ROYAUME DU MAROC
Université Mohammed V - RABAT -
جامعة محمد الخامس – الرباط –
Faculté des sciences
كلية العلوم
CENTRE D'ETUDES DOCTORALES - SCIENCES ET TECHNOLOGIES

N° d'ordre : 3335

THESE

En vue de l'obtention du : ***DOCTORAT***

Centre de recherche : CEREMAR-FS
Structure de Recherche : Laboratoire Mathématiques, Statistique et Applications
Discipline : Mathématiques appliquées
Spécialité : Cryptographie et sécurité de l'information

Présentée et soutenue le : 15/09/2020 par :

Khalid CHARIF

Nouveaux Cryptosystèmes Basés sur le Billard Chaotique de Sinaï

JURY

Nadia BOUDI	*PES*	*Faculté des sciences, Université Mohammed V-Rabat*	*Président*
Zine El Abidine GUENNOUN	*PES*	*Faculté des sciences, Université Mohammed V-Rabat*	*Directeur de thèse*
Fouzia OMARY	*PES*	*Faculté des sciences, Université Mohammed V-Rabat*	*Rapporteur / Examinateur*
Jilali ANTARI	*PES*	*Faculté polydisciplinaire de Taroudant Université Ibn Zohr-Agadir*	*Rapporteur / Examinateur*
Abdelalim SADIQ	*PH*	*Faculté des sciences, Université Ibn Tofail-Kénitra*	*Rapporteur / Examinateur*
Soumia ZITI	*PH*	*Faculté des sciences, Université Mohammed V-Rabat*	*Rapporteur / Examinateur*
Mohammed ZIANI	*PH*	*Faculté des sciences, Université Mohammed V-Rabat*	*Rapporteur / Examinateur*
Ahmed DRISSI	*PA*	*École nationale des sciences appliquées de Tanger Université Abdelmalek Essaâdi-Tétouan*	*Invité*

Année Universitaire : **2019 / 2020**

Remerciements

Les travaux présentés dans cette thèse ont été réalisés au sein de la structure de recherche Mathématiques, Statistique et Applications au département des mathématiques à la Faculté des Sciences de Rabat, sous la direction du Professeur Zine El Abidine GUENNOUN, Professeur de l'Enseignement Supérieur à la faculté des Sciences de Rabat de L'université Mohamed V-Rabat.

Je tiens à exprimer ma profonde et sincère gratitude à mon encadrant, le Professeur et chef du département de mathématiques **Zine El Abidine GUENNOUN**, PES à la faculté des sciences de Rabat, pour m'avoir donné l'opportunité de travailler sous sa direction et pour m'avoir fourni des conseils précieux tout au long du voyage qui a abouti à cette thèse. Son dynamisme, sa vision, sa sincérité et sa motivation m'ont profondément inspiré. Il m'a appris la méthodologie pour mener à bien la recherche et présenter les travaux le plus clairement possible. Je lui suis très reconnaissant pour ce qu'il m'a offert, il m'avoir continuellement soutenu dans tous les domaines, qu'ils soient scientifiques ou non. Ce fut un grand privilège et un honneur de travailler et d'étudier sous sa direction. Je n'aurais pas pu imaginer avoir un meilleur directeur de thèse.

Je voudrais exprimer ma gratitude à Madame le Professeur **Nadia BOUDI**, PES à la faculté des sciences de Rabat, pour ce grand honneur qu'elle nous faites, en acceptant de présider ce jury. Elle est une personne de principe et de rigueur.

Je voudrais sincèrement remercier, Madame le Professeur **Fouzia OMARY**,

PES à la faculté des sciences de Rabat, pour ce grand honneur qu'elle nous a fait en acceptant d'être un rapporteur de cette thèse. Ses commentaires opportuns et perspicaces ont été extrêmement utiles pour peaufiner le texte des versions préliminaires de la thèse. Ce n'est pas justifié si je la remercie simplement de m'avoir guidé lors de la rédaction de la version finale. Elle mérite une mention spéciale pour m'avoir accueilli et m'avoir permis de profiter de ses conseils, de ses directives et de son savoir-faire.

Je remercie profondément Professeur **Jilali ANTARI**, PES à la faculté polydisciplinaire de Taroudant, qui a gentiment accepté de faire partie du jury de cette thèse et de servir de rapporteur externe pour cette thèse.

Je dois aussi remercier le Professeur **Abdelalim SADIQ**, PH à la faculté des sciences de Kénitra, pour l'honneur qu'il nous a fait, en acceptant d'être un rapporteur et membre de jury malgré ses multiples occupations.

Je suis également reconnaissant à Madame le Professeur **Soumia ZITI**, PH à la faculté des sciences de Rabat, pour l'intérêt qu'il a manifesté en participant en tant que rapporteur, pour le temps consacré à la lecture de ce travail ainsi que pour ses efforts.

Je dois aussi remercier notre Professeur **Mohammed ZIANI**, PH à la faculté des sciences de Rabat, qui a bien voulu juger une grande partie de ce travail en tant que rapporteur.

Mes remerciements s'adressent en particulier à mon co-auteur, le Professeur **Ahmed DRISSI**, PA à l'école nationale des sciences appliquées de Tanger, pour avoir accepter notre invitation à participer à ce jury. Je le remercie encore pour avoir m'accompagné tout au long de mon travail, notamment pour sa contribution sur nos articles. Je lui exprime ma gratitude pour sa grande disponibilité, ses nombreux conseils et le suivi de mon travail. Je le remercie également pour son amitié et son empathie.

Je n'oublierai jamais un grand homme, le Professeur **Aboubakr LBEKKOURI**, qui nous a quittés. J'ai eu le grand plaisir et l'honore d'être un de ses étudiants,

il s'est toujours montré à l'écoute et disponible pour répondre à nos questions avec gentillesse. Son style d'enseignement et son enthousiasme m'ont fortement impressionné et j'ai toujours gardé avec moi des souvenirs positifs de ses cours. Que le bon Dieu l'ait en sa sainte miséricorde.

Plus important encore, rien de tout cela n'aurait pu se produire sans ma famille et mes amis. Ils ne m'ont pas laissé et je leur en suis éternellement reconnaissant. Cette thèse témoigne de leur amour et de leurs encouragements inconditionnels jusqu'à présent. Merci pour tout !

Liste des publications et des communications scientifiques realisées dans le cadre de ce travail

Publications

1) Khalid Charif, Ahmed Drissi and Zine El Abidine Guennoun. **A Pseudo Random Number Generator Based on Chaotic Billiards**. *International Journal of Network Security* 19, 3 (2017), 479–486.
2) Charif Khalid and Guennoun Zine El Abidine. **A novel image encryption algorithm based on chaotic billiards**. *Journal of Discrete Mathematical Sciences and Cryptography*, December 2019, 1-26.

Communications

1) La conférence internationale MOCASIM2017, Marrakech. Présentation d'une communication intitulée : **A Pseudo Random Number Generator based on Sinai Billiard**.
2) Le Congrès International SM2A2017, Meknès. Présentation d'une communication intitulée : **Un Générateur de nombre Pseudo Aléatoire basé sur le billard de Sinaï**.
3) La conférence internationale MOCASIM2019 Marrakech. Présentation d'une communication intitulée : **Un nouvel algorithme de chiffrement d'images basé sur le Sinaï Billard**.

Résumé

Dans cette thèse, nous avons profité de la marche aléatoire et de l'imprévisibilité du mouvement des particules dans un billard chaotique (Billard de Sinaï), pour concevoir de nouveaux algorithmes cryptographiques à clé secrète de longueur arbitraire.

Malgré ses propriétés chaotiques bien développées, les systèmes des billards n'ont pas pris plus d'attention de la part des cryptographes, parmi les raisons est la forme complexe de ces systèmes.

Notre première contribution propose un nouveau générateur de nombres pseudo-aléatoires, en se reposant sur les systèmes de deux particules. Tandis que la deuxième, est la conception d'un nouveau schéma de chiffrement des images selon l'architecture de confusion-diffusion, en utilisant les systèmes de trois particules. Dans notre concept, nous avons proposé de nouvelles techniques pour résoudre des problèmes de sécurité et profité le maximum possible du chaos offerte par le billard de Sinaï.

Les résultats de simulation montrent que les algorithmes proposés, sont simples à mettre en œuvre, rapides et hautement sécurisés. Ils sont très sensibles à un changement de bit dans la clé, robuste contre les attaques différentielles et passent tous les tests statistiques de validation.

Mots-clés : Générateur de nombres pseudo-aléatoires, NIST, Billard de Sinaï, Confusion, Diffusion, Permutation, Chiffrement des images.

Abstract

In this thesis, we took advantage of the random walk and the unpredictability of the movement of particles in a chaotic billiards (Billiards of Sinai), to design new cryptographic algorithms with secret key of arbitrary length.
Despite its well-developed chaotic properties, billiard systems have not been given more attention by cryptographers, among the reasons is the complex form of these systems.
Our first contribution proposes a new pseudo-random number generator, based on systems of two particles. While the second is the design of a new image encryption scheme according to the confusion-diffusion architecture, using three-particle systems. In our concept, we have proposed new techniques to solve security problems and took full advantage of of the chaos offered by the Sinai billiards. The simulation results show that the proposed algorithms are simple to implement, fast and highly secure. They are very sensitive to a bit change in the key, robust against differential attacks, and pass all statistical validation tests.

Keywords : Pseudo-random number generator, NIST, Sinaï Billiard, Confusion, Diffusion, Permutation, Image encryption.

Abréviations

GNPA	Générateur de nombre pseudo-aléatoire.
GBPAC	Générateur de bit pseudo-aléatoire Cryptographique.
DH	La distance Hamming.
DES	Le Data Encryption Standard .
AES	L'Advanced Encryption Standard.
IDEA	International Data Encryption Algorithm ;
XOR	Le ou exclusive.
ECB	Electronic Codebook.
CFB	Ciphertext feedback.
OFB	Output Feedback.
CBC	Cipher Block Chaining.
SD	Système dynamique.
SC	Système chaotique.
BS	Billard de Sinaï.
EL	L'exposant de Lyapunov.
ASCII	L'American Standard Code for Information Interchange.
NCPR	The number of changing pixel rate.
UACI	The unified averaged changed intensity.
NIST	National Institute of Standards and Technology.
CD	Le nombre de rondes de confusion-diffusion.

Notations

u_i	Le $i^{\text{ème}}$ état d'un GNPA.
SF	La séquence aléatoire $SF = u_1\, u_2\, \ldots$.
K	La clé secrète.
L	La longueur du clé.
$[\,]$	La partie entière.
$\overline{b}$	L'inverse du bit b.
A_i	Le point de collision d'une particule à la $i^{\text{ème}}$ collision.
$\overrightarrow{v_i}$	La vitesse d'une particule à la $i^{\text{ème}}$ collision.
$\overrightarrow{N_i}$	Le vecteur normale unitaire au point A^i.
S^i	L'état de la particule à la $i^{\text{ème}}$ collision.
$pro_{(x,y)}(S^i)$	La projection de l'état S^i sur le plan (Oxy).
$D(O, D_n)$	La distance de O à D_n.
$x \longleftarrow y$	Expression d'affectation, écrire la valeur y dans la variable x.
Δ_n	Le discriminant.
f	La fonction de transition.
P	Permutation.
$Pr(B)$	La probabilité de la réalisation de l'événement B.
$\bigoplus$	L'opération du ou exclusive .
$\|\|$	L'opération Concaténation.
H	La fonction d'entropie.
h_{KS}	L'entropie de Kolmogorov-Sinaï.

