N° d'ordre :

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE & POPULAIRE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR & DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ DJILLALI LIABES
FACULTÉ DES SCIENCES EXACTES
SIDI BEL ABBÈS

THESE DE DOCTORAT EN SCIENCE

Présentée par

BOUCHAKOUR ERRAHMANI Hichem

Spécialité : Informatique
Option : Systèmes d'information Web (SIW)

Intitulée

Sur la sécurité de l'information par le biais des courbes elliptiques

Soutenue le 06 décembre 2018
Devant le jury composé de :

Président : Dr. Boukli Hacène Sofiane, M.C.A. – U.D.L. Sidi Bel Abbès
Examinateurs: Dr. Badr Benmammar, M.C.A. – Université de Tlemcen
Dr. Keskes Nabil, M.C.A. – E.S.I. Sidi Bel Abbès
Dr. Hamou Reda mohamed, M.C.A. – Université de Saïda
Dr. Ali Cherif moussa, M.C.A. – U.D.L. Sidi Bel Abbès
Directeur de thèse : Pr. Faraoun Kamel Mohamed, Professeur – U.D.L. SBA

Année universitaire 2018/2019

On The Information Security Through Elliptic Curves

Hichem **Bouchakour Errahmani**

2018

Résumé

La cryptographie prend de plus en plus de place dans la société actuelle. Les cartes bancaires, DVD, achats en ligne... nécessitent des systèmes de protection sûrs et rapides. Ce qui était autrefois réservé aux armés et aux gouvernements entre peu à peu dans la vie quotidienne. Les systèmes cryptographiques à base de courbes elliptiques sont aujourd'hui de plus en plus employés dans les protocoles utilisant la cryptographie à clef publique. Ceci est particulièrement vrai dans le monde de l'embarqué qui est soumis à de fortes contraintes de coût, de ressources et d'efficacité, car la cryptographie à base de courbes elliptiques permet de réduire significativement la taille des clefs utilisées par rapport à d'autres systèmes cryptographiques tels que RSA.

Dans cette thèse, nous présentons une nouvelle approche de chiffrement d'image en utilisant un chiffrement par flot, le générateur de nombres pseudo-aléatoires est basé sur les automates cellulaires unidimensionnels élémentaires (AC) et les courbes elliptiques. En effet, nous avons exploré les transitions de l'AC avec les coordonnées d'un point appartenant à une courbe elliptique ; ce point est le résultat d'un autre point qui a été multiplié par un scalaire, connu sous le nom du problème de logarithme discret de courbe elliptique (PLDCE). Ce dernier complique la génération d'un point, et rend impossible de trouver son antécédent. AC offre aussi des qualités d'ambiguïté et de chaos, combinant ainsi les deux concepts ; nous avons construit un générateur qui génère un flux de clé, utilisé dans notre approche. Ce travail constitue une analogie avec des travaux impliquant les systèmes dynamiques comme les suites logistiques avec les courbes elliptiques. Notre générateur a montré de bonnes propriétés cryptographiques, car il est basé sur le PLDCE. Nous avons testé les images cryptées, et il s'avère que les résultats sont de haute performance.

Mots clés : Cryptographie par les courbes elliptiques, Automates cellulaires, Générateur de nombres pseudo-aléatoires, Chiffrement d'image.

Abstract

Cryptography is gaining more and more place in today's society. Bank cards, DVDs, online purchases ... require safe and fast protection systems. What was formerly reserved for armies and governments gradually enters everyday life. Cryptographic systems based on elliptic curves are nowadays increasingly used in protocols using public key cryptography. This is particularly true in the embedded world, which is subject to high cost, resource and efficiency constraints, since elliptic curve cryptography significantly reduces the size of the keys used compared to other cryptographic systems such as RSA.

In this thesis, we present a new approach of image encryption using a stream cipher, the pseudo-random number generator is based on the elementary one dimensional cellular automata (CA) and elliptic curves. Indeed, we explored the transitions of the CA with the coordinates of a point belonging to an elliptic curve; outcome from another point which was multiplied by a scalar, known as the Elliptic Curve Discrete Logarithm Problem (ECDLP). This last complicates the generation of a point, and makes it impossible to find its antecedent. CA also offers the qualities of ambiguity and chaos, so combining the two concepts; we have constructed a PRNG that generates a key stream, used by the way in our approach. This work constitutes an analogy to works that involved dynamic systems like logistics map with elliptic curves. Our PRNG showed good cryptographic properties, since it is based on the ECDLP. We tested the encrypted images, and it turns out that the results are high performance.

Keywords: Elliptic Curve Cryptography, Cellular Automata, Pseudo-Random Number Generator, Image Encryption.

Table des matières

Table des figures

Liste des tableaux

Listes de symboles et abréviations

$\mathbb{Z}$: L'ensemble des entiers relatifs

$\mathbb{R}$: L'ensemble des réels

$\mathbb{Q}$: L'ensemble des rationnels

$\mathbb{N}$: L'ensemble des entiers naturels

C : L'ensemble des nombres complexes

F_p : Corps fini d'ordre p

$E(F_p)$: Courbe elliptique E définie sur F_p

$\#E(F_p)$: Nombres de points de la courbe $E(F_p)$

PLD : Problème du logarithme discret

PLDCE : Problème du logarithme discret sur les courbes elliptiques

PEDH : Problème elliptique de Diffie-Hellman

CCE : Cryptgraphie par les courbes elliptiques

RSA : Cryptosystème asymétrique Rivest Shamir Adleman

AC : Automate cellulaire

PRNG : Pseudo Random Number Generator

LFSR : Linear Feedback Shift Register

Chapitre 1

Introduction générale

Sommaire

1.1 Contexte général

Dans cette thèse, nous allons étudier la combinaison des courbes elliptiques et les automates cellulaires pour construire un générateur de séquences pseudo-aléatoires, qui sera par la suite, mis en œuvre dans le chiffrement par flot des images numériques.

1.1.1 Cryptographie et courbe elliptique

Depuis toujours la communication entre les peuples a existé, et s'est développé au cours des siècles. Que ce soit dans son contenu qui était seulement des écritures qui se sont développées vers des images, des vidéo, et au multimédia d'aujourd'hui, ou par son moyen de transport.

Et c'est ainsi qu'avec les tensions qui régnaient entre les peuples, ce qui a engendré des conflits, des concurrences voir même des guerres, tous cela a poussé les parties en conflits à protéger leurs communications et leurs données, d'où l'apparition de la *cryptographie* ; une science qui permet de chiffrer les données et les rendre incompréhensibles pour une personne non autorisée à y accéder.

Par conséquent, la technologie n'a cessé de croître en performance et en miniaturisation, les techniques aussi, la plus récente d'entre elles est les *courbes elliptiques*, ce sont des objets mathématiques qui ont montré leur efficacité dans la cryptographie, surtout en réduisant la taille des clés de chiffrement, et leurs capacités d'être utilisé dans des systèmes embarquées tels que les téléphones mobiles, les tablettes, les capteurs etc.

Étant donné que notre siècle est caractérisé par son impulsion technologique de multimédia, des images, des vidéos. Leurs communications nécessite une forte protection avec des outils de chiffrement qui devront être facile à implémenter et moins coûteuse. La solution était d'utiliser un système de *chiffrement par flot* considéré comme étant le plus sûr et le plus facile à implémenter, surtout dans les systèmes embarqués où les performances du processeur et de la mémoire sont limitées.

1.1.2 Automates cellulaires

Les automates cellulaires sont un modèle puissant de calcul, introduits au début des années 1950 par le mathématicien *John Von Neumann*, qui s'intéressait alors à l'autoreproduction des systèmes artificiels.

Depuis leur création, ils ont été étudiés dans divers domaines comme par exemple la physique, la biologie, où ils permettent de modéliser et de simuler divers phénomènes qui ne peuvent pas être analysés directement. Ils ont aussi été considérés en tant que systèmes dynamiques discrets car, bien que leur définition soit très simple,

ils sont capables de produire des comportements complexes difficiles à prévoir, et fournissent ainsi un outil théorique pour l'étude des systèmes complexes discrets. Ces comportements complexes générés à partir de règles simples rendent les automates cellulaires des candidats idéals pour la conception de primitives cryptographiques.

Les automates cellulaires possèdent une structure intéressante pour atteindre des débits maximum. Cependant, malgré leur simplicité, prédire le comportement d'un automate cellulaire en fonction de la règle utilisée est très difficile.

1.1.3 Motivation

Vu l'essor emblématique des systèmes embarquées tel que les smartphones, les tablettes, et les réseaux de capteurs, et la nécessite d'établir une communication sûre entre ces diverses composants, il a fallut voir sous un autre angle vis-à-vis de la cryptographie pour améliorer et mettre en œuvre des cryptosystèmes adaptés à la limites des performance qu'offre ces systèmes.

Ainsi, le multimédia, représenté par sa fameuse unité d'*image* qui fait le support de communication d'aujourd'hui au lieu des textes et manuscrits d'autrefois a besoin d'être protégée via des canaux de transmission qui ne sont généralement pas sécurisés. Par conséquent, la cryptographie symétrique offre une solution pour notre problème décrit ci-après.

1.2 Problématique

Les courbes elliptiques sont indépendants des propriétés que présentent les générateurs de nombres pseudo-aléatoires qui sont utilisés dans le chiffrement par flot. Cela soulève des questions quant à la pertinence du problème de logarithme discret dans les courbes elliptiques et son impact dans la prédictibilité des générateurs de nombres pseudo-aléatoires, sauf si on fait combiner les courbes elliptiques avec les systèmes dynamiques qui présente un aspect aléatoire à titre d'exemple *les automates cellulaires*.

Dans cette thèse, nous abordons les questions suivantes :

- Comment les automates cellulaires peuvent fournir un caractère aléatoire pour les générateurs de nombres pseudo-aléatoires ?
- Comment combiner les automates cellulaires avec les courbes elliptiques pour générer un Keystream pour le chiffrement par flôt ?
- Comment utiliser ce chiffrement dans une plate-forme contenant des images numériques ?

1.3 Démarche de la recherche et contribution

Pour répondre à cette multitude de questions, les principales contributions de ce travail de thèse sont les suivantes :

- L'élaboration d'un état de l'art détaillé survolant les différentes approches de génération de séquences pseudo-aléatoires, ainsi que les approches de chiffrement par flot des images numériques en utilisant les courbes elliptiques.
- Proposition d'une nouvelle approche de chiffrement par flot, dont le générateur de nombres pseudo-aléatoires est basé sur l'utilisation des coordonnées de point générée aléatoirement à partir d'une courbe elliptique, et leur mappage dans une grille construite par les transitions d'un automate cellulaires élémentaires.
- Implémentation d'un prototype pour montrer l'intérêt et la faisabilité de notre approche.

1.4 Organisation de la thèse

Le manuscrit est structuré comme suit :

Le chapitre 2 introduit le lecteur à la notion vertèbre de notre champ de recherche qui est les courbes elliptiques et la cryptographie appliquée dessus. Le troisième chapitre est consacré à la description des automates cellulaires comme un système dynamique performant.

Nous mènerons dans le quatrième chapitre un état de l'art sur les différentes approches qui ont été élaborées pour le développement des générateurs de séquences pseudo-aléatoires et quelques applications dans le chiffrement. Dans le cinquième chapitre, nous introduisons notre approche de développement d'un cryptosystème basée sur le chiffrement par flot en utilisant les courbes elliptiques et les automates cellulaires.

Enfin, Ce mémoire s'achève par une conclusion générale où nous présentons le bilan et les perspectives de notre travail.

Chapitre 2

Courbes elliptiques et Cryptographie

Sommaire

2.1 Introduction

Au cours des dernières années, un sujet sur la théorie des nombres et la géométrie algébrique appelé *courbes elliptiques* a trouvé un champ d'application en cryptographie[HPSS08].

Nous allons présenter un rappel sur les groupes, les définitions de base et les règles sur les courbes elliptiques. Nous n'incluons que l'étendue minimale requis pour comprendre les applications à la cryptographie, mettant l'accent sur des exemples et des descriptions concrètes au détriment des preuves et de la généralité.

2.2 Groupe

Définition 1.

Un groupe $(G, *)$ est un ensemble G muni d'une loi de composition $*$ vérifiant les quatre axiomes suivants :

1. $\forall x, y \in G, x * y \in G$ ($*$ est une loi de composition interne)
2. $\forall x, y, z \in G, (x * y) * z = x * (y * z)$ (la loi est associative)
3. $\exists e \in G$ *tel que* $\forall x \in G, x * e = x$ *et* $e * x = x$ (e est l'élément neutre)
4. $\forall x \in G, \exists \bar{x} \in G$ *tel que* $x * \bar{x} = \bar{x} * x = e$ ($\bar{x}$ est l'inverse de x et est noté x^{-1})

Définition 2 (Groupe abélien)**.**

Si de plus l'opération $*$ vérifie :

$$\forall x, y \in G, x * y = y * x$$

Alors on dit que G est un groupe *commutatif* (ou *abélien*).

Exemples :

Voici des ensembles et des opérations bien connus qui ont une structure de groupe.

- $(\mathbb{Z}, +)$, $(\mathbb{Q}, +)$, $(\mathbb{R}, +)$, et $(\mathbb{C}, +)$ sont des groupes commutatifs. Ici $+$ est l'addition habituelle.

- $(\mathbb{R}^*, \times)$, $(\mathbb{Q}^*, \times)$, $(\mathbb{C}^*, \times)$ sont des groupes commutatifs., $\times$ est la multiplication habituelle.

Voici deux exemples qui **ne sont pas** des groupes :

- $(\mathbb{Z}^*, \times)$ n'est pas un groupe. Car si 2 avait un inverse (pour la multiplication $\times$) ce serait $\frac{1}{2}$ qui n'est pas un entier.
- $(\mathbb{N}, +)$ n'est pas un groupe. En effet l'inverse de 3 (pour l'addition $+$) devrait être -3 mais $-3 \notin \mathbb{N}$.

2.2.1 Ordre

Définition 3.

Soit $(G, *)$ un groupe fini (C'est un groupe dont le nombre d'éléments est fini). Le cardial de l'ensemble G noté $card(G)$ est appelé l'**ordre** du groupe.

Proposition 1 (Théorème de Lagrange)**.**

Soit $(G, *)$ un groupe fini, et H un sous-groupe de G. Alors l'ordre de H divise l'ordre de G.

Exemples :

- Dans n'importe quel groupe, l'ordre de l'élément neutre est 1, et c'est le seul élément d'ordre 1 du groupe.
- Dans $(\mathbb{Z}, +)$, tous les entiers sont d'ordre infini à l'exception de 0 qui est d'ordre 1.
- Dans $(\mathbb{Z}/6\mathbb{Z}, +)$ on a $\dot{2} + \dot{2} + \dot{2} = 0$ et $\dot{2} + \dot{2} = 4 \neq 0$ ce qui montre que 2 est d'ordre 3.
- Dans $(\mathbb{Z}/20\mathbb{Z}, +)$ on a $ord(12) = card(<12>) = 5$.

Corollaire 1 (Déduit du Théorème de Lagrange)**.**

Soit $(G, *)$ un groupe fini, et soit g un élément quelconque de G, l'ordre de g divise le cardinal de G. En particulier $g^{card(G)} = e$ pour tout $g \in G$.

2.2.2 Notion de générateur

Définition 4.

On dit qu'un groupe G est monogène s'il est engendré par l'un de ses éléments g, donc si $G =<g>= \{g^m,\ m \in \mathbb{Z}\}$ (ou $\{mg,\ m \in \mathbb{Z}\}$ en notation additive). On dit alors que g est un **générateur** de G.

2.3 Corps finis

Définition 5 (Notion de corps)**.**

Un corps est un ensemble F muni des opérations de multiplication et d'addition, et qui satisfont les règles familières suivantes :

- l'associativité et la commutativité de l'addition et de la multiplication,
- la loi est distributive,
- l'existence d'un élément neutre additive 0 et d'un élément neutre multiplicative 1,
- l'existence des inverses additifs et des inverses multiplicatives pour Tout élément sauf le 0.

Les exemples suivants de corps sont fondamentaux dans de nombreux domaines de la mathématique :

- le corps $\mathbb{Q}$ constitué de tous les nombres rationnels ;
- le corps $\mathbb{R}$ des nombres réels ;
- le corps $\mathbb{C}$ des nombres complexes ;
- le corps $(\mathbb{Z}/n\mathbb{Z})$ des entiers modulo un nombre premier p.

2.4 Concepts de base sur les courbes elliptiques

Définition 6 (Courbe Elliptique)**.**

Une *courbe elliptique* est l'ensemble des solutions satisfaisant l'équation :

$$Y^2 = X^3 + AX + B \tag{2.1}$$

Les équations de ce type s'appellent les équations de *Weierstrass* [BSSS99] d'après le mathématicien qui les a étudiés au cours du $19^{ème}$ siècle.

Soient deux exemples de courbes elliptiques,

$$E_1 : \quad Y^2 = X^3 - 3X + 3 \quad et \quad E_2 : \quad Y^2 = X^3 - 6X + 5$$

illustrées dans la figure 2.1.

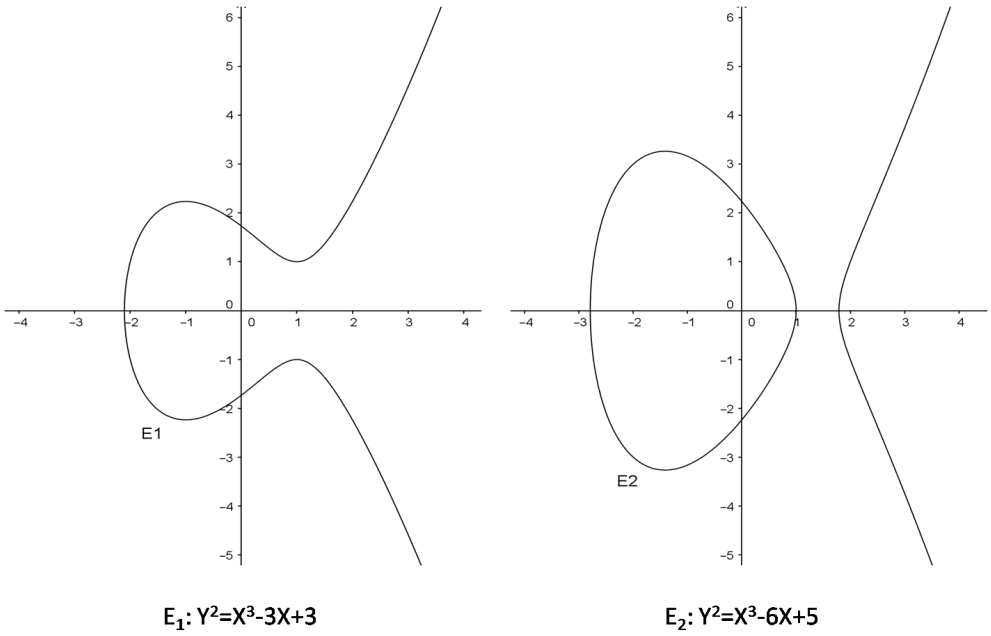

Figure 2.1 – Deux exemples de courbes elliptiques.

Une caractéristique étonnante des courbes elliptiques est qu'il existe un moyen naturel de prendre deux points sur une courbe elliptique et de les "additionner" pour donner un troisième point [HPSS08]. Mettons des guillemets autour du mot «additionner» parce que nous parlons d'une opération qui combine deux points, d'une manière analogue à l'addition (elle est commutatif, associatif, et il y a un élément neutre), mais très différente de l'addition traditionnelle. La manière la plus normale de décrire la «loi additionnelle» sur les courbes elliptiques consiste à utiliser la géométrie.

Soit P et Q deux points sur une courbe elliptique E, comme l'illustre la figure 2.2. Commençons par dessiner la ligne L qui traverse P et Q. Cette ligne L coupe E en trois points, P, Q et R. Prenons ce point R et projetons le sur l'axe des X (c-à-d., multiplions sa coordonnée Y par -1) pour obtenir un nouveau point R'. Le point R' s'appelle la "somme de P et Q".

Pour l'instant, désignons cette étrange loi d'addition par le symbole $\oplus$. Ainsi, nous écrivons :

$$P \oplus Q = R'$$

.

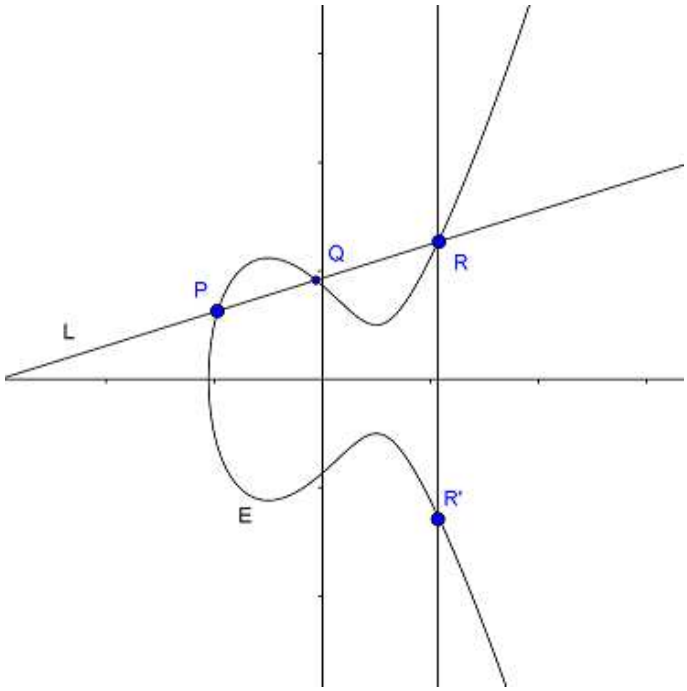

Figure 2.2 – La loi d'addition sur les courbes elliptiques.

Il existe quelques subtilités concernant l'addition des points sur les courbes elliptiques qui doivent être abordées.

D'abord, que se passe-t-il si nous voulons additionner un point P à lui-même ? Imaginez ce qui arrive à la ligne L reliant P et Q si le point Q glisse le long de la courbe et se rapproche de P. Dans la limite, comme Q approche P, la ligne L devient la ligne tangente de E à P.

