus. L'approche utilisée par Brigitte Vallée dans [Val98a] pour contourner ces problèmes a été d'utiliser les espaces de Hardy. On considère ici un disque ouvert D, de diamètre $]0,2\rho[$ où ρ est strictement compris entre 0 et 1. Si on note δ la frontière de D et δ_r le cercle de centre ρ et de rayon r, alors l'espace de Hardy d'ordre 2 $\mathcal{H}_2(D)$ est l'espace des fonctions f analytiques sur D telles que la quantité

$$||f||_{\mathcal{H}} = \sup_{0\leq r<\rho} \frac{1}{2\pi\rho} \int_\delta |f(z)|^2 dz$$

est finie. Cette quantité définit une norme, qui fait de $\mathcal{H}_2(D)$ un espace de Banach. Sur cet espace, on peut montrer que l'opérateur est compact et qu'il vérifie toutes les propriétés requises pour la fin de l'analyse.

La situation est encore différente pour le système Plus-Moins. On ne peut en effet plus se placer sur un espace de fonctions analytiques du fait de la présence de fonctions indicatrices, qui interdisent même d'utiliser des fonctions continues. L'espace des fonctions à variation bornée n'est pas plus indiqué, du fait de la remarque précédente. On ne dispose à ce jour d'aucun espace fonctionnel adéquat sur lequel faire agir l'opérateur : il faut trouver un espace permettant de traiter les discontinuités mais qui soit plus "gros" que BV, sans pour autant atteindre la taille de L^1, sur lequel le spectre de l'opérateur est continu. C'est donc toujours un problème ouvert.

Chapitre 6

Application à l'Algorithme de Lehmer-Euclide

Sommaire

Nous abordons maintenant le dernier chapitre de cette thèse, dans lequel nous appliquons les techniques d'analyse dynamique à un algorithme de structure plus complexe que ceux étudiés jusqu'à présent, l'algorithme de Lehmer-Euclide. Cet algorithme est historiquement le premier algorithme "rapide" de calcul de pgcd (l'appellation "rapide" est trompeuse, car la complexité en bits de l'algorithme est toujours quadratique, il faut plutôt parler ici d'algorithme "accéléré"), puisqu'il a été introduit dès les années 30 par D. H. Lehmer. L'idée de base de l'algorithme, simuler les divisions successives de "grands entiers" par des divisions de "petits entiers", trouve un écho dans l'informatique moderne : on simule des divisions "multi-précision" par des divisions "simple-précision" (plus quelques multiplications multi-précision). Ce principe conduit à une version accélérée de l'algorithme d'Euclide, implémentée dans de nombreuses bibliothèques multi-précision ou logiciels de calcul formel.
De plus, le principe général de l'algorithme a été exploité avec succès dans les années 70 par A. Schönhage, qui a mis au point un algorithme "réellement" rapide, puisque de complexité en bits en $O(n \log^2 n \log\log n)$, basé sur l'idée générale de Lehmer, une approche Diviser pour Régner et la multiplication rapide. Cette approche n'est réellement efficace que pour de très grands entiers, puisque les expérimentations de Cesari [Ces98] montrent par exemple que jusqu'à plusieurs centaines de bits, la constante cachée dans le $O(n \log^2 n \log\log n)$ fait que cet algorithme n'est pas le plus approprié. Notons cependant que les récents travaux de Stehlé et Zimmermann, qui ont utilisé cette approche en remplaçant l'algorithme d'Euclide par l'algorithme LSB contribuent à réduire considérablement la taille de cette constante, de sorte qu'on obtient finalement un algorithme plus rapidement "utilisable".

Nous étudions donc ici l'algorithme de Lehmer-Euclide, première étape vers une analyse future de l'algorithme de Schönhage (la récursivité de cet algorithme ajoute des difficultés supplémentaires). Les paramètres à étudier dans cet algorithme ne sont plus exactement les mêmes

que précédemment : en particulier, le nombre total de divisions effectuées par l'algorithme est le même que pour l'algorithme d'Euclide. Une analyse fine requiert ici de distinguer la nature de chaque opération. En particulier on distinguera les divisions, qui sont des opérations simple-précision, des multiplications, qui sont multi-précision.

Ce chapitre est constitué de trois sections : dans la première, nous décrivons l'algorithme Lehmer-Euclide $\mathcal{LE}$, ainsi que la version paramétrée que nous étudions $\mathcal{LE}_\mu$. Nous énonçons le résultat obtenu dans le théorème 6.1. La dernière section est consacrée à la preuve de ce théorème. Remarquons finalement que nous utiliserons à de nombreuses reprises les résultats obtenus dans les chapitres précédents sur les algorithmes d'Euclide interrompus.

6.1 Description de l'algorithme $\mathcal{LE}_\mu$

Cet algorithme est abondamment décrit dans la littérature, comme par exemple dans [Knu98]. Nous le présentons ici de la même manière que dans [DV04], c'est-à-dire comme une succession de "phases", chacune de ces phases comportant trois étapes. La paramétrisation de l'algorithme intervient dans la première étape de chaque phase.

Remarque Le principe l'algorithme étant de simuler la division de deux entiers à l'aide des parties dominantes de ces entiers, nous serons amenés à manipuler en permanence des "grands" entiers et des "petits" entiers. Pour cela, toute grandeur relative aux grands entiers sera en majuscules, et toute grandeur relative aux petits en minuscules (les entiers seront (U, V), U_i et (u, v), u_i, les quotients Q_i et q_i, etc...).

L'algorithme $\mathcal{LE}$ repose sur le fait que le quotient d'une division euclidienne standard dépend essentiellement des bits de poids forts des entiers. Plus précisément, soient U et V deux entiers de tailles respectives $\ell(U)$ et $\ell(V)$, et u et v les entiers formés respectivement des m bits de poids fort de U et des $m - (\ell(U) - \ell(V)$ bits de poids fort de V. Alors si $m \leq \ell(V)$ est choisi suffisamment grand, on vérifie l'égalité

$$\left\lfloor \frac{U}{V} \right\rfloor = \left\lfloor \frac{u}{v} \right\rfloor .$$

Nous allons utiliser ce principe de la manière suivante. Soit (U, V) une entrée de l'algorithme d'Euclide, pour laquelle celui-ci fait p itérations. Soit $Q_1, \ldots Q_p$ la suite de quotients engendrée par l'algorithme, et $\mathcal{M}_{[Q_1]}, \ldots \mathcal{M}_{[Q_p]}$ la suite de matrices associée, donnée pour $1 \leq i \leq p$ par

$$\mathcal{M}_{[Q_i]} = \begin{pmatrix} 0 & 1 \\ 1 & Q_i \end{pmatrix}.$$

Rappelons que la suite (A_i, B_i) de coefficients calculée par l'algorithme d'Euclide étendu est notamment donnée par

$$\mathcal{M}_{[Q_1]} \cdot \mathcal{M}_{[Q_2]} \cdots \mathcal{M}_{[Q_i]} = \begin{pmatrix} B_{i+1} & A_{i+1} \\ B_i & A_i \end{pmatrix},$$

et que pour tout $1 \leq i \leq p$ on vérifie

$$UA_i + VB_i = U_i. \tag{6.1}$$

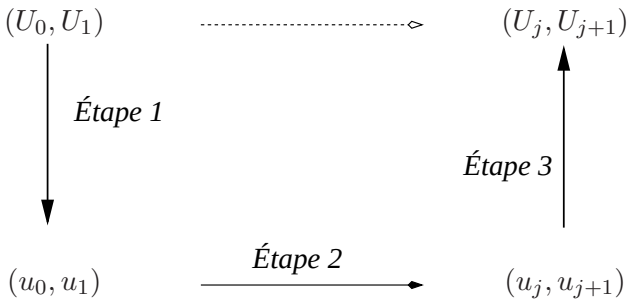

FIG. 6.1 – Une phase de l'algorithme $\mathcal{LE}$

Soit maintenant un entier $k \leq \ell(V)$, et considérons les entiers u et v définis par

$$u = \left\lfloor \frac{U}{2^{\ell(V)-k}} \right\rfloor \qquad v = \left\lfloor \frac{V}{2^{\ell(V)-k}} \right\rfloor,$$

donc déduits de U et V en supprimant les $\ell(V) - k$ bits de poids faible. L'algorithme d'Euclide appliqué à u et v engendre les suites (u_i), (q_i), $(\mathcal{M}_{[q_i]})$, et (a_i, b_i).
Il existe un entier $j \geq 0$ pour lequel l'égalité $q_i = Q_i$ est valide pour tout i compris entre 1 et j. On en déduit donc les égalités

$$\mathcal{M}_{[q_i]} = \mathcal{M}_{[Q_i]}, \qquad (a_{i+1}, b_{i+1}) = (A_{i+1}, B_{i+1})$$

pour tout $0 \leq i \leq j$. En particulier, la relation (6.1) permet de retrouver les restes U_j, U_{j+1} puisqu'on a les égalités

$$\begin{aligned} Ua_j + Vb_j &= U_j, \\ Ua_{j+1} + Vb_{j+1} &= U_{j+1}. \end{aligned} \tag{6.2}$$