Table des matières

Table des figures

Liste des tableaux

Introduction générale

Aujourd'hui, l'échange d'informations privées sur Internet est devenu une habitude quotidienne. Surtout, après le développement rapide des technologies de l'information et des réseaux de communication, la majorité des individus et les organisations tels que les entreprises et les gouvernements sont désormais connectés à Internet. Pendant cette période où Internet fournit une communication essentielle entre des milliards de personnes, un flux grandissant de données est en circulation sur le réseau public. En effet, la sécurité des données est devenue une nécessité incontournable et de plus en plus pertinente pour les organisations modernes. Par exemple, les informations sensibles et confidentielles contenues dans une conversation téléphonique, un message électronique ou un fichier transféré peuvent être exploités ou modifiés, s'il tombe entre les mains d'un adversaire.

Avec l'avancement des technologies, la sécurité de l'information est sortie de son rôle d'origine dans le domaine militaire et les services secrets. Aujourd'hui, elle est devenue un problème extrêmement important pour chaque utilisateur. La plupart des applications nécessitent un niveau de sécurité pour garantir la protection des données dans différents canaux de communication. En outre, la confidentialité, l'authenticité et l'intégrité des informations stockées ou transmises sont les services essentiels à assurer, ce qui oblige les chercheurs à proposer de nouvelle technique et développer des méthodes pour résoudre les problèmes de sécurité.

Depuis l'antiquité jusqu'à nos jours, la science de la cryptographie, détient toujours les solutions fondamentales de tous les problèmes de sécurité posés par les technologies modernes de l'information et de la communication. La cryptographie est une science qui étudie comment protéger les données en utilisant des algorithmes, à titre d'exemple, les systèmes de chiffrement tels que les algorithmes de chiffrement d'images qui sont utilisés sur plusieurs applications de communications. Mais aussi les signatures numériques ainsi que les fonctions de hachage pour assurer l'intégrité des données et la protection des mots de passe. Aussi les générateurs de nombres pseudo-aléatoires (GNPA) qui jouent un rôle très important dans la génération des clés pour le chiffrement des données. Pratiquement, la cryptographie a intégré la vie quotidienne de chaque personne.

Les algorithmes de chiffrement performants ont la propriété de mélange et de sensibilité aux conditions initiales, qui correspond aux propriétés de confusion-diffusion mentionnées par Shannon. En effet, ces deux propriétés sont considérées comme les concepts fondamentaux des systèmes cryptographiques. De même, plusieurs chercheurs ont souligné l'existence d'une similitude entre les propriétés des systèmes chaotiques tels que : l'ergodicité, le mélange, le caractère aléatoire, l'imprévisibilité et la sensibilité aux conditions initiales, d'une part, et les besoins des systèmes cryptographiques de l'autre part. En conséquence, une telle relation étroite a conduit à un nouveau domaine de recherche appelé cryptographie à base du chaos. Jusqu'à aujourd'hui, plusieurs algorithmes cryptographiques à base du chaos ont été proposés, en implémentant des systèmes chaotiques différents tels que : Logistique, Henon généralisé, Tinkerbell, Standard, Lorenz.

Au cours des dernières décennies, les billards chaotiques sont devenus l'un des domaines de recherche les plus actifs. Cela a commencé avec un article fondateur du mathématicien russe Yakov Sinaï en 1970 où il a proposé une table de billard à $2D$ (billard de Sinaï) pour simplifier l'étude du comportement du gaz de Lorentz. La théorie a été bien développée au sein de la théorie des systèmes

dynamiques après une explosion d'articles dans des revues de mathématiques et de physiques. Il a été prouvé que le billard de Sinaï a le plus haut degré de chaos (les propriétés chaotiques les plus fortes).

Malgré les propriétés chaotiques intéressantes des billards, leurs applications dans le domaine de la cryptographie restent négligeables. Dans ce travail, nous avons proposé une nouvelle approche, en se basant sur l'application de billard de Sinaï (BS), afin d'améliorer les schémas de communication existants. Les propriétés offertes par le BS, sont bien exploitées pour concevoir et analyser systématiquement de nouveaux schémas de chiffrement en introduisant de nouvelles techniques cryptographiques.

La première contribution de la thèse est la proposition d'un GNPA [19] efficaces, robustes et simple à mettre en œuvre. L'algorithme proposé a deux paramètres d'entrée, une clé secrète (graine) de longueur arbitraire et un entier qui est la longueur de la suite de sortie. En fait, dans ce concept, nous nous reposons sur la marche aléatoire de deux particules, où les coordonnées des points de collision avec le carré du billard sont convertis en nombres aléatoires. En outre, entre deux nombres générés consécutivement, les particules effectuent un nombre variable de collisions. Une telle technique permet au générateur d'être imprévisible de telle sorte qu'il est impossible de prédire les suites à venir sans connaitre la clé, et très sensible sensible au changement d'un bit de la clé. Enfin, le générateur est validé par une analyse statistique, où l'algorithme montre un niveau de sécurité bien élevé.

La deuxième contribution est un nouveau schéma de chiffrement des images numériques [20]. Dans ce concept, nous avons utilisé les systèmes de trois particules et reposé sur plusieurs rondes de confusion-diffusion. En effet, Le mécanisme de diffusion basé sur nombres aléatoires générés par le GNPA déjà proposé, et la confusion pixels de l'image est effectuée par une permutation générée par le système d'une particule. Les résultats de la simulation et de l'analyse de sécurité montrent que l'algorithme de chiffrement proposé présente de bonnes perfor-

mances en termes de sécurité, de robustesse et de vitesse de chiffrement bien élevée.

Dans le chapitre 1, nous donnons les connaissances préliminaires sur la cryptographie et la théorie du chaos. Tandis que dans le chapitre 2, nous présentons la conception d'un GNPA basé sur le système d'un billard chaotique, y compris la validation par les tests de sécurité. Le chapitre 3 est consacré à une proposition d'un schéma de chiffrement des images en se basant sur les systèmes du billard de Sinaï, une analyse des performances de sécurité est effectuée pour le schéma proposé. Enfin, les résultats de la thèse sont résumés dans le conclusion à la fin de la thèse où les perspectives pour la recherche future sont également définies.

Chapitre 1

L'état de l'art

Ce chapitre introduit les notions de base de la cryptologie et les systèmes dynamiques. Cette introduction exposera aussi le fonctionnement des billards chaotiques, en particulier le billard de Sinaï (BS). Le chapitre terminera par mettre l'accent sur les propriétés qui semblent pertinentes pour son application à la cryptographie, différents types de cryptosystèmes chaotiques, leurs principales caractéristiques et les meilleures pratiques et directives pour concevoir de bons algorithmes cryptographiques.

1.1 La cryptologie

La cryptologie est une science du secret qui se décompose en cryptographie et cryptanalyse. La cryptographie est une discipline qui regroupe les domaines des mathématiques, l'informatique, la théorie de l'information et les études d'ingénierie, dont l'objet est l'étude des techniques permettant d'assurer les services de confidentialité, d'intégrité et d'authenticité. Par contre, la cryptanalyse est une discipline qui consiste à trouver des faiblesses dans les algorithmes cryptographiques et les utiliser pour acquérir des connaissances concernant la communication sécurisée. Dans ce contexte, la confidentialité est préservée à l'aide des cryptosystèmes.

Cryptosystème [103] est un système de chiffrement composé de $5-$uplet $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ ayant les propriétés suivantes :

1) $\mathcal{P}$ est l'ensemble des textes clairs.
2) $\mathcal{C}$ est l'ensemble des textes chiffrés ou cryptogramme.
3) $\mathcal{K}$ est l'espace des clés.
4) $\mathcal{E} = \{E_{k_e}; k_e \in \mathcal{K}\}$ est une famille de fonctions $E_{k_e} : \mathcal{P} \longrightarrow \mathcal{C}$, ses éléments sont les fonctions de chiffrement, $\mathcal{D} = \{D_{k_d}; k_d \in \mathcal{K}\}$ est une famille de fonctions $D_{k_d} : \mathcal{C} \longrightarrow \mathcal{P}$; ses éléments sont les fonctions de déchiffrement et à chaque clé $k_e \in \mathcal{K}$, il existe une clé $k_d \in \mathcal{K}$ une fonction D_{k_d} et une fonction E_{k_e} telles que $D_{k_d}(E_{k_e}(x)) = x$, pour tout message x de $\mathcal{P}$.

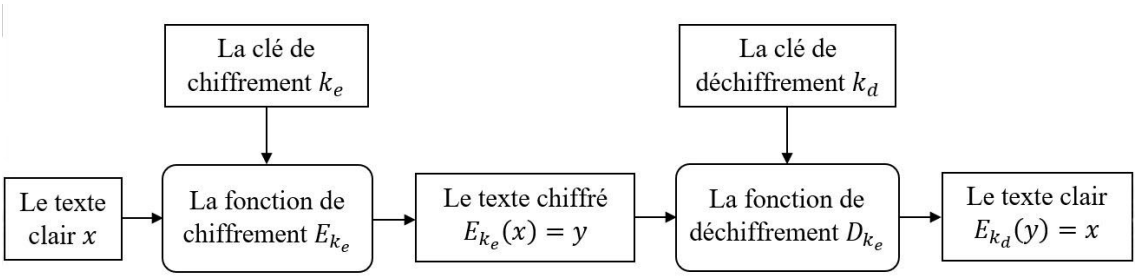

FIGURE 1.1 – Schéma d'un Cryptosystème

La cryptographie peut être divisée en deux champs principaux selon les clés des cryptosystèmes, la cryptographie symétrique et la cryptographie asymétrique. En cryptographie symétrique, l'expéditeur et le destinataire doivent se mettre d'accord préalablement sur une clé k qui doit être gardée secrète et utiliser exactement la même clé $k = k_e = k_d$ pour chiffrer et déchiffrer des données. Le DES [11] et le AES [45] sont parmi les cryptosystèmes symétriques les plus utilisés. Tandis que dans la cryptographie asymétrique, chaque entité de communication possède un couple de clés distinctes (clé privée k_s, clé publique k_p). Ces deux clés sont liées mais presque impossible de trouver la clé k_s à partir de la clé k_p. Les données reçues par l'entité sont chiffrées par $k_e = k_p$ qui est connue de tous, et le déchiffrement des données s'effectue par $k_d = k_s$ qui n'est connue que de l'entité. Les

cryptosystèmes asymétriques les plus connus sont RSA [90], ElGamal [35] et Rabin [89].

La cryptographie à clé symétrique est classifiée en deux types de chiffrements : les chiffrements par blocs et les chiffrements par flux.

Le chiffrement par flux est un chiffrement opérant sur un flux de données, où le texte clair est combiné à un flux de clés, pour donner un flux du texte chiffré. Le chiffrement de Vernam (one-time-pad) est parmi les cryptosystèmes de chiffrement par flux les plus connus.