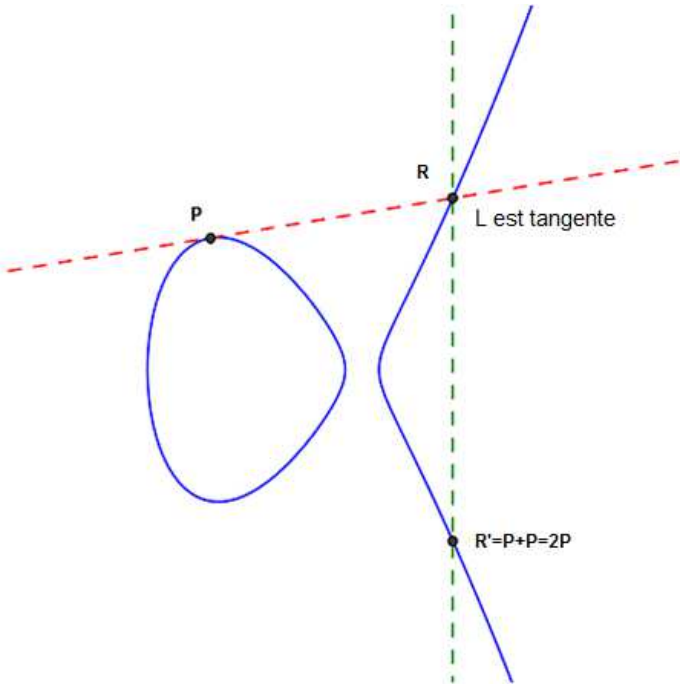

Figure 2.3 – L'addition de P à lui-même.

Ainsi, pour ajouter P à lui-même, prenons simplement la ligne L pour tangente à E en P, comme illustré dans la figure 2.3. Ensuite, L coupe E en P et en un autre point R. Dans un certain sens, L coupe toujours E en trois points, mais P compte deux fois d'entre eux.

Un deuxième problème potentiel se pose avec la «loi d'addition», est que si nous essayons d'additionner un point $P = (a, b)$ à son point symétrique $P' = (a, -b)$ sur l'axe des X [Kob12]. La ligne L qui traverse P et P' est la ligne verticale $x = a$, et cette ligne coupe E en deux points seulement P et P'.

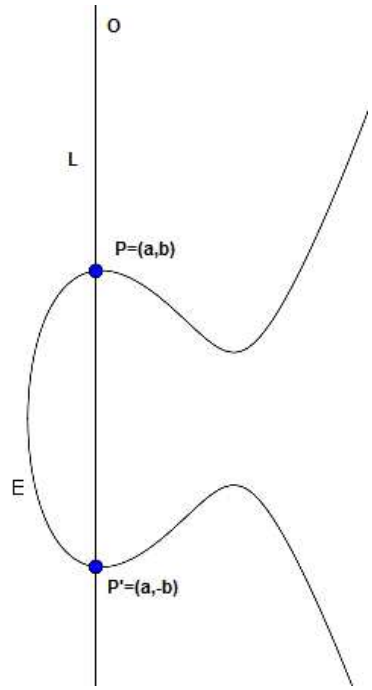

Figure 2.4 – La ligne L traversant P et P'

(Voir la figure 2.4) Il n'y a pas de troisième point d'intersection, il semble donc que nous soyons bloqués ! Mais il y a une sortie. La solution est de créer un extra-point O qui se situe «à l'infini». Plus précisément, le point O n'existe pas dans le plan XY, mais nous prétendons qu'il se trouve sur chaque ligne verticale. Nous posons ainsi :

$$P \oplus P' = O$$

Nous devons également trouver comment additionner O avec un point ordinaire $P = (a, b)$ sur E. La ligne L reliant P à O est la ligne verticale par P, puisque O se trouve sur les lignes verticales, et cette ligne verticale coupe E aux points P, O et $P' = (a, -b)$. Pour additionner P et O, nous allons faire la projection du point P' sur l'axe X, ce qui nous ramène à P. En d'autres termes, $P \oplus O = P$, donc O agit comme le zéro de l'addition pour les courbes elliptiques.

Définition 7.

Une courbe elliptique E est l'ensemble des solutions à une équation de Weierstrass :

$$E: \ Y^2 = X^3 + AX + B$$

avec un extra-point O, où les constantes A et B doivent satisfaire :

$$4A^3 + 27B^2 \neq 0$$

Remarque. Quelle est cette condition supplémentaire $4A^3 + 27B^2 \neq 0$? La quantité $\Delta_E = 4A^3 + 27B^2$ est appelée le discriminant de E [Men12].

La condition $\Delta_E \neq 0$ est équivalente à la condition que le polynôme cubique $X^3 + AX + B$ ne possède pas de racines répétées, c'est-à-dire que si nous factorisons $X^3 + AX + B$ complètement comme

$$X^3 + AX + B = (X - e_1)(X - e_2)(X - e_3)$$

Où e_1, e_2, et e_3 sont autorisés à être des nombres complexes, alors $4A^3 + 27B^2 \neq 0$ si et seulement si e_1, e_2, et e_3 sont distincts.

Les courbes avec $\Delta_E = 0$ ont des points singuliers [Sil05]. La loi d'addition ne fonctionne pas bien sur ces courbes. C'est pourquoi nous ajoutons une condition selon laquelle $\Delta_E \neq 0$ dans la définition de la courbe elliptique.

Théorème 1.

Soit E une courbe elliptique. Alors la loi d'addition sur E possède les propriétés suivantes :

1. Identité : $\forall P \in E, \quad P + O = O + P = P$
2. Inverse : $\forall P \in E, \quad P + (-P) = O$
3. Associativité : $\forall P, \ Q, \ R \in E, \quad (P + Q) + R = P + (Q + R)$
4. Commutativité : $\forall P, \ Q \in E, \quad P + Q = Q + P$

En d'autres termes, la loi d'addition sur les points de la courbe elliptique E forme un groupe abélien.

Démonstration. Comme nous l'avons expliqué plus haut, la loi d'identité (1) et la loi inverse (2) sont vraies parce que O réside sur toutes les lignes verticales. La loi

commutative (4) est facile à vérifier, puisque la ligne qui passe par P et Q est la même qui passe par Q et P, donc l'ordre des points n'a pas d'importance.

La partie restante du théorème est la loi associative (3). On pourrait ne pas penser que cela est difficile à prouver, mais si on dessine une figure et on met toutes les lignes nécessaires pour vérifier (3), on constate que c'est assez compliqué. Après avoir développé des formules explicites pour la loi d'addition sur E (Théorème 2), on peut utiliser ces formules pour vérifier la loi d'associativité par un calcul direct.

Notre prochaine tâche est de trouver des formules explicites pour nous permettre d'additionner et de soustraire des points sur une courbe elliptique. La dérivation de ces formules utilise la géométrie analytique élémentaire [Was03], un peu de calcul différentiel [Kob94] pour trouver une ligne tangente et une certaine quantité de manipulation algébrique.

Nous indiquons les résultats sous la forme d'un algorithme, puis nous indiquons brièvement la preuve [HPSS08]. □

Théorème 2 (Algorithme d'addition dans les courbes elliptiques)**.**

Soit

$$E : X^3 + AX + B$$

une courbe elliptique, et P_1, P_2 deux point sur E :

1. Si $P_1 = O$ alors $P_1 + P_2 = P_2$;
2. Sinon, si $P_2 = O$ alors $P_1 + P_2 = P_1$;
3. Autrement, écrire $P_1 = (x_1, y_1)$ et $P_2 = (x_2, y_2)$;
4. Si $x_1 = x_2$ et $y_1 = -y_2$ alors $P_1 + P_2 = O$;
5. Sinon, définir λ par :

$$\lambda = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1} & si \quad P_1 \neq P_2, \\ \\ \frac{3x_1^2 + A}{2y_1} & si \quad P_1 = P_2. \end{cases}$$

Et soit :

$$x_3 = \lambda^2 - x_1 - x_2 \qquad et \qquad y_3 = \lambda(x_1 - x_3) - y_1.$$

Alors :

$$P_1 + P_2 = (x_3, y_3)$$

Démonstration. Les cas (1) et (2) sont claires, (4) est le cas où la ligne traversant P_1 et P_2 est verticale, donc $P_1 + P_2 = O$. (Notez que si $y_1 = y_2 = 0$, la ligne tangente est verticale, ainsi, ce cas fonctionne aussi). Pour (5), on note que si $P_1 \neq P_2$, alors λ est la pente de la ligne traversant P_1 et P_2 , Et si $P_1 = P_2$, alors λ est la pente de la ligne tangente à $P_1 = P_2$. Dans les deux cas, la ligne L est donnée par l'équation $Y = \lambda X + v$ avec $v = y_1 - \lambda x_1$. La substitution de l'équation de L dans l'équation de E donne :

$$(\lambda X + v)^2 = X^3 + AX + B,$$

donc

$$X^3 - \lambda^2 X^2 + (A - 2\lambda v)X + (B - v^2) = 0.$$

Nous savons que cette équation cubique a x_1 et x_2 comme deux de ses racines. Si nous appelons la troisième racine x_3, elle sera factorisée comme suit :

$$X^3 - \lambda^2 X^2 + (A - 2\lambda v)X + (B - v^2) = (X - x_1)(X - x_2)(X - x_3)$$

Développons maintenant le côté droit et regardons le coefficient de X^2 de chaque côté. Le coefficient de X^2 sur le côté droit est $-x_1 - x_2 - x_3$, qui est égal à $-\lambda^2$ (le coefficient de X^2 sur le côté gauche). Cela nous permet de résoudre $x_3 = \lambda^2 - x_1 - x_2$, puis la coordonnée Y du troisième point d'intersection de E et L est donnée par $\lambda x_3 + v$.

Enfin, pour obtenir $P_1 + P_2$, nous devons faire projection sur l'axe des X, ce qui signifie que le remplacement de la coordonnée Y est négatif. □

2.5 Courbes elliptiques sur les corps finis

Dans la section précédente, nous avons développé la théorie des courbes elliptiques géométriquement. Par exemple, la somme de deux points distincts P et Q sur une courbe elliptique E est définie en dessinant la ligne L reliant P à Q, puis nous trouvons le troisième point où L et E se croisent, comme illustré à la Figure 2.2.

Cependant, pour appliquer la théorie des courbes elliptiques à la cryptographie, il faut considérer des courbes elliptiques dont les points ont des coordonnées dans un corps fini F_p. Cela est facile à faire. Nous définissons simplement une courbe elliptique sur F_p par une équation de la forme [Eng12] :

$$E : \ Y^2 = X^3 + AX + B$$

avec $A,\ B \in F_p$ satisfaisant $4A^3 + 27B^2 \neq 0$.

Aussi, nous considérons les points sur E avec des coordonnées en F_p, que nous désignons par :

$$E(F_p) = \{(X, Y) : X, Y \in F_p \ \ tel\ que : \ \ Y^2 = X^3 + AX + B\} \cup \{O\}.$$

Exemple :

Considérons la courbe elliptique E, défini sur le corps F_{13} [HPSS08] :

$$E : \ Y^2 = X^3 + 3X + 8$$

On peut trouver les points de $E(F_{13})$ en prenant toutes les valeurs possibles de $X = 0, 1, 2, \ldots, 12$ et vérifier pour quelle valeur X la quantité $X^3 + 3X + 8$ est un carré modulo 13.

Par exemple, prenons $X = 0$, ceci donne 8 et 8 n'est pas un carré modulo 13. Ensuite, essayons $X = 1$, ce qui donne $1 + 3 + 8 = 12$. Il s'avère que 12 est un carré modulo 13 ; En fait, il a deux racines carrées,

$$5^2 \equiv 12 (mod\ 13) \qquad and \qquad 8^2 \equiv 12 (mod\ 13).$$

Cela donne deux points $(1, 5)$ et $(1, 8)$ dans $E(F_{13})$. En continuant de cette façon, nous finissons avec une liste complète,

$$E(F_{13}) = \{O, (1, 5), (1, 8), (2, 3), (2, 10), (9, 6), (9, 7), (12, 2), (12, 11)\}.$$

Ainsi, $E(F_{13})$ se compose de neuf points.

Supposons maintenant que P et Q sont deux points dans $E(F_p)$ et que nous voulons "additionner" les points P et Q. Une possibilité est de développer une théorie de la géométrie en utilisant le corps F_p au lieu de $\mathbb{R}$. Ensuite, nous pourrions imiter nos constructions antérieures pour définir $P + Q$. Cela peut être fait, et cela conduit à un domaine fascinant des mathématiques appelé *géométrie algébrique* [Sil05].

Cependant, dans l'intérêt de la brièveté de l'exposition, nous utilisons plutôt les formules explicites données dans le théorème 2 pour additionner des points dans $E(F_p)$. Mais notons que si l'on veut acquérir une compréhension plus profonde de la théorie des courbes elliptiques, il est nécessaire d'utiliser une partie du formalisme de la géométrie algébrique.

Soit $P = (x_1, y_1)$ et $Q = (x_2, y_2)$ des points dans $E(F_p)$. Définissons la somme $P + Q$ à être le point (x_3, y_3) obtenu en appliquant l'algorithme d'addition dans les courbes elliptiques (Théorème 2). Notez que dans cet algorithme, les seules opéra-

tions utilisées sont l'addition, la soustraction, la multiplication et la division impliquant les coefficients de E et les coordonnées de P et Q. Puisque ces coefficients et ces coordonnées sont dans le corps F_p, nous nous retrouvons avec un point (x_3, y_3) dont les coordonnées sont en F_p. Bien sûr, il n'est pas tout à fait clair que (x_3, y_3) est un point dans $E(F_p)$.

Théorème 3.

Soient E une courbe elliptique sur F_p, et P, Q deux points dans $E(F_p)$.
- L'algorithme d'addition de courbe elliptique (Théorème 2) appliqué à P et Q donne un point dans $E(F_p)$. Nous désignons ce point par $P + Q$.
- Cette loi d'addition sur $E(F_p)$ satisfait toutes les propriétés répertoriées dans le Théorème 1. En d'autres termes, cette loi d'addition permet à $E(F_p)$ d'être un groupe fini [HPSS08].

Démonstration. Les formules (5) du théorème 2 sont dérivées en substituant l'équation d'une droite dans l'équation de E et en résolvant pour X, de sorte que le point résultant est automatiquement un point sur E, c'est-à-dire une solution de l'équation définissant E. Cela montre pourquoi (1) est vrai, bien que lorsque $P = Q$, un petit argument supplémentaire soit nécessaire pour indiquer pourquoi le polynôme cubique résultant a une double racine. Pour (2), la loi d'identité découle des étapes (1) et (2) de l'algorithme d'addition, la loi inverse est claire à partir de l'étape d'algorithme d'addition (4), et la loi commutative est simple, puisqu'il s'agit d'une brève vérification de l'algorithme d'addition qui montre que la permutation des deux points conduit au même résultat. Malheureusement, la loi associative n'est pas si claire. Il est possible de vérifier la loi d'associativité directement en utilisant les formules de l'algorithme d'addition, bien qu'il existe de nombreux cas particuliers à considérer. □

Exemple :

Voici un tableau qui résume toutes les additions possibles des points de la courbe elliptique dont l'équation est :

$$E : \ Y^2 = X^3 + 3X + 8 \qquad \textit{sur le corps } F_{13}$$

Il est clair que l'ensemble des points $E(F_p)$ est un ensemble fini, car il n'y a que de nombreuses possibilités finies pour les coordonnées X et Y. Plus précisément, il

+	$\mathcal{O}$	(1,5)	(1,8)	(2,3)	(2,10)	(9,6)	(9,7)	(12,2)	(12,11)
$\mathcal{O}$	$\mathcal{O}$	(1,5)	(1,8)	(2,3)	(2,10)	(9,6)	(9,7)	(12,2)	(12,11)
(1,5)	(1,5)	(2,10)	$\mathcal{O}$	(1,8)	(9,7)	(2,3)	(12,2)	(12,11)	(9,6)
(1,8)	(1,8)	$\mathcal{O}$	(2,3)	(9,6)	(1,5)	(12,11)	(2,10)	(9,7)	(12,2)
(2,3)	(2,3)	(1,8)	(9,6)	(12,11)	$\mathcal{O}$	(12,2)	(1,5)	(2,10)	(9,7)
(2,10)	(2,10)	(9,7)	(1,5)	$\mathcal{O}$	(12,2)	(1,8)	(12,11)	(9,6)	(2,3)
(9,6)	(9,6)	(2,3)	(12,11)	(12,2)	(1,8)	(9,7)	$\mathcal{O}$	(1,5)	(2,10)
(9,7)	(9,7)	(12,2)	(2,10)	(1,5)	(12,11)	$\mathcal{O}$	(9,6)	(2,3)	(1,8)
(12,2)	(12,2)	(12,11)	(9,7)	(2,10)	(9,6)	(1,5)	(2,3)	(1,8)	$\mathcal{O}$
(12,11)	(12,11)	(9,6)	(12,2)	(9,7)	(2,3)	(2,10)	(1,8)	$\mathcal{O}$	(1,5)

Table 2.1 – Table d'addition des points pour $E : y^2 = x^3 + 3x + 8$ sur $\mathbb{F}_{13}$

existe p possibilités pour X, puis pour chaque X, l'équation

$$Y^2 = X^3 + AX + B$$

montre qu'il existe au plus deux possibilités pour Y. En ajoutant l'extra point O, cela montre que le nombre de points de E noté $\#E(F_p)$ [Kob12] a au plus $2p + 1$ point. Toutefois, cette estimation est considérablement supérieure à la taille réelle.

Lorsque nous saisissons une valeur pour X, il existe trois possibilités pour la valeur de la quantité

$$X^3 + AX + B.$$

Premièrement, il peut s'agir d'un résidu quadratique modulo p, auquel cas il a deux racines carrées et nous obtenons deux points dans $E(F_p)$. Cela se produit environ 50% du temps.

Deuxièmement, il peut s'agir d'un non résidu modulo p, auquel cas nous éliminons X. Cela se produit également environ 50% du temps.

Troisièmement, il pourrait être égal à 0, auquel cas nous obtenons un point dans $E(F_p)$, mais ce cas se produit très rarement [1].

Nous pourrions donc nous attendre à ce que le nombre de points dans $E(F_p)$ soit approximativement

$$\#E(F_p) \approx (50\%.2p) + 1 = p + 1.$$

Un théorème célèbre de *Hasse* [Was03], plus tard généralisé par *Weil* et *Deligne*, dit que cela est vrai en fonction des fluctuations aléatoires.

1. La congruence $X^3 + AX + B \equiv 0 \ (mod\ p)$ comporte au plus trois solutions, et si p est grand, la chance de choisir au hasard l'une d'entre elles est très faible.

Théorème 4 (Hasse)**.**

Soit E une courbe elliptique sur F_p, alors

$$\#E(F_p) = p + 1 - t_p \qquad avec \quad |t_p| \leq 2\sqrt{p}.$$

Définition 8.

La quantité

$$t_p = p + 1 - \#E(F_p)$$

Apparaissant dans le Théorème 4 s'appelle la trace de Frobenius de E/F_p.

2.6 Problème du Logarithme Discret Elliptique

Afin de créer un cryptosystème basé sur le problème du logarithme discret (PLD) [BSSS99] dans le corps fini F_p^*, Alice publie deux nombres g et h, et son secret est l'exposant x qui résout la congruence

$$h \equiv g^x \ (mod\ p).$$

Voyons maintenant comment Alice peut faire quelque chose de similaire avec une courbe elliptique E défini sur F_p. Si Alice considère g et h comme éléments du groupe F_p^*, alors le problème du logarithme discret exige que Éve l'adversaire d'Alice, trouve un x tel que

$$h \equiv \underbrace{g \,.\, g \,.\, g \ \ldots \ g}_{x \text{ multiplications}} \ (mod\ p)$$

En d'autres termes, Éve doit déterminer combien de fois g doit être multiplié par lui-même pour arriver à h.

Avec cette formulation, il est clair qu'Alice peut faire la même chose avec le groupe des points $E(F_p)$ d'une courbe elliptique E sur un corps fini F_p. Elle choisit et publie deux points P et Q de $E(F_p)$, et son secret est un entier n qui vérifie :

$$Q = \underbrace{P \ + \ P \ + \ P \ \ldots \ + P}_{n \text{ additions sur } E} = nP$$

Alors Eve doit savoir combien de fois P doit être additionner à lui-même pour obtenir Q.

Gardez à l'esprit que, bien que la «loi d'addition» sur une courbe elliptique soit classiquement écrite avec un signe plus, l'addition dans E est en fait une opération très compliquée, de sorte que cet analogue elliptique du problème du logarithme discret peut être assez difficile à résoudre.

Définition 9.

Soit E une courbe elliptique sur le corps fini F_p et soient P et Q des points de $E(F_p)$.
Le problème du logarithme discret dans les courbe elliptique (PLDCE) est le problème de trouver un entier n tel que $Q = nP$. Par analogie avec le problème du logarithme discret dans F_p^*, on dénote cet entier n par

$$n = log_P(Q)$$

Et nous appelons n le logarithme discret elliptique de Q par rapport à P.

Remarque. Notre définition de $log_P(Q)$ n'est pas assez précise. La première difficulté est qu'il peut y avoir des points $P, Q \in E(F_p)$ tels que Q n'est pas un multiple de P. Dans ce cas, $log_P(Q)$ n'est pas défini. Cependant, à des fins cryptographiques, Alice commence avec un point public P et un entier privé n et elle calcule et publie la valeur de $Q = nP$. Donc, dans les applications pratiques, $log_P(Q)$ existe et sa valeur est le secret d'Alice.