La démarche que nous venons de décrire correspond à une phase de l'algorithme de Lehmer-Euclide. Les trois étapes qui la constituent sont donc les suivantes :

- (1) on "tronque" les entiers U et V pour obtenir u et v,
- (2) on applique l'algorithme d'Euclide à la paire (u, v) jusqu'au premier indice j tel que $q_{j+1} \neq Q_{j+1}$,
- (3) on calcule la paire (U_j, U_{j+1}) à l'aide des relations (6.2).

Une fois calculée la paire (U_j, U_{j+1}), on réitère le même procédé. Nous décrivons maintenant précisément l'algorithme de Lehmer-Euclide paramétré $\mathcal{LE}_\mu$ en détaillant chacune des trois phases.

L'algorithme $\mathcal{LE}_\mu$

Étape 1 C'est ici que le paramètre μ de troncature de notre algorithme intervient. En effet, dans le principe général de l'algorithme énoncé plus haut, le nombre de bits des entiers (u, v) est fixe. En pratique, on fait en sorte que u corresponde à un mot machine (64 ou 32 bits) ou à deux mots machine (il est même judicieux de choisir deux mots machines, comme l'observe T. Jebelean dans [Jeb95]). Pour les besoins de l'analyse, nous supposerons que les entiers (u, v) sont de tailles proportionnelles à l'entrée (U, V) de l'algorithme, et nous poserons

$$m := \ell(u) = \lfloor \mu \ell(U) \rfloor,$$

où le paramètre μ est un réel appartenant à l'intervalle $]0,1]$. L'opération de troncature est donc définie par l'application T_μ :

$$T_m(U,V) := \begin{cases} (\frac{U}{2^{\ell(V)-m}}, \frac{V}{2^{\ell(V)-m}}) & \text{si } \ell(V) \geq m, \\ (0,0) & \text{sinon.} \end{cases}$$

Étape 2 C'est bien sûr l'étape centrale de la phase, puisqu'il faut ici résoudre le problème suivant : comment savoir si l'égalité $q_i = Q_i$ est vérifiée sans calculer explicitement Q_i ?
Il existe plusieurs manières de résoudre ce problème. Plus exactement, on dispose de plusieurs conditions qui, si elles sont vérifiées, garantissent que l'égalité souhaitée est vérifiée. La solution initialement proposée par Lehmer consistait à calculer simultanément les suites de quotients relatives aux paires $(u-1, v)$ et $(u, v+1)$. Il est clair que tant que les deux suites de quotients concordent, alors l'égalité $q_i = Q_i$ est vérifiée. Cependant, cette condition n'est pas optimale. On peut préférer par exemple le test "exact" de Jebelean [Jeb95] : on a l'égalité $q_i = Q_i$ tant que la condition suivante est vérifiée,

$$u_{i+1} \geq -a_{i+1} \; et \; u_i - u_{i+1} \geq b_{i+1} - b_i \text{ si } i \text{ est pair},$$

$$u_{i+1} \geq -b_{i+1} \; et \; u_i - u_{i+1} \geq a_{i+1} - a_i \text{ si } i \text{ est impair}.$$

Nous préférerons un autre test, plus léger, dû à Collins [Col] et à Jebelean [Jeb95], qui est basé sur la relation suivante :

$$\text{si } u_j > u_0^{1/2} \text{ alors } Q_i = q_i, \text{ pour tout } i \leq j-2.$$

Munie de cette condition d'arrêt, l'étape 2 de l'algorithme $\mathcal{LE}_\mu$ est donc exactement $\mathcal{E}_{1/2}$, l'algorithme d'Euclide interrompu de paramètre $1/2$.

Étape 3 Cette étape n'appelle que peu de commentaires... Remarquons cependant que c'est l'étape la plus coûteuse de la phase : en effet, les opérations de la seconde phase sont menées sur des entiers "petits", alors qu'ici on effectue 4 multiplications dans lesquelles interviennent des "grands" entiers, à savoir U et V. D'une manière générale, les coefficients (a_i, b_i) et (a_{i+1}, b_{i+1}) utilisés dans ces multiplications sont de taille $m/2$ (c'est d'ailleurs à partir de cette observation que Jebelean propose de choisir u d'une taille de deux mots machines).

Finalement, l'algorithme $\mathcal{LE}_\mu$ est décrit dans la figure 6.2 suivante.

6.2 Analyse probabiliste de l'algorithme $\mathcal{LE}_\mu$

Comme nous l'avons déjà remarqué, le nombre d'itérations de l'algorithme n'est pas ici un paramètre pertinent. On peut étudier le nombre et la longueur des phases. C'est par exemple ce que fait Sorenson dans [Sor95], ou il étudie le pire des cas de ces paramètres. Nous exhiberons la comportement moyen de ces paramètres par la suite. Mais une étude judicieuse de l'algorithme prend nécessairement en compte la complexité en bits de l'algorithme. Plus précisément, nous chercherons ici à distinguer les différentes opérations effectuées sur les algorithmes, qui peuvent être des divisions, effectuées lors de l'étape 2 de chaque phase sur des petits entiers, ou des multiplications, effectuées soit dans l'étape 2 lors du calcul des coefficients a_i, b_i, soit dans l'étape trois. Nous allons donc attribuer à chacune de ces opérations un coût différent :

Entrée : (U, V) avec $0 \leq V \leq U$.
Sortie : $\text{pgcd}(U, V)$

Initialisation $n := \ell(U)$; $m := \lfloor \mu n \rfloor$; $U_0 := U$; $U_1 := V$;
`Tant que` $\ell(U_0) > m$
  (1). $u_0 := T_m(U_0)$; $u_1 := T_m(U_1)$;
  (2). $i := 1$; $u_0 := 1$; $u_1 := 0$; $v_0 := 0$; $v_1 := 1$;
    `Tant que` $u_i > u_0^{1/2}$
      $q_i := u_{i-1} \operatorname{div} u_i$; $u_{i+1} := u_{i-1} \mod u_i$;
      $a_{i+1} := -a_i q_i + a_{i-1}$; $b_{i+1} := -b_i q_i + b_{i-1}$;
      $i := i + 1$;
  (3). $U' := a_{i-3} U_0 + b_{i-3} U_1$; $V' := a_{i-2} U_0 + b_{i-2} U_1$;
    $U_0 := U'$; $U_1 := V'$;
$i := 1$;
`Tant que` $U_i > 0$
  $U_{i+1} := U_{i-1} \mod U_i$;
  $i := i + 1$;
`Renvoyer` U_{i-1}

FIG. 6.2 – L'algorithme de Lehmer-Euclide de paramètre de troncature μ.

- Une multiplication "générique" entre deux entiers u et v à pour coût $M \cdot \ell(u) \cdot \ell(v)$, M étant donc la constante relative à la multiplication,
- un échange entre deux entiers u et v a pour coût $M \cdot (\ell(u) + \ell(v))$,
- finalement, une division euclidienne $v = uq + r$ à un coût égal à $D \cdot \ell(u) \cdot \ell(q)$, D étant ici une constante relative à la division.