Cryptosystéme : One-Time-Pad [103]

Soit un entier $n \geq 1$, soit$\mathcal{P} = \mathcal{C} = \mathcal{K} = \{0,1\}^n$, pour une clé $K = (k_1, k_2, \cdots, k_n)$ et un texte clair $M = (m_1, m_2, \cdots, m_n)$, la fonction de chiffrement E_K est définie par :

$$E_K(M) = (m_1 \oplus k_1, m_2 \oplus k_2, \ldots, m_n \oplus k_n),$$

la fonction de déchiffrement D_K est identique à E_K, si $C = (c_1, c_2, \ldots, c_n)$ donc :

$$D_K(M) = (c_1 \oplus k_1, c_2 \oplus k_2, \ldots, c_n \oplus k_n).$$

Comme indiqué, le texte clair est Xoré avec un flux de clé pour avoir le texte chiffré. En fait, le flux de clé est généré par les générateurs des nombres pseudo-aléatoires cryptographiques.

Le chiffrement par bloc est un type de chiffrement qui découpe le texte clair $M = M_1M_2M_3 \ldots M_n$ en blocs de taille fixe $\mid M_1 \mid \geq 64$, les blocs sont ensuite chiffrés les uns après les autres en utilisant la même clé secrète K pour générer des blocs de texte chiffré $C = C_1C_2C_3 \ldots C_n$, tels que $C_i = E_k(M_i)$ pour $i \leq n$, le déchiffrement s'effectue comme suit $M_i = D_k(C_i)$. Pour masquer les modèles qui existent dans les données chiffrées, des modifications sont introduite pour mélanger des blocs de texte en clair avec les blocs de texte chiffré. En effet, ces modifications s'appellent les modes d'opération de chiffrement par bloc. Le chiffrement selon le mode d'enchaînement des blocs (CBC) est $C_i = E_K(C_{i-1} \oplus M_i)$ avec C_0 est un

vecteur d'initialisation choisi au hasard, le déchiffrement $M_i = D_K(C_i) \oplus C_{i-1}$. Le Chiffrement à rétroaction (CFB) est très similaire au CBC où $C_i = E_K(C_{i-1}) \oplus M_i$, bien que le déchiffrement est $M_i = E_K(C_{i-1}) \oplus C_i$. Le chiffrement à rétroaction de sortie (OFB) est similaire à CBC et CFB où les résultats du chiffrement du bloc précédent sont utilisés dans le chiffrement du bloc suivant, le chiffrement selon ce mode est $C_i = O_i \oplus M_i$ où $O_i = E_K(O_{i-1})$ et $O_0 = E_K(C_0)$ alors que le déchiffrement est $M_i = O_i \oplus C_i$ où $O_i = E_K(O_{i-1})$ et $O_0 = E_K(C_0)$.

La cryptographie moderne est basée sur le principe de kerchoffs [55], le principe selon lequel la sécurité d'un système cryptographique doit dépendre exclusivement de la clé et non de la sécurité d'une autre partie du système.

Aujourd'hui, la conception des cryptosystème est basée sur le principe de Kerchoffs. Il est déconseillé d'utiliser des algorithmes non standardisés et conçu en secret, seuls les algorithmes testés publiquement par des cryptanalystes doivent être utilisés. En général, le cryptanalyste met à l'épreuve les algorithmes cryptographiques pour déterminer les faiblesses et des fuites d'informations par des attaques selon les données disposées. il y a plusieurs types d'attaques rangés du plus difficile au plus simple [16, 75, 103] :

- **Attaque sur texte chiffré seul** (ciphertext-only) : le cryptanalyste possède des exemplaires du cryptogramme , il peut faire des hypothèses sur les messages originaux qu'il ne possède pas.
- **Attaque à texte clair connu** (known-plaintext attack) : le cryptanalyste possède des messages ou des parties de messages en clair ainsi que les versions chiffrées. La cryptanalyse linéaire fait partie de cette catégorie.
- **Attaque à texte clair choisi** (chosen-plaintext attack) : le cryptanalyste possède des messages en clair, il peut créer les versions chiffrées de ces messages par un accès à l'algorithme de chiffrement. La cryptanalyse différentielle est un exemple d'attaque à texte clair choisi.
- **Attaque à texte chiffré choisi** (chosen-ciphertext attack) : le cryptanalyste possède des messages chiffrés et demande la version en clair de certains

de ces messages pour mener l'attaque.

Les cryptanalystes évaluent tous les types de sécurité d'un cryptosystème tels que la sécurité informatique, la sécurité prouvable ou la sécurité inconditionnelle.

Le mathématicien américain Claude Shannon [94] a formulé un principe similaire au principe de Kerckhoff (peut-être basé sur le même principe) connu aujourd'hui sous le nom de maxime de Shannon. il est largement adopté par les cryptographes.

Selon shannon, un schéma de chiffrement est parfaitement secret si on ne peut obtenir aucun information sur le texte clair en observant le texte chiffré. Autrement dit, si pour chaque distribution M (resp. C) sur l'ensemble des texte clair (resp. chiffré), chaque texte clair m et chaque texte chiffré c pour lequel $Pr[C = c] > 0$, on a $Pr[M = m \mid C = c] = Pr[M = m]$.

Shannon a identifié deux propriétés de fonctionnement d'un schéma de chiffrement sécurisé : la *confusion* et la *diffusion*. En termes simples, la confusion rend la relation entre la clé et le texte chiffré aussi complexe que possible, et ainsi le texte chiffré ne donne aucun information sur le texte clair. En effet, la confusion est assuré en s'étendant une partie du texte clair sur plusieurs partie du texte chiffré, par exemple lorsqu'un seul bit du texte clair est modifié, il doit affecter l'ensemble du texte chiffré. Alors que la diffusion est un processus qui disperse les données afin que la redondance dans le texte clair est dissipée dans le texte chiffré. Dans un cryptosystème avec une propriété de diffusion, un bit d'entrée doit dépendre des bits de sortie de manière imprévisible ou pseudo-aléatoire. Cela signifie que, pour une entrée choisie au hasard, si l'en retourne le bit b_i, alors la probabilité que le bit b_j de sortie change soit d'une probabilité égale à $1/2$, pour tout i et j.

Depuis la publication des articles [93, 94] à aujourd'hui, les deux propriétés de la confusion et de la diffusion restent les principaux directeurs pour la conception des algorithmes cryptographiquement sûrs. Pour garantir ces deux principes dans d'un schéma de chiffrement, de nombreux cryptosystèmes utilisent plusieurs cycles d'une combinaison de substitution ou de masquage des bits par

des générateurs de nombres pseudo-aléatoires selon de chiffrement de Vernam, et par la manipulation de l'ordre des éléments en utilisant des permutations.

1.2 Le chaos

Cette section donne des définitions et des notions préliminaires sur les systèmes dynamiques (SD), les systèmes chaotiques (SC) et leurs propriétés. Elle se terminera par la définition de l'exposant de Lyapunov et l'entropie de Kolmogorov-Sinaï, les deux grandeurs qui permettent de mesurer le degré de "sensibilité aux conditions initiales".

1.2.1 Systèmes dynamiques

En mathématiques, un SD est un système dans lequel une fonction décrit la dépendance temporelle d'un point dans un espace d'une manière déterministe. En général, un système dynamique se compose d'un ensemble d'états permettant de décrire l'évolution d'un point au cours du temps.
Un système dynamique [14] est un triplet (E, T, φ) où S est l'espace des phases, T est un domaine temporel et $\varphi : E \times T \longrightarrow E$ est un opérateur décrivant l'évolution temporelle d'un système tel que :

$$\begin{cases} \varphi(x, 0) = x \\ \varphi\big(\varphi(x, t_1), t_2\big) = \varphi(x, t_1 + t_2) \end{cases}$$

Pour tout $x \in E$ et $t_1, t_2 \in T$.
Un système dynamique en temps continu peut être modélisé mathématiquement par un système d'équations différentielles :

$$\begin{cases} x(t) = \frac{dx(t)}{dt} = f(x(t)) \\ x(0) \text{ condition initial} \end{cases}$$

avec $f : E \longrightarrow E$ une application d'un espace E, dit espace des phases, dans lui-même, et $t \in \mathbb{R}$.

Un système dynamique discret est un système d'équations aux différences finies, où l'évolution des variables est mesurée par des étapes discrètes de la forme :

$$\begin{cases} x_{k+1} = f(x_k) \\ x_0 \text{ condition initial} \end{cases}$$

avec $f : E \longrightarrow E$ une application d'un espace d'états dans lui-même et x_t est l'état du système à l'instant discrète k avec $k \in \mathbb{N}$, ou $k \in \mathbb{Z}$ dans le cas où f est inversible. On note $f^n(x)$ la composition de la fonction avec elle-même une n fois en un point x, on peut écrire aussi :

$$f^0(x) = x,\ f^1(x) = f(x),\ f^2(x) = f\big(f(x)\big),\ f^n(x) = f\big(f^{n-1}(x)\big)$$

et

$$x_0, \quad x_1 = f(x_0), \quad x_2 = f^2(x_0), \quad x_k = f^k(x_0)$$

L'orbite (positive) de x par le système dynamique de f est l'ensemble $\{f^n(x)/n \in \mathbb{N}\}$. Si f est bijective, l'orbite de x est l'ensemble $\{f^n(x)/n \in \mathbb{Z}\}$. Dans un système dynamique, un point x est dit point fixe d'un système dynamique, si $f(x) = x$, et le point est dit point périodique, s'il existe $p \in \mathbb{N}^*$ tel que :

$$f^p(x) = x \tag{1.1}$$

la période du point périodique x est le plus petit entier $p \geq 1$ tel que 1.1 est vérifiée. La fonction f possède un cycle d'ordre p s'il existe x tel que les itérés de f reviennent à la valeur de départ au bout de p itérations. Une partie A de E est dite invariante par f (ou f-invariante) si $f(A) \subset A$. Un point a est dit un point

attractif s'il existe un ouvert I contenant a tel que si $x \in I$, alors

$$\lim_{n \to +\infty} f^n(x) = a$$

1.2.2 Systèmes dynamiques chaotiques

Le chaos est utilisé pour désigner l'absence d'ordre et d'organisation, il a été observé dans divers phénomènes naturels [60, 77, 101, 105]. Le chaos a été adopté par le mathématicien James Yorke dans les années 60 et s'est popularisé après, ce qui donne à la théorie du chaos le statut de champs de recherche. En 1974 le mathématicien Anglais Lord Robert a publié un article [77] pour désigner le comportement déterministe de quelques équations différentielles non linéaires [80]. Le comportement chaotique apparait spécifiquement dans un cas particulier des systèmes dynamiques, à savoir, la caractéristique la plus importante des systèmes dynamiques chaotiques est la sensibilité aux conditions initiales. En général, un système dynamique $x_{t+1} = f(x_t)$ où $f : E \longrightarrow E$ est dit chaotique si les conditions suivantes sont vérifiés :

- L'application $f : E \longrightarrow E$ est sensible aux conditions initiales

  $\exists \delta > 0, \forall x \in E, \forall V$ voisinage de $x, \exists y \in V, n > 0$ telle que $|f^n(x) - f^n(y)| > \delta$

- La fonction f est topologiquement transitive, c'est-à-dire pour chaque deux ensembles ouverts non vides $U, V \subset E$, il existe $n \geq 0$ tel que

  $$f^n(U) \cap V \neq \emptyset.$$

Les systèmes dynamiques chaotiques sont définies par des équations non-linéaires déterministes. Ils ont la propriété que toutes les trajectoires divergent rapidement de n'importe quel point de l'espace d'état.