La deuxième difficulté est que s'il y a une valeur de n satisfaisant $Q = nP$, il existe de nombreuses valeurs de ce type. Pour voir cela, on note tout d'abord qu'il existe un entier positif s tel que $sP = O$. (voir Proposition sur l'ordre d'un élément). Puisque $E(F_p)$ est fini, les points de la liste $P, 2P, 3P, 4P, \ldots$ Ne peuvent pas tous être distincts. Par conséquent, il existe des entiers $k > j$ tels que $kP = jP$, et nous pouvons prendre $s = k - j$. Le plus petit entier s tel que $s \geq 1$ est appelé l'ordre de P. (Le corollaire 1 nous dit que l'ordre de P divise $\#E(F_p)$.) Ainsi, si s est l'ordre de P et si n_0 est un nombre entier tel que $Q = n_0P$, alors les solutions à $Q = nP$ sont les entiers $n = n_0 + is$ avec $i \in \mathbb{Z}$. Cela signifie que la valeur de $log_P(Q)$ est vraiment un élément de $\mathbb{Z}/s\mathbb{Z}$, c'est-à-dire que $log_P(Q)$ est un entier modulo s, où s est l'ordre de P. Pour le concret, nous pourrions définir $log_P(Q)$ égal à n_0. Cependant, l'avantage de définir les valeurs dans $\mathbb{Z}/s\mathbb{Z}$ est que le logarithme discret elliptique satisfait ensuite :

$$log_P(Q_1 + Q_2) = log_P(Q_1) + log_P(Q_2) \quad pour\ tout \quad Q_1, Q_2 \in E(F_p). \tag{2.2}$$

Notez l'analogie avec le logarithme ordinaire $log(\alpha\beta) = log(\alpha) + log(\beta)$ et le logarithme discret dans F_p^*. Le fait que le logarithme discret pour $E(F_p)$ satisfasse l'équation 2.2 signifie qu'il vérifie la loi d'addition lorsque le groupe $E(F_p)$ est mappé (ou est associé) au groupe $\mathbb{Z}/s\mathbb{Z}$.

Cette association du log_P définit un homomorphisme de groupe.

Exemple :

Considérons la courbe elliptique

$$E: \ Y^2 = X^3 + 8X + 7 \ \ sur \ \ F_{73}.$$

Les points $P = (32, 53)$ et $Q = (39, 17)$ sont à la fois dans $E(F_{73})$, et il est facile de vérifier que :

$$Q = 11P, \ \ donc \ \ log_P(Q) = 11.$$

De même, $R = (35, 47) \in E(F_{73})$ et $S = (58, 4) \in E(F_{73})$, et après un certain calcul, on trouve qu'ils satisfont $R = 37P$ et $S = 28P$, donc

$$Log_P(R) = 37 \ \ et \ \ log_P(S) = 28.$$

Enfin, nous mentionnons que $\#E(F_{73}) = 82$, mais P satisfait $41P = O$. Ainsi P a l'ordre $41 = 82/2$, donc seulement la moitié des points dans $E(F_{73})$ sont des multiples de P. Par exemple, $(20, 65)$ est dans $E(F_{73})$, mais ce n'est pas un multiple de P.

2.7 L'algorithme Double-and-Add

Il semble être assez difficile de récupérer la valeur de n des deux points P et $Q = nP$ dans $E(F_p)$, c'est-à-dire qu'il est difficile de résoudre le PLDCE.

Nous en dirons plus sur la difficulté du PLDCE dans la section suivante. Toutefois, pour utiliser la fonction :

$$\mathbb{Z} \longrightarrow E(F_p), \qquad n \longmapsto nP,$$

pour la cryptographie, nous devons calculer efficacement nP à partir des valeurs connues de n et P. Si n est grand, nous ne voulons certainement pas calculer nP en calculant $P, 2P, 3P, 4P, \ldots$.

Le moyen le plus efficace de calculer nP est très similaire à la méthode pour

calculer la puissances $a^n(mod\ N)$. Cependant, puisque l'opération sur une courbe elliptique est décrite comme addition plutôt que multiplication, on l'appelle "double-and-add" au lieu de "square-and-multiply" [Men12].

L'idée sous-jacente est la même. Écrivons d'abord n sous forme binaire comme

$$n = n_0 + n_1.2 + n_2.4 + n_3.8 + \cdots + n_r.2^r \quad avec \quad n_0,\ n_1, \ldots,\ n_r \in \{0, 1\}.$$

(Supposons également que $n_r = 1$.) Ensuite, calculons les quantités suivantes :

$$Q_0 = P, \quad Q_1 = 2Q_0, \quad Q_2 = 2Q_1, \quad \ldots, \quad Q_r = 2Q_{r-1}.$$

Notez que Q_i est simplement le double du précédent Q_{i-1}, donc

$$Q_i = 2^i P$$

Ces points sont appelés multiples à 2 puissances de P, et le calcul nécessite des doublages. Enfin, calculons nP en utilisant au plus r opérations d'additions supplémentaires,

$$nP = n_0Q_0 + n_1Q_1 + n_2Q_2 + \ldots + n_rQ_r.$$

Considérons l'addition des points dans $E(F_p)$ comme une opération ponctuelle. Ainsi, le temps total pour calculer nP est au plus $2r$ opérations ponctuelles dans $E(F_p)$. Notez que $n \geq 2^r$, donc il ne faut pas plus de $2log_2(n)$ opérations ponctuelles pour calculer nP. Cela rend possible de calculer nP même pour de très grandes valeurs de n. Ci-après l'algorithme double-et-add :

Entrée. Point $P \in E(F_p)$ et un entier $n \geq 1$.

1. Faire $Q = P$ et $R = O$.
2. Tant que $n > 0$.
   - si $n \equiv 1\ (mod\ 2)$ alors $R = R + Q$
   - $Q = 2Q$
     $n = [n/2]$
3. Retourner le point $R = nP$.

Exemple :

Utilisons l'Algorithme Double-and-Add comme décrit précédemment pour calculer nP dans $E(F_p)$ pour :

$$n = 947, \qquad E:\ Y^2 = X^3 + 14X + 19, \qquad p = 3623, \qquad P = (6, 730).$$

L'expansion binaire de n est :

$$n = 947 = 1 + 2 + 2^4 + 2^5 + 2^7 + 2^8 + 2^9.$$

Le calcul étape par étape, qui nécessite neuf opérations de doublages et six opérations d'additions, est donné dans le tableau 2.2. Le résultat final est $947P = (3492, 60)$. (La colonne n du tableau se réfère au n utilisé dans l'algorithme décrit précédemment).

Étape i	n	$Q = 2^i P$	R
0	947	(6,730)	$\mathcal{O}$
1	473	(2521,3601)	(6,730)
2	236	(2277,502)	(2149,196)
3	118	(3375,535)	(2149,196)
4	59	(1610,1851)	(2149,196)
5	29	(1753,2436)	(2838,2175)
6	14	(2005,1764)	(600,2449)
7	7	(2425,1791)	(600,2449)
8	3	(3529,2158)	(3247,2849)
9	1	(2742,3254)	(932,1204)
10	0	(1814,3480)	(3492,60)

Table 2.2 – Calcul de $947.(6, 730)$ sur $Y^2 = X^3 + 14X + 19$ modulo 3623

2.8 La difficulté du PLDCE

Soit le groupe des points elliptiques $E(F_p)$. Pour résoudre $Q = nP$, Eve choisit des entiers aléatoires $j_1, \ldots, j_r$ et $k_1, \ldots, k_r$ entre 1 et p et fait deux listes de points :
List #1 :

$$j_1 P,\ j_2 P,\ j_3 P,\ \ldots,\ j_r P,$$

List #2 :

$$k_1 P + Q,\ k_2 P + Q,\ k_3 P + Q,\ \ldots,\ k_r P + Q.$$

Dès qu'elle trouve un match (collision) entre les deux listes, elle s'arrête, car si elle trouve $j_u P = k_v P + Q$, alors $Q = (j_u - k_v)P$ fournit la solution.

Si r est un peu plus grand que $\sqrt{p}$, disons $r \approx 3\sqrt{p}$, alors il y a de très bonnes chances qu'il y ait une collision.

Cet algorithme de collision naïf nécessite beaucoup de stockage pour les deux listes. Cependant, il n'est pas difficile d'adapter la méthode ρ de Pollard [HPSS08] pour concevoir un algorithme de collision sans stockage avec un temps d'exécution

similaire. En tout cas, il existe certainement des algorithmes qui résolvent le PLDCE pour $E(F_p)$ dans $O(\sqrt{p})$ étapes.

Nous savons qu'il existe des moyens beaucoup plus rapides pour résoudre le problème du logarithme discret dans F_p^*. En particulier, le calcul d'index [HPSS08] qui a un temps d'exécution sous-exponentiel, c'est-à-dire que le temps d'exécution est $O(p^\epsilon)$ pour tout $\epsilon > 0$. La principale raison pour laquelle les courbes elliptiques sont utilisées en cryptographie est le fait qu'il n'existe pas d'algorithmes de calcul d'indice connus pour le PLDCE, et en effet, il n'existe pas d'algorithmes généraux qui résolvent le PLDCE en moins de $O(\sqrt{p})$ étapes. En d'autres termes, malgré la nature hautement structurée du groupe $E(F_p)$, les algorithmes connus les plus rapides pour résoudre le PLDCE ne sont pas meilleurs que l'algorithme générique qui fonctionne aussi bien pour résoudre le problème du logarithme discret dans n'importe quel groupe. Ce fait est suffisamment important pour le souligner [Kob94].

L'algorithme connu le plus rapide pour résoudre le PLDCE dans $E(F_p)$ prend environ $\sqrt{p}$ étapes.

Ainsi, le PLDCE semble être beaucoup plus difficile que le PLD. Rappelons, cependant, qu'il existe certains premiers p pour lesquels le PLD dans F_p^* est relativement simple. Par exemple, si $p-1$ est un produit de petits nombres premiers, alors l'algorithme Pohlig-Hellman [HPSS08] donne une solution rapide au PLD dans F_p^*. D'une manière similaire, il existe des courbes elliptiques et des nombres premiers pour lesquels le PLDCE dans $E(F_p)$ est relativement simple.

2.9 La cryptographie par les courbes elliptiques

Il est enfin temps d'appliquer les courbes elliptiques à la cryptographie. Nous allons commencer par la plus simple application, l'échange de clés Diffie-Hellman [BSSS99, Kob12, Eng12], qui implique un peu plus le remplacement du problème de logarithme discret dans le corps fini F_p avec le problème du logarithme discret dans une courbe elliptique $E(F_p)$.

Nous décrivons ensuite une analogie elliptique du cryptosystème de clé publique ElGamal [BSSS99, Kob12, Was03].

2.9.1 Échange de clés elliptique de Diffie-Hellman

Alice et Bob se mettent d'accord pour utiliser une courbe elliptique particulière $E(F_p)$ et un point particulier $P \in E(F_p)$. Alice choisit un entier secret n_A et Bob choisit un entier secret n_B.

Génération des paramètres publics	
Quelqu'un de confiance choisit et publie un (grand) nombre premier $\boldsymbol{p}$*, une courbe elliptique* $\boldsymbol{E}$ *sur* F_p *et un point* $\boldsymbol{P}$ *dans* $E(F_p)$.	
Calcul privé	
Alice	**Bob**
Elle choisit un entier secret n_A *Elle calcule le point* $Q_A = n_A P$	*Il choisit un entier secret* n_B. *Il calcule le point* $Q_B = n_B P$.
L'échange public des valeurs	
Alice envoie Q_A *à Bob* $\longrightarrow$	Q_A
Q_B $\longleftarrow$	*Bob envoie* Q_B *à Alice*
D'autres calculs privés	
Alice	***Bob***
Elle calcule le point $n_A Q_B$.	*Il calcule le point* $n_B Q_A$.
La valeur du secret partagé est: $n_A Q_B = n_A(n_B P) = n_B(n_A P) = n_B Q_A$.	

FIGURE 2.5 – Échange de clé Diffie-Hellman en utilisant les courbes elliptiques

Ils calculent les multiples associés

$$\overbrace{Q_A = n_A P}^{\text{Alice calcule ceci}} \qquad et \qquad \overbrace{Q_B = n_B P}^{\text{Bob calcule ceci}}$$

Et ils échangent les valeurs de Q_A et Q_B. Alice utilise son multiplicateur secret pour calculer $n_A Q_B$, et Bob calcule de façon similaire $n_B Q_A$. Ils ont maintenant la valeur secrète partagée

$$n_A Q_B = (n_A n_B)P = n_B Q_A,$$

qu'ils peuvent utiliser comme clé pour communiquer en privé via un chiffrement symétrique. La figure 2.5 résume l'échange de clés Elliptique de Diffie-Hellman.

Exemple : Alice et Bob décident d'utiliser le protocole elliptique de Diffie-Hellman avec le nombre premier, la courbe et le point suivants :

$$p = 3851, \qquad E: \ Y^2 = X^3 + 324X + 1287, \qquad P = (920, 303) \in E(F_{3851}).$$

Alice et Bob choisissent les valeurs secrètes respectives $n_A = 1194$ et $n_B = 1759$, puis

$$Alice \ caclule \ \ Q_A = 1194P = (2067, 2178) \in E(F_{3851}),$$

$$Bob \ calcule \ \ Q_B = 1759P = (3684, 3125) \in E(F_{3851}).$$

Alice envoie QA à Bob et Bob envoie QB à Alice. Finalement,

$$\textit{Alice caclule} \quad n_A Q_B = 1194(3684, 3125) = (3347, 1242) \in E(F_{3851}),$$

$$\textit{Bob calcule} \quad n_B Q_A = 1759(2067, 2178) = (3347, 1242) \in E(F_{3851}).$$

Bob et Alice ont échangé le point secret $(3347, 1242)$. Comme cela sera expliqué dans la remarque qui va suivre, ils doivent ignorer la coordonnée y et traiter uniquement la valeur $x = 3347$ comme valeur du secret partagé.

Un moyen pour Eve de découvrir le secret d'Alice et Bob est de résoudre le PLDCE

$$n_A P = Q_A,$$

Car si Eve peut résoudre ce problème, elle connaît n_A et peut l'utiliser pour calculer $n_A Q_B$. Bien sûr, il pourrait y avoir une autre façon pour Eve de calculer leur secret sans réellement résoudre le PLDCE. Le problème précis que Eve doit résoudre est l'analogue elliptique du problème de Diffie-Hellman.

Définition 10.

Soit $E(F_p)$ une courbe elliptique sur un corps fini et soit $P \in E(F_p)$.
Le problème Elliptique de Diffie-Hellman (PEDH) est le problème du calcul de la valeur de $n_1 n_2 P$ à partir des valeurs connues de $n_1 P$ et $n_2 P$.

Remarque : L'échange de clés elliptique de Diffie-Hellman nécessite que Alice et Bob échangent des points sur une courbe elliptique. Un point Q dans $E(F_p)$ se compose de deux coordonnées $Q = (x_Q, y_Q)$, où x_Q et y_Q sont des éléments du corps fini F_p, il semble qu'Alice doit envoyer à Bob deux nombres de F_p. Cependant, ces deux nombres modulo p ne contiennent pas autant d'informations que deux nombres arbitraires, puisqu'ils sont liés par la formule

$$y_Q^2 = x_Q^3 + A x_Q + B \qquad \textit{dans} \quad F_p.$$

Notez que Eve connaît A et B, donc si elle peut deviner la valeur correcte de x_Q, alors il n'y a que deux valeurs possibles pour y_Q et, en pratique, il n'est pas trop difficile pour elle de calculer réellement les deux valeurs de y_Q.

Il y a donc peu de raisons pour lesquelles Alice envoie les deux coordonnées de Q_A à Bob, car la coordonnée y contient si peu d'informations supplémentaires. Au lieu de cela, elle envoie à Bob uniquement la coordonnée x de Q_A. Bob calcule et

utilise l'une des deux coordonnées possibles de y. Si il choisit le y "correct", alors il utilise Q_A, et s'il choisit le y "incorrect" (qui est le négatif du bon y), alors il utilise $-Q_A$. En tout cas, Bob finit par calculer l'un des

$$\pm n_B Q_A = \pm(n_A n_B)P.$$

De même, Alice finit par calculer l'un des $\pm(n_A n_B)P$. Alors Alice et Bob utilisent la coordonnée x comme leur valeur secrète partagée, puisque cette coordonnée x est la même quelle que soit le y qu'ils utilisent.

Exemple : Alice et Bob décident d'échanger une autre valeur secrète en utilisant les mêmes paramètres publics que dans l'exemple précédent :

$$p = 3851, \qquad E: \ Y^2 = X^3 + 324X + 1287, \qquad P = (920, 303) \in E(F_{3851}).$$

Cependant, cette fois-ci, ils veulent envoyer moins de bits les uns aux autres. Alice et Bob choisissent respectivement de nouvelles valeurs secrètes $n_A = 2489$ et $n_B = 2286$, et comme précédemment,

$$\textit{Alice calcule} \ \ Q_A = n_A P = 2489(920, 303) = (593, 719) \in E(F_{3851}),$$

$$\textit{Bob computes} \ \ Q_B = n_B P = 2286(920, 303) = (3681, 612) \in E(F_{3851}.$$

Cependant, plutôt que d'envoyer les deux coordonnées, Alice envoie seulement $x_A = 593$ à Bob et Bob envoie seulement $x_B = 3681$ à Alice.

Alice remplace $x_B = 3681$ dans l'équation de E et trouve que

$$y_B^2 = x_B^3 + 324x_B + 1287 = 3681^3 + 324 \times 3681 + 1287 = 997.$$

(Rappelez-vous que tous les calculs sont effectués dans F_{3851}.) Alice doit calculer la racine carrée de 997 modulo 3851. Donc,

$$y_B \equiv 612 (mod\ 3851).$$

Il apparaît qu'elle obtienne le même point $Q_B = (x_B, y_B) = (3681, 612)$ que Bob a utilisé, et elle calcule $n_A Q_B = 2489(3681, 612) = (509, 1108)$.

De même, Bob remplace $x_A = 593$ dans l'équation de E et prend une racine carrée,

$$y_A^2 = x_A^3 + 324x_A + 1287 = 593^3 + 324 \times 593 + 1287 = 927,$$

$$y_A \equiv 3132(mod\ 3851).$$

Bob utilise alors le point $\acute{Q}_A = (593, 3132)$, qui n'est pas le point Q_A d'Alice, pour calculer $n_B\acute{Q}_A = 2286(593, 3132) = (509, 2743)$. Bob et Alice se retrouvent avec des points négatifs l'un de l'autre dans $E(F_p)$, mais tout va bien, puisque leur valeur secrète partagée est la coordonnée $x = 593$, qui est la même pour les deux points.

2.9.2 Cryptosystème elliptique d'ElGamal

Il est facile de créer un analogue direct du cryptosystème ElGamal. Brièvement,

1. Alice et Bob se mettent d'accord à utiliser un nombre premier p particulier, une courbe elliptique E et un point $P \in E(F_p)$.
2. Alice choisit un multiplicateur secret n_A et publie le point $Q_A = n_AP$ comme sa clé publique.
3. Le texte en clair de Bob est un point $M \in E(F_p)$. Il choisit un entier k pour être sa clé éphémère et calcule

$$C_1 = kP \qquad et \qquad C_2 = M + kQ_A.$$

4. Il envoie les deux points (C_1, C_2) à Alice, qui calcule

$$C_2 - n_AC_1 = (M + kQ_A) - n_A(kP) = M + k(n_AP) - n_A(kP) = M$$

Pour récupérer le texte en clair.

Remarque Rappelons que dans le cryptosystème ElGamal, le texte en clair est un entier m entre 2 et $p-1$, tandis que le texte chiffré se compose de deux entiers c_1 et c_2 dans la même plage. Ainsi, en général, il faut deux fois plus de bits pour écrire le texte chiffré qu'il le faut pour écrire le texte en clair. Nous disons que ElGamal a une expansion de message de 2 à 1.

En principe, le cryptosystème elliptique ElGamal fonctionne bien, mais il existe des difficultés pratiques.

1. Il n'y a aucun moyen évident d'attacher des messages texte en points dans $E(F_p)$.
2. Le cryptosystème elliptique ElGamal a une expansion de message de 4 à 1, par rapport au rapport d'expansion de 2 à 1 d'ElGamal dans F_p.

La raison pour laquelle le cryptosystème elliptique d'ElGamal a une expansion de message de 4 à 1 réside dans le fait que le texte en clair M est un point unique dans

$E(F_p)$. Par le théorème de Hasse, il existe approximativement p points différents dans $E(F_p)$, donc seulement sur p différents textes en clairs. Cependant, le texte chiffré (C_1, C_2) se compose de quatre nombres modulo p, puisque chaque point dans $E(F_p)$ a deux coordonnées.

Diverses méthodes ont été proposées pour résoudre ces problèmes. La difficulté d'associer les textes en clairs aux points peut être contournée en choisissant M de façon aléatoire et en l'utilisant comme un masque pour le texte en clair réel.

Une autre façon d'améliorer l'expansion des messages est d'envoyer uniquement les coordonnées x de C_1 et C_2. Malheureusement, puisque Alice doit calculer la différence $C_2 - n_A C_1$, elle a besoin les valeurs correctes des coordonnées x et y de C_1 et C_2. (Notez que les points $C_2 - n_A C_1$ et $C_2 + n_A C_1$ sont tout à fait différents !) Cependant, la coordonnée x d'un point détermine la coordonnée y en modifiant le signe, donc Bob peut envoyer un bit supplémentaire, par exemple

$$Extra \;\; bit = \begin{cases} 0 & \text{si } 0 \leq y < \frac{1}{2}p, \\ 1 & \text{si } \frac{1}{2}p < y < p. \end{cases}$$

De cette façon, Bob doit envoyer uniquement les coordonnées x de C_1 et C_2, plus deux bits supplémentaires. Cette idée est parfois appelée *compression de points* [HPSS08].

2.10 L'évolution de la cryptographie à clé publique

L'invention de RSA à la fin des années 1970 a catapulté le problème de factorisation des grands nombres entiers, ce qui a conduit à des méthodes de factorisation améliorées telles que les cribles quadratiques et numériques. En 1984, Hendrik Lenstra Jr. a publié un manuscrit décrivant une nouvelle méthode de factorisation à l'aide des courbes elliptiques. L'algorithme de Lenstra [HPSS08] est un analogue elliptique de l'algorithme de factorisation $p-1$ de Pollard et exploite le fait que le nombre de points dans $E(F_p)$ varie selon qu'on choisit les différentes courbes elliptiques. Bien que moins efficace que les méthodes des cribles pour les problèmes de factorisation qui se produisent dans la cryptographie, l'algorithme de Lenstra a aidé à introduire les courbes elliptiques dans la communauté cryptographique.