Ainsi, étant donnés une entrée (u, v) de l'algorithme d'Euclide, le coût B exprimant la complexité en bits s'écrit maintenant

$$B(u, v) := \sum_{i=1}^{p} \ell(u_i) \cdot b(q_i), \qquad \text{avec} \qquad b(q) := D \cdot \ell(q) + 2 \cdot M,$$

et le coût supplémentaire lié au calcul d'un coefficient a_i est

$$X(u, v) := \sum_{i=1}^{p} \ell(a_i) \cdot c(q_i), \qquad \text{avec} \qquad c(q) := (\ell(q) + 2) \cdot M.$$

Avec ces notations, la complexité en bits moyenne de l'algorithme d'Euclide donnée par exemple au paragraphe 1.7.0.0, (1.30), s'écrit

$$\mathbb{E}_N[B] \sim (L_1 M + L_2 D) \cdot N^2, \qquad \mathbb{E}_N[X] \sim (L_1 + L_2) \cdot M \cdot N^2,$$

avec

$$L_1 = \frac{12 \log^2 2}{\pi^2} \sim 0.58, \qquad L_2 = \frac{6 \log 2}{\pi^2} \log \prod_{k=0}^{\infty} (1 + \frac{1}{2^k}) \sim 0.66,$$

et la constante L_1 est donc relative aux multiplications et échanges, alors que la constante L_2 est relative aux multiplications. Enfin nous notons L_μ le coût total d'une exécution de l'algorithme

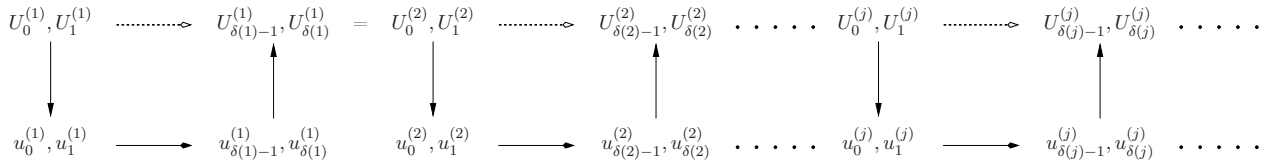

FIG. 6.3 – Une exécution de l'algorithme $\mathcal{LE}$

$\mathcal{LE}_\mu$. La quantité $L_\mu(U,V)$ fait donc intervenir les constantes M et D. Le principal résultat de ce chapitre, obtenu en collaboration avec Brigitte Vallée, [DV04], traite du comportement moyen de cette variable aléatoire. Il est décrit dans le théorème suivant.

Théorème 6.1 *Soit $\mathcal{LE}_\mu$ l'algorithme Lehmer-Euclide de paramètre de troncature μ, et $\Omega, \widetilde{\Omega}$ les ensembles d'entrées valides de l'algorithme d'Euclide. Alors la complexité en bits moyenne de l'algorithme sur les entrées de taille N est asymptotiquement quadratique en N, et est donnée par*

$$\mathbb{E}_N[L_\mu] \sim \left[\frac{3}{2}(L_1M + L_2D)\mu + (L_1 + L_2)M\mu + (2-\mu)M\right] N^2.$$

Discussion du théorème 6.1

Comparons maintenant les complexités en bits des algorithmes Lehmer-Euclide et Euclide. Il n'est avantageux d'utiliser le premier que si une grande division coûte plus qu'une grande multiplication. Plus précisément, l'algorithme $\mathcal{LE}_\mu$ est meilleur si $2M < L_1M + L_2D$, ce qui est généralement vérifié (dès qu'on a $D \geq 5M$). Dans ce cas, toutes les valeurs de μ ne sont pas pertinentes. Par exemple, si $D = 5M$ il faut alors choisir μ supérieur ou égal à 0.3. D'une manière générale, si le ratio entre le coût d'une division et celui d'une multiplication est ρ, alors la valeur maximale μ_0 de μ est donnée par

$$\mu_0 = 2\frac{L_1 + L_2\rho - 2}{5L_1 + L_2(3\rho+2) - 2}.$$

Finalement, si on pose $\rho = 15$ et $\mu = 1/3$, alors le rapport entre les complexités en bits est proche de 0.7. Pour $\rho = 30$ et $\mu = 1/10$, il est proche de $1/4$.

6.3 Preuve du théorème 6.1

L'idée générale de la preuve est en quelque sorte l'opposée de celle qui a conduit à la construction de l'algorithme : nous allons simuler la complexité en bits de l'algorithme sur une exécution de l'algorithme d'Euclide "normal" sur l'entrée (U,V). En effet, la plupart des quantités intervenant dans la complexité en bits $L_\mu(U,V)$ peuvent se lire sur la ligne "supérieure" de la figure 6.3 : les quotients sont par construction les mêmes et les coefficients de Bezout sont donc accessibles. Seuls les tailles des "petits" restes ne peuvent être obtenues directement. Nous disposons cependant de suffisamment de contrôle pour pouvoir les approximer à l'aide des tailles des "grands" restes.
Nous procédons ici en plusieurs temps. Tout d'abord, nous montrons comment générer ces tailles. Puis nous étudions chacune des phases de l'algorithme, en particulier leur longueur, et nous montrons qu'elles ont un comportement régulier. Plus précisément, nous prouvons que si l'algorithme d'Euclide fait P itérations sur l'entrée (U,V), alors "en moyenne", la longueur de la j-ème phase

est de $\lfloor(\mu/2)P\rfloor$ itérations. Nous introduisons par la suite un algorithme "artificiel", $\overline{\mathcal{LE}}_\mu$, dont chaque phase effectue exactement $\lfloor(\mu/2)P\rfloor$ itérations. Cet algorithme est aisé à analyser, et nous montrons finalement que sa complexité en bits moyenne et celle de l'algorithme $\mathcal{LE}_\mu$ sont asymptotiquement les mêmes.

Notations Nous allons utiliser ici les notations de la figure 6.3, à savoir que la j-ème phase de l'algorithme débute avec la paire d'entiers $(U_0^{(j)}, U_1^{(j)})$, qui génère via l'application T_m la paire $(u_0^{(j)}, u_1^{(j)})$. La i-ème paire de restes apparaissant dans la j-ème phase est notée $(u_{i-1}^{(j)}, u_i^{(j)})$, et la paire correspondante obtenue si on avait appliqué l'algorithme d'Euclide à (U, V) est notée $(U_{i-1}^{(j)}, U_i^{(j)})$. Nous noterons $p(j)$ l'indice de début de la j-ème phase, c'est à dire l'indice de $U_0^{(j)}$ dans la suite $U_0, \ldots U_P$ et $\delta(j)$ la longueur de la j-ème phase, c'est-à-dire son nombre d'itérations. Finalement, nous noterons $a[r, t]$ le coefficient a calculé entre les indices r et t, c'est-à-dire le coefficient obtenu si on avait appliqué l'algorithme d'Euclide étendu avec comme entrée la paire (U_r, U_{r+1}) et qu'on s'était arrêté au reste U_t. Enfin, pour un paramètre μ fixé, et pour une entrée (U, V) avec $\ell(U) = N$, nous noterons m la taille des entiers $u_0^{(j)}$ obtenus après troncation, c'est-à-dire $m = \lfloor \mu \cdot N \rfloor$.

Nous commençons par relier les tailles $\ell(u_i)$ aux tailles $\ell(U_i)$.

Lemme 6.2 *Soit $(u_{i-1}^{(j)}, u_i^{(j)})$ la i-ème paire de restes apparaissant dans la j-ème phase et $(U_{i-1}^{(j)}, U_i^{(j)})$ la paire correspondante. Alors*

$$|\ell(u_i^{(j)}) - \ell(U_i^{(j)}) + \ell(U_0^{(j)}) - m| \leq 2.$$

Preuve : Soit une phase j (nous oublierons l'indice j dans la suite de la preuve). Soit $\mathcal{M}_i$ la matrice correspondant aux i quotients déjà calculés lors de cette phase. On a

$$\begin{pmatrix} U_{i+1} \\ U_i \end{pmatrix} = \mathcal{M}_i^{-1} \begin{pmatrix} U_1 \\ U_0 \end{pmatrix}, \qquad \begin{pmatrix} u_{i+1} \\ u_i \end{pmatrix} = \mathcal{M}_i^{-1} \begin{pmatrix} u_1 \\ u_0 \end{pmatrix}. \tag{6.3}$$