1.2.3 Propriétés fondamentales des systèmes chaotiques

Les systèmes dynamiques chaotiques suivent certaines lois particulières des systèmes déterministes non linéaires. Le chaos apparaît lors de l'évolution du système, il a un aspect désordonnée qui satisfait certains critères mathématiques. Il existe un ensemble de propriétés qui résument les caractéristiques observées dans les systèmes chaotiques [33, 41]. Les plus appropriés sont :

- Sensibilité à l'état initial : également appelée effet de papillon, où un changement négligeable dans son état initial peuvent générer des états complètement différents.
- Mélange topologique : cela signifie que le système évoluera dans le temps de sorte que toutes les régions d'états soient transformées avec toute autres régions donnée.
- Apériodicité : le système évolue sur une orbite qui ne se répète jamais sur lui-même, c'est-à-dire que ces orbites ne sont jamais périodiques.
- Orbites périodiques denses : cela signifie que le système suit une dynamique qui peut approcher arbitrairement de près chaque état asymptotique possible.
- Ergodicité : les mesures statistiques des variables donnent des résultats similaires, qu'elles soient effectuées dans le temps ou dans l'espace.
- Auto-similitude : l'évolution du système, dans le temps ou dans l'espace, montre la même apparence à différentes échelles d'observation. Cette caractéristique fait apparaître le système auto-répétitif à différentes échelles d'observation.

1.2.4 Exposant de Lyapunov

Les exposants de Lyapunov (EL) sont des grandeurs qui mesurent et quantifient le taux de divergence entre deux orbites issues de deux conditions initiales différentes [15]. Dans le cas des systèmes discrets, l'EL est le taux de sé-

paration moyenne des trajectoires issues de condition initiales différents et permettent de quantifier la sensibilité aux conditions initiaux d'un système chaotique (SC) [41,46].

Soit le système dynamique non-linéaire suivant :

$$x_{k+1} = f(x_k) \text{ où } f : E \longrightarrow E$$

Pour deux conditions initiales x_0 et x'_0, La trajectoire issue de la condition initiale x_0 (respectivement. x'_0) est $x_k = f^k(x_0)$ (respectivement. $x'_k = f^k(x'_0)$). En supposant que les deux trajectoires x_k et x'_k s'écarte exponentiellement après k itérations, alors :

$$|x_k - x'_k| = |x_0 - x'_0| exp(k\lambda)$$

λ correspond au taux de divergence des deux trajectoires dont l'expression est la suivante :

$$\lambda = \frac{1}{k} \ln |\frac{x_k - x'_k}{x_0 - x'_0}|$$

Pour x_0 et x'_0 très proche, leurs différences $\epsilon = |x_0 - x'_0|$ tend vers 0, lorsque k tend vers l'infini, on obtient :

$$\lambda_L = \lim_{k\to\infty} \lim_{\epsilon\to 0} \ln |\frac{x_k - x'_k}{x_0 - x'_0}|$$

cela donne :

$$\begin{aligned}
\lambda_L &= \lim_{k\to\infty} \lim_{\epsilon\to 0} \sum_{i=0}^{k-1} \ln |\frac{x_k - x'_k}{x_0 - x'_0}| && (1.2)\\
&= \lim_{k\to\infty} \lim_{\epsilon\to 0} \sum_{i=0}^{k-1} \ln |\frac{x_{i+1} - x'_{i+1}}{x_i - x'_i}| && (1.3)\\
&= \lim_{k\to\infty} \lim_{\epsilon\to 0} \sum_{i=0}^{k-1} \ln |\frac{f(x_i) - f(x'_i)}{x_i - x'_i}| && (1.4)\\
&= \lim_{k\to\infty} \sum_{i=0}^{k-1} \ln |\frac{df(x_i)}{dx_i}| && (1.5)
\end{aligned}$$

λ_L est appelé l'exposant de Lyapunov. Il montre la vitesse de divergence des trajectoires les plus proches pendant le temps discret k. Si $\lambda_L > 0$ pour une valeur particulière à une condition initial ou un paramètre de contrôle, alors les trajectoires voisines divergent et l'évolution est sensible aux conditions initiales et donc le système est chaotique.

La relation 1.2 se généralise au système de dimension $n > 1$, l'EL possède n composants $\lambda_L^{(i)}(i = 1, 2, \ldots, n)$, chacun d'entre eux mesure le taux de divergence suivant un des axes de l'espace de phase. Pour $x_k = f^k(x_0)$ avec $x_k = [x_k^{(1)} \ldots x_k^{(n)}]^T \in \mathbb{R}^n$ et $f = [f_1 \ldots f_n]^n$. L'EL est calculé par l'expression suivante :

$$\lambda_L^{(i)} = \lim_{k \to \infty} \frac{1}{k} \ln |\lambda_i(j_k \ldots j_1)|, i = 1, \ldots, n$$

où $\lambda_i(j_k \ldots j_1)$ représente la $i^{\text{ème}}$ valeur propre des produits des matrices $(j_k \ldots j_1)$. Les J_k sont les matrices jacobiennes des dérivées partielles du premier ordre de la fonction f au point x_k :

$$Df(x_i) = \begin{pmatrix} \frac{\partial f_1(x_i)}{\partial x^{(1)}} & \cdots & \frac{\partial f_1(x_i)}{\partial x^{(n)}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(x_i)}{\partial x^{(1)}} & \cdots & \frac{\partial f_n(x_i)}{\partial x^{(n)}} \end{pmatrix}$$

Un système dynamique multidimensionnel est chaotique, s'il admet au moins un exposnat de Lyapunov positif.

L'EL mesure l'efficacité des systèmes cryptographiques. Plus la valeur de l'exposant est élevée, plus le nombre des itérations est petit pour atteindre le degré requis de diffusion et de confusion des informations. Pour tenir compte de la précision de l'observation, des informations plus utiles donnent l'entropie de Kolmogorov-Sinaï.

1.2.5 Entropie de Kolmogorov-Sinaï

L'entropie Kolmogorov-Sinaï h_{ks} est apparue dans l'article de Kolmogorov [59]. C'était une époque où Kolmogorov était intéressé par le travail sur plusieurs problèmes de la théorie de l'information. Si on considère la trajectoire $x(t) = (x_1(t), x_2(t), \ldots, x_N(t))$ et partitionnons l'espace des phases en n hypercubes de côté ϵ. Soit $P_{i_0,i_1,\ldots}$, la probabilité conjointe que le point $x(0)$ se trouve dans la $i_0^{\text{ème}}$ cellule, $x(\tau)$ dans la $i_1^{\text{ème}}$ cellule, ..., et $x(n\tau)$ se trouve dans la cellule $i_n^{\text{ème}}$. Selon l'entropie de Shannonn, soit K_n définie par :

$$K_n = -\sum_{i_0 i_1 \ldots i_n} P_{i_0,i_1,\ldots,i_n} \ln P_{i_0,i_1,\ldots,i_n}$$

K_n la mesure de la quantité d'informations nécessaires pour spécifier la trajectoire à l'intérieur d'une précision ϵ. Il s'ensuit que $K_{n+1} - K_n$ est la quantité supplémentaire d'informations requise pour spécifier dans quelle cellule $x(n\tau + \tau)$ elle tombera. L'entropie h_{ks} est définie comme suit [8] :

$$\begin{aligned} h_{ks} &= \lim_{\tau\to 0}\lim_{\epsilon\to 0}\lim_{N\to\infty}\sum_{n=0}^{N-1} K_{n+1} - K_n \\ &= \lim_{\tau\to 0}\lim_{\epsilon\to 0}\lim_{N\to\infty}\sum_{i_0 i_1 \ldots i_n} P_{i_0,i_1,\ldots,i_n} \ln P_{i_0,i_1,\ldots,i_n} \end{aligned}$$

L'entropie h_{ks} est définie comme le taux moyen de perte d'informations. La quantité $h_{ks} = 0$ pour les systèmes non chaotiques, c'est-à-dire qu'il n'y a pas de perte d'information, car initialement les points proches sur une trajectoire restent proches les uns des autres à mesure que le temps d'évolution. Cependant, $h_{ks} > 0$ pour les SCs, car les points proches initialement se séparent de façon exponentielle en moyenne, et donc les probabilités conjointes pour les occupations de cellules diminuent exponentiellement avec le temps. Donc, l'entropie h_{ks} est utile pour distinguer le comportement régulier du comportement chaotique.
Bien que h_{ks} et EL sont deux critères différents et aient été définies indépendamment, déterminent le même type de comportement chaotique. En effet, on peut

s'attendre à ce qu'il existe une relation entre eux dans un certain sens. En fait, Ruelle a prouvé une relation entre les deux mesures. Il a montré [34] que h_{ks} est inférieure ou égale à la somme des exposants positifs. L'inégalité inverse a été prouvée par Pesin [87] dans certaines conditions restreintes.

1.3 Les Systèmes des billards

Les billards sont des modèles mathématiques pour de nombreux phénomènes physiques dans lesquels une ou plusieurs particules se déplacent dans un récipient et entrent en collision avec ses parois et / ou les unes avec les autres. Les propriétés dynamiques de tels modèles sont déterminées par la forme des parois du conteneur et peuvent être totalement chaotiques. les billards chaotiques comprennent les modèles classiques de balles durement étudiés par L. Boltzmann au XIXe siècle, le gaz de Lorentz introduit pour décrire l'électricité en 1905. La théorie mathématique du billard chaotique est née en 1970 lorsque Ya. Sinaï a publié son article précurseur [98]. Mais au cours de ces années, il a grandi et s'est développé à une vitesse remarquable et est devenu un domaine bien établi et florissant au sein de la théorie moderne des systèmes dynamiques et de la mécanique statistique.

En général, la dynamique du billard est définie comme un domaine dans les régions planaires $\mathcal{D} \in \mathbb{R}^2$, dont les frontières sont lisses ou par morceau lisses. Un système de billard correspond au mouvement libre d'une particule ponctuelle dans $\mathcal{D}$ avec des réflexions spéculaires sur la frontière $\partial\mathcal{D}$. La frontière $\partial\mathcal{D}$ est une union finie de courbes lisse :

$$\partial\mathcal{D} = \Gamma = \bigcup_{i}^{r} \Gamma_i$$

Précisément, chaque courbe Γ_i est définie par une carte $f_i : [a_i, b_i] \longrightarrow \mathbb{R}^2$. Nous appelons $\mathcal{D}$ une table de billard et des parois en $\Gamma_1, \ldots, \Gamma_r$ ou des composants de

$\partial\mathcal{D}$. Si $f_i(a_i) = f_i(b_i)$, alors nous appelons Γ_i un arc et notons $\partial\Gamma_i = \{f_i(a_i), f_i(b_i)\}$. Si $f_i(a_i) = f_i(b_i)$, alors Γ_i est une courbe fermée, qui peut être défini sur un cercle plutôt que $[a_i, b_i]$, et nous posons $\partial\Gamma_i = \varnothing$.

Les composantes limites Γ_i ne peuvent se croiser qu'à leurs extrémités, c'est à dire.

$$\Gamma_i \bigcap \Gamma_j \subset \partial\Gamma_i \bigcup \partial\Gamma_j \text{ pour } i \neq j.$$

nous mettons

$$\Gamma_* = \partial\Gamma_1 \bigcup \ldots \bigcup \partial\Gamma_r \text{ et } \tilde{\Gamma} = \Gamma \setminus \Gamma_*.$$

Nous appelons les points singuliers Γ_* de $\mathcal{D}$ (Fig 1.2. s_1 et s_2) et $\tilde{\Gamma}$ les points réguliers de la frontière.