L'importance des algorithmes de factorisation pour la cryptographie est qu'ils sont utilisés pour briser RSA et d'autres cryptosystèmes similaires. En 1985, *Neal Koblitz* et *Victor Miller* ont proposé de manière indépendante l'utilisation des courbes elliptiques pour créer des cryptosystèmes. Ils ont suggéré que le problème du logarithme discret de la courbe elliptique pourrait être plus difficile que le problème de

logarithme discret classique modulo p. Ainsi, l'échange de clés de Diffie-Hellman et le cryptosystème de clé publique ElGamal, implémenté à l'aide des courbes elliptiques, peuvent nécessiter des clés plus petites et fonctionner plus efficacement que RSA car on pourrait utiliser des nombres plus petits.

Koblitz et Miller [Men12] ont chacun publié leurs idées en tant que documents académiques, mais aucun d'eux n'a poursuivi les aspects commerciaux de la cryptographie des courbes elliptiques. En effet, à l'époque, il n'y avait pratiquement aucune recherche sur le PLDCE, il était donc difficile de dire avec certitude que le PLDCE était en effet beaucoup plus difficile que le PLD classique. Cependant, *Scott Vanstone* et *Ron Mullin*, qui avaient lancé une société cryptographique appelée *Certicom* en 1985, ont noté le potentiel de ce qu'on appelle la cryptographie par les courbes elliptiques (CCE). Ils ont rejoint d'autres chercheurs du milieu universitaire et le monde des affaires pour promouvoir la CCE comme une alternative à RSA et à ElGamal.

Tout n'était pas une navigation en douceur. Par exemple, à la fin des années 1980, plusieurs cryptographes ont proposé d'utiliser des courbes elliptiques dites super-singuliaires pour une efficacité accrue, mais en 1990, l'algorithme MOV [Kob12] a montré que les courbes super-singuliaires sont vulnérables aux attaques. Certains ont vu cela comme un acte d'accusation de CCE dans son ensemble, tandis que d'autres ont souligné que RSA a également des cas faibles qui doivent être évités, par exemple, RSA doit éviter d'utiliser des nombres qui peuvent être facilement factorisés par la méthode $p-1$ de Pollard.

La question pure mathématique de savoir si la CCE fournissait une alternative sûre et efficace à RSA était obscurcie par le fait qu'il y avait des enjeux commerciaux et financiers. Afin d'avoir un succès commercial, les méthodes cryptographiques doivent être normalisées pour être utilisées dans des domaines tels que les communications et la banque. RSA a été le leader initial, puisqu'il a été inventé en premier, mais RSA a été breveté et certaines entreprises ont résisté à l'idée que les normes approuvées par les groupes commerciaux ou les organismes gouvernementaux devraient exiger l'utilisation d'une technologie brevetée. ElGamal, après avoir été inventé en 1985, a fourni une alternative sans redevance, tant de normes ont spécifié ElGamal comme alternative à RSA. Dans l'intervalle, la CCE était de plus en plus grande, mais même en 1997, plus d'une décennie après son introduction, les principaux experts ont exprimé leurs doutes quant à la sécurité de la CCE [2].

2. En 1997, la société RSA a publié la citation suivante du co-inventeur *Ron Rivest* sur son site web : "Mais la sécurité des cryptosystèmes basés sur les courbes elliptiques n'est pas bien comprise, en raison en grande partie de la nature abstruse des courbes elliptiques...

Au fil du temps, cela peut changer, mais pour l'instant, essayer d'obtenir une évaluation de la

Un dilemme majeur qui imprègne le domaine de la cryptographie est que personne ne connaît la difficulté réelle des problèmes supposés être difficiles sur lesquels elle se fonde.

À l'heure actuelle, la sécurité des cryptosystèmes à clé publique dépend de la perception et du consensus des experts sur la difficulté de problèmes tels que la factorisation entière et les logarithmes discrets.

Tout ce que l'on peut dire est que "*Un tel problème a été largement étudié pour N années, et voici la méthode la plus rapide connue pour le résoudre.*" Les promoteurs de cryptosystèmes fondés sur la factorisation soulignent le fait que, dans un certain sens, les gens ont essayé de factoriser les nombres depuis l'antiquité ; Mais en vérité, la théorie moderne de la factorisation nécessite des dispositifs informatiques de grande vitesse et c'est à peine que prédomine l'invention de RSA. L'étude sérieuse du problème du logarithme discret dans les courbes elliptiques a commencé à la fin des années 1980, de sorte que les méthodes de factorisation modernes ont une avance de 10 à 15 ans sur le PLDCE.

RSA, le premier cryptosystème à clé publique, a été breveté par ses inventeurs. La question des brevets en cryptographie est très controversée. On pourrait soutenir que le brevet RSA, qui a duré de 1983 à 2000, a permis de reculer l'utilisation de la cryptographie en obligeant les utilisateurs à payer les droits de licence. Cependant, il est également vrai que, pour construire une entreprise, un inventeur a besoin d'investisseurs disposés à risquer leur argent, et il est beaucoup plus facile de collecter des fonds s'il existe un produit exclusif à offrir. En outre, le fait que RSA était à l'origine «*le seul produit au marché*» signifiait qu'il recevait automatiquement un examen minutieux de la communauté universitaire, ce qui a permis de valider sa sécurité.

L'invention et la mise en œuvre commerciale éventuelle de la CCE ont suivi un chemin différent. Puisque ni Koblitz ni Miller n'ont demandé un brevet, l'idée sous-jacente fondamentale de CCE est librement disponible pour tous. Cela a conduit Certicom et d'autres entreprises à postuler à des brevets améliorant l'idée de base de la CCE. Certaines de ces améliorations ont été basées sur de nouvelles idées de recherche importantes, tandis que d'autres ont été moins innovantes et pourraient presque être qualifiées de problèmes de devoirs de routine [3]. Malheureusement, l'Of-

sécurité d'un cryptosystème à courbe elliptique est un peu comme essayer d'obtenir une évaluation de la poésie chaldéenne récemment découverte. Jusqu'à ce que les courbes elliptiques aient été approfondies et évaluées, je conseillerais de ne mettre en place aucune application à grande échelle en fonction de celles-ci."

3. Par exemple, à la fin de la section 2.9.2, nous avons décrit comment sauvegarder la bande passante dans le cryptosystème elliptique d'ElGamal en envoyant la coordonnée x et un bit supplémentaire pour spécifier la coordonnée y. Cette idée s'appelle "*compression de points*" et est

fice américain des brevets et des marques (the United States Patents and Trademark Office USPTO) n'a pas l'expertise nécessaire pour évaluer efficacement l'inondation des demandes de brevet cryptographique qu'il reçoit. Le résultat a été une grande incertitude sur le marché sur quelles versions des CCE étaient gratuites et lesquelles nécessitaient des licences, même si tous les brevets délivrés peuvent résister à une contestation judiciaire [HPSS08].

2.11 Conclucion

Dans ce chapitre, nous avons présenté les concepts mathématiques liés aux courbe elliptiques, et l'application de ces dernières dans la cryptographie, notamment dans l'échange de clé entre individus, aussi nous avons défini le fameux problème du logarithme discret sur les courbes elliptiques et sa complexité qui a contribué à l'évolution de la cryptographie par la naissance de la cryptographie par les courbes elliptiques (CCE).

Dans le chapitre suivant, nous allons continuer à définir un autre objet mathématique appartenant à la catégorie des systèmes dynamiques, à savoir les automates cellulaires.

couverte par le brevet $US6,141,420$.

Chapitre 3

Les automates cellulaires

Sommaire

3.1 Introduction

Rencontrés fréquemment dans de nombreux domaines de recherche (physique, biologie, sociologie, économie, informatique etc.), les grands réseaux d'entités en interaction que l'on peut regrouper sous l'appellation "systèmes complexes" constituent aujourd'hui un enjeu scientifique majeur.

Ils posent en effet un certain nombre de questions fondamentales qui dépassent largement les spécificités des domaines où ils sont rencontrés et demandent un effort de recherche propre. La complexité de tels systèmes tient à ce que la connaissance des entités qui les composent ne suffit en général pas à comprendre le comportement du système dans son ensemble.

Les *automates cellulaires* sont un modèle particulier de systèmes dynamiques discrets. Ce sont des réseaux réguliers de cellules toutes identiques, possédant chacune et à chaque instant un état parmi un ensemble fini, et qui évoluent par application synchrone et uniforme d'une règle de mise à jour définie localement. Ils sont utilisés pour la modélisation de phénomènes naturels variés (jusqu'aux "lois de l'univers"), mais ils constituent avant tout un modèle en soi, remarquable par sa simplicité formelle et néanmoins capable de produire des comportements d'une grande richesse et souvent difficiles à prévoir. À ce titre, ils se présentent comme un cadre idéal pour étudier les problématiques transversales issues des "systèmes complexes".

Au delà des constats empiriques de complexité, aujourd'hui abondants, la compréhension du modèle des automates cellulaires nécessite le développement d'outils théoriques adaptés, qu'il s'agisse de confirmer ou d'infirmer l'intuition qui naît des observations (nécessairement partielles), ou encore de délimiter ce qui est hors d'atteinte du calcul et donc de toute prédiction exacte, ou enfin, et surtout, de saisir de manière unifiée et explicite, par des objets mathématiques, ce qui apparaît de manière diffuse à travers une somme d'expériences hétérogènes.

Notre travail s'inscrit avant tout dans cette démarche théorique, où la recherche de définitions formelles pertinentes au vu des phénomènes observés et l'établissement de liens entre celles-ci par la preuve sont deux activités étroitement liées.

3.2 Historique

On fait généralement remonter l'histoire des automates cellulaires aux années quarante et à *Stanislas Ulam*. Ce mathématicien s'est intéressé à l'évolution de constructions graphiques engendrées à partir de règles simples. La base en était un espace à deux dimensions divisé en "cellule", soit une sorte de feuille quadrillée.

Chacune des cellules pouvait avoir deux états : allumé ou éteint. Partant d'une

configuration donnée, la génération suivante était déterminée en fonction de règles de voisinage. Par exemple, si une cellule donnée était en contact avec deux cellules allumées elle s'allumait sinon elle s'éteignait. Ulam, qui utilisait l'un des premiers ordinateurs, a rapidement constaté que ce mécanisme permettait de générer des figures complexes et esthétiques et que dans certains cas, ces figures pouvaient se répliquer. Des règles extrêmement simples permettaient de construire des structures très complexes. À partir de là, se posait la question suivante : ces mécanismes récursifs - c'est-à-dire en l'occurrence dépendant de leur propre état antérieur peuvent-ils expliquer la complexité du réel ? Cette complexité n'est elle qu'apparente, les lois fondamentales étant elles-mêmes simples [Heu94] ?

En parallèle, John Von Neumann - fort des travaux de Alan Turing s'intéressait à la théorie des automates autoréplicateurs et travaillait à la conception d'une machine autoréplicatrice le *kinématon* Une telle machine devait être capable, à partir de matériaux trouvés dans l'environnement, de produire n'importe quelle machine décrite dans son programme, y compris une copie d'elle-même. Von Neumann montrait ici comment résoudre le problème de l'auto-référence de la description. Pour s'autorépliquer, la machine devrait en effet contenir une description d'elle-même, mais pour être complète, cette description doit également être décrite, etc. La solution réside dans la capacité donnée à la machine d'interpréter sa description à la fois comme un programme, une séquence d'instruction, et comme un composant. La description sera d'abord interprétée pour construire la nouvelle machine, elle sera ensuite simplement copiée afin de donner à la nouvelle machine une description d'elle-même. Ce mécanisme correspond de fait à l'interprétation actuelle du fonctionnement de la molécule d'ADN découverte après les travaux de Von Neumann.

A.C. Clarke a rendu les machines de Von Neumann célèbre avec la série "2001 Odyssée de l'espace". Pour transformer Jupiter en étoile, un premier monolithe se reproduit, les descendants font de même, la population croît ainsi de manière exponentielle pour atteindre rapidement la taille nécessaire à la réalisation d'une aussi gigantesque tâche.

C'est S. Ulam qui a suggéré à Von Neumann d'utiliser ce qu'il appelait les "espaces cellulaires" (cellular spaces) pour construire sa machine autoréplicatrice.

Il pouvait ainsi s'affranchir des conditions physiques réelles pour travailler dans un univers extrêmement simplifié pourtant apte à engendrer une haute complexité.

Le passage à cet univers formel l'a amené à constater : "En axiomatisant les automates autoréplicateurs de cette manière, on a jeté la moitié du problème par la fenêtre et c'est peut-être la moitié la plus importante. On s'est résigné à ne pas expliquer comment ces éléments sont constitués de choses réelles, particulièrement

comment ces éléments sont constitués de particules élémentaires ou même de molécules, on considérera simplement que des particules élémentaires dotées de certaines propriétés existent. La question à laquelle on espère répondre, ou au moins examiner, est : Quels principes sont mis en œuvre dans l'organisation de ces molécules dans les êtres vivants fonctionnels ? On discuterai de tout cela seulement de ce point de vue limité [VNB96]."

Sur cette base, il conçut un automate cellulaire de quelques 200.000 cellules à 29 états contenant un copieur universel, une description de lui-même et une machine de Turing pour la supervision.

Les automates cellulaires sont sortis des laboratoires en 1970 avec le désormais fameux Jeu de la vie (Life Game) de John Horton Conway.

3.3 Définitions de l'automate cellulaire

3.3.1 Formellement

Un automate cellulaire est un quadruplet (L, S, N, R) où :

- L est l'espace cellulaire (appelé aussi grille, réseau, etc...). C'est en pratique un ensemble de coordonnées labellisant la cellule c. En $2D$, $c = (i, j)$ où i et j sont des entiers.
- S est l'ensemble d'état discret, de cardinal k.
- $N = v_1, \ldots, v_n$ est le voisinage : un sous-ensemble de L qui peut être donné par :

$$N : \quad L \longrightarrow 2^L$$

$$c \longmapsto N(c) = c + v_1, \ldots, c + v_n$$

  où $c+v_i$ est le translaté de la cellule c suivant la direction v_i et 2^L est l'ensemble des parties de L. n est le cardinal de $N(c)$ et définit la taille du voisinage.
- R est la règle de transition de l'automate cellulaire. C'est une fonction qui calcule le nouvel état $s_{t+1}(c)$ comme fonction des états des cellules du voisinage à l'itération t :

$$s_{t+1}(c) = R(s_t(N(c))).$$

3.3.2 Intuitivement

Un automate cellulaire consiste en une grille régulière de "cellules" contenant chacune un "état" choisi parmi un ensemble fini et qui peut évoluer au cours du temps. L'état d'une cellule au temps $t + 1$ est fonction de l'état au temps t d'un nombre

fini de cellules appelé son "voisinage". À chaque nouvelle unité de temps, les mêmes règles sont appliquées simultanément à toutes les cellules de la grille, produisant une nouvelle "génération" de cellules dépendant entièrement de la génération précédente.

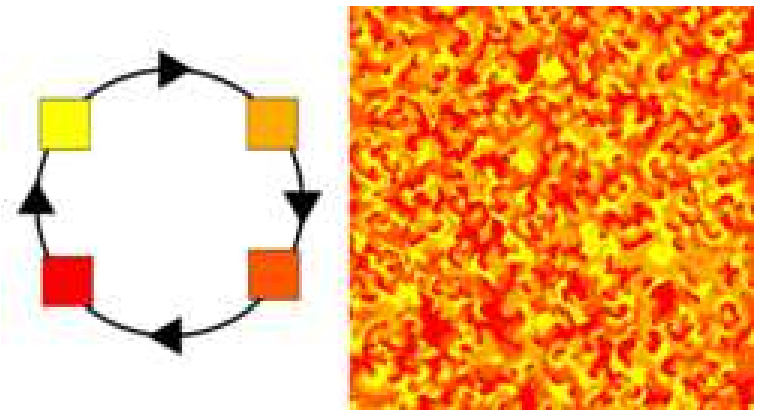

Figure 3.1 – Exemple d'un automate cellulaire.

Étudiés en mathématiques et en informatique théorique, les automates cellulaires sont à la fois un modèle de système dynamique discret et un modèle de calcul.

Le modèle des automates cellulaires est remarquable par l'écart entre la simplicité de sa définition et la complexité que peuvent atteindre certains comportements macroscopiques : l'évolution dans le temps de l'ensemble des cellules ne se réduit pas (simplement) à la règle locale qui définit le système. À ce titre il constitue un des modèles standards dans l'étude des systèmes complexes.

À gauche dans la figure 3.1, une règle locale simple : une cellule passe d'un état (i) au suivant $(i+1)$ dans le cycle d'états dès que $i+1$ est présent dans au moins 3 cellules voisines.

À droite, le résultat (complexe) de l'application répétée de cette règle sur une grille de cellules. Ce type d'automates cellulaires a été découvert par *D. Griffeath.*

3.4 Les concepts clé des automates cellulaires

Voici quelques concepts clé que se partagent la plupart des automates cellulaires, et qui permettent de cerner leur mode de fonctionnement. Notons toutefois qu'on ne peut réduire exclusivement la définition d'automate cellulaire à ces quelques concepts. Si certains d'entre eux comme le parallélisme ou le voisinage semblent être liés intrinsèquement à leur définition, d'autres comme déterminisme ou homogénéité peuvent laisser planer un doute.

3.4.1 Voisinage

Le nouvel état de chaque cellule est déterminé à partir de sa position spatiale dans l'univers de l'automate, en examinant les états des cellules voisines. Les transitions d'un état à l'autre de l'automate se font localement pour chaque cellule.

3.4.2 Parallélisme

Toutes les cellules constituant l'univers de l'automate sont mises à jour de manière simultanée et synchrone (recours à un buffer en pratique pour simuler le pseudo-parallélisme / architectures parallèles).

3.4.3 Déterminisme

Pour une cellule, la donnée des états des cellules voisines détermine à elle seule le nouvel état. Certains automates cellulaires dits stochastiques introduisent un facteur probabilité dans la transition : une même configuration de voisinage pourra conduire à différentes nouvelles configurations.

3.4.4 Homogénéité (AC uniforme)

On dit qu'un automate cellulaire est homogène s'il satisfait les conditions suivantes :

- La topologie du réseau cellulaire est régulière.
- La fonction qui calcule le voisinage doit être uniforme pour toutes les cellules.
- L'évolution de l'automate se définit par une seule règle de transition qui s'applique à toutes les cellules.

3.4.5 Hétérogénéité (AC non-uniforme)

Les cellules ne sont pas homogènes puisqu'elles sont soumises à des règles de transition différentes de l'espace de règle de l'automate cellulaire.

3.4.6 Discrétisation

Un automate cellulaire se déroule dans le temps de manière discrète, génération après génération, en opposition avec la plupart des phénomènes physiques continus.

3.5 Les caractéristiques des automates cellulaires

Un automate cellulaire (AC) peut-être décrit par les quatre composantes suivantes.

3.5.1 La dimension

Le plus généralement 1 (AC élémentaire) ou 2 (AC type Life), il n'y a pas de limite à la dimension d'un automate, si ce n'est la puissance de calcul des machines sensées le reproduire.

Figure 3.2 – Les dimensions d'un automate cellulaire.

Les exemples pratiques d'automates en $3D$ sont rares. Passé les 3 dimensions, on doit recourir à des projections sur $R3$ ou $R2$ pour visualiser correctement l'hypermatrice.

3.5.2 Le voisinage d'une cellule

Celui-ci définit l'ensemble des cellules qui auront une influence sur la cellule étudiée. En pratique, le voisinage est souvent limité à la cellule cible et aux cellules adjacentes.

La géométrie des cellules est également étroitement liée à son voisinage. Une cellule peut-être un hyper-cube (cas le plus fréquent), mais également un autre hypervolume ou désorienté (blind neighbourhood) : un voisinage "aveugle" est un voisinage pour lequel chaque cellule appartenant à ce voisinage joue un rôle identique.

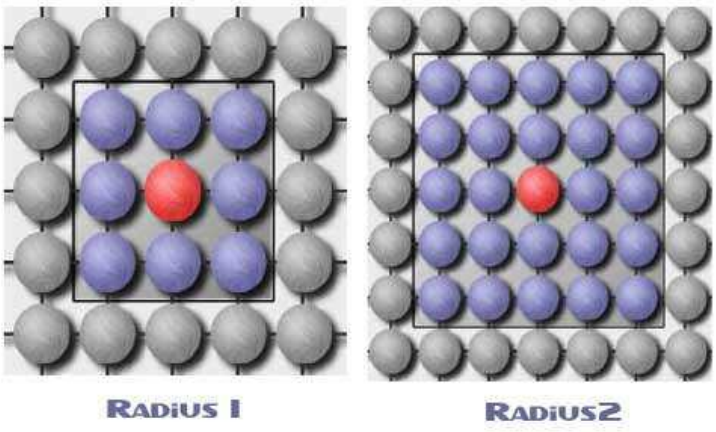

Figure 3.3 – Le voisinage d'une cellule.

La cellule cible dont on étudie le voisinage a "connaissance" des différents états de ces voisines, mais ne sait pas à quelle cellules correspond quel état.

La plupart des automates cellulaires définissent sans le préciser des voisinages "aveugles".

3.5.3 L'espace d'états

Cet espace correspond à l'ensemble des états que peut prendre une cellule. Le plus souvent limité à 2, il n'y a aucune limite théorique. Pour exemple, Von Neumann a étudié mathématiquement un automate à 29 états. Pratiquement, ces états sont représentés par des couleurs, qui permettent de suivre les évolutions de l'automate.

Lors de la modélisation des systèmes, les états des cellules correspondent à des états physiques locaux. Par exemple, dans le Jeu de la Vie, une cellule est soit "morte" soit "vivante", mais on pourrait très bien imaginer des états transitoires de dégénérescence d'une cellule, en augmentant le nombre d'états de l'automate.

3.5.4 La fonction de transition

C'est l'ensemble des règles qui permettent de déterminer le nouvel état d'une cellule en fonction de son état précédent et de l'état précédent de son voisinage. Pour un automate à n états et avec un voisinage de k cellules, il peut y avoir n^k configurations de voisinage différentes :

- pour $n = 2$ et $k = 3$, $n^k = 8$ voisinages différents (AC élémentaire de wolfram).
- pour $n = 2$ et $k = 9$, $n^k = 512$ voisinages différents (jeu de la vie).