D'un autre côté, puisque la paire (u_0, u_1) est la troncature de la paire (U_0, U_1),

$$\begin{pmatrix} U_1 \\ U_0 \end{pmatrix} = 2^{\ell(U_0)-m} \left[\begin{pmatrix} u_1 \\ u_0 \end{pmatrix} + \begin{pmatrix} \delta_1 \\ \delta_0 \end{pmatrix} \right],$$

où δ_0, δ_1 satisfont $0 \leq \delta_0, \delta_1 < 1$. On a donc

$$\begin{pmatrix} U_{i+1} \\ U_i \end{pmatrix} = 2^{\ell(U_0)-m} \left[\begin{pmatrix} u_{i+1} \\ u_i \end{pmatrix} + \mathcal{M}_i^{-1} \begin{pmatrix} \delta_1 \\ \delta_0 \end{pmatrix} \right].$$

De plus, la relation $u_0 = |a_{i-1}|\, u_i + |a_i|\, u_{i-1}$, entraîne avec les inégalités $u_i, u_{i-1} > \sqrt{u_0}$ que les valeurs absolues des coefficients des matrices $\mathcal{M}_i, \mathcal{M}_i^{-1}$ sont inférieurs à $(1/2)\sqrt{u_0} \leq (1/2)u_i$. Enfin, la relation

$$|U_i - 2^{\ell(U_0)-m} u_i| \leq 2^{\ell(U_0)-m} \frac{u_i}{2}$$

prouve le lemme. ∎

Considérons maintenant la complexité en bits de la j-ème phase. Elle se décompose en trois coûts. Le premier (type 1) est lié à l'algorithme d'Euclide interrompu de l'étape 2 de la phase. Le second (type 2) est le coût supplémentaire lié au calcul des suites de coefficients, et le troisième correspond aux quatre multiplications de l'étape 3 de la phase. Les deux derniers coûts utilisent des quantités directement accessibles avec la ligne supérieure (algorithme d'Euclide appliqué à (U,V)). Les séries de Dirichlet relatives à ces coûts s'expriment donc à l'aide de l'opérateur de transfert $\mathbf{G}_{s,0}$ relatif au système $S_{\mathcal{E}}$. Le premier, dans lequel apparaissent les restes $u_i^{(j)}$ n'est plus exprimable directement, mais le lemme précédent montre qu'il peut être approché en utilisant les restes $U_i^{(j)}$. Finalement, nous allons étudier les trois coûts suivants :

$$\text{Coût de type 1 :} \quad \sum_{j=1}^{J} \sum_{i=p(j)}^{p(j+1)-1} [\ell(U_i) - \ell(U_{p(j)}) + m] \cdot b(q_i)$$

$$\text{Coût de type 2 :} \quad 2 \sum_{j=1}^{J} \sum_{i=p(j)}^{p(j+1)-1} \ell(a[p(j), i]) \cdot c(q_i)$$

$$\text{Coût de type 3 :} \quad 4 \sum_{j=1}^{J} \ell(U_{p(j)}) \cdot \ell(a[p(j), p(j+1)])$$

Le coût de type 1 se décompose lui-même en trois différents coûts (quand on développe $[\ell(U_i) - \ell(U_{p(j)}) + m] \cdot b(q_i)$). Le premier correspond à la complexité en bits de l'algorithme d'Euclide (restreint à la j-ème phase) sur entrée (U,V). La dernier coût a été analysé dans les chapitres précédents, il correspond à la valeur moyenne du coût à croissance modérée b. Enfin, le second coût, dit de type 1', est donné par

$$\text{Coût de type } 1' : \sum_{j=1}^{J} \ell(U_{p(j)}) \cdot \sum_{i=p(j)}^{p(j+1)-1} b(q_i).$$

Chacune de ces différentes expressions dépend de l'indice $p(j)$ du début de la phase, ainsi que de sa longueur $\delta(j)$. Nous montrons dans le lemme suivant qu'en moyenne, chaque phase est de longueur $\bar{p} := \lfloor (\mu/2)p \rfloor$. Ce lemme est une adaptation du théorème 5.43 sur le nombre d'itérations de l'algorithme interrompu.

Lemme 6.3 *Soit $\delta(j)$ la longueur de la j-ème phase. Alors pour tout $\varepsilon > 0$, il existe $K < 1$ tel que quand $n \to \infty$*

$$\mathbb{P}_n \left[\left| \frac{\delta(j)}{P} - \frac{\mu}{2} \right| > \varepsilon \right] = O(K^n).$$

Preuve : Si une phase commence à la r-ème itération de l'algorithme d'Euclide, elle se termine à la $(r+t)$-ème dès que la suite u_i satisfait une certaine condition. Le lemme précédent montre qu'il existe des relations entre les restes U_i et les restes u_i, ce qui fait que le test d'arrêt de la j-ème phase est

$$U_{r+t} \sim \frac{U_r}{U_0^{\mu/2}}.$$

On utilise maintenant la même démarche que dans l'analyse de l'algorithme interrompu. On commence par utiliser l'inégalité de Markov, qui nous amène à étudier la série de Dirichlet relative au coût

$$N(U,V) := \left(\frac{U_{r+t}}{U_r U_0^{-\mu/2}} \right)^{\gamma},$$

pour $\gamma > 0$. Si on utilise les mêmes notations que dans le chapitre 5.4, à savoir

$$s^+ = s + \alpha\gamma, \qquad s^- = s - \beta\gamma,$$

avec $\alpha = 1 - (\mu/2)$ et $\beta = \mu/2$, alors la séries de Dirichlet relative au coût N s'exprime avec l'opérateur $\mathbf{G}_{s,0}$. Plus exactement, cette expression utilise les opérateurs suivants (on s'inspire de la proposition 4.4),

$$\mathbf{G}_{s^-,0}^{p-r-t} \circ \mathbf{G}_{s^+,0}^{t} \circ \mathbf{G}_{s^-,0}^{r}.$$

Remarquons que la partie dominante de ces fonctions ne dépend pas de l'indice r du début de phase. Si on note $p := (c+d)k, t := ck$, alors la fonction ϕ est la même fonction qu'au chapitre précédent, c'est-à-dire

$$\phi(s) := \lambda^d(s^-, 0)\lambda^c(s^+, 0).$$

Ceci implique que l'étude de singularité se fait exactement de la même manière que dans la proposition 5.46. On en déduit ainsi le résultat. ■

Nous venons ainsi de montrer que la longueur des phases de l'algorithme $\mathcal{LE}_\mu$ possède une certaine régularité. Considérons maintenant un algorithme "artificiel", l'algorithme normalisé $\overline{\mathcal{LE}}_\mu$, dont la longueur de chaque phase est exactement $\bar{p} := \lfloor (\mu/2)p \rfloor$. Nous allons faire deux choses avec cet algorithme : tout d'abord, calculer sa complexité en bits moyenne, puis montrer que celle-ci est asymptotiquement celle de l'algorithme $\mathcal{LE}_\mu$.

Lemme 6.4 *La complexité en bits de la j-ème phase de l'algorithme $\overline{\mathcal{LE}}_\mu$ sur les ensembles $\Omega_N, \widetilde{\Omega}_N$ est asymptotiquement*

$$\frac{3}{4}(L_1 M + L_2 D)\mu^2 N^2 + \frac{1}{2}(L_1 + L_2)M\mu^2 N^2 + 4\frac{\mu}{2}(1 - (j-1)\frac{\mu}{2})MN^2.$$

Preuve : Exprimons tout d'abord les différents coûts intervenant ici. Ils sont essentiellement de trois types, et sont donnés ci-dessous (on suppose que l'algorithme d'Euclide fait p itérations avec entrée (U, V)).

$$\text{Coût de type } 1' : \quad \sum_{i=(j-1)\bar{p}}^{j\bar{p}-1} \ell(U_{(j-1)\bar{p}}) \cdot b(q_i),$$

$$\text{Coût de type } 2 : \quad 2\sum_{i=(j-1)\bar{p}}^{j\bar{p}-1} \ell(a[(j-1)\bar{p}, i]) \cdot c(q_i),$$

$$\text{Coût de type } 3 : \quad 4\ell(U_{(j-1)\bar{p}}) \cdot \ell(a[(j-1)\bar{p}, j\bar{p}]).$$