L'orientation de chaque Γ_i est fixé de sorte que $\mathcal{D}$ se trouve à gauche de Γ_i. Donc, chaque Γ_i est paramétré par sa longueur d'arc, ainsi les vecteurs tangents unitaires $\|\overrightarrow{N_i}\| = 1$. En conséquence, nous distinguons trois types de murs selon la forme courbure de Γ_i comme indiqué dans la figure1.2 : parois plats (Γ_2 et Γ_4), parois de dispersion (Γ_1) et parois de focalisation(Γ_3).

la dynamique sur une table de billard ne sera pas une tâche simple à construire, car il y aura plusieurs cas où la construction échouera et la trajectoire d'une particule de billard ne pourra pas être définie. Soit $A \in \mathcal{D}$ la position de la particule en mouvement et $v \in \mathbb{R}^2$ son vecteur vitesse. Bien sûr, $A = A(t)$ et $v = v(t)$ sont des fonctions du temps $t \in \mathbb{R}$. Lorsque la particule se déplace à l'intérieur du billard, elle maintient une vitesse constante $\|v\| = cst$. Lorsque la particule entre en collision avec la partie régulière de la frontière, c'est-à-dire $\tilde{\Gamma}$, son vecteur vitesse se reflète instantanément sur la tangente à T au point A. Ceci est spécifié par la règle classique "*l'angle d'incidence est égal à l'angle de réflexion par rapport à la normale au point de collisions*" (Fig. 1.2), et il peut être exprimé par :

$$v^+ = v^- - 2(v^-.n)n,$$

où v^+ et v^- font respectivement référence aux vitesses postcollisionnelle et précollisionnelle et n désigne le vecteur normal unitaire à $\tilde{\Gamma}$ au point A.
Si la particule heurte un point singulier A, c'est-à-dire $A \in \Gamma_*$, elle s'arrête et son mouvement ne sera plus défini au-delà de ce point. C'est l'une des complications qui rendent difficile l'analyse de la dynamique du billard.

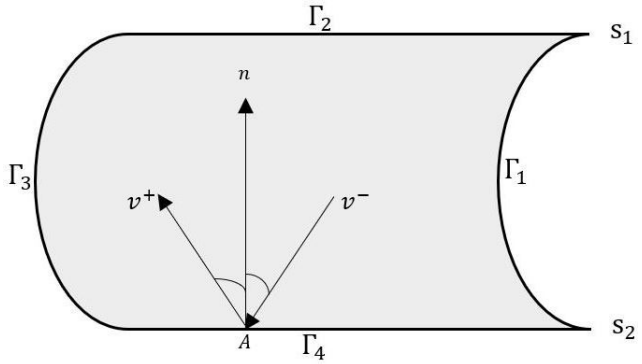

FIGURE 1.2 – Une table de billard.

1.3.1 Le billard de Sinaï

Le billard Sinaï a été proposé par Yakov G. Sinaï en 1970 [98], est un carré plat de côté $2a$ et une barrière circulaire de rayon $r < a$ comme indiqué sur la figure 1.3. Il est surgi pour simplifier l'étude du comportement de deux disques (molécule de gaz) rebondissant par collisions mutuelles dans un carré.
Le billard est une zone plane $\mathcal{D} \in \mathbb{R}^2$ avec des bordures $\partial\mathcal{D} = \Gamma = \cup_{i=1}^{5}\Gamma_i$. L'ensemble $s = \cup_{i\neq j}^{4}\Gamma_i \cap \Gamma_j$ est la partie singulière de la frontière, il se compose de quatre points et $\Gamma \setminus s$ est l'ensemble des points réguliers de la frontière. En chaque point régulier, il y a un vecteur normal interne $\overrightarrow{N}$. Une particule ponctuelle se déplace dans le billard à une vitesse de norme constante $v = \left\| \overrightarrow{v} \right\| = 1$ et lorsqu'il atteint la frontière $\Gamma \setminus s$, il subit une collision élastique à réflexion spéculaire selon la loi de réflexion : "l'angle d'incidence est égal à l'angle de réflexion" par rapport à $\overrightarrow{N}$ le vecteur normal à la frontière au point de collision (si la particule atteint l'un des quatre coins, elle arrête, son mouvement n'est pas défini dans ce cas).

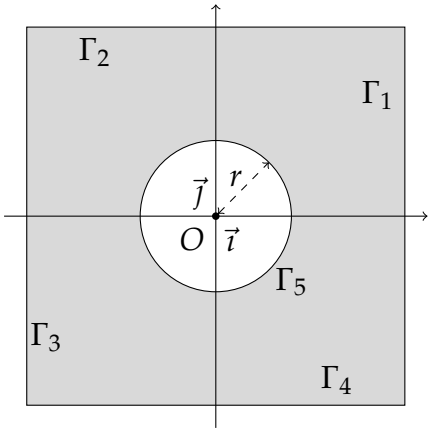

FIGURE 1.3 – Billard de Sinaï

Les propriétés chaotiques du billard de Sinaï

Le billard de Sinaï a fait l'objet de nombreuses études ultérieures par divers chercheurs depuis 1970 [10,17,18,25–27,29,31,38,40,116] en utilisant une variété de techniques pour déterminer son comportement détaillé. Le système présente un comportement complètement chaotique et possède des propriétés statistiques intéressantes.

Per Dahlqvist [31] a calculé explicitement l'expression du λ_L de BS. L'indicateur λ_L est positif pour toutes les valeurs du rayon r de la barrière circulaire. Le système du billard est sensible aux petites changement dans les états initiales, et donc il est très difficile de prédire les trajectoires d'une particule en mouvement dans le billard.

L'entropie de Kolmogorov-Sinaï de BS h_{ks} positive [38], qui est une condition suffisante pour le chaos et l'imprévisibilité du système.

Sinaï a montré à [98] que tout billard avec parois de dispersion est toujours très chaotique. De plus, Sinaï a développé une méthode pour prouver que les billards du Sinaï sont hyperboliques (exposant de Lyapunov non nul presque partout), ergodiques, mélanges et K-mélanges .

Gallavotti et Ornstein [40] ont prouvé que les billards du Sinaï sont des systèmes de Bernoulli. La propriété de Bernoulli est la plus forte parmi les propriétés d'er-

godicité, elle implique K -mélange, mélange et ergodicité.
Le BS a les propriétés chaotiques les plus fortes possibles, étant ergodique, mélangeant, Bernoulli, ayant une entropie Kolmogorov-Sinaï positive. Toutes ces propriétés sont assurées par l'un des mécanismes fondamentaux du chaos (l'hyperbolicité) qui s'appelle le mécanisme de dispersion. Si des particules avec des trajectoires parallèles heurte une frontière de dispersion, les trajectoires divergent directement après réflexion et ainsi le distance entre eux augmente.

Le système du billard de Sinaï

Soit $(O, \vec{\imath}, \vec{\jmath})$ un repère orthonormé du plan. Nous considérons un BS avec O est le centre de la barrière circulaire (Figure 1.3). Soit p une particule ponctuelle se déplaçant dans le billard à une vitesse constante $\left\| \overrightarrow{v} \right\| = 1$ avec $\overrightarrow{v} = \cos(\theta)\vec{\imath} + \sin(\theta)\vec{\jmath}$ et $\theta = (\overrightarrow{\vec{\imath}, \overrightarrow{v}})$ où $\theta \in \Theta = [0, 2\pi[$. L'état $S = (A, \overrightarrow{v})$ de la particule en mouvement est spécifiée par sa position $A \in \mathcal{D}$ et sa vecteur de vitesse unitaire $\overrightarrow{v} \in \mathcal{V}^1$. L'espace de phase du système Ω est défini comme suit

$$\Omega = \left\{ (A, \overrightarrow{v}) \right\} = \mathcal{D} \times \Theta = \mathcal{D} \times \mathcal{V}^1.$$

Nous nous intéressons aux états tridimensionnels dont la frontière est

$$\partial\mathcal{D} \times \mathcal{V}^1 = \Gamma \times \mathcal{V}^1.$$

Entre deux collisions consécutives sur des points non singuliers, nous identifions deux états de la particule $S_n = (A_n, \overrightarrow{v_n})$ and $S_{n+1} = (A_{n+1}, \overrightarrow{v_{n+1}})$ où les deux orientations sont reliées par la règle de collision suivante

$$\overrightarrow{v_{n+1}} = \overrightarrow{v_n} - 2(\overrightarrow{N_{n+1}}.\overrightarrow{v_n})\overrightarrow{N_{n+1}}.$$

Après la forme géométrique du billard, nous avons

$$\overrightarrow{N_{n+1}} = \begin{cases} -\dfrac{x_{n+1}}{|x_{n+1}|}\vec{\imath} & \text{si } A_{n+1} \in \Gamma_1 \bigcup \Gamma_3 \\ -\dfrac{y_{n+1}}{|y_{n+1}|}\vec{\jmath} & \text{si } A_{n+1} \in \Gamma_2 \bigcup \Gamma_4 \\ \dfrac{x_{n+1}\vec{\imath} + y_{n+1}\vec{\jmath}}{\sqrt{x_{n+1}^2 + y_{n+1}^2}} & \text{si } A_{n+1} \in \Gamma_5 \end{cases}.$$

Nous définissons f, la fonction de transition de S_n à S_{n+1}, à savoir,

$$\begin{array}{rcl} f : [-a\,;a]^2 \times [0\,;2\pi[& \to & [-a;a]^2 \times [0\,;2\pi[\\ (x_n, y_n, \overrightarrow{v_n}) & \mapsto & (x_{n+1}, y_{n+1}, \overrightarrow{v_{n+1}}) \\ S_n & \mapsto & S_{n+1} \end{array}$$

Entre les deux états S_n et S_{n+1}, la trajectoire de la particule est rectiligne et satisfait

$$\overrightarrow{A_nA_{n+1}} = t\overrightarrow{v_n}, \quad t \in \mathbb{R}^+, \text{ alors } \begin{cases} x_{n+1} = tv_{n,x} + x_n \\ y_{n+1} = tv_{n,y} + y_n \end{cases}.$$

L'équation du mouvement d'une particule est

$$(D_n): \quad v_{n,y}x - v_{n,x}y - v_{n,y}x_n + v_{n,x}y_n = 0.$$

Par conséquent, il est important de calculer les coordonnées du point de collision $A_{n+1}(x_{n+1}, y_{n+1})$. Si $A_n \in \Gamma_i$ où $i \in \{1, 2, \ldots, 5\}$ donc $\exists! j \in \{1, 2, \ldots, 5\} \setminus \{i\}$ tel que $A_{n+1} \in \Gamma_j$. Pour $v_{n,x} \neq 0$ et $v_{n,y} \neq 0$, on distingue deux cas : $A_n \in \Gamma_5$ et $A_n \notin \Gamma_5$.