De plus, la fonction de transition est crée en associant à chaque voisinage un état de sortie. Il y a donc potentiellement n^{n^k} fonctions de transition pour un automate

cellulaire à n états ayant un voisinage de k cellules.

La plupart du temps, ces règles ne sont pas explicitées, mais résumées. Elles sont synthétisées sous la forme de méta-règles (règles du jeu de la vie).

Les règles possibles pour définir un automate cellulaires sont très nombreuses, même avec un petit nombre d'états et un petit voisinage : Le modèle des automates

	2 états	**3 états**	**4 états**	**5 états**
2 voisins	8	19683	4 294 967 296	$> 10^{17}$
3 voisins	256	7 625 597 484 987	$> 10^{38}$	$> 10^{87}$
4 voisins	65536	$> 10^{38}$	$> 10^{154}$	$> 10^{436}$
5 voisins	4 294 967 296	$> 10^{115}$	$> 10^{616}$	$> 10^{2184}$
6 voisins	$> 10^{19}$	$> 10^{347}$	$> 10^{2466}$	$> 10^{10921}$

Table 3.1 – Espace des règles possibles d'automate cellulaire

cellulaire offre donc un terrain d'exploration immense. Il n'est pas difficile de programmer un simulateur d'automates cellulaires et la Toile regorge de réalisations plus ou moins abouties.

3.6 Automates cellulaires unidimensionnels élémentaires

Le réseau peut être représenté sous la forme d'une droite, ou d'un cercle (les conditions "aux bords" deviennent alors périodiques). La structure d'interaction la plus simple consiste alors à considérer les deux plus proches voisins. La structure d'interaction comprend alors deux ou trois entrées selon que l'on considère que l'automate interagit avec lui même.

Dans le cas où $k = 2$, on a : $2^2 = 4$ configurations d'entrée possibles, et donc $2^4 = 16$ règles différentes possibles pour les changements d'état.

On peut coder ces règles de 0 à 15 en classant les sorties selon leur valeur binaire. Dans le cas où $k = 2$ il est intéressant de classer les configurations d'entrées selon leur valeur binaire, en plaçant les valeurs les plus faibles à gauche. On remarque alors que les règles sont symétriques.

Les 8 dernières règles sont les complémentaires des huit premières et peuvent être obtenues en inversant les valeurs de sorties.

Pour $k = 3$, il y a $2^3 = 8$ configurations d'entrée possible et donc $2^8 = 256$ règles de transition différentes. Ces réseaux ont été étudiés de manière approfondie par *Wolfram*. De même que précédemment, on peut coder ces règles de 0 à 255, en classant les configurations d'entrée et de sortie suivant leur valeur binaire.

Configuration d'entrée	**0,0**	**0,1**	**1,0**	**1,1**
Sortie(règle 0)	0	0	0	0
Sortie(règle 1) "ET"	0	0	0	1
Sortie(règle 2)	0	0	1	0
Sortie(règle 3) entrée de gauche	0	0	1	1
Sortie(règle 4)	0	1	0	0
Sortie(règle 5) entrée de droite	0	1	0	1
Sortie(règle 6) "OU" exclusive	0	1	1	0
Sortie(règle 7) "OU"	0	1	1	1
Sortie(règle 8) non "OU"	1	0	0	0
Sortie(règle 9) non "OU" exclusif	1	0	0	1
Sortie(règle 10) inverse (entrée de droite)	1	0	1	0
Sortie(règle 11)	1	0	1	1
Sortie(règle 12) inverse (entrée de gauche)	1	1	0	0
Sortie(règle 13)	1	1	0	1
Sortie(règle 14) non "ET" exclusif	1	1	1	0
Sortie(règle 15)	1	1	1	1

Table 3.2 – Les règles de transition de l'AC de Wolfram

Wolfram [Wol84] a décrit de manière détaillée le comportement des AC unidimensionnels élémentaires. Un AC unidimensionnel élémentaire consiste en une ligne de cellules, chacune colorée de noir ou de blanc. À chaque période de temps, une règle définie détermine la couleur d'une cellule donnée sur base de la couleur de cette cellule et de ses voisines immédiates de gauche et de droite à l'étape précédente" (traduit de Wolfram [WGeH03]). La superposition des états de chaque cellule au fur et à mesure du temps sur un même diagramme permet l'observation des trajectoires de configurations spatiales. La figure 3.4 en présente trois exemples. L'exemple de droite est le plus simple puisque la règle utilisée conduit à une structure complètement homogène où chaque cellule possède le même état. À gauche, une structure émerge, elle est stable et une périodicité peut être identifiée. Au milieu le pattern est chaotique. Des structures y émergent mais sans aucune périodicité.

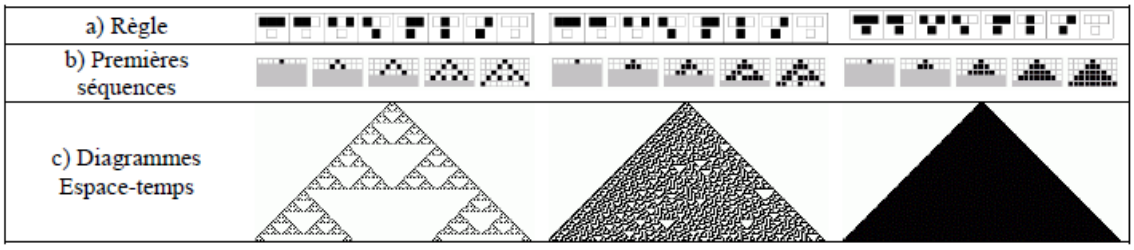

Figure 3.4 – Exemple d'automates cellulaires élémentaires aux règles 90,30 et 254.

3.6.1 L'automate cellulaire élémentaire -Règle 30-

L'automate cellulaire non-trivial le plus simple que l'on puisse concevoir consiste en une grille unidimensionnelle de cellules ne pouvant prendre que deux états ("0" ou "1"), avec un voisinage constitué pour chaque cellule, d'elle-même et des deux cellules qui lui sont adjacentes.

Chacune des cellules pouvant prendre deux états, il existe $2^3 = 8$ configurations (ou motifs) possibles d'un tel voisinage. Pour que l'automate cellulaire fonctionne, il faut définir quel doit être l'état à la génération suivante d'une cellule pour chacun de ces motifs. Il y a $2^8 = 256$ façons différentes de s'y prendre, soit donc 256 automates cellulaires différents de ce type.

Les automates de cette famille sont dits *élémentaires*. On les désigne souvent par un entier entre 0 et 255 dont la représentation binaire est la suite des états pris par l'automate sur les motifs successifs 111, 110, 101, etc.

À titre d'exemple, considérons l'automate cellulaire défini par la table suivante, qui donne la règle d'évolution :

Motif initial (t)	111	110	101	100	011	010	001	000
Valeur suivante de la cellule centrale ($t+1$)	0	0	0	1	1	1	1	0

Table 3.3 – La règle de transition 30

Cela signifie que si par exemple, à un temps t donné, une cellule est à l'état "1", sa voisine de gauche à l'état "1" et sa voisine de droite à l'état "0" (3è*me* colonne du tableau), au temps $t + 1$ elle sera à l'état "0".

Si l'on part d'une grille initiale où toutes les cellules sont à l'état "0" sauf une, on aboutit à : Où chaque ligne est le résultat de la ligne précédente.

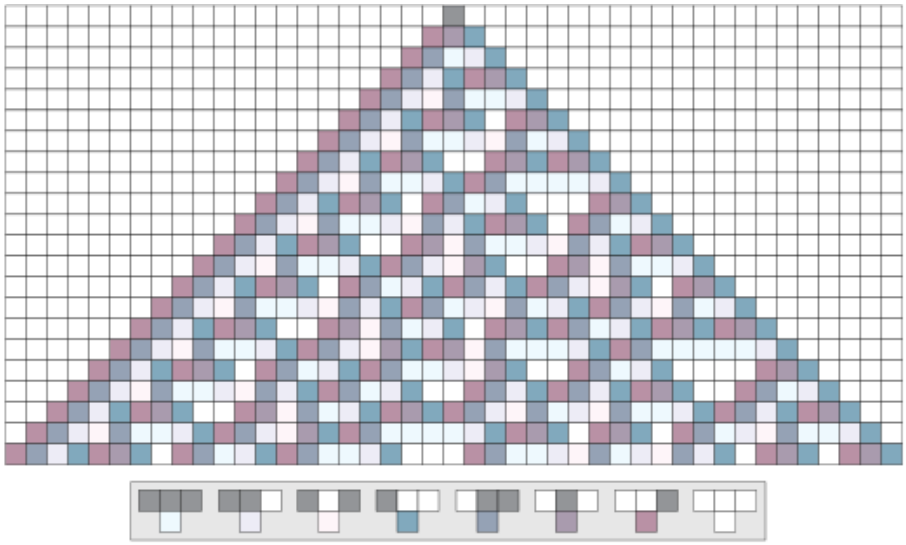

Figure 3.5 – Diagramme espace-temps de la règle 30.

Par convention cette règle est nommée "règle 30", car 30 en décimale s'écrit 00011110 en binaire et 00011110 est la deuxième ligne du tableau ci-dessus, décrivant la règle d'évolution.

3.6.2 Le comportement chaotique de la règle 30

La règle 30 est une règle de transition de l'automate cellulaire unidimensionnel présenté par *Stephen Wolfram* en 1983 [Wol83]. D'après la classification de Wolfram, la règle 30 est une règle de classe III, qui a une structure apériodique, et un comportement chaotique.

Cette règle est particulièrement intéressante, car elle produit des motifs complexes, apparemment aléatoires à partir de règles simples et bien définies. Pour cette raison, Wolfram estime que la règle 30 et les automates cellulaires en général sont la clé pour comprendre comment les règles simples produisent des structures et des comportements complexes dans la nature.

Le schéma suivant (figure 3.6) apparaît à partir d'un état initial dont une seule cellule est à l'état 1 (représenté en noir) est entouré par des cellules à l'état 0 (blanc).

Ici, l'axe vertical représente le temps et l'axe horizontal de la figure 3.6 représente l'état de toutes les cellules à un moment précis de l'évolution de l'automate. Plusieurs motifs sont présents dans cet automate, tels que l'apparition fréquente des triangles blancs et le motif rayé qui est bien défini sur le côté gauche, mais cette structure dans son ensemble n'a pas de tendance perceptible. Le nombre de cellules noires à la génération n est donné par la séquence $1, 3, 3, 6, 4, 9, 5, 12, 7, 12, 11, 14,$

$12, 19, 13, 22, 15, 19, \ldots$ (sequence A070952 in OEIS [1]) et est d'environ n.

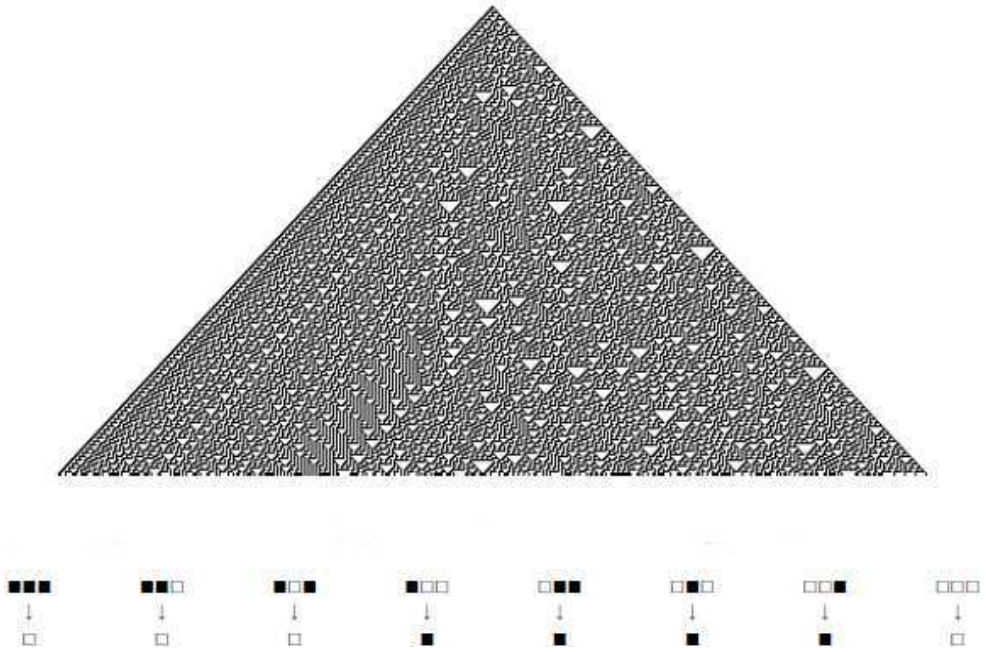

Figure 3.6 – L'évolution dynamique d'un AC unidimensionnel obéissant à la règle 30.

La règle 30 peut être utilisée comme un générateur de nombre aléatoire malgré l'absence de tout ce qui pourrait raisonnablement être considérée comme entrée aléatoire. Stephen Wolfram a proposé d'utiliser la colonne centrale d'un automate cellulaire basé sur la règle comme un générateur de nombres pseudo-aléatoires (PRNG), qui a pu passer de nombreux tests standards pour l'aléatoire.

Wolfram a classifié la règle 30 comme chaotique (classe III) sur la base de son aspect visuel, mais il a été démontré plus tard pour répondre à des définitions plus rigoureuses de chaos proposées par Devaney [Dev08] et Knudson [Knu94], que la règle 30 indique une dépendance sensible aux conditions initiales (deux configurations initiales qui ne diffèrent que dans un petit nombre de cellules divergent rapidement), ce qui signifie que si deux configurations C et D sont différentes dans l'état d'une seule cellule à la position i, après une seule étape, les nouvelles configurations seront différentes à la cellule $i + 1$ [CFM00].

1. The On-Line Encyclopedia of Integer Sequences (OEIS) : Nombre de 1 à la n-ième génération de 1-D AC applique la règle 30, commence avec un seul 1.

3.7 Automates cellulaires de dimension 2

Le premier automate cellulaire a été construit en dimension 2 par Von Neuman pour représenter la logique d'un processus d'autoreproduction. Le monde artificiel de Von Neuman se présente sous la forme d'un réseau composé de mailles carrées ou les cellules peuvent prendre place à chaque intersection. On peut aussi concevoir d'autres formes de structures de réseaux, la seule contrainte étant de conserver à la fois une symétrie de translation et une symétrie de rotation autour des nœuds du réseau. Ainsi une symétrie de rotation d'ordre 3 donnera des figures triangulaires, une symétrie de rotation d'ordre 6 des alvéoles hexagonales etc...

Dans les réseaux à maillage carré, ou treillis carré, les structures d'interaction les plus répandues sont :

- le voisinage de Von Neuman, où chaque automate n'a que ses 4 voisins directs
- le voisinage de Moore où l'on inclut les voisins situés sur les diagonales (ce qui revient à construire une sous-structure d'interaction triangulaire)
- le voisinage élargi *MvonN* qui correspond à un voisinage de Moore auquel on ajoute les 4 voisins les plus proches des voisins de Von Neuman qui ne sont pas déjà inclus dans le voisinage de Moore. Ce voisinage revient à construire des sous-structures d'interaction établissant des liens directs avec tous les voisins séparés de l'automate par au plus un automate sur le treillis carré.

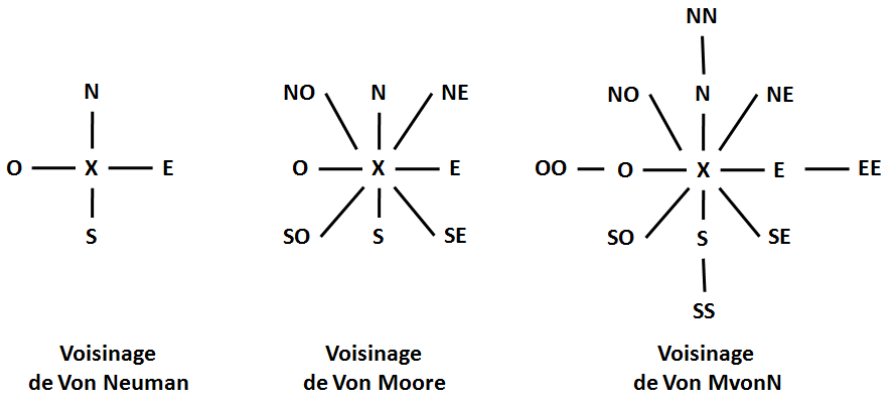

Figure 3.7 – Type de voisinage.

3.7.1 Le Jeu de la vie

À l'origine, le Jeu de la vie fut présenté comme un jeu mathématique. Sa description va nous permettre de matérialiser et mieux comprendre ce que sont les automates cellulaires.

À l'instar des espaces cellulaires d'Ulam, le Jeu de la vie se présente sous la forme

d'une grille constituée de cellules, par exemple :

00	01	02	03	04
10	11	12	13	14
20	21	22	23	24

Figure 3.8 – Exemple de configuration de départ.

L'univers est limité ici à un rectangle de 5 par 3. Pour faciliter l'explication, nous avons numéroté les cellules de 0 à 4 en horizontal et de 0 à 2 en vertical. Les cellules claires sont actives. Dans le Jeu de la vie, est considérée comme voisine toute cellule contiguë, y compris les diagonales.

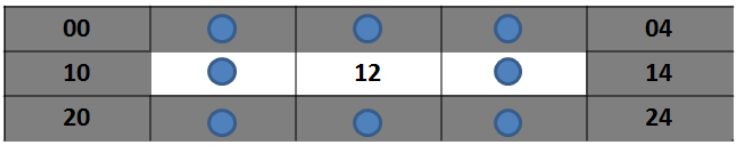

Figure 3.9 – Détermination du voisinage.

La figure ci-dessus montre le voisinage de la cellule 12. En l'occurrence, sur les huit voisins, deux sont actifs.

Les règles du Jeu de la vie sont simples :

- Une cellule inactive entourée de 3 cellules actives devient active ("naît") ;
- Une cellule active entourée de 2 ou 3 cellules actives reste active ;
- Dans tous les autres cas, la cellule "meurt" ou reste inactive.

On peut interpréter ces règles en considérant qu'une naissance nécessite un certain rassemblement de population (3 en l'occurrence), que les cellules ne peuvent survivre à un trop grand isolement (moins de 2 voisines) et qu'une trop forte concentration (plus de 3 voisines) les étouffe.

Les automates cellulaires fonctionnent de manière discrète. C'est-à-dire que le temps s'écoule par à-coups. Ceci signifie dans notre cas qu'à la génération t, chaque cellule examine son environnement et détermine son état futur. Quand l'ensemble des cellules a été traité, et seulement à ce moment là, toutes les cellules passent à l'état calculé. On simule ainsi un traitement parallèle. Illustrons ce mécanisme à partir de la configuration précédente :

1	2	3	2	1
1	1	2	1	1
1	2	3	2	1

Figure 3.10 – Valeurs de voisinage.

Dans le schéma précédent, le nombre de voisins actifs est noté pour chaque cellule :

- Les cellules inactives 00, 04, 10, 14, 20 et 24 ont une voisine active et restent donc en cet état.
- Les cellules inactives 01, 03, 21 et 23 ont deux voisines, elles ne changent donc pas.
- Les deux cellules inactives restantes (02 et 22) ont trois voisines actives, la règle 1 s'applique : elles naissent.
- Les cellules actives 11 et 13 n'ont qu'une voisine active : elles meurent.
- Enfin la cellule active 12 ayant deux voisines actives elle reste en vie.

À la génération suivante, seules les cellules 02, 12 et 22 seront donc actives.

00	01	02	03	04
10	11	12	13	14
20	21	22	23	24

Figure 3.11 – Seconde génération.

3.8 Les propriétés des automates cellulaires

3.8.1 La reproduction

Le but de Von Neumann, à l'origine, lorsqu'il a conçu le principe des automates cellulaires, était de concevoir un mécanisme copiant la vie dans le sens où il pourrait se reproduire de lui même.

Ainsi certains automates cellulaires sont capables de produire des copies d'eux-mêmes, propriété particulièrement intéressante lorsqu'il s'agit de modéliser la vie d'êtres vivants. Ce point de vue de la reproduction permet de classer les automates cellulaires en deux grandes catégories :

- Les automates cellulaires actifs, c'est à dire auto-reproducteurs.
- Les automates cellulaires passifs.

Les automates cellulaires actifs sont auto-reproducteurs dans le sens où ils contiennent une sous-configuration qui se comporte en "copieur universel" dirigeant activement la réplication via une fonction de transition assurant la réplication.

Les automates cellulaires passifs sont ainsi nommés car leur reproduction est provoqué par la règle de transition et non pas par les caractéristiques de la configuration initiale. Le Jeu de la Vie est un exemple d'automate cellulaire passif.

3.8.2 L'inversibilité

On appelle automate inverse l'automate qui permet de revenir aux états précédents.

Un exemple simple d'automate inverse est celui de l'automate *Déplacement Est*. Chaque case peut avoir deux états, 0 et 1 (vide ou plein). L'automate regarde l'état de la case voisine Ouest, s'en souvient et agit en le prenant pour nouvel état de la case. Un réseau d'automates *Déplacement Est* sur un plan a pour effet, d'une génération à l'autre, de déplacer d'une case vers l'Est le motif initial. Son inverse est l'automate *Déplacement Ouest*.

Cet automate possède deux états : 0 et 1, représentés l'un par une case blanche, l'autre par une case noire. D'une génération à l'autre, chaque automate du réseau regarde l'état de son voisin Ouest et le prend pour lui-même. Le résultat est que le dessin se déplace vers l'Est.