On déduit aisément l'expression des séries de Dirichlet associées à ces trois coûts (on engendre en réalité ici les coûts approximant, où $\ell(u_i)$ est remplacé par $\log_2 u_i$ par exemple ; nous savons que cette approximation ne modifie pas le résultat final). Elles sont données par (toujours à nombre p d'itérations fixé) :

$$\text{Coût de type } 1' : \quad \sum_{i=1}^{\bar{p}} \Delta\left[\mathbf{G}_{s,0}^{p-j\bar{p}} \circ \mathbf{G}_{s,0}^{\bar{p}-i} \circ \mathbf{G}_{s,0}^{[b]} \circ \mathbf{G}_{s,0}^{i-1}\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}}[1](0)$$

$$\text{Coût de type } 2 : \quad 2\sum_{i=1}^{\bar{p}} \mathbf{G}_{s,0}^{p-j\bar{p}} \circ \mathbf{G}_{s,0}^{\bar{p}-i} \circ \mathbf{G}_{s,0}^{[c]} \circ \Delta\left[\mathbf{G}_{s,0}^{i-1}\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}}[1](0)$$

$$\text{Coût de type } 3 : \quad 4\Delta\left[\mathbf{G}_{s,0}^{p-j\bar{p}} \circ \Delta\left(\mathbf{G}_{s,0}^{\bar{p}}\right)\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}}[1](0)$$

Quand finalement on somme sur le nombre p d'itérations possibles, les termes dominants des séries contiennent un produit de deux facteurs : le premier est commun à toutes les séries et est donné par

$$\sum_{p\geq 0} p^2 \lambda(s,0)^p \sim 2\left(\frac{1}{1-\lambda(s,0)}\right)^3. \tag{6.4}$$

Les autres facteurs sont respectivement

$$\begin{aligned}
&\text{Coût de type } 1' : && \tfrac{\mu}{2}\left[1-\tfrac{\mu}{2}(j-1)\right]\ \mathbf{P}_{s,0}\circ\Delta\mathbf{G}_{s,0}\circ\mathbf{P}_{s,0}\circ\mathbf{G}_{s,0}^{[b]}\circ\mathbf{P}_{s,0}[1](0)\\
&\text{Coût de type } 2 : && 2\tfrac{1}{2}\left(\tfrac{\mu}{2}\right)^2\ \mathbf{P}_{s,0}\circ\mathbf{G}_{s,0}^{[c]}\circ\mathbf{P}_{s,0}\circ\Delta\mathbf{G}_{s,0}\circ\mathbf{P}_{s,0}[1](0)\\
&\text{Coût de type } 3 : && 4\tfrac{\mu}{2}\left[1-\tfrac{\mu}{2}(j-1)\right]\ \mathbf{P}_{s,0}\circ\Delta\mathbf{G}_{s,0}\circ\mathbf{P}_{s,0}\circ\Delta\mathbf{G}_{s,0}\circ\mathbf{P}_{s,0}[1](0)
\end{aligned} \tag{6.5}$$

On peut appliquer le théorème taubérien à toute ces fonctions, ce qui donne le résultat. ■

Si on somme sur tous les indices j, correspondant aux numéros de phases, pour j variant entre 1 et $\bar{J} := \lceil 2/\mu \rceil$, on obtient la complexité en bits de l'algorithme $\overline{\mathcal{LE}}_\mu$. Cette complexité est exactement la même que celle du théorème 6.1. Il ne reste donc plus qu'à montrer que les complexités des algorithmes $\overline{\mathcal{LE}}_\mu$ et $\mathcal{LE}_\mu$ sont (en moyenne) asymptotiquement les mêmes, ce que nous faisons dans le lemme suivant.

Lemme 6.5 *Quand N tend vers l'infini, les complexités en bits moyennes des algorithmes $\overline{\mathcal{LE}}_\mu$ et $\mathcal{LE}_\mu$ sur les ensembles $\Omega_N, \widetilde{\Omega}_N$ sont les mêmes.*

Le lemme 6.3 montre que les longueurs $\delta(j)$ des phases sont proches de $\bar{p} := \lfloor (\mu/2)p \rfloor$. Nous allons séparer l'ensemble Ω des entrées de l'algorithme en deux sous-ensembles : l'ensemble formé des entrées dont le comportement est "exceptionnel" au regard de la longueur des phases, et l'ensemble d'entrées "ordinaires". Soit $\varepsilon > 0$, et considérons l'évènement

$$D(\varepsilon) := [\exists j \leq J,\ \ |\delta(j) - \bar{p}| > \varepsilon p]. \tag{6.6}$$

Il existe un $K < 1$ tel que $\mathbb{P}_N[D(\varepsilon)] = O(K^N)$, donc l'ensemble $D(\varepsilon)$ est bien un ensemble exceptionnel. Il est donc suffisant d'étudier la complexité en bits de l'algorithme $\mathcal{LE}$ sur le complémentaire de $D(\varepsilon)$. Les longueurs "ordinaires" de phases varient entre deux quantités,

$$\bar{p}_- := \left\lfloor \left(\frac{\mu}{2} - \varepsilon\right) p \right\rfloor, \qquad \bar{p}_+ := \left\lceil \left(\frac{\mu}{2} + \varepsilon\right) p \right\rceil,$$

et donc les indices de départ $p(j), p(j+1)$ vérifient

$$j\bar{p}_- \leq p(j) \leq j\bar{p}_+,$$

$$[j\bar{p}_+, (j+1)\bar{p}_-] \subset [p(j), p(j+1)] \subset [j\bar{p}_-, (j+1)\bar{p}_+].$$

Remarquons enfin que le nombre J de phases vérifie

$$J^- := \frac{2}{\mu + 2\varepsilon} \leq J \leq J^+ := \frac{2}{\mu - 2\varepsilon} \tag{6.7}$$

et que la longueur du grand intervalle $[j\bar{p}_-, (j+1)\bar{p}_+]$ est inférieure à $\bar{p}_{++} := \bar{p} + J^+\varepsilon p$, alors que la longueur du petit intervalle est supérieure à $\bar{p}_{--} := \bar{p} - J^-\varepsilon p$. L'application $i \mapsto \ell(U_i)$ est décroissante, et la fonction $[r,t] \mapsto a[r,t]$ est une fonction croissante. Nous pouvons donc fournir des bornes inférieures et supérieures aux trois coûts à étudier,

Bornes supérieures.

$$\text{Coût de type } 1' : \quad \sum_{j=1}^{J^+} \ell(U_{(j-1)\bar{p}_-}) \cdot \sum_{i=(j-1)\bar{p}_-}^{j\bar{p}_+-1} b(q_i)$$

$$\text{Coût de type } 2 : \quad 2\sum_{j=1}^{J^+} \sum_{i=(j-1)\bar{p}_-}^{j\bar{p}_+-1} \ell(a[(j-1)\bar{p}_-, i]) \cdot c(q_i)$$

$$\text{Coût de type } 3 : \quad 4\sum_{j=1}^{J^+} \ell(U_{(j-1)\bar{p}_-}) \cdot \ell(a[(j-1)\bar{p}_-, j\bar{p}_+])$$

Bornes inférieures.

$$\text{Coût de type } 1' : \quad \sum_{j=1}^{J^-} \ell(U_{(j-1)\bar{p}_+}) \cdot \sum_{i=(j-1)\bar{p}_+}^{j\bar{p}_--1} b(q_i)$$

$$\text{Coût de type } 2 : \quad 2\sum_{j=1}^{J^-} \sum_{i=(j-1)\bar{p}_+}^{j\bar{p}_--1} \ell(a[(j-1)\bar{p}_+, i]) \cdot c(q_i)$$

$$\text{Coût de type } 3 : \quad 4\sum_{j=1}^{J^-} \ell(U_{(j-1)\bar{p}_+}) \cdot \ell(a[(j-1)\bar{p}_+, j\bar{p}_-])$$

A chacun des coût ci-dessus, on peut donc associer une série de Dirichlet, qu'on relie aisément à l'opérateur $\mathbf{G}_{s,0}$, de la manière suivante (par simplicité, nous n'indiquons ici que le coût relatif à la j-ème phase).