Le cas $A_n \in \Gamma_5$. $A_n \in \Gamma_5$ alors $A_{n+1} \in \cup_{i=1}^4 \Gamma_i$, soit

$$t_1 = \frac{a - x_n}{v_{n,x}}, \quad t_2 = \frac{a - y_n}{v_{n,y}}, \quad t_3 = \frac{-a - x_n}{v_{n,x}} \text{ et } t_4 = \frac{-a - y_n}{v_{n,y}},$$

alors il y a deux nombres distincts i et j de $\{1,2,3,4\}$ tels que :

$$t_i, t_j > 0,$$

soit

$$t = min(t_i, t_j),$$

donc

$$(x_{n+1}, y_{n+1}) = (tv_{n,x} + x_n, tv_{n,y} + y_n).$$

Le cas $A_n \notin \Gamma_5$. $A_n \notin \Gamma_5$ alors $A_{n+1} \in \cup_{i=1}^{5}\Gamma_i$. La distance de O à (D_n) est :

$$d\big(O,(D_n)\big) = |-x_n v_{n,x} + y_n v_{n,y}|.$$

Si $d\big(O,(D_n)\big) > r$ alors $A_{n+1} \in \cup_{i=1}^{4}\Gamma_i$, donc (x_{n+1}, y_{n+1}) est calculé de la même manière que dans le cas où $A_n \in \Gamma_5$.
Si $d(O, D_n) < r$ alors $A_{n+1} \in \Gamma_5$, par conséquent

$$\begin{cases} x_{n+1} = tv_{n,x} + x_n \\ y_{n+1} = tv_{n,y} + y_n \\ x_{n+1}^2 + y_{n+1}^2 = r^2 \end{cases},$$

ensuite nous avons

$$t^2 + 2(x_n v_{n,x} + y_n v_{n,y})t + x_n^2 + y_n^2 - r^2 = 0. \tag{1.6}$$

$\triangle_n^{'}$ les discriminent de (1.6) est défini comme suit

$$\triangle_n^{'} = r^2 + (x_n v_{n,x} + y_n v_{n,y})^2 - x_n^2 - y_n^2.$$

Par conséquent, deux solutions possibles sont

$$t_1 = x_n v_{n,x} + y_n v_{n,y} - \sqrt{\triangle_n'} \text{ et } t_2 = x_n v_{n,x} + y_n v_{n,y} + \sqrt{\triangle_n'}.$$

Soit

$$t = min(t_1, t_2),$$

alors

$$(x_{n+1}, y_{n+1}) = (tv_{n,x} + x_n, tv_{n,y} + y_n).$$

Pour $v_{n,x} = 0$ $(v_{n,y} = 1)$ ou $v_{n,y} = 0$ $(v_{n,x} = 1)$, on a

$$(x_{n+1}, y_{n+1}) = \begin{cases} \left(x_n, \frac{v_{n,y}}{|v_{n,y}|}a\right) & \text{si } A_n \in \Gamma_5 \text{ et } |v_{n,y}| = 1 \\ \left(\frac{v_{n,x}}{|v_{n,x}|}a, y_n\right) & \text{si } A_n \in \Gamma_5 \text{ et } |v_{n,x}| = 1 \\ \left(x_n, \frac{v_{n,y}}{|v_{n,y}|}\sqrt{r^2 - x_n^2}\right) & \text{si } A_n \notin \Gamma_5 \text{ et } |v_{n,y}| = 1 \text{ et } -r < x_n < r \\ \left(x_n, \frac{v_{n,y}}{|v_{n,y}|}a\right) & \text{si } A_n \notin \Gamma_5 \text{ et } |v_{n,y}| = 1 \text{ et } |x_n| \geqslant r \\ \left(\frac{v_{n,x}}{|v_{n,x}|}\sqrt{r^2 - y_n^2}, y_n\right) & \text{si } A_n \notin \Gamma_5 \text{ et } |v_{n,x}| = 1 \text{ et } -r < y_n < r \\ \left(\frac{v_{n,x}}{|v_{n,x}|}a, y_n\right) & \text{si } A_n \notin \Gamma_5 \text{ et } |v_{n,x}| = 1 \text{ et } |y_n| \geqslant r \end{cases}.$$

1.4 La cryptographie à base du chaos

La cryptographie à base du chaos est de mettre œuvre la théorie du chaos au service de la cryptographie. L'étude des SCs et de leurs applications à la cryptographie a fait l'objet d'une attention considérable au cours des dernières années dans une partie de la communauté scientifique. Surtout après la déclaration du chercheur American Claude Shannon en 1949 dans son article [94] qui a écrit :

"Good mixing transformations are often formed by repeated products of two simple non-commuting operations. Hopf has shown, for example, that pastry dough can be mixed by such a sequence of operations. The dough is first rolled out into a thin slab, then folded over, then rolled, and then folded again, etc."

En effet, la cryptographie à base du chaos devient un sujet d'étude [57, 62, 70], de nombreux chercheurs ont montré l'existence d'une relation directe entre les propriétés des SCs et les deux propriétés de confusion et diffusion mentionnés par Shannon.
Cette partie porte sur la compréhension de la cryptographie basée sur le chaos. La première section la relation entre la théorie du chaos et la cryptographie, tandis que la deuxième montre les similitudes entre les algorithmes cryptographiques et les SDs. Enfin, Le chapitre se terminera par une liste des règles nécessaires pour la conception des cryptosystèmes à basés du chaos plus sécurisés.

1.4.1 Relation entre le chaos et la cryptographie

Les SCs non linéaires déterministes sont caractérisés selon par [41] : l'effet d'avalanche, le mélange topologique, l'a-périodicité, les orbites denses non périodiques et l'ergodicité. Or un système cryptographique performant ou sécurisé doit être capable de produire un comportement pseudo-aléatoire. Autrement dit, il doit garantir les deux exigences de la confusion et la diffusion.
En se référant aux propriétés discutées des SCs, il est clair que les propriétés d'ergodicité et de mélange topologique sont directement liées à la confusion. La dynamique dans l'attracteur chaotique est donnée par des orbites apériodiques qui génèrent des modèles statistiques similaires. Ces modèles peuvent être utilisés pour masquer des messages clairs au moyen de techniques analogues à la substitution. D'autre part, la diffusion est intrinsèquement liée à la sensibilité aux conditions initiales des SCs. La diffusion produit l'effet d'avalanche lorsqu'une petite différence entre des entrées dans le cryptosystème donne des sortie complètement différentes. Les paramètres d'entrées dans un algorithme cryptographique peuvent produire l'effet d'avalanche. La table 1.1 résume le lien entre les propriétés cryptographie et le chaos [3] :

Propriétés chaotiques	**Propriétés cryptographique**	**Description**
Ergodicité	Confusion	La sortie a la même distribution pour n'importe quelle entrée
Sensibilité aux conditions initiales / paramètre de contrôle	Diffusion avec un petit changement dans le texte en clair / clef secrète	Une petite déviation de l'entrée peut entraîner une modification importante de la sortie
Propriété de mélange	Diffusion avec un petit changement dans un bloc simple de tout le texte en clair	Une petite déviation dans la zone locale peut provoquer un changement important dans tout l'espace
Dynamique déterministe	Pseudo-aléa déterministe	Un processus déterministe peut provoquer un comportement pseudo-aléatoire
Structure complexe	Complexité de l'algorithme (attaque)	Un processus simple a une très grande complexité

TABLE 1.1 – Comparaison des propriétés du chaos et de la cryptographie

La relation profonde entre le chaos et la cryptographie est résumé dans la table 1.2. La table présente la similitude étroite entre les SCs et les algorithmes cryptographiques. La différence importante entre le chaos et la cryptographie ré-

Algorithmes cryptographiques	**Systèmes chaotiques**
Espace de phase : ensemble fini d'entiers	Espace de phase : (sous) ensemble de nombres réels
Méthodes algébriques	Méthodes analytiques
Rondes	Itérations
Key (Boolean) - Espace clé discret	Paramètres (réels) - Espace de clé continu
La diffusion	Sensibilité à un changement de condition initiale / paramètres
Réalisations numériques en arithmétique entière	Réalisation numérique par arithmétique non entière qui se rapproche du continu

TABLE 1.2 – Similarités et différences entre les systèmes chaotiques et les algorithmes cryptographiques.

side dans le fait que les transformations de chiffrement sont définies sur des ensembles finis, tandis que le chaos n'a de sens que sur des nombres réels. En bref, le lien étroit entre les propriétés des deux théories, a encouragé les chercheurs à

proposer une grande variété de cryptosystèmes à base du chaos, en reposant sur des techniques différentes.

1.4.2 Techniques de chiffrement basées sur le chaos

Dans la littérature, plusieurs cryptosystèmes basés sur le chaos ont été proposés [5,44,56,97,114]. Pour avoir un vue générale des principales techniques et de leurs concepts de fonctionnement, des aperçus résumés avec des références des résultats obtenus peuvent être trouvés auprès de [2,71]. En général, il existe deux classes de cryptosystèmes à base du chaos : analogiques et numériques [3]. Le but des cryptosystèmes analogiques est de fournir une communication sécurisée sur un canal bruyant. il font intervenir plusieurs techniques de synchronisation du chaos. Les cryptosystèmes analogiques basés sur le chaos [12,86]. En revanche, les cryptosystèmes basés sur le chaos numérique sont conçus pour les ordinateurs numériques, où une ou plusieurs cartes chaotiques sont implémentées avec une précision de calcul finie. La recherche présentée dans cette thèse se concentre sur la cryptographie numérique. Le chiffrement d'un message clair s'effectue par bloc ou par flux par plusieurs manières, y compris les suivantes :

— **Chiffrement chaotique par flot**

- Chiffrement par flot basé sur des générateurs de nombres pseudo-aléatoires chaotiques [9,47] : Le message est masqué en appliquant un Xor avec un flot de clé généré par des GNPA.
- Chiffrement par flot par l'intermédiaire de l'approche système inverse [119,120] : Ce chiffrement est proposé par Feldmann [36], où Un signal de message est ajouté à la sortie du signal chaotique, qui a été alimenté par le signal de message chiffré dans les instants précédents. Ce chiffrement est peu sécurisé contre les attaques connues.

— **Chiffrement chaotique par bloc**

- Chiffrement par bloc basé sur des SCs inverses [37,43,76] : Ce chiffre-

ment est proposé par Habutsu et al [43] utilise la suite en accent circonflexe déviée (skew Tent map).

- Chiffrement par bloc basé sur les fonctions chaotiques arrondies ou S-Box [50, 82] : Ce chiffrement utilise des S-Box chaotiques pour chiffré les données. Il repose essentiellement sur l'architecture de Feistel en utilisant une table de substitution non linéaire.
- Chiffrement chaotique par Bloc pour Réseaux de Capteurs : Ce chiffrement est proposé par Chen [23], il est basé sur une fonction chaotique.

1.4.3 Règles de conception pour la cryptographie basée sur le chaos

Un système cryptographique basé sur le chaos doit être surement robuste, seulement si les propriétés offertes par les SCs sont bien exploitées au cours de sa conception. Grâce à des études bien approfondies [2, 58, 102] et cryptanalyse des cryptosystèmes à base du chaos, une liste des règles et exigences à été accumulée pour s'assurer que les mêmes erreurs ne se reproduisent pas lors de la création des cryptosystèmes. La liste des règles et exigences à respecter est trouvés dans [2, 3] :

- Le cryptosystème basé sur le chaos devrait être comparable en coût et en rapidité aux chiffrements cryptographiques classiques.
- Une description détaillée précise du fonctionnement des algorithmes chaotiques devrait être fournie.
- Définition exhaustive et rigoureuse de la clé et de l'espace clé, à partir duquel les clés valides doivent être choisies, doit être spécifié avec précision et éviter les régions non chaotiques.
- Sans perte de sécurité, le cryptosystème devrait être facile à mettre en œuvre avec un coût et une vitesse acceptables.
- La connaissance partielle d'une clé ne doit pas révéler d'informations sur

le texte en clair ou les clés inconnues restantes.