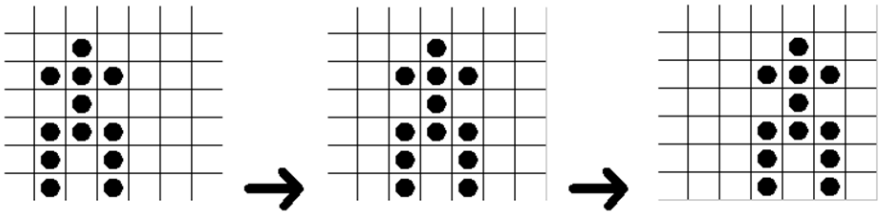

Figure 3.12 – Propriété de l'inversibilité d'un AC-déplacement Est.

On a démontré que tous les automates n'ont pas nécessairement un inverse. Pour démontrer qu'un automate n'a pas d'inverse il suffit de trouver deux configurations différentes qui aboutissent à la même configuration.

Cet automate, utilisant les règles du jeu de la vie, commence par un pentamino et évolue en 11 étapes. Au bout de la onzième, l'étape 12 est la même que la dixième. On a donc deux configurations différentes qui donnent la même configuration : l'automate de *Conway* (qui est celui du jeu de la vie) n'est pas inversible.

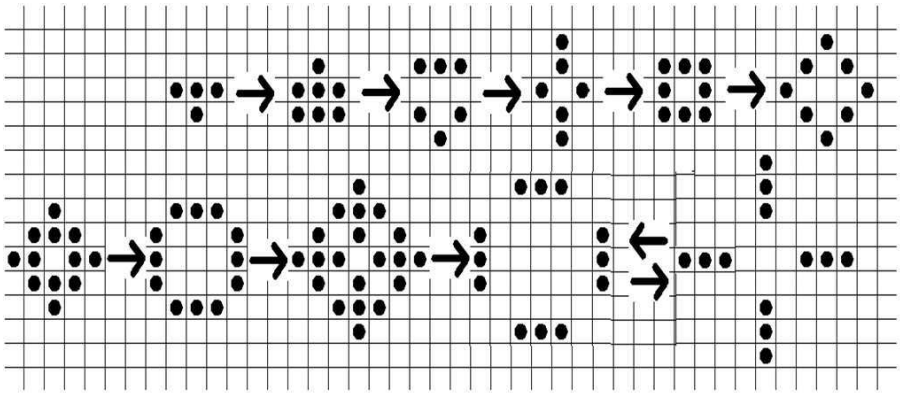

Figure 3.13 – L'inversibilité d'un AC.

La question qui se pose alors est, lorsqu'on a construit un automate, de savoir si celui-ci est inversible. Et bien cette question est un problème indécidable.

3.8.3 L'indécidabilité

L'une des caractéristiques importantes des automates cellulaires est le caractère d'indécidabilité qui touche nombre de leurs propriétés.

Ainsi, déterminer si un automate cellulaire possède un inverse est indécidable : il ne sera jamais possible d'écrire un programme prenant en paramètres un automate quelconque et pouvant décider si oui ou non cet automate possède un inverse.

De la même façon, l'avenir d'un automate est indécidable. On n'a pas de méthode générale permettant de déterminer si un automate ne va pas s'éteindre au bout d'un certain nombre de générations ou s'il va se stabiliser.

Les jardins d'Eden

Un *Jardin d'Eden* est une configuration qui ne possède pas d'antécédent. Cela ne se produit bien entendu pas avec les automates inversibles. De même, par exemple, une configuration Jardin d'Eden ne peut être un attracteur car elle ne peut apparaître que comme première configuration d'une suite de configurations. L'aspect remarquable de ce concept est que la question de l'existence de Jardins d'Eden est indécidable. Cela a été démontré par J. Kari [Kar92] en 1990.

Un autre résultat intéressant avait déjà été trouvé en 1962 par E. Moore [Moo62] et en 1963 par J. Myhill [Myh63] : un automate possède des Jardins d'Eden si et seulement si deux configurations finies donnent le même résultat. Cette précision pour souligner le fait que cette question a su tenir en haleine les informaticiens, et peut-être plus généralement, les logiciens pendant longtemps.

Cette configuration (figure 3.14) n'a pas de prédécesseur : c'est un jardin d'Eden.

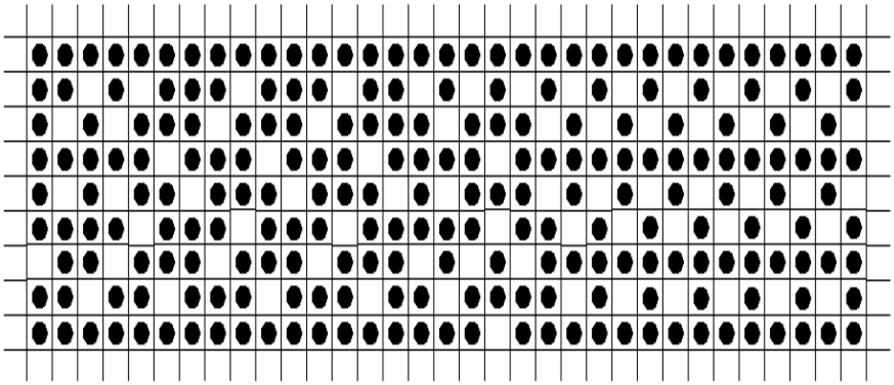

Figure 3.14 – Une configuration de Jardin d'Eden.

3.8.4 Les attracteurs

Les attracteurs des automates sont des configurations qui reviennent indéfiniment. Ce sont des cas particuliers d'automates, ils permettent des études intéressantes sur les automates et la démonstration de propriétés. Ainsi, J. Kari [Kar92] a démontré que toute propriété de l'ensemble limité (l'ensemble des attracteurs) qui est vraie pour certains automates et fausse pour d'autres est indécidable. Ce résultat est finalement le plus extraordinaire de tous, car il nous montre que nous ne saurons jamais rien à l'infini des réseaux d'automates cellulaires.

3.9 Classification de Wolfram

L'histoire des AC tient ses origines depuis les travaux de recherches effectués par Stephen Wolfram sur les AC à l'*Institute for Advanced Study*.

Stephen Wolfram est un physicien et mathématicien anglais, né à Londres, le 29 Août 1959. Il a publié un certain nombre d'articles sur la physique des particules en étant adolescent. Á l'âge de vingt ans, il obtient son doctorat en physique des particules de l'université de Cal Tech et rejoint l'*Institute for Advanced Study* de Princeton en 1982. C'est là, en cherchant des modèles de la façon dont les galaxies s'étaient formées à partir d'un état initial chaotique qu'il s'intéressa aux AC. C'est ce choix qui lui a valu d'être le premier chercheur qui a effectué une étude systématique de la totalité d'un espace d'AC.

En 1984, Wolfram publie un article sous le titre de "L'universalité et la complexité des automates cellulaires" [Wol84]. Il a proposé la première classification des AC selon leur comportement dynamique, en s'inspirant de la théorie des systèmes dynamiques. Partant d'une étude systématique des AC unidimensionnels à deux états sur un espace constitué de 256 AC, il a mis en exergue le résultat suivant :

Si chaque règle conduit à des motifs qui diffèrent dans le détail, tous les AC semblent pouvoir appartenir à seulement quatre classes qualitatives distinctes [Wol84].

3.9.1 Classe I

L'évolution de l'automate après certains pas de temps, pour la quasi totalité des différents états initiaux, tend vers un état homogène où les cellules ont la même valeur. Toute information existante sur l'état initial du système sera complètement détruite, la prédictibilité d'évolution est donc évidente. Partant de l'état initial, le système évolue toujours vers un état homogène.

Comme exemple, voici les règles qui appartiennent à la classe I : 0, 32, 40, 160, 172, 234 et 250.

3.9.2 Classe II

L'évolution de l'automate conduit à un ensemble de structures stables ou périodiques (petite période), mais en tous cas simples et séparées. La prédictibilité d'évolution reste faisable. Les effets d'une cellule se propagent à un nombre fini de voisins. La modification d'une cellule de l'état initial n'affectera qu'une région finie de son entourage.

Comme exemple, voici les règles qui appartiennent à la classe II : 4, 108, 218 et 232.

3.9.3 Classe III

L'évolution de l'automate conduit à un motif chaotique caractérisé par des "attracteurs étranges et des structures apériodiques". Au cours de l'évolution, les cellules propagent les informations à vitesse constante, contrairement à ce qu'on peut observer dans les automates de la classe II. Connaître l'état d'une cellule après un nombre assez grand de pas de temps d'évolution exige la connaissance des états initiaux d'un nombre très grand de cellules. La prédiction de l'évolution n'est donc possible qu'à partir d'un nombre infini d'états initiaux.

Comme exemple, voici les règles qui appartiennent à la classe III : 22, 30, 101, 126, 150, 182.

3.9.4 Classe IV

L'évolution de l'automate conduit à un état mort en permettant l'apparition d'un petit nombre de structures complexes stables ou périodiques, parfois persis-

tantes dans le temps. Le jeu de la vie en est le plus représentatif. Le degré du non prédictibilité est encore plus important que dans les automates de la classe III. Cette classe est de loin la plus intéressante pour la Vie Artificielle.

3.9.5 Illustration

Figure 3.15 – Classification de Wolfram.

Les trois premières classes ont été inspirées des catégories qui apparaissent dans l'étude des systèmes dynamiques, la quatrième étant la plus intéressante car elle est spécifique au domaine des AC. Cependant, il semble être très facile de détecter les insuffisances de cette classification du fait que la classe d'un automate ne peut être déterminée à priori, ici l'accent est mis sur ce que peut être un comportement dynamique d'un AC plutôt que toucher à comment l'aboutir.

D'autres chercheurs ont également critiqué cette classification en disant qu'il est impossible de décider si un automate appartient à la classe III ou à la classe IV sans intervention de la vue d'oeil. D'ailleurs, même si l'on accepte l'hypothèse de Wolfram selon laquelle les automates de la classe IV sont supposés avoir la capacité de réaliser la calculabilité universelle, des résultats d'indécidabilité peuvent être obtenus : il serait alors impossible d'arriver à trouver une méthode systématique pour décider de la classe d'un AC.

Une conclusion est que prévoir la classe d'un AC à partir de sa fonction de transition reste encore un travail à faire.

3.10 Conclusion

Beaucoup de concepts fondamentaux dans les automates cellulaires telles que le mélange, et des mesures préservants les transformations et la sensibilité ont été déjà appliqués pendant longtemps dans la cryptographie.

Le tableau suivant résume les similitudes et les différences entre les automates cellulaires et les algorithmes cryptographiques. Les automates cellulaires et les cryptosystèmes ont quelques propriétés semblables telles que la sensibilité à un changement dans les conditions initiales et les états initiaux d'évolutions.

Les ronds de chiffrage d'un algorithme cryptographique mènent aux propriétés désirées de diffusion et de confusion de l'algorithme. Les itérations d'un automate cellulaire écartent la région initiale au-dessus de l'espace de phase entier.

L'état initial d'évolutions d'un automate cellulaire peut représenter la clé de l'algorithme de chiffrement.

Les systèmes dynamiques d'automates cellulaires possèdent beaucoup de points communs avec les cryptosystèmes puissants :

Système d'AC	**Cryptosystème**
Ensemble d'états discrets	Ensemble de valeurs entières (discrets)
Itérations	Rounds
État initiale d'évolutions	Clé de chiffrement
Distribution aléatoire	Cypher-text aléatoire et confus
Sensibilité aux conditions initiales	Principe de diffusion
Réversibilité difficile sans certains paramètres	Déchiffrement difficile sans clé de chiffrement
Parallélisme implicite	Parallélisations souhaitable
Représentation de données par blocs ou flot	Représentation de données par blocs ou flot

Table 3.4 – Les similitudes et les différences entre les ACs et les algorithmes cryptographiques

Chapitre 4

Chiffrement par flot avec les courbes elliptiques : État de l'art

Sommaire

4.1 Introduction

Dans ce chapitre, nous proposons un état de l'art sur les courbes elliptiques et les générateurs de nombres pseudo-aléatoires (PRNG), et plus précisément sur la génération des nombres pseudo-aléatoires en utilisant les courbes elliptiques et son application à la cryptographie.

En 1985, *Neal Koblitz* et *Victor Miller* [HPSS08] ont proposé de manière indépendante l'utilisation des courbes elliptiques pour créer des cryptosystèmes. Ils ont déclaré que le problème du logarithme discret sur les courbes elliptiques pourrait être plus difficile que le problème du logarithme discret classique dans $\mathbb{Z}_p$. Ainsi, des approches impliquant les courbes elliptiques dans la cryptographie n'ont cessé d'apparaître, particulièrement dans le chiffrement par flot où la sécurité repose sur la sécurité du générateur des nombres pseudo-aléatoires.

Dans cet état de l'art, nous allons présenter des travaux qui ont abordé le problème de génération du Keystream, ainsi que son application dans le chiffrement de *Vernam* pour les images numériques.

4.2 PRNG basés sur les courbes elliptiques

Gong et al. ont construit un générateur de séquences pseudo-aléatoires basées sur la multiplication d'un scalaire par un point et sur la fonction de trace.

En effet, ils ont commencé par générer un point aléatoire, le multiplier par $i = 1, \cdots, 2^n$ pour avoir un point de coordonnées x_i et y_i, ensuite ils ont calculé $a_i = trace(x_i)$ et $b_i = trace(y_i)$. Par conséquent, la séquence générée est la concaténation de a_i et b_i [GBS99].

Beelen et al. ont utilisé les courbes elliptiques d'une autre manière pour la génération des séquences pseudo aléatoire.

Il se sont basés sur les caractères additives et multiplicatives des courbes elliptiques, et aussi, en utilisant les relations de récurrence linéaires [BD02].

Bayer a construit un PRNG appelé ECPRNG basé sur les travaux de Kaliski [Kal86, Kal88] qui a utilisé des courbes elliptiques torsadées.

Il choisit une courbe elliptique E tel que $E(F_p)$ et sa torsadé $E^{torsadé}(F_p)$ ont un ordre primal, il déclare que son cryptosystème est sûr. Cela est dû à la difficulté de résoudre le problème du logarithme discret dans les courbes elliptiques [Bai03].

Lee et al. ont construit un générateur de nombre pseudo aléatoires basés sur les opérations d'addition de points dans une courbe elliptique.

Initialement, ils ont choisit :

– un corps fini F,

- une courbe elliptique E,
- un point de cette courbe noté P,
- et une graine k_1.

Après avoir calculé le point $k_1.P$, son abscisse x, qui va représenter la séquence pseudo aléatoire, est ajouté au nombre d'itérations effectuées à savoir 1 (au début) pour avoir la nouvelle graine k_2, ainsi :

$$k_{n+1} = x_n + n$$

et successivement ce k_{n+1} sera multiplié par le point P [LW04].

Barker et al. ont proposé un générateur de nombres pseudo-aléatoire appelé « The Dual Elliptic Curve Pseudorandom Generator DECPRG » basée entièrement sur les points d'une courbe elliptique, ils déclarent que la sécurité de ce générateur est sûre puisqu'elle est fortement liée à la résolution du problème du logarithme discret dans les courbes elliptique ECDLP [BK07].

En effet, les bits pseudo aléatoires sont extraits à partir d'un point aléatoire d'une courbe elliptique, en exploitant les 240 bits les moins significatifs de l'abscisse de ce point, la courbe elliptique est définie sur un corps fini F_p avec $log_2\ p = 256$. Ils montrent que ces bits sont distincts des 240 bits pseudo aléatoires d'une distribution uniforme. Leur démarche a été comme suit :

Soient deux points P et Q tel que $P = \alpha Q$, α est difficile à retrouver, due à l'insolubilité du problème du logarithme discret dans les courbe elliptique. Ainsi la graine du générateur est un nombre pseudo aléatoire $s_0 \in \{0, 1, \cdots, \#E(F_p) - 1\}$, où $\#E(F_p)$ est le nombre de points de la courbe.

Soit $x : E(F_p) \longrightarrow F_p$ une fonction qui retourne l'abscisse d'un point d'une courbe.

Soit $lsb_i(s)$ une fonction qui retourne i bits les moins significatifs d'un entier s. Par exemple $lsb_3(23) = 7$, puisque $23 = 10111$ en binaire.

Le générateur DECPRG transforme la graine en une séquence pseudo-aléatoire de longueur $240k$, $k > 0$, comme suit :

Données : $s_0 \in \{0, 1, \cdots, \#E(F_p) - 1\}$, k>0
Résultat : $240k$ *bits*
pour $i = 1$ **a** k **faire**
  $s_i \leftarrow x(s_{(i-1)}.P)$;
  $r_i \leftarrow lsb_{240}(x(s_i.Q))$;
fin
retourner $r_1 r_2 \cdots r_k$.

Dans son papier, **Laszlo Merai** propose un PRNG basé sur la génération congruente

des points de la courbe elliptique, à travers une fonction rationnelle qui dépend des coordonnées du point généré, et qui retourne une séquence pseudo aléatoire [Mér12].

Tout d'abord, un générateur congruent de points de la courbe elliptique est défini selon la relation suivante :

$$P_n = G + P_{n-1} = nG + P_0$$

Avec une valeur initiale pour P_0.

Ainsi la séquence pseudo aléatoire s_n est donnée par la fonction suivante :

$$s_n = f(P_n) = f(nG + P_0)$$

Payingat et al. ont proposé un générateur de nombres pseudo-aléatoires pour un chiffrement par flot basé sur les courbes elliptiques.

La séquence est obtenue en combinant l'opération traditionnelle de multiplication d'un scalaire secret avec un point et la transition d'un registre LFSR, donc la séquence représente un ensemble de bits extraits d'un point généré [PP15], l'algorithme est décrit comme suit :

Données : point P, clé secrète $e = 2m\ bits$

Résultat : un bit de séquence s_i

(1) Obtenir e_1 et e_2 de e;

(2) Générer $Q = K_1.P$;

(3) $k_0 = e_1$est la clé initiale, $C_0 = e_2$ est la graine du registre LFSR;

(4) Générer une clé pour la $i^{ème}$ itération : $k_i = X(k_{i-1}.P) + C_{i-1}$;

(5) Faire avancer le compteur du LFSR C_i;

(6) Le $i^{ème}$ point de sortie $S_i = k_i.P + Q$;

(7) Le bit de séquence $s_i = Trunc(X(S_i))$;

(8) Retourner s_i;

(9) Aller à l'étape (4);

Reyad et al. ont proposé une nouvelle construction basée sur les systèmes chaotiques et les courbes elliptiques de caractéristique 2 c-à-d $F(2^m)$.

L'ajout de systèmes chaotiques a rendu le cryptosystème plus rapide et sémantiquement plus sûr. La séquence pseudo-aléatoire constitue un point $U(x, y)$ généré par la relation suivante :

$$U_i = G + U_{i-1} = [i]G + U_0$$

où G est un point donné dans $E(F_p)$ ayant un très haut ordre et U_0 est un point

choisi initialement [RK16].

Dans un autre article, **Laszlo Merai** propose un générateur de séquences (x_n) basé sur les congruences linéaires dans les courbes elliptiques, ce générateur est défini par la relation suivante :

$$x_n = x(W_n) = x(W_{n-1} + G) = x(nG + W_0)$$

tel que W_0 est un point initial dans une courbe elliptique $E(F_p)$ [Mér17].

4.3 Chiffrement par flot des images basés sur les courbes elliptiques

Peu de travaux ont été proposés pour l'utilisation des courbes elliptiques dans le cryptage des images, ceci n'empêche pas de citer quelques travaux qui ont été vus comme :

Sathyanarayana et al. ont élaboré un système de chiffrement par flot des images numériques [SKB11].

Ils ont commencé par construire une courbe elliptique cyclique, c'est une courbe qui possède un point P dont l'ordre est égale aux nombre de points de la courbe elliptique, appelé aussi un point générateur, ensuite ils ont généré une graine k par le générateur $LFSR$, qui par la suite, sera multiplié par P, ainsi pour chaque graine pseudo aléatoire, on a un point pseudo aléatoire dont ses coordonnées seront utilisé comme clé de chiffrement sous différents aspects dans un chiffrement additive et affine. Le principe étant comme suit :

Soit le point pseudo aléatoire généré $k.P$ dont ses coordonnées sont (x, y). L'opération de chiffrement est :

$$chiffré = pixel \oplus clé$$

Pour cela Sathyanarayana et al. ont proposé plusieurs schémas qui différencient sur l'utilisation des coordonnées (x, y) comme clé, par exemple :
$clé = x$;
$clé = y$;
$clé =$ alternation entre x et y
$clé = lsb_i(x)$;
$clé = lsb_i(y)$;
$clé =$ alternation entre $lsb_i(x)$ et $lsb_i(y)$.

Abdellatif et al. ont présenté un chiffrement d'image en utilisant les suites

logistique chaotiques combinées avec les courbes elliptiques cycliques [ELN13].

Leur générateur de flux de clé commence par produire une graine k du corps F_p à l'aide d'un générateur classique, ensuite il multiplie cette graine par un point P de la courbe, ce qui donne une séquence $k.P$ dont les coordonnées x, y seront combinées dans une addition avec la sortie pseudo aléatoire d'un schéma basée sur les suites logistiques chaotiques. Ce schéma combine en addition l'entré (la clé secrète de l'utilisateur) avec deux autres opérandes : la sortie du schéma après une itération, ainsi que le flux de clé généré après aussi une itération.

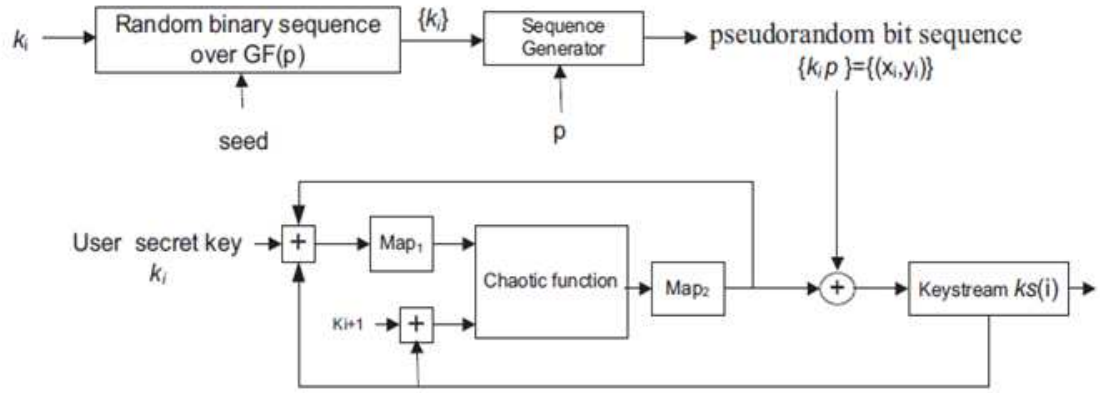

Figure 4.1 – La construction de Abdellatif et al.