Séries de Dirichlet pour les bornes supérieures

$$\text{Coût de type } 1' : \quad \sum_{i=1}^{\bar{p}_{++}} \Delta\left[\mathbf{G}_{s,0}^{p-(j-1)\bar{p}_- -i} \circ \mathbf{G}_{s,0}^{[b]} \circ \mathbf{G}_{s,0}^{i-1}\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}_-}[1](0)$$

$$\text{Coût de type } 2 : \quad 2\sum_{i=1}^{\bar{p}_{++}} \mathbf{G}_{s,0}^{p-(j-1)\bar{p}_- -i} \circ \mathbf{G}_{s,0}^{[c]} \circ \Delta\left[\mathbf{G}_{s,0}^{i-1}\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}_-}[1](0)$$

$$\text{Coût de type } 3 : \quad 4\Delta\left[\mathbf{G}_{s,0}^{p-j\bar{p}_+} \circ \Delta\left(\mathbf{G}_{s,0}^{\bar{p}_{++}}\right)\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}_-}[1](0)$$

Séries de Dirichlet pour les bornes inférieures

$$\text{Coût de type } 1' : \quad \sum_{i=1}^{\bar{p}_{--}} \Delta\left[\mathbf{G}_{s,0}^{p-(j-1)\bar{p}_+ -i} \circ \mathbf{G}_{s,0}^{[b]} \circ \mathbf{G}_{s,0}^{i-1}\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}_+}[1](0)$$

$$\text{Coût de type } 2 : \quad 2\sum_{i=1}^{\bar{p}_{--}} \mathbf{G}_{s,0}^{p-(j-1)\bar{p}_+ -i} \circ \mathbf{G}_{s,0}^{[c]} \circ \Delta\left[\mathbf{G}_{s,0}^{i-1}\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}_+}[1](0) \tag{6.8}$$

$$\text{Coût de type } 3 : \quad 4\Delta\left[\mathbf{G}_{s,0}^{p-j\bar{p}_-} \circ \Delta\left(\mathbf{G}_{s,0}^{\bar{p}_{--}}\right)\right] \circ \mathbf{G}_{s,0}^{(j-1)\bar{p}_+}[1](0)$$

Si on somme sur tous les nombres d'itérations p possibles, les termes dominants des séries des deux derniers tableaux contiennent un produit de deux facteurs. Le premier est commun à toutes les séries, et est de la forme

$$\sum_{p\geq 0} p^2 \lambda(s,0)^p \sim 2\left(\frac{1}{1-\lambda(s,0)}\right)^3.$$

Il reste à étudier le second membre, et à le comparer à la complexité en bits de l'algorithme $\overline{\mathcal{LE}}_\mu$. Les opérateurs apparaissant dans (6.5) et (6.8) sont les mêmes, seules les constantes changent. Plus précisément, les constantes relatives à $\overline{\mathcal{LE}}_\mu$, c'est-à-dire

$$A(\mu) := \frac{\mu}{2}\left[1 - \frac{\mu}{2}(j-1)\right], \qquad B(\mu) := \frac{1}{2}\left(\frac{\mu}{2}\right)^2$$

sont remplacées par $A(\mu) + O(\varepsilon), B(\mu) + O(\varepsilon)$. Sommons maintenant sur toutes les valeurs possibles de l'indice j. Celui-ci varie selon les cas entre 1 et J^- ou entre 1 et J^+ (ces valeurs sont définies en (6.7)). On en déduit maintenant le résultat.

6.4 Conclusion

Nous venons de procéder à la première analyse en moyenne d'un algorithme à la structure complexe, et cette approche met en évidence la proportion de divisions (sur des petits entiers) et de multiplications (sur des grands entiers). Pour obtenir une analyse qui prend en compte l'aspect simple ou multi précision des calculs, il faut pouvoir travailler avec une taille $m := \lfloor \mu N \rfloor$ fixe, et donc avec un paramètre μ de la forme $\mu = K/N$, ou K est la taille d'un entier simple-précision par exemple. Les arguments employés jusqu'ici ne permettent pas de traiter ce cas rigoureusement.

Conclusion

Nous venons, au travers d'une description générale de la méthode, de montrer des applications très variées de l'analyse dynamique d'algorithmes. Le champ d'application de cette méthodologie s'est en effet élargi : les différents algorithmes étudiés ici nous mènent tous dans des cadres différents, et pour chacun d'entre eux nous avons dû développer des techniques spécifiques. Ceci nous a permis d'enrichir le domaine dans des directions différentes.

L'analyse des algorithmes α-euclidiens a imposé une approche différente de celles présentées dans [Val97, Val98b, Val00, Val03, Val98a] par exemple de l'étude fonctionnelle des opérateurs. Les discontinuités obligent à travailler sur un espace fonctionnel dans lequel l'analyse est différente, et l'approche utilisée, basée sur les théorèmes de Ionescu-Tulcea et Marinescu ou d'Hennion, s'avère particulièrement maniable et fructueuse. C'est une approche relativement souple puisqu'elle s'applique à différents espaces fonctionnels, comme nous l'avons montré dans cette thèse. Elle permet ainsi d'étudier des systèmes dynamiques de natures diverses, et donc des algorithmes euclidiens variés.

Nous avons également étudié des algorithmes dont l'extension naturelle n'est pas une extension réelle, comme c'est le cas pour l'algorithme LSB. C'est la première instance d'analyse dynamique qui conduit à cette situation, et la méthode s'avère être suffisamment robuste puisqu'elle s'applique également. Au passage, nous avons utilisé des arguments nouveaux dans notre domaine, basés sur les systèmes de fonctions itérées et les produits de matrices aléatoires. Cette approche s'avère être pertinente, puisque le comportement de l'algorithme s'exprime en fonction d'un exposant de Lyapunov, grandeur caractéristique d'un système de produits de matrices aléatoires.

Enfin, l'analyse dynamique s'applique dorénavant à des algorithmes à la structure plus complexe, comme l'algorithme de Lehmer-Euclide. Elle permet d'avoir une compréhension fine des différents mécanismes de l'algorithme. En particulier, elle permet d'accéder à des paramètres non étudiés jusqu'alors, relatifs aux algorithmes interrompus. Cette étude se situe à un autre niveau que les précédentes, puisque l'étude ne concerne plus un paramètre global de l'algorithme, mais un paramètre local. Là encore, nous avons montré que l'analyse dynamique permet de répondre aux questions qu'on se pose dans ce cadre.

Finalement, nous avons aussi montré les limites actuelles de l'analyse dynamique, puisque les méthodes présentées dans cette thèse ne s'appliquent pas à l'algorithme Plus-Moins, pourtant si proche de l'algorithme Binaire...

Nous avons également apporté des réponses parfois surprenantes aux questions posées : nous avons en effet précisé la classification des algorithmes faite par Brigitte Vallée, puisque nous avons montré que, parmi l'ensemble des algorithmes α-euclidiens, seul l'algorithme Par-Excès appartient à la classe des algorithmes lents. Nous avons de plus soulevé un point intrigant, puisque la dépendance en α du comportement de ces algorithmes n'est pas uniforme sur $[0, 1]$, et que de manière plus surprenante encore cette dépendance est nulle pour certaines valeurs de α.

Nous avons présenté dans cette thèse la première analyse de l'algorithme LSB. Son fonctionnement est maintenant bien compris, et nous avons mis en évidence quelques-une de ses particularités, observées expérimentalement par Stehlé [Ste]. En particulier, nous avons confirmé la pertinence de cet algorithme comme brique de base d'un algorithme Diviser pour Régner, ceci grâce aux propriétés de stabilité que nous avons montrées (densité invariante préservée, etc...). Nous avons de même rencontré au cours de l'analyse une différence de comportement étonnante entre l'algorithme et son extension continue : c'est la première fois que nous observons un tel phénomène, qui mériterait d'être mieux compris.

Enfin nous avons effectué la première étude précise d'un algorithme à la structure plus complexe. Même si cette étude ne prend pas en compte certains des principaux paramètres du calcul multi-précision, nous avons pu mesurer la part des différentes opérations de l'algorithme, et confirmer que sous certaines hypothèses généralement vérifiées, celui-ci est bien plus rapide que l'algorithme d'Euclide.

Ce travail soulève également des interrogations, et peut se prolonger dans plusieurs directions. On peut tout d'abord tenter de répondre aux quelques questions évoquées précédemment, sur les algorithmes α-euclidiens ou l'algorithme LSB. Mais on peut également essayer d'utiliser la méthodologie présentée dans cette thèse dans un cadre encore plus vaste.