- Les SCs devraient avoir un degré élevé de sensibilité à l'inadéquation des paramètres
- Pour les SCs mis en œuvre sous forme numérique, les effets négatifs de la dégradation dynamique doivent être pris en considération avec une évaluation minutieuse.
- L'entropie du cryptosystème doit être analysée.
- Des SCs avec des fonctions de densité invariantes uniformes et une mesure invariante indépendante du paramètre doivent être utilisées.
- Il ne devrait y avoir aucun moyen de reconstruire la dynamique du SD ou le texte en clair à travers le texte chiffré.
- Les systèmes cryptographiques basés sur le chaos doivent résister aux attaques classiques et aux applications spécifiques.
- Le temps de chiffrement / déchiffrement ne doit pas dépendre de la valeur de la clé secrète d'un cryptosystème basé sur le chaos.
- fournir une sécurité suffisante contre les attaques par force brute, la taille de l'espace clé doit être supérieur à 2^{100}.
- Il convient de vérifier si le cryptosystème peut être rompu par toutes les attaques spécifiques La résistance à la cryptanalyse différentielle et linéaire doit être prouvée ou vérifiée très soigneusement dans les chiffrements par blocs numériques.
- Une connaissance partielle de la clé ne doit jamais révéler d'informations partielles sur le texte en clair ni la partie inconnue de la clé.
- La région chaotique utile, c'est-à-dire l'espace clé K, doit être discrétisée de manière à garantir l'effet d'avalanche.

Chapitre 2

La conception d'un générateur de nombres pseudo-aléatoires basé sur le billard de Sinaï

Les nombres pseudo-aléatoires ont un rôle important non seulement pour la simulation stochastique mais aussi pour divers applications telles que : les expériences statistiques, l'analyse numérique, les algorithmes probabilistes, les jeux informatiques et les protocoles cryptographiques et bien d'autres. En pratique, la génération de tels nombres pseudo-aléatoires avec des propriétés de caractère aléatoire est une tâche complexe. En général, on dit des nombres pseudo-aléatoires pour désigner les nombres au hasard qui apparaissent au cours de deux étapes. Ces nombres aléatoires sont simulés par des algorithmes déterministes appelés des générateurs de nombres pseudo-aléatoires. À partir d'une seule graine initiale, ces générateurs produiront toujours les mêmes séquences de nombres dont le comportement est très difficile à distinguer ou à prédire. Les atouts de tels générateurs sont : un temps d'exécution rapide, la répétabilité et la reproductibilité des séquences pseudo-aléatoires. Une famille de générateurs basé sur des techniques basique sont souvent utilisés, telle que Générateur congruentiel linéaire [65], générateur de Fibonacci [13], générateur par Registre à décalage [64],

générateur de Marsaglia-Zaman [74], générateur BBS (Blum Blum Shub) [54] et bien d'autre [51]. Cependant, ces générateurs ne sont pas pratiques au niveau de d'application à la cryptographie et sont également inefficaces. Généralement, la sécurité d'un générateur cryptographique est basée sur la difficulté de résoudre le problème mathématique associé.

Les algorithmes cryptographiques à base du chaos ont montré de bonnes performances pour le chiffrement des données telles que les images, les vidéos ou les données audio [1,52,72,92], cela dû à la similitudes entres les besoins de la cryptographie et les propriétés offertes par le chaos [57,61,70]. Les GNPAs basés sur le chaos ont retenu plus l'attention. Le premier GNPA a été proposé par Oishi et Inoue [78] en 1982, en utilisant les équations différentielles non linéaires de premier ordre chaotiques. Après cet article, les GNPAs basées sur le chaos ont connu un grand développement. En effet, plusieurs GNPA ont été suggérésen se basant sur divers SCs. Des propositions à base du système logistique dans [6, 42, 85]. Dans [118], un générateur basé sur la carte généralisée de Henon. En utilisant le système Lorenz, un nouveau générateur pour le chiffrement des données vocales est conçu dans [1]. Un système standard chaotique a été appliqué dans la conception du générateur dans [84].

Notre travail se concentre sur une approche alternative basée sur la mise en œuvre d'un système plus concret qui possède des propriétés chaotiques intéressantes, ce sont les systèmes de billard chaotique en deux dimensions [28]. Ils font partie des classes de systèmes simples, qui explorent toujours le chaos. La théorie mathématique du billard a été introduite par Yakov Sinai en 1970 [98]. Il est développé et évolué avec une vitesse remarquable pour devenir un bien ancré dans la théorie des systèmes dynamiques et de la mécanique statistique. Plusieurs études ont été consacrées spécifiquement au billard chaotique. Le billard de Sinaï est la première classe des billards chaotiques, il est aussi appelé le système de dispersion. Un disque circulaire à l'intérieur du billard provoque des trajectoires divergentes et imprévisible, ce qui nous a encouragés à l'utiliser dans la construc-

tion d'un nouveau GNPA.

Ce chapitre est organisé comme suit. Dans la section 1, nous donnons une introduction générale sur les générateurs de nombres pseudo-aléatoires et une discussion. Dans la section 2, nous donnons une description détaillée de GNPA. Une validation du GNPA par des tests de sécurité sont décrites dans la section 4. Dans la dernière section, une conclusion est tirée.

2.1 Générateurs de nombres pseudo-aléatoires

Les GNPAs sont des algorithmes déterministes, initialisés par une gaine et utilisés pour générer des nombres ou de bits de manière aléatoire dont le comportement est très difficile à distinguer ou prédire. Mathématiquement, un GNPA est défini [63] comme une structure (S, U, f, s_0, E), où :

- S : est un ensemble fini d'états ;
- U : est un ensemble des valeurs de sortie ;
- $f : S \longrightarrow S$ est la fonction de transition ;
- s_0 : est état initial ;
- $E : S \longrightarrow U$ est la fonction de sorties.

L'état évolue en fonction de la récurrence f (fonction de transition) $s_i = f(s_{i-1})$, pour tout entier $i \geq 1$, et la sortie à l'étape i est $u_i = E(s_i) \in U$. Le choix de l'état initial s_0 (la graine) détermine entièrement la suite de nombres pseudo-aléatoires produite par le générateur. Généralement s_0 est déterminée par l'utilisateur ou à l'aide de l'horloge de la machine. Les valeurs de sortie $u_1, u_2, u_3, \ldots$ sont soi-disant les nombres aléatoires produits par le GNPA qui semblent indépendant et identiquement distribués sur U. Dans le cas des générateur de bit pseudo aléatoires $u_i = b_1 b_2 b_3, \ldots b_n$ où $|u_i| = n$ où $n \in \{8, 16, 32, \ldots\}$. Le plus petit entier $p \in \mathbb{N}$, tel que $\forall n \in \mathbb{N}$, $u_{n+p} = u_n$ est la période du GNPA.

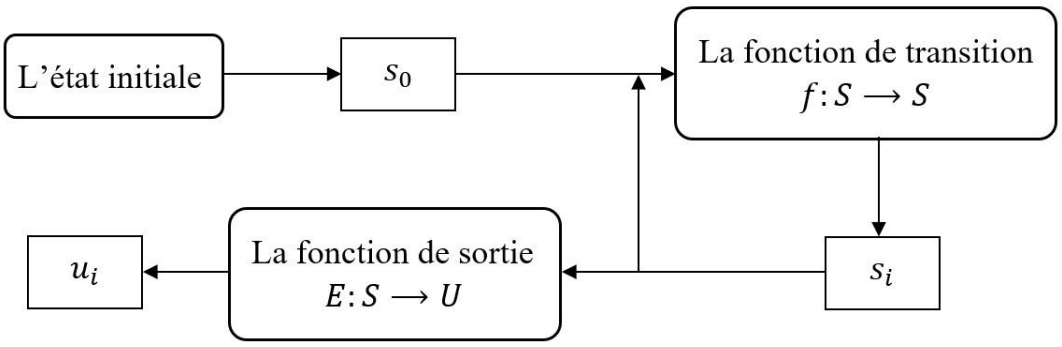

FIGURE 2.1 – Architecture du générateur de nombres aléatoires

2.1.1 Générateur de bits pseudo-aléatoires Cryptographique

Un générateur de bits pseudo-aléatoires cryptographique (GBPAC) est GNPA avec des exigences supplémentaires qui le rendent convenable à l'utilisation en cryptographie. En fait, Le générateur doit réussir les tests statistiques d'aléatoire [91] ; et résiste bien en cas d'attaque, même lorsque l'attaquant dispose d'une partie de leur suite. Par conséquent, un GBPAC produit des séquences de bits aléatoires $SF = b_1 b_2 b_3, \ldots$ qui possède les propriétés suivantes :

— Le générateur non biaisé : $Pr(b_i = 0) = 1/2$ pour $i \geq 1$.

— Les bits ne sont pas corrélés : $Pr(b_i = 0 | b_1, b_2, b_3 \ldots b_{i-1}) = 1/2$.

Autrement dit, un GBPAC est GNPA qui est imprévisible. Cela signifie que, si on dispose n bits consécutifs $b_i, b_{i+1}, \ldots, b_{i+n-1}$ d'une séquence SF, il n'existe pas d'algorithme en temps polynomial qui permet de prédire le prochain bit $b_n, b_{n+1} \ldots$ où les bits précédents $b_{i-1}, b_{i-2}, \ldots$ avec une probabilité de succès supérieure à $1/2$. Notez que la nécessité de l'imprévisibilité des GBPAC est unique à la cryptographie.

2.1.2 Discussion

Les GBPACs basés sur des SCs génèrent séquence binaire SF pour le chiffrement. En effet, le flux de clés SF est utilisé pour masquer le message clair via une opération XOR binaire selon le chiffrement de Vernam. La suite binaire SF est

composée de la concaténation des sou-suites comme suit :

$$SF = u_1 \| u_2 \| \dots \| u_N$$

avec $\{u_i\}_{1 \leq i \leq N}$ une sou-suite binaire de longueur fixe $n \in \{8, 16, 32, \dots\}$, u_i est générée à l'étape i tel que :

$$u_i = E\big(f(s_{i-1})\big) \text{ et } s_i = f(s_{i-1}) \text{ pour tout } i \geq 1$$

Avec f est la fonction de transition du SCs et E est la fonction de sortie qui transforme les états du système en nombres binaires. Les conceptions existantes de GNPA chaotiques utilisent différentes techniques pour passer du continuum au monde binaire. Les plus pertinents sont :

- Extraire des bits de chaque état le long des orbites chaotiques [30, 88].
- Partitionner l'espace de phase en m sous-espaces et produire un nombre binaire $i = 0, 1, \dots, m$ si l'orbite chaotique visite le $i^{\text{ème}}$ sous-espace [104, 106].
- Combiner les sorties de deux ou plusieurs SCs pour générer les nombres pseudo-aléatoires [66, 96].

Le texte chiffré de certains cryptosystèmes permet de reconstruire un système de retour du SC sous-jacent. La façon la plus directe d'estimer les paramètres de contrôle à partir d'une orbite chaotique est de tracer s_{n+1} en fonction de s_n, qui est en fait, la fonction de transition chaotique elle-même. Si une telle fonction de retour est significative, un attaquant peut être en mesure d'inférer les valeurs des paramètres de contrôle qui régissent l'évolution du SC. Dans [100], une attaque de texte chiffré choisi est utilisée pour construire une version discrétisée de la carte logistique qui conduit en outre à l'estimation du paramètre de contrôle. Une solution contre ce type d'attaque consiste à :

- Ne pas utiliser tous les états de l'orbite pour produire la séquence pseudo-

aléatoire ;

- Mélanger et tronquer l'orbite chaotique avant de l'utiliser pour le chiffrement, ce qui randomise le tracé de la carte de retour ;
- Utiliser des systèmes fortement chaotiques et qui n'ont pas de forme explicite.