Dawahdeh et al. ont développé un cryptosystème pour les images qui combine les courbes elliptiques et le chiffrement de HILL,ce dernier est un algorithme symétrique connu pour être non sécurisé.

L'idée est de transformer l'aspect symétrique du chiffrement de HILL en un autre qui est asymétrique, et ce afin de renforcer sa sécurité. Les résultats ont montré une bonne performance en particulier concernant l'entropie [DYbO18].

4.4 Synthèse et comparaison

Toutes ces approches que nous venons décrire permettent la génération de séquences pseudo-aléatoires, chacune a sa démarche, les auteurs ont exploité les caractéristiques et les services qu'offrent les courbes elliptiques en les combinant d'une façon ou d'une autre avec d'autres systèmes complexes tel que les systèmes chaotiques.

Dans le tableau suivant (tableau 4.1), nous présentons les caractéristiques des différentes approches déjà vu dans ce chapitre.

Approche	**Opération et type de courbe**	**Complexité**
Gong [GBS99]	Multiplication d'un scalaire par un point. Fonction de trace.	PLD
Beelen [BD02]	Relations de récurrence linéaire.	
Bayer [Bai03]	Courbe torsadée.	PLD
Lee [LW04]	Multiplication d'un scalaire par un point.	PLD
Barker [BK07]	L'abscisse d'un point. $lsb_i(s)$.	PLD
Merai [Mér12]	Les congruences.	
Payingat [PP15]	Multiplication d'un scalaire par un point. Transition d'un registre LFSR.	PLD
Reyad [RK16]	Systèmes chaotiques. Courbe de carctéristique 2.	\
Merai [Mér17]	les congruences linéaires.	\
Sathyanarayana [SKB11]	Multiplication d'un scalaire par un point. Courbe cyclique.	PLD PLD
Abdellatif [ELN13]	Suites logistiques chaotiques. Multiplication d'un scalaire par un point. Courbe cyclique.	PLD
Dawahdeh [DYbO18]	Chiffrement de HILL.	\

Table 4.1 – Comparaison entre les différentes approches

4.5 Conclusion

Dans cet état de l'art, nous avons entamé différentes approches pour le développement des PRNG basé sur l'implication des courbes elliptiques. Par conséquent, nous avons pu mettre en place notre propre architecture pour le développement d'un PRNG en utilisant les courbes elliptiques et les automates cellulaires dans le chiffrement par flôt, et qui fera l'objet du chapitre suivant.

Chapitre 5

Chiffrement par flot elliptique et Automates Cellulaires

Sommaire

5.1 Introduction

Dans le chapitre précédant, nous avons vu les différentes approches connues pour le développement de générateurs des séquences pseudo-aléatoires en utilisant les courbes elliptiques.

Dans ce chapitre, nous proposons une nouvelle approche pour le développement d'un chiffrement par flot des images numériques. La nouveauté dans cette approche est comment faire la génération du *keystream* en combinant à la fois les systèmes dynamiques comme les automates cellulaires avec les courbes elliptiques.

5.2 Algorithme de chiffrement d'image proposé

5.2.1 Générateur de séquences pseudo-aléatoire

Le générateur proposé (figure 5.1) combine les automates cellulaires avec les courbes elliptiques en exploitant sa fameuse opération de multiplication d'un point par un scalaire [HMk18].

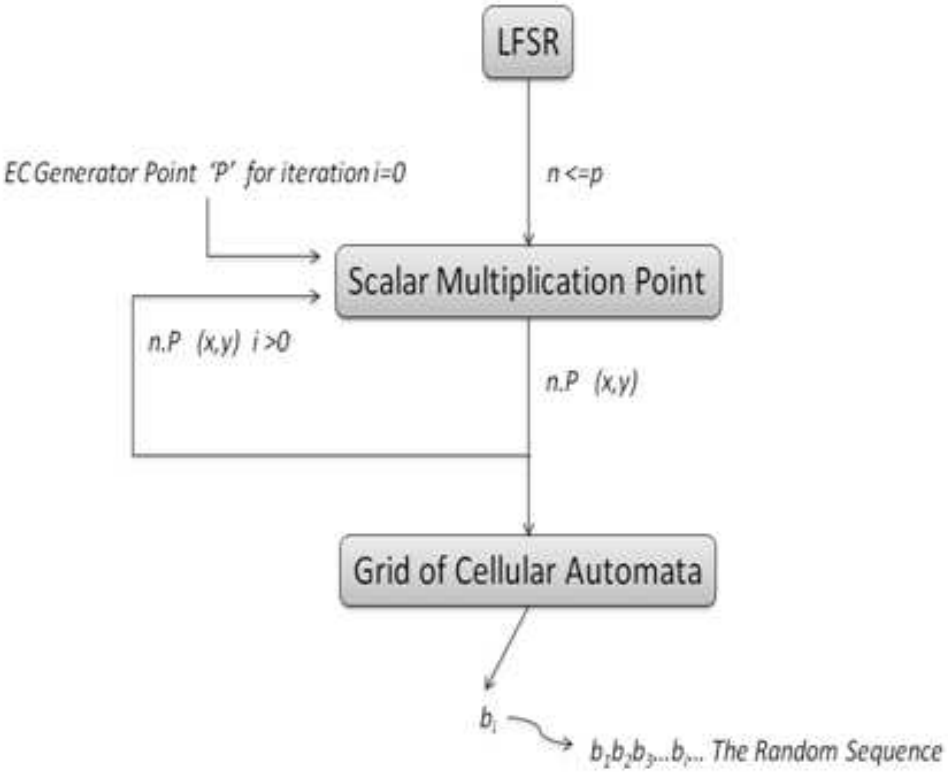

Figure 5.1 – Schéma du générateur de séquences pseudo-aléatoires

D'abord une graine générée par un générateur conventionnel, pour notre cas nous

avons opté pour le générateur LFSR si répandu en recherche, nous aurions pu utiliser un autre générateur de n'importe quel langage tel que la classe *Random* du langage Java.

Cette graine est ensuite multipliée par un point de base de la courbe elliptique cyclique. Le résultat étant simplement les coordonées x, y d'un point dans $E(F_p)$ qui correspondront respectivement à la ligne et à la colonne de la grille statique générée par un automate cellulaire, la cellule désignée qui contient soit 1 soit 0 fera l'objet d'un nouveau bit de notre séquence aléatoire (voir figure 5.2).

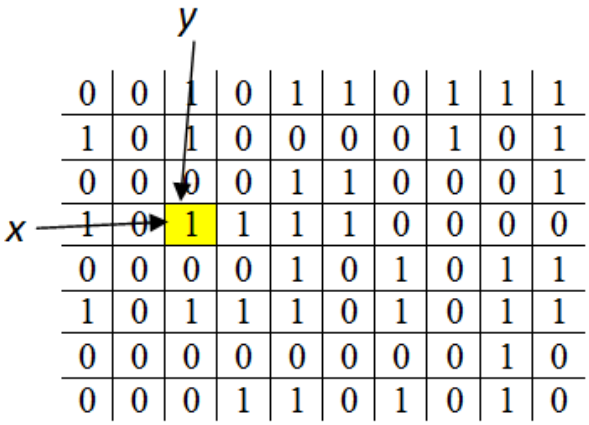

Figure 5.2 – Exemple : $Grid(x, y) = 1$

Le processus est répété autant de fois jusqu'à obtenir une séquence aléatoire de k bits.

Remarques :

- L'algorithme utilisé pour la multiplication d'un point par un scalaire est le fameux Double-and-Add à cause de sa simplicité,
- La taille des automates cellulaires est de 1024,
- Aussi le nombre de transition est assez grand, pour notre cas, nous avons choisi 1024,
- La grille générée par les automates est une matrice carrée.
- La cellule indiquée par les coordonnées (x, y) du point $n_i.P$ correspond au bit de la séquence aléatoire.

Données : P un point de la courbe elliptique, n une graine
Résultat : $b_1 b_2 \cdots b_i \cdots b_k$
pour $i = 1$ **a** k **faire**
  $P \leftarrow n.P$; //(x,y) sont les coordonnées du point $n.P$
  $b_i \leftarrow Grid(x, y)$;
fin
retourner $b_1 b_2 \cdots b_k$.

Algorithme 1: Génération de la séquence pseudo-aléatoire

Où $Grid$ représente la transition des automates cellulaires obtenus comme une matrice.

Nous pouvions proposer plusieurs schémas pour calculer b_i comme :
$b_i \longleftarrow Grid(x, x)$ ou ;
$b_i \longleftarrow Grid(y, y)$ ...etc.

5.2.2 Schéma de chiffrement

Ce générateur sera injecter dans le chiffrement par flot des images, ce chiffrement est considéré comme parfait et sûr, il est aussi simple à implémenter, donc une fois la séquence aléatoire générée, elle est utilisée comme un keystream dans le Xor du pixel (figure 5.3), ainsi :

$$c_i = k_i \oplus p_i$$

où :
p_i est un pixel en clair, c_i est un pixel chiffré, et k_i est une partie du keystream.

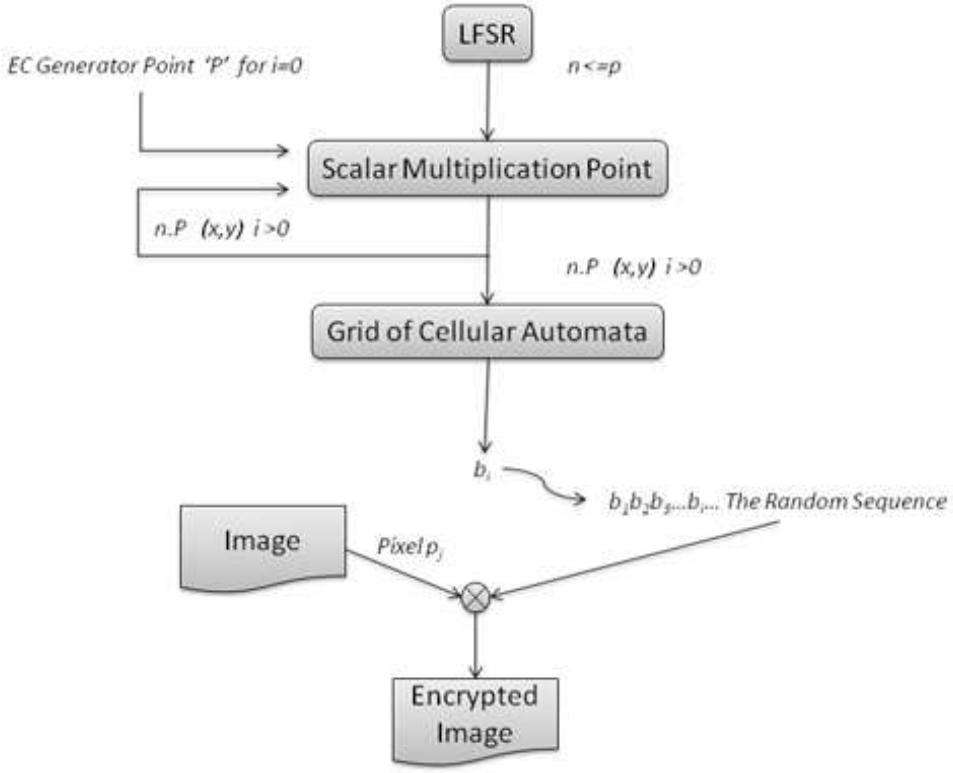

Figure 5.3 – Le cryptosystème proposé

5.3 Analyse et résultats expérimentales

5.3.1 Aspect sécuritaire de l'approche proposée

Analyse du keystream

Notre PRNG est imprédictible, en effet, si nous possédons les premiers i bits $b_1, b_2, \cdots, b_i$ de la sortie du PRNG alors, on ne peut pas calculer le reste des bits $b_{i+1}, b_{i+2}, \cdots, b_n$, car b_{i+1} dépend de l'état précédent de la grille tel que $b_{i+1} = Grid(n.P)$ et il est prouvé être sécurisé, car pour cela, il faut résoudre le problème du logarithme discret elliptique (ECDLP) de $n.P$ pour obtenir n.

Espace des clés

Pour avoir un cryptosystème sécurisé, qui repose sur le secret de la clé, l'espace de celle-ci doit être suffisamment grand pour empêcher l'effet de l'attaque par force brute dans un délai raisonnable.

Notre PRNG dispose d'un espace de clé large et flexible, en effet, la clé est construite à partir d'une paire (n, P) et sa dimension est la suivante :

- Le nombre des courbes elliptiques distinctes sur $GF(2^m)$ est de $2(2^m-1)$ (pour notre cas $m = 24$),
- Le nombre possible de points P est 2^{2m},
- Le nombre de valeurs possibles pour n dépend de la taille du registre LFSR d, soit 2^d (pour notre cas $d = 32$).

Par conséquent, le nombre total des clés possibles est égal au produit de ce que nous venons de déclarer :

$$2(2^m - 1) \times 2^{2m} \times 2^d$$

5.4 Résultats du chiffrement d'une image

5.4.1 Illustration de l'approche

Nous avons testé notre approche [HMk18] sur une image numérique représentant des pingouins, avec la résolution de 1024×768, et la taille de 759 Ko, la figure 5.4 représente l'image en clair et son image chiffrée, en comparant les deux images, on constate qu'il n'y a pas d'informations visibles dans l'image cryptée sur l'image en clair.

Cependant, la vérification visuelle n'est pas assez convaincante pour juger la performance d'un système. Pour cela, nous allons introduire quelques tests statistiques pour prouver la qualité de notre image cryptée.

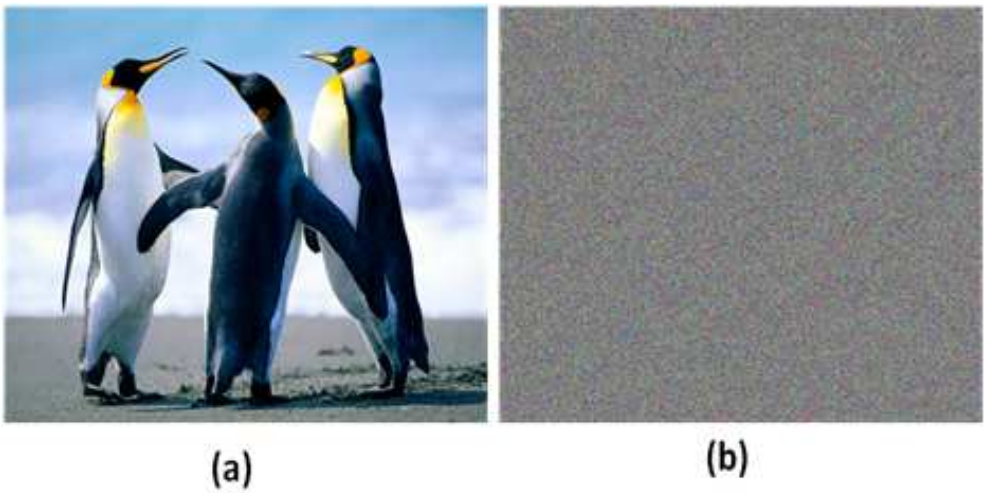

Figure 5.4 – (a) Image en clair. (b) Image chiffrée

5.4.2 Histogrammes

L'histogramme d'une image montre la distribution des pixels de niveau gris, la figure 5.5 (a) illustre la distribution des pixels de l'image en clair, cette distribution dépend bien du contenu de l'image, la figure 5.5 (b) montre la distribution des pixels de niveau gris de l'image chiffrée, on constate que la distribution est uniforme et loin d'être similaire à l'histogramme (a). Cela montre une bonne propriété statistique qui stipule que tous les pixels de l'image cryptée apparaissent avec presque la même probabilité, et montre également qu'un attaquant ne peut pas avoir d'informations sur l'image en clair (Cipher Image Only Attack), ce qui implique une sécurité contre l'analyse statistique.

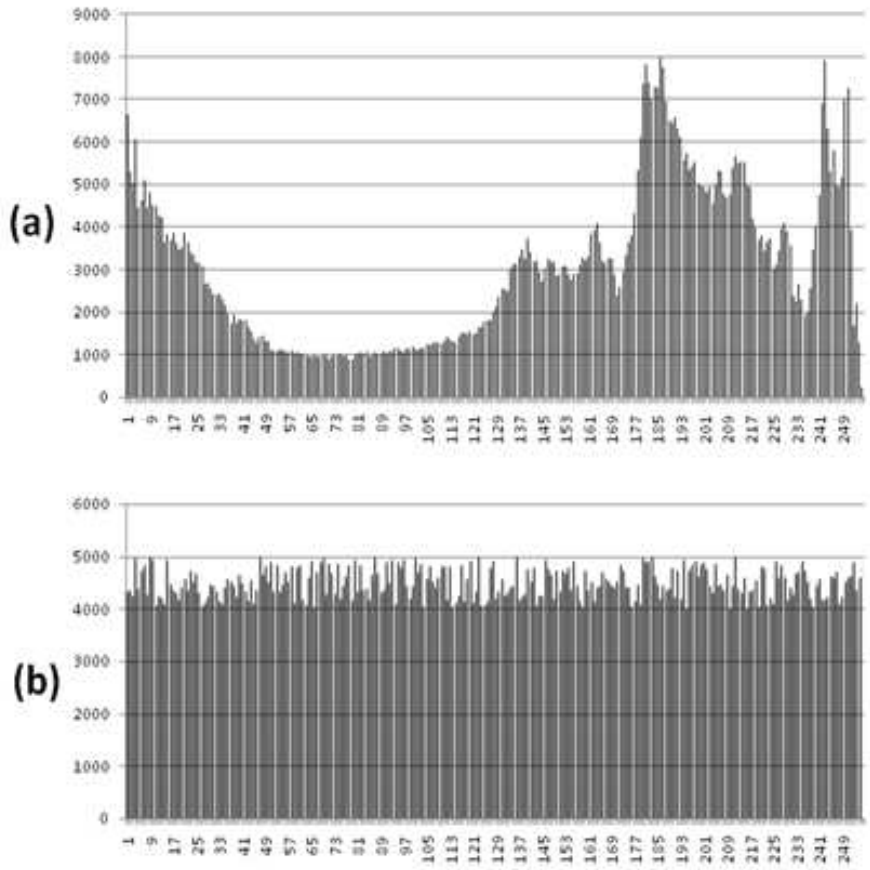

Figure 5.5 – (a) Histogramme de l'image en clair. (b) Histogramme de l'image chiffrée

5.4.3 Corrélation

Dans toute image normale, chaque pixel est fortement corrélé avec ses pixels adjacents ; soit verticalement, horizontalement ou en diagonale. Cependant, les images chiffrées ne doivent pas avoir cette propriété, car leur coefficient de corrélation est

inférieur à celui de l'image simple, pour cela nous avons calculé le coefficient de corrélation des trois niveaux (horizontal, vertical et Diagonal) pour les trois couleurs (rouge, vert et bleu) [ALI+13].

La figure 5.6 montre le graphique des corrélations de l'image en clair, on peut voir que le coefficient de corrélation est régulier dans les trois niveaux de couleur et varie entre 0, 97 et 0, 98, très proche de 1. Comme pour la figure 5.7, qui montre le graphe de corrélation de l'image chiffrée, on peut voir que les coefficients de corrélation sont moins inférieurs aux précédents.