Il existe en effet d'autres algorithmes dont l'analyse n'a pas été faite. On pense en particulier à la classe des algorithmes Mixtes, dans laquelle seul l'algorithme Binaire et les algorithmes pseudo-euclidiens ont été analysés. Un problème encore ouvert est celui de l'analyse de l'algorithme Plus-Moins. On peut raisonnablement penser qu'une fois cet algorithme analysé, les analyses des généralisations de Weber, Sorenson ou Jebelean seront accessibles avec les méthodes dynamiques.

Le fonctionnement des algorithmes Diviser pour Régner, comme ceux de Schönhage ou de Stehlé et Zimmermann, est encore mal compris. L'étude de l'algorithme de Lehmer-Euclide est clairement un premier pas vers l'analyse de ces algorithmes, mais ce premier pas est pour l'instant insuffisant, puisque notre analyse ne nous fournit pas assez de précision pour analyser un algorithme récursif. En particulier ce type d'analyse nécessite d'avoir des termes de restes très précis, ce que ne fournit pas l'analyse en moyenne. Il est possible à partir des travaux présentés dans ma thèse de définir deux approches possibles. La première consiste à adapter les travaux de Baladi et Vallée pour faire une analyse en distribution des algorithmes interrompus. Ce type d'analyse permet usuellement d'obtenir les termes de restes souhaités, et on peut donc envisager de s'attaquer de cette manière à l'algorithme de Schönhage. D'un autre côté, il peut être plus judicieux d'adopter la même démarche à partir de l'algorithme LSB. D'une part, ceci enrichit notre compréhension d'un algorithme d'un type particulier, et d'autre part la stabilité de l'algorithme facilite l'étude de la version récursive, en particulier quand on se penche sur l'évolution des distributions. Quoiqu'il en soit, une question pertinente concerne la comparaison des algorithmes rapides en fonction de la division utilisée. En effet, cette comparaison est partiellement faite dans ma thèse pour les algorithmes "simples", et il est intéressant de savoir comment elle se répercute à un niveau supérieur.

Une des dernières extensions évidentes de ces analyses porte sur la généralisation de cette méthode à des algorithmes fonctionnant dans des dimensions plus élevées. En effet, les algorithmes euclidiens se généralisent en dimension supérieure en algorithmes de réduction des réseaux comme l'algorithme de Gauss en dimension 2 ou l'algorithme LLL. Un grand pas en avant serait fait si l'analyse dynamique permettait d'analyser ces algorithmes. En réalité, l'algorithme de Gauss a déjà été analysé, grâce à des méthodes dynamiques, par Daudé, Flajolet et Vallée [DFV97, Val97], mais la généralisation à des dimensions supérieure reste un problème ouvert.

Bibliographie

[AV00] A. Akhavi et B. Vallée. Average bit–complexity of euclidean algorithms. In *Proceedings of ICALP'2000*, volume 1853, pages 373–387, 2000.

[Bal00] V. Baladi. *Positive Transfer operators and decay of correlations*, volume 16 of *Advanced Series in non linear dynamics*. World Scientific, 2000.

[BDV02] J. Bourdon, B. Daireaux, et B. Vallée. Dynamical analysis of α-euclidean algorithms. *Journal of Algorithms*, 44 :246–285, 2002.

[BG97] A. Boyarsky et P. Gora. *Laws of Chaos, Invariant measures and dynamical systems in one dimension*. Probability and its applications. Birkhauser, 1997.

[BK85] R. P. Brent et H.T. Kung. A systolic VLSI array for integer GCD computation. In K. Hwang, éditeur, *ARITH-7, Proceedings of the Seventh Symposium on Computer Arithmetic*, pages 118–125. IEEE CS Press, 1985.

[BK90] V. Baladi et G. Keller. Zeta functions and transfer operators for piecewise monotone transformations. *Commun. Math. Phys.*, 127 :459–477, 1990.

[BL85] P. Bougerol et J. Lacroix. *Products of Random Matrices with Applications to Schrödinger Operators*, volume 8 of *Progress in Probability and Statistics*. Birkhäuser, Boston, 1985.

[BNV01] J. Bourdon, M. Nebel, et B. Vallée. On the stack-size of general tries. *ITA*, 35(2) :163–185, 2001.

[Bou01] J. Bourdon. Size and path length of Patricia tries : Dynamical sources context. *Random Structures and Algorithms*, 19(3-4) :289–315, 2001.

[Bre76] R.P. Brent. *Analysis of the Binary Euclidean algorithm*. Algorithms and Complexity, New directions and recent results. Academic Press, 1976.

[Bro78] J. Browkin. Continued fractions in local fields. I. *Demonstratio Math.*, 11 :67–82, 1978.

[Bro96] A. Broise. Transformations dilatantes de l'intervalle et théorèmes limites. *Asterisque*, 238 :5–109, 1996.

[Bro01] J. Browkin. Continued fractions in local fields. II. *Math. Comp.*, 70(235) :1281–1292, 2001.

[BV02] J. Bourdon et B. Vallée. Generalized pattern matching statistics. In *Proceedings of the "colloquium on Mathematics and Computer Science : Algorithms and Trees"*, Trends in Mathematics, pages 249–265. Birkhauser, 2002.

[BV04] V. Baladi et B. Vallée. Euclidean algorithms are gaussian. *Journal of Number Theory*, 2004.

[Ces98] G. Cesari. Parallel implementation of Schönhage's integer GCD algorithm. In *Proceedings of ANTS-III*, volume 1423, pages 64–76, 1998.

[CFV98] J. Clément, P. Flajolet, et B. Vallée. The analysis of hybrid trie structures. In *SODA*, pages 531–539, 1998.

[CFV01] J. Clément, P. Flajolet, et B. Vallée. Dynamical sources in information theory : A general analysis of trie structures. *Algorithmica*, 29(1) :307–369, 2001.

[Clé00] J. Clément. *Arbres Digitaux et Sources Dynamiques.* Thèse de doctorat, Université de Caen, 2000.

[Col] G.E. Collins. Lecture notes on arithmetic algorithms, univ. of wisconsin.

[Col96] P. Collet. Some ergodic properties of maps of the interval. In *Dynamical systems, Proceedings of the first UNESCO CIMPA School on Dynamical and Disordered Systems, (Temuco, Chile, 1991).* Hermann, 1996.

[Dai04] B. Daireaux. Dynamical analysis of the GCD algorithm directed by Least Significant Bits. Rapport technique, GREYC, Université de Caen, 2004.

[Del54] H. Delange. Généralisation du théorème d'Ikehara. *Ann. Sc. ENS*, 71 :213–242, 1954.

[DFV97] H. Daudé, P. Flajolet, et B. Vallée. An average-case analysis of the Gaussian algorithm for lattice reduction. *Combinatorics, Probability & Computing*, 6(4) :397–433, 1997.

[Dix70] J.D. Dixon. The number of steps in the Euclidean algorithm. *Journal of Number Theory*, 2 :414–422, 1970.

[DMDV05] B. Daireaux, V. Maume-Deschamps, et B. Vallée. The Lyapunov tortoise and the Dyadic hare. *Discrete Mathematics and Theoretical Computer Science*, 2005. à paraitre.

[DS58] N. Dunford et J. Schwartz. *Linear Operators. Part I. General Theory*, volume 7 of *Pure and Applied Mathematics.* Interscience Publishers, Inc., New York ; Interscience Publishers, Ltd., London, 1958.

[DV04] B. Daireaux et B. Vallée. Dynamical analysis of the parameterized Lehmer-Euclid algorithm. *Combinatorics, Probability and Computing*, 2004.

[Fla] P. Flajolet. Personal communication.

[Fla92] P. Flajolet. Analytic analysis of algorithms. In W. Kuich, éditeur, *Proceedings of the 19th International Colloquium "Automata, Languages and Programming", Vienna, July 1992*, volume 623, pages 186–210, 1992.

[Fla02] P. Flajolet. Singular combinatorics. In *Proceedings of the International Congress of Mathematicians 2002*, volume III, pages 561–571. World Scientific, 2002.

[FS] P. Flajolet et R. Sedgewick. Analytic combinatorics. `http ://algo.inria.fr/flajolet/Publications/books.html`.

[FS96] P. Flajolet et R. Sedgewick. *An introduction to the analysis of algorithms.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1996.

[Fur63] H. Furstenberg. Non commuting random products. *Trans. amer. Math. Soc.*, 108 :377–428, 1963.

[FV98] P. Flajolet et B. Vallée. Continued fraction algorithms, functional operators, and structure constants. *Theor. Comput. Sci.*, 194(1-2) :1–34, 1998.