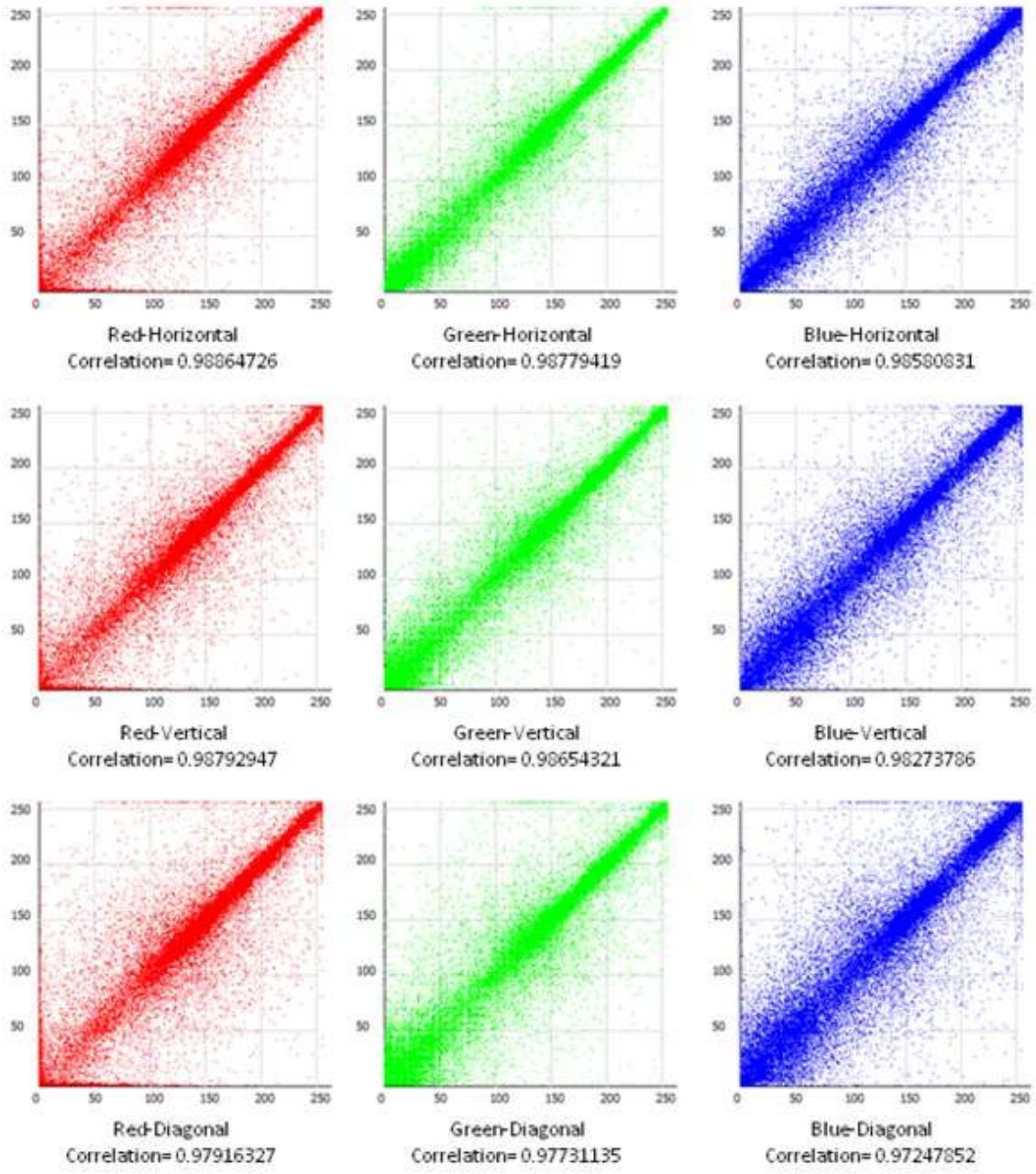

Figure 5.6 – Graphe de corrélation de l'image en clair

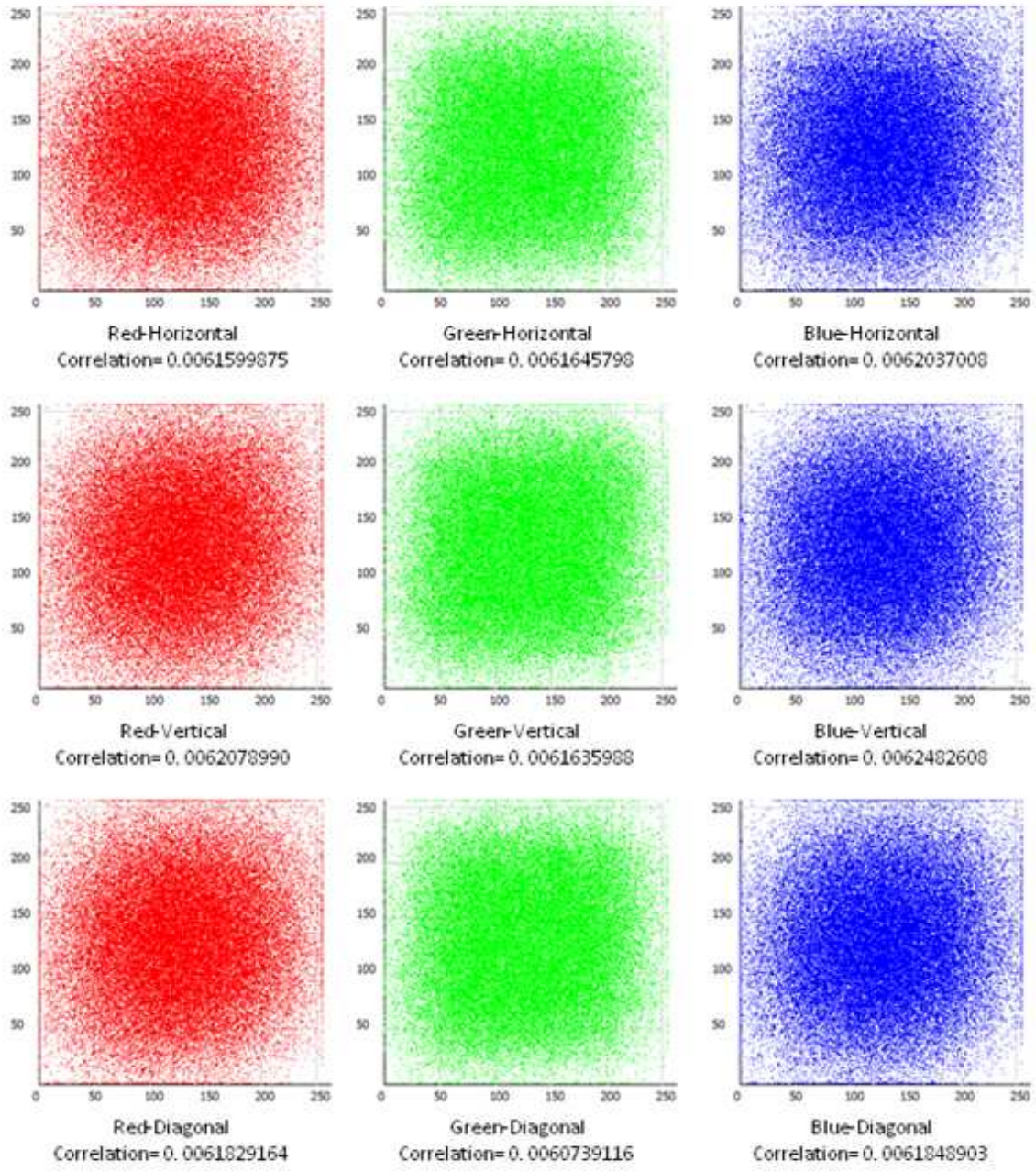

Figure 5.7 – Graphe de corrélation de l'image chiffrée

5.4.4 Entropie de l'information

L'entropie est une mesure aléatoire de la théorie de l'information ; elle est calculé en utilisant l'équation suivante :

$$H = -\sum_{i=0}^{2m-1} p_i \; log_2(p_i)$$

où p est la distribution de probabilité des différentes valeurs du niveau de gris d'une image (de 0 à 255).

Une entropie élevée indique un degré élevé de caractère aléatoire, de sorte que l'entropie de valeur élevée d'un message codé sur m bits est m. Puisque la couleur est codée sur 24 bits/pixel (8 bits pour chaque couleur), la valeur d'entropie optimale pour chaque couleur est de 8 dans les images chiffrées.

Le tableau 5.1 montre les différentes valeurs d'entropie pour les images (en clair et chiffrée), il est clair que l'image chiffrée a une valeur d'entropie proche de l'optimum, ce qui implique que le schéma de chiffrement est sécurisé contre l'attaque basée sur l'entropie.

Image en clair			**Image chiffrée**		
Rouge	**Vert**	**Bleu**	**Rouge**	**Vert**	**Bleu**
7.1326	7.0915	7.0726	7.9907	7.9890	7.9800

Table 5.1 – Entropie de l'image en clair/chiffrée pour le cryptosystème proposé.

5.5 Comparaison avec d'autres approches

Le cryptosystème que nous venons de proposer est original dans sa conception, puisqu'il combine deux objets complètement différents dans la construction du générateur de nombres pseudo-aléatoires, à savoir les courbes elliptiques et les automates cellulaires, d'autres approches ont été proposées dans ce sens comme [ELN13].

Comme notre système de chiffrement a prouvé sa sécurité contre la cryptanalyse statistique, et a montré de bons résultats de chiffrement, l'étude comparative porte sur les valeurs des coefficients de corrélation et de l'entropie pour comparer les performances de notre approche proposée avec celles proposées et comparées dans [ELN13, SKB11].

Le tableau 5.2 montre les résultats de la comparaison en termes de corrélation et d'entropie. À partir des valeurs présentées dans le tableau cité ci-avant, il apparaît que notre approche présente une performance concurrente aux autres approches.

5.6 Conclusion

Dans ce chapitre, un nouveau système de chiffrement a été proposé, basé sur des objets purement mathématiques tel que les courbes elliptiques et des systèmes dynamiques comme les automates cellulaires qui ont fait leurs preuves.

Approche	**Entropie**	**Coéfficient de corrélation horizointal**	**Coéfficient de corrélation vertical**	**Coéfficient de corrélation diagonal**
Approche proposée	7.9865	0.0061	0.0062	0.0061
[ELN13]	7.9973	0.0010	0.0017	0.0125
AES	7.91	0.046	0.066	0.056
AES + W7	8	0.02	0.03	0.025
AES + A5/1	7.96	0.056	0.077	0.067
Basée sur le chaos	7.92	0.0308	0.0304	0.0317
[ELN13] schéma 1	7.9331	-0.0037	0.0032	0.0055
[ELN13] schéma 2	7.9520	0.0013	0.0044	0.0080
[ELN13] schéma 3	7.9718	0.0031	0.0029	7.59e-005
[ELN13] schéma 4	7.9966	7.6340e-004	0.0045	-4.24e-005
[ELN13] schéma 5	7.9915	4.9460e-005	-7.20e-004	-0.0011
[ELN13] schéma 6	7.9964	-7.98e-004	-0.0013	-0.0046
[ELN13] schéma 7	7.9997	0.0030	0.0030	0.0027
[ELN13] schéma 8	7.9996	-0.0027	-0.0028	0.0026

Table 5.2 – Comparaison des résultats en termes de corrélation et d'entropie.

Cette combinaison a permit de générer des séquences pseudo-aléatoires. En effet, le générateur conçu utilise les coordonnées d'un point généré aléatoirement, et les associe à une matrice construite par les transitions d'un automate cellulaire unidimensionnelle. L'efficacité du générateur est prouvée, puisqu'il faut résoudre l'ECDLP. Le cryptosystème a été testé et les résultats étaient performants.

Chapitre 6

Conclusion générale

Depuis son apparition, l'informatique s'est développée massivement et surtout très rapidement. L'expansion de l'Internet a profondément changé nos méthodes de travail. Les communications sont alors devenues omniprésentes et indispensables dans tous les domaines, et sont de plus en plus utilisées dans la vie quotidienne.

Pour sécuriser ces communications et les rendre confidentielles, la cryptographie a accompagné ce développement et a connu quand à elle un essor exponentiel. Depuis l'ère du chiffrement conventionnel tels que les cryptosystèmes symétriques, en passant par les cryptosystèmes asymétriques introduits dans les années 70, ce qui a bouleversé le monde de la cryptographie et a rendu cette dernière plus impliquées dans les domaines industriels aussi bien que dans les domaines militaires et diplomatiques.

Ce développement à connu en parallèle, le progrès de la technologie de l'information jusqu'au multimédia, où tout est devenu *image* et *vidéo*. Par conséquent, il fallait lier ces technologies avec la cryptographie et les rendre encore aussi confidentielles et intègres que leurs ancêtres écrits. Plusieurs techniques furent émergées et mises au point, avec des objets mathématiques prometteurs tels que les courbes elliptiques. Depuis leurs introduction comme domaine fertile de cryptographie, les courbes elliptiques n'ont cessé de rapporter de performance et de bon résultats sur les cryptosystèmes modernes, comme leurs homologues des systèmes dynamiques, en particulier les automates cellulaires qui ont prouvé une caractéristique plus que satisfaisante dans les générateurs des nombres pseudo-aléatoires.

Dans cette thèse, nous avons exploité les travaux combinant les cryptosystèmes basés sur les courbes elliptiques, ainsi que les générateurs de séquences pseudo-aléatoires pour donner naissance à un nouveau cryptosystème hybridant les courbes elliptiques et les automates cellulaires élémentaires.

Un état de l'art a été élaboré survolant les différentes approches et mettant en

œuvre les générateurs des nombres pseudo-aléatoires basées sur les courbes elliptiques, ainsi que des approches du chiffrement par flot pour les images numériques en support avec les courbes elliptiques.

Une approche de développement de générateur de nombres pseudo-aléatoires appliquée dans le chiffrement par flot des images numériques a été proposée. Cette approche vise à choisir aléatoirement des cases d'une grille dessinée par les transitions d'un automate cellulaire élémentaire unidimensionnel. Ce choix est basé sur les coordonnées d'un point générée à partir de la multiplication d'un scalaire avec un autre point qui représente le point de base ou point générateur de la courbe, le processus est répété autant de fois que la séquence pseudo-aléatoires est construite.

Enfin, un prototype a été implémenté pour montrer l'intérêt et la faisabilité de notre approche dans un chiffrement par flot ayant comme des entrées des images numériques.

Ce travail a fait l'objet d'un article scientifique parût dans le journal : "*Journal of Information Security Research*" dans son numéro 1 du volume 9 en mars 2018.

Parmi les travaux futurs à l'approche, nous pouvons citer :

- Améliorer l'approche en utilisant des automates cellulaires plus robustes tels que les automates cellulaires programmables.
- Utiliser le générateur ainsi construit dans la conception de nouveaux fonctions de hachage et signatures numériques.
- Implémenter l'approche dans un système réel tel que les systèmes embarqués.

Bibliographie

[ALI+13] AA Abdo, Shiguo Lian, IA Ismail, M Amin, and H Diab. A cryptosystem based on elementary cellular automata. *Communications in Nonlinear Science and Numerical Simulation*, 18(1) :136–147, 2013.

[Bai03] Harald Baier. *A Fast Java Implementation of a Provably Secure Pseudorandom Bit Generator Based on the Elliptic Curve Discrete Logarithm Problem*. Inst. für Theoretische Informatik, 2003.

[BD02] PHT Beelen and JM Doumen. Pseudorandom sequences from elliptic curves. In *Finite Fields with Applications to Coding Theory, Cryptography and Related Areas*, pages 37–52. Springer, 2002.

[BK07] Elaine B Barker and John Michael Kelsey. *Recommendation for random number generation using deterministic random bit generators (revised)*. US Department of Commerce, Technology Administration, NIST, 2007.

[BSSS99] Ian Blake, Gerald Seroussi, Gadiel Seroussi, and N Smart. *Elliptic curves in cryptography*, volume 265. Cambridge university press, 1999.

[CFM00] Gianpiero Cattaneo, Michele Finelli, and Luciano Margara. Investigating topological chaos by elementary cellular automata dynamics. *Theoretical computer science*, 244(1-2) :219–241, 2000.

[Dev08] Robert Devaney. *An introduction to chaotic dynamical systems*. Westview press, 2008.

[DYbO18] Ziad E Dawahdeh, Shahrul N Yaakob, and Rozmie Razif bin Othman. A new image encryption technique combining elliptic curve cryptosystem with hill cipher. *Journal of King Saud University-Computer and Information Sciences*, 30(3) :349–355, 2018.

[ELN13] Ahmed A Abd El-Latif and Xiamu Niu. A hybrid chaotic system and cyclic elliptic curve for image encryption. *AEU-International Journal of Electronics and Communications*, 67(2) :136–143, 2013.

[Eng12] Andreas Enge. *Elliptic curves and their applications to cryptography : an introduction*. Springer Science & Business Media, 2012.

[GBS99] Guang Gong, Thomas A Berson, and Douglas R Stinson. Elliptic curve pseudorandom sequence generators. In *International Workshop on Selected Areas in Cryptography*, pages 34–48. Springer, 1999.

[Heu94] Jean-Claude Heudin. *La vie artificielle*. Hermes, 1994.

[HMk18] Bouchakour Errahmani Hichem and Faraoun Mohamed kamel. Towards a hybrid approach based on elliptic curves and cellular automata to encrypt images. *Journal of Information Security Research*, 9(1) :1–14, 2018.

[HPSS08] Jeffrey Hoffstein, Jill Catherine Pipher, Joseph H Silverman, and Joseph H Silverman. *An introduction to mathematical cryptography*, volume 1. Springer, 2008.

[Kal86] Burton S Kaliski. A pseudo-random bit generator based on elliptic logarithms. In *Conference on the Theory and Application of Cryptographic Techniques*, pages 84–103. Springer, 1986.

[Kal88] Burton Stephen Kaliski. *Elliptic curves and cryptography : A pseudorandom bit generator and other tools*. PhD thesis, Massachusetts Institute of Technology, 1988.

[Kar92] JJ Kari. Decision problems concerning cellular automata. 1992.

[Knu94] Carsten Knudsen. Chaos without nonperiodicity. *The American Mathematical Monthly*, 101(6) :563–565, 1994.

[Kob94] Neal Koblitz. *A course in number theory and cryptography*, volume 114. Springer Science & Business Media, 1994.

[Kob12] Neal Koblitz. *Algebraic aspects of cryptography*, volume 3. Springer Science & Business Media, 2012.

[LW04] Lap-Piu Lee and Kwok-Wo Wong. A random number generator based on elliptic curve operations. *Computers & Mathematics with Applications*, 47(2-3) :217–226, 2004.

[Men12] Alfred J Menezes. *Elliptic curve public key cryptosystems*, volume 234. Springer Science & Business Media, 2012.

[Mér12] László Mérai. Remarks on pseudorandom binary sequences over elliptic curves. *Fundamenta Informaticae*, 114(3-4) :301–308, 2012.

[Mér17] László Mérai. Predicting the elliptic curve congruential generator. *Applicable Algebra in Engineering, Communication and Computing*, 28(3) :193–203, 2017.

[Moo62] Edward F Moore. Machine models of self-reproduction. In *Proceedings of symposia in applied mathematics*, volume 14, pages 17–33. American Mathematical Society New York, 1962.

[Myh63] John Myhill. The converse of moore's garden-of-eden theorem. *Proceedings of the american mathematical society*, 14(4) :685–686, 1963.

[PP15] Jilna Payingat and Deepthi P Pattathil. Pseudorandom bit sequence generator for stream cipher based on elliptic curves. *Mathematical Problems in Engineering*, 2015, 2015.

[RK16] Omar Reyad and Zbigniew Kotulski. Pseudo-random sequence generation from elliptic curves over a finite field of characteristic 2. In *Computer Science and Information Systems (FedCSIS), 2016 Federated Conference on*, pages 991–998. IEEE, 2016.

[Sil05] Joseph H Silverman. Elliptic curves and cryptography. In *PROCEEDINGS OF SYMPOSIA IN APPLIED MATHEMATICS*, volume 62, page 91, 2005.

[SKB11] SV Sathyanarayana, M Aswatha Kumar, and KN Hari Bhat. Symmetric key image encryption scheme with key sequences derived from random sequence of cyclic elliptic curve points. *IJ Network Security*, 12(3) :137–150, 2011.

[VNB96] John Von Neumann and Arthur Walter Burks. *Theory of self-reproducing automata*. University of Illinois Press Urbana, 1996.

[Was03] Lawrence C Washington. *Elliptic curves : number theory and cryptography*. Chapman and Hall/CRC, 2003.

[WGeH03] Stephen Wolfram and M Gad-el Hak. A new kind of science, 2003.

[Wol83] Stephen Wolfram. Statistical mechanics of cellular automata. *Reviews of modern physics*, 55(3) :601, 1983.

[Wol84] Stephen Wolfram. Universality and complexity in cellular automata. *Physica D : Nonlinear Phenomena*, 10(1-2) :1–35, 1984.

Résumé :

Dans cette thèse, nous présentons une nouvelle approche de chiffrement d'image en utilisant un chiffrement par flot, le générateur de nombres pseudo-aléatoires est basé sur les automates cellulaires unidimensionnels élémentaires (AC) et les courbes elliptiques. En effet, nous avons exploré les transitions de l'AC avec les coordonnées d'un point appartenant à une courbe elliptique; ce point est le résultat d'un autre point qui a été multiplié par un scalaire, connu sous le nom du problème de logarithme discret de courbe elliptique (PLDCE). Ce dernier complique la génération d'un point, et rend impossible de trouver son antécédent. AC offre aussi des qualités d'ambiguïté et de chaos, combinant ainsi les deux concepts; nous avons construit un générateur qui génère un flux de clé, utilisé dans notre approche. Ce travail constitue une analogie avec des travaux impliquant les systèmes dynamiques comme les suites logistiques avec les courbes elliptiques. Notre générateur a montré de bonnes propriétés cryptographiques, car il est basé sur le PLDCE. Nous avons testé les images cryptées, et il s'avère que les résultats sont de haute performance.

Mots clés: Cryptographie par les courbes elliptiques, Automates cellulaires, Générateur de nombres pseudo-aléatoires, Chiffrement d'image.

Abstract :

In this thesis, we present a new approach of image encryption using a stream cipher, the pseudo-random number generator is based on the elementary one dimensional cellular automata (CA) and elliptic curves. Indeed, we explored the transitions of the CA with the coordinates of a point belonging to an elliptic curve; outcome from another point which was multiplied by a scalar, known as the Elliptic Curve Discrete Logarithm Problem (ECDLP). This last complicates the generation of a point, and makes it impossible to find its antecedent. CA also offers the qualities of ambiguity and chaos, so combining the two concepts; we have constructed a PRNG that generates a key stream, used by the way in our approach. This work constitutes an analogy to works that involved dynamic systems like logistics map with elliptic curves. Our PRNG showed good cryptographic properties, since it is based on the ECDLP. We tested the encrypted images, and it turns out that the results are high performance.

Keywords: Elliptic Curve Cryptography, Cellular Automata, Pseudo-Random Number Generator, Image Encryption.

ملخص:

في هذه الرسالة ، نقدم طريقة جديدة لتشفير الصور باستخدام تشفير دفق، يعتمد مولد الرقم العشوائي الزائف على الأوتوماتية الخلوية الأولية والمنحنيات الإهليلجية. في الواقع ، لقد استكشفنا تحولات الأوتوماتية الخلوية مع إحداثيات نقطة ، تنتمي إلى منحنى إهليلجي ، هذه النقطة هي نتيجة لنقطة أخرى تم ضربها بواسطة عدد، تُعرف باسم مشكلة اللوغاريتم المنفصلة في المنحنيات الإهليلجية. هذا الأخير يعقد توليد نقطة ، ويجعل من المستحيل العثور على سابقة لها. تقدم الأوتوماتية الخلوية أيضا صفات الغموض والفوضى، وبالتالي الجمع بين المفهومين، بنينا مولدًا يولد مفتاحًا، يستخدم في نهجنا. هذا العمل مماثل للعمل الذي ينطوي على أنظمة ديناميكية مثل المتتاليات اللوجيستية والمنحنيات الإهليلجية. أظهر مولدنا خصائص تشفير جيدة، لأنه مبني على مشكلة اللوغاريتم المنفصل في المنحنيات الإهليلجية. اختبرنا الصور المشفرة، واتضح أن النتائج عالية الأداء.

الكلمات المفتاحية: التشفير المنحنى الاهليلجي ، الأوتوماتية الخلوية ، مولد الأرقام‌الشبه عشوائية ، تشفير الصور