[Ga02] T. Granlund et al. GNU Multiple Precision arithmetic library 4.1.2, December 2002. `http ://swox.com/gmp/`.

[GR85] Y. Guivarc'h et A. Raugi. Frontières de Furstenberg, propriétés de contraction et théorèmes de convergence. *Zeit. fur Wahrscheinlichkeitstheorie und Verw. Gebiete*, 69 :187–242, 1985.

[Hei69] H. Heilbronn. On the average length of a class of continued fractions. *Number Theory and Analysis*, pages 87–96, 1969.

[Hen93] H. Hennion. Sur un théorème spectral et son application aux noyaux lipschitziens. *Proc. Amer. Math. Soc.*, 118 :627–634, 1993.

[Hen94] D. Hensley. The number of steps in the Euclidean algorithm. *Journal of Number Theory*, 49(2) :142–182, 1994.

[Hwa98] H. K. Hwang. On convergence rates in the central limit theorem for combinatorial structures. *European Journal of Combinatorics*, 19 :329–343, 1998.

[ITM50] C.T. Ionescu Tulcea et G. Marinescu. Théorie ergodique pour des classes d'opérations non complètement continues. *Ann. of Math.*, 52(2) :140–147, 1950.

[Jeb93a] T. Jebelan. Comparing several GCD algorithms. In E. E. Swartzlander, M. J. Irwin, et J. Jullien, éditeurs, *Proceedings of the 11th IEEE Symposium on Computer Arithmetic*, pages 180–185, Windsor, Canada, June 1993. IEEE Computer Society Press, Los Alamitos, CA.

[Jeb93b] T. Jebelean. A generalization of the Binary GCD algorithm. In *ISSAC*, pages 111–116, 1993.

[Jeb95] T. Jebelean. A double-digit Lehmer-Euclid algorithm for finding the GCD of long integers. *J. Symb. Comput.*, 19(1-3) :145–157, 1995.

[Kat80] T. Kato. *Perturbation Theory for Linear Operators*. Springer-Verlag, 1980.

[Knu98] D.E. Knuth. *The art of Computer programming*, volume 2. Addison Wesley, Reading, Massachussets, 3rd édition, 1998.

[Kra64] A. Krasnoselskii. *Positive solutions of operators equations*. Groninger : The Netherlands : P.Noordhoff Ltd., 1964.

[Kuz28] R. Kuzmin. Sur un problème de Gauss. In *Atti Congr. Inter. Bologna*, volume 6, pages 83–89, 1928.

[Ler97] R. Lercier. *Algorithmique des courbes elliptiques dans les corps finis*. Thèse de doctorat, Ecole Polytechnique, 1997.

[Lho05] L. Lhote. Normality of bit-complexities for euclidean algorithms. Rapport technique, GREYC, Université de Caen, 2005.

[LM94] A. Lasota et M. Mackey. *Chaos, Fractals and Noise ; Stochastic Aspects of Dynamics*, volume 97 of *Applied Mathematical Science*. Springer, 1994.

[LP82] E. Le Page. *Théorèmes limites pour les produits de matrices aléatoires*, volume 928 of *Lectures Notes in Math.*, pages 258–303. Springer, New-York, 1982.

[LY73] A. Lasota et J.A. Yorke. On the existence of invariant measures for piecewise monotonic transformations. *Trans. Amer. Math. Soc.*, 186 :481–488, 1973.

[Lév29] P. Lévy. Sur les lois de probabilités dont dépendent les quotients complets et incomplets d'une fraction continue. *Bull. Soc. Math. France*, 57 :178–194, 1929.

[May91] D. Mayer. Continued fractions and related transformations. In T. Bedford, M. Keane, et C. Series, éditeurs, *ergodic theory, symbolic dynamics and hyperbolic spaces*, Chapitre 7, pages 175–222. Oxford science Publications, 1991.

[MCM99] P. Moussa, A. Cassa, et S. Marmi. Continued fractions and Brjuno functions. *Journal of Computational and Applied Mathematics*, 105 :403–415, 1999.

[Nak81] H. Nakada. Metrical theory for a class of continued fraction transformations and their natural extensions. *Tokyo J. Math.*, 4 :399–426, 1981.

[Nus70] R. Nussbaum. The radius of the essential spectrum. *Duke Math. J.*, 37 :473–478, 1970.

[Phi70] W. Philipp. Some metrical results in number theory II. *Duke Math J.*, 38 :447–488, 1970.

[RE83] J. Rousseau-Egele. Un théorème de la limite locale pour une classe de transformations dilatantes et monotones par morceaux. *Anaals of Probability*, 3 :772–788, 1983.

[Rie78] G. J. Rieger. Uber die mittlere schrittanzahl bei divisionalgorithmen. *Math. Nachr.*, pages 157–180, 1978.

[Rub70] A. A. Ruban. Certain metric properties of p-adic numbers. *Sibiesk. Math. Zh.*, 11 :222–227, 1970.

[Rud87] W. Rudin. *Real and complex analysis, 3rd ed.* McGraw-Hill, Inc., New York, NY, USA, 1987.

[Rue78] D. Ruelle. *Thermodynamic formalism.* Addison Wesley, 1978.

[Sed01] S.M. Sedjelmaci. On a parallel Lehmer-Euclid GCD algorithm. In *ISSAC*, pages 303–308, 2001.

[Sha90] J. Shallit. On the worst–case of the three algorithmss for computing the Jacobi symbol. *Journal of Symbolic Computation*, 10 :593–610, 1990.

[SL97] S.M. Sedjelmaci et C. Lavault. Improvements on the accelerated integer GCD algorithm. *Inf. Process. Lett.*, 61(1) :31–36, 1997.

[Sor94] J. Sorenson. Two fast GCD algorithms. *J. Algorithms*, 16(1) :110–144, 1994.

[Sor95] J. Sorenson. An analysis of Lehmer's euclidean GCD algorithm. In *Proceedings of ISSAC'95*, pages 254–258, 1995.

[Ste] D. Stehlé. Web page. `http ://www.loria.fr/~stehle/BINARY.html`.

[Ste67] J. Stein. Computational problems associated with Racah algebra. *Journal of Computational Physics*, 1 :397–405, 1967.

[SZ04] D. Stehlé et P. Zimmermann. A Binary recursive Gcd algorithm. In *Proceedings of the 6th Algorithmic Number Theory Symposium*, volume 3076, 2004.

[Val97] B. Vallée. Opérateurs de Ruelle-Mayer généralisés et analyse en moyenne des algorithmes d'Euclide et de Gauss. *Acta Arithmetica*, 2(LXXXI) :101–144, 1997.

[Val98a] B. Vallée. Dynamics of the binary euclidean algorithm : Functional analysis and operators. *Algorithmica*, 22(4) :660–685, 1998.

[Val98b] B. Vallée. Fractions continues à contraintes périodiques. *Journal of Number Theory*, 72 :183–235, 1998.

[Val00] B. Vallée. Digits and continuants in euclidean algorirthms. ergodic versus tauberian theorems. *Journal de Théorie des Nombres de Bordeaux*, 12 :531–570, 2000.

[Val01] B. Vallée. Dynamical sources in information theory : Fundamental intervals and word prefixes. *Algorithmica*, 29(1) :262–306, 2001.

[Val03] B. Vallée. Dynamical analysis of a class of euclidean algorithms. *Theoretical Computer Science*, 1-3(297) :447–486, 2003.

[Var] I. Vardi. Continued fractions. preprint, book in preparation.

[VVZ94] V. S. Vladimirov, I. V. Volovich, et E. I. Zelenov. *P-Adic Analysis and Mathematical Physics*, volume 1 of *Series on Soviet & East European Mathematics*. 1994.

[Web95] K. Weber. The accelerated integer GCD algorithm. *ACM Trans. Math. Softw.*, 21(1) :111–122, 1995.

[Wir74] E. Wirsing. On the theorem of Gauss-Kuzmin-Levy and a Frobenius type theorem for function spaces. *Acta Arithm.*, 24 :507–528, 1974.

[YK75] A.C. Yao et D.E. Knuth. Analysis of the subtractive algorithm for greatest common divisors. In *Proc. Nat. Acad. Sc. USA*, volume 72, pages 4720–4722, 1975.