Nous proposons dans la suite un GBPAC, nous reposons sur un système purement chaotique et qui a de forme complexe, en introduisant de nouvelles techniques. Afin de se débarrasser de la corrélation existante entre les états consécutifs, à chaque étape i un nombre n_i des itérations effectués, ce nombre est contrôlé par la clé. En effet, entre deux sous-suites u_i et u_{i+1} il peut avoir jusqu'à 4 itérations. Pour une suite binaire SF généré par notre algorithme, on ne peut pas faire une prédiction sur les des orbites à partir de sous-suites des séquences binaires si on ne connaît pas la clé. En conséquent, notre algorithme n'offre aucune information à partir de sa sortie, on peut dire alors qu'il est plus que sûre par rapport à la plupart des algorithmes déjà proposés.

2.2 Conception d'un GNPA basé sur le billard de Sinaï

Il s'agit d'un générateur déterministe des nombres pseudo-aléatoires initialisé par une clé K de taille arbitraire, dont la sortie est une séquence binaire cryptographiquement sûrs.

Nous considérons deux particules ponctuelles qui se déplacent dans le billard de Sinaï avec une vitesse constante $\left\|\overrightarrow{v_1}\right\| = \left\|\overrightarrow{v_2}\right\| = 1$, sans interaction entre elles. L'état initial de la première particule (resp. Seconde particule) est $S_{0,1} = (O_1, \overrightarrow{v_{0,1}})$ (resp. $S_{0,2} = (O_2, \overrightarrow{v_{0,2}})$ avec $\overrightarrow{OO_1} = \frac{3a}{4}\vec{\imath}$ (resp. $\overrightarrow{OO_2} = -\frac{3a}{4}\vec{\imath}$) et $\overrightarrow{v_{0,1}} = \cos(\theta_{0,1})\vec{\imath} + \sin(\theta_{0,1})\vec{\jmath}$ (resp. $\overrightarrow{v_{0,2}} = \cos(\theta_{0,2})\vec{\imath} + \sin(\theta_{0,2})\vec{\jmath}$ où :

$$0 \leq \theta_{0,1}, \theta_{0,2} < 2\pi.$$

Les angles $\theta_{0,1}$ et $\theta_{0,2}$ sont calculés à partir du clé K, en utilisant une technique basée sur un pointeur. Le pointeur se déplace d'une position à l'autre en fonction d'une congruence linéaire sur la représentation ASCII du K comme indiquée dans l'algorithme 1.

Après l'initialisation du système, les deux particules effectuent un nombre prédéterminé de collisions n_0. À chaque étape i, le système effectue n_i itérations. Les valeurs à virgules flottantes prises par les coordonnées du système sont converties par une fonction a valeur entière E, pour générer deux sous-suites $s_{(i,1)}$ et $s_{(i,2)}$ nécessaires pour la construction de la suite finale S. La suite finale $S = S_1\, S_2 \ldots S_i \ldots$ avec $S_i = s_{i,1} \bigoplus s_{i,2}$ comme montrée dans l'algorithme 2.

2.2.1 Le calcul des états initiaux

Par une clé $K = (k_0 k_2 k_1 \ldots k_{L-1})_2$, de longueur arbitraire L, nous calculons les angles d'initialisation $\theta_{0,1}$ and $\theta_{0,2}$. Pour chaque angle, nous avons besoin d'extraire 64 bits du clé K. Nous considérons un pointeur pt qui prend des valeurs indiquant les positions des bits du K. La suite des positions est définie comme suite :

$$\begin{cases} pt(0) & = 1 \\ pt(i+1) & = \left(\left([\frac{L}{2}] + 1 \right) \times pt(i) + 1 \right) \mathrm{mod}(L) \text{ pour } i \geq 0 \end{cases} \tag{2.1}$$

Le pointeur se déplace sur la clé K de passe et à chaque fois se positionne sur un nouveau bit k_i et lit l'information 0 ou 1 nécessaire pour le calcul du $\theta_{0,1}$ et $\theta_{0,2}$. Nous trouvons $b_{0,1}$ et $b_{0,2}$ ($0 \leq b_{0,1}, b_{0,2} < 2^{64}$) comme suit :

$$\begin{aligned} b_{0,1} &= (\overline{p}_{pt(63)}\, p_{pt(62)} \cdots p_{pt(2)}\, \overline{p}_{pt(1)}\, p_1)_2 \\ &= p_1 + \sum_{i=0}^{31} \overline{p}_{pt(2\times i+1)} \times 2^i + \sum_{i=1}^{31} p_{pt(2\times i)} \times 2^i \end{aligned}$$

et

$$\begin{aligned} b_{0,2} &= (p_{L-1-pt(63)}\,\overline{p}_{L-1-pt(62)}\cdots p_{L-1-pt(1)}\,\overline{p}_{L-2})_2 \\ &= \overline{p}_{L-2} + \sum_{i=0}^{31} p_{pt(2\times i+1)} \times 2^i + \sum_{i=1}^{31} \overline{p}_{pt(2\times i)} \times 2^i \end{aligned}$$

finalement

$$\theta_{0,1} = \frac{2\pi \times b_{0,1}}{2^{64}} \quad \text{et} \quad \theta_{0,2} = \frac{2\pi \times b_{0,2}}{2^{64}}$$

Comme indiqué dans l'équation 2.1, on a $\left([\frac{L}{2}]+1\right) \wedge L = 1$, alors la période de la suite des positions prise par le pointeur est maximale, et donc les bits du clé sont presque tous pris dans la calcul des états initiaux.

Nous appelons **initialiser** l'algorithme 1 de calcul des orientations initiales.$\theta_{0,1}$ et $\theta_{0,2}$.

Algorithm 1 Le calcule des $\theta_{0,1}$ et $\theta_{0,2}$ (initialiser)

Début
Entrée : Une clé $K = (k_0 k_1 \ldots k_{L-2} k_{L-1})_2$ de longueur arbitraire L.
Sortie : Des orientations $\theta_{0,1}$ et $\theta_{0,2}$.
$pt \leftarrow 1$
$b_{0,1} \leftarrow p_1$
$b_{0,2} \leftarrow p_{L-2}$
for $i = 1$ to 63 **do**
    $pt \leftarrow \left(\left([\frac{L}{2}]+1\right) \times pt + 1\right) \mod (L)$
    if i est pair **then**
        $b_{0,1} \leftarrow b_{0,1} + \overline{p}_{pt} \times 2^i$
        $b_{0,2} \leftarrow b_{0,2} + p_{L-1-pt} \times 2^i$
    else {i est impair}
        $b_{0,1} \leftarrow b_{0,1} + p_{pt} \times 2^i$
        $b_{0,2} \leftarrow b_{0,2} + \overline{p}_{L-1-pt} \times 2^i$
    end if
end for
$\theta_{0,1} \leftarrow \frac{2\pi \times b_{0,1}}{2^{64}}$
$\theta_{0,2} \leftarrow \frac{2\pi \times b_{0,2}}{2^{64}}$
Fin

2.2.2 Génération de la séquence pseudo-aléatoire

Après avoir calculé les angles d'initialisation $\theta_{0,1}$ et $\theta_{0,2}$, les deux particules sont prêtes à parcourir le billard. Avant de commencer à générer les individus, les particules effectuent n_0 collisions avec les frontières du billard, où e $(0 \leq n_0 \leq 255)$ un entier est déterminé à partir des derniers 8 bits de la clé K tel que :

$$n_0 = \sum_{i=0}^{7} p_i \times 2^i$$

nous obtenons

$$(x_k^0, y_k^0, \theta_k^0) = f^{n_0}(x_{0,k}, y_{0,k}, \theta_{0,k}) \text{ avec } k = 1, 2$$

À chaque étape $i \geq 1$, nous effectuons $(n_i + 1)$ collisions pour les deux particules avec $n_i (0 \leq n_i \leq 3)$ est déterminé par 2 bits pris directement de la clé K comme suit :

$$n_i = 2 \times p_{j+1} + p_j \tag{2.2}$$

où

$$j = 2 \times i \bmod (L - 1)$$

Après $(n_i + 1)$ collisions, de nouveaux états sont obtenus

$$(x_k^i, y_k^i, \theta_k^i) = f^{n_i+1}(x_k^{i-1}, y_k^{i-1}, \theta_k^{i-1}).$$

Nous nous intéressons aux coordonnées de collision avec la bordure carrée du billard (c'est-à-dire $\bigcup_{i=1}^{4} \Gamma_i$) en ignorant les collisions avec le cercle. Si $A_k^i \in \Gamma_5$ (c'est-à-dire $(x_k^i)^2 + (y_k^i)^2 = r^2$), la $k^{\text{ème}}$ est invitée à effectuer une collision addi-

tionnelle et avoir une nouvelle état :

$$(x_k^i, y_k^i, \theta_k^i) \longleftarrow f^{n_i+2}(x_k^{i-1}, y_k^{i-1}, \theta_k^{i-1}),$$

puis deux sou-suites sont générés des valeurs :

$$\begin{aligned} E(x_k^i, y_k^i) &= \begin{cases} \left[2^{32}\frac{x_k^i}{a}\right] & \text{si } x_k^i \geq 0 \text{ et } \left|y_k^i\right| = a \\ \left[2^{32}(1+\frac{x_k^i}{a})\right] & \text{si } x_k^i < 0 \text{ et } \left|y_k^i\right| = a \\ \left[2^{32}\frac{y_k^i}{a}\right] & \text{si } y_k^i \geq 0 \text{ et } \left|x_k^i\right| = a \\ \left[2^{32}(1+\frac{y_k^i}{a})\right] & \text{si } y_k^i < 0 \text{ and } \left|x_k^i\right| = a \end{cases} \\ &= u_{i,k} = (b_{31}^{i,k} b_{30}^{i,k} \dots b_1^{i,k} b_0^{i,k})_2 \end{aligned}$$

La sortie SF du GNPA est la concaténation des sous-séquences $u_1, u_2, \dots, u_i \dots$ donc :

$$SF = u_1 u_2 \dots u_i \dots,$$

avec

$$u_i = u_{i,1} \bigoplus u_{i,2},$$

où $u_{i,1}$ et $u_{i,2}$ deux sous-séquences générés à i^{th} étape.

En général, le processus de la plupart des générateurs reposent sur itérations successive pour générer les nombres. En effet l'utilisation de la clé reste limité seulement pour générer les états initiaux, et les sou-suites sont générés par des itérations successive. Ces générateurs sont générateurs vulnérables contre les attaques. Dans Notre algorithme, entre deux sou-suites u_i et u_{i+1}, il y a un nombre n_i d'itérations à effectuer contrôlé par la clé secrète comme indiqué dans 2.2. Donc, sans la clé secrète, il est impossible de faire une prédiction avec une probabilité supérieurs à $1/2$ sur les sou-suites à venir même si on dispose des d'un nombre importants des séquences.

L'algorithme a deux paramètres d'entrée, une clé K et un entier N qui indique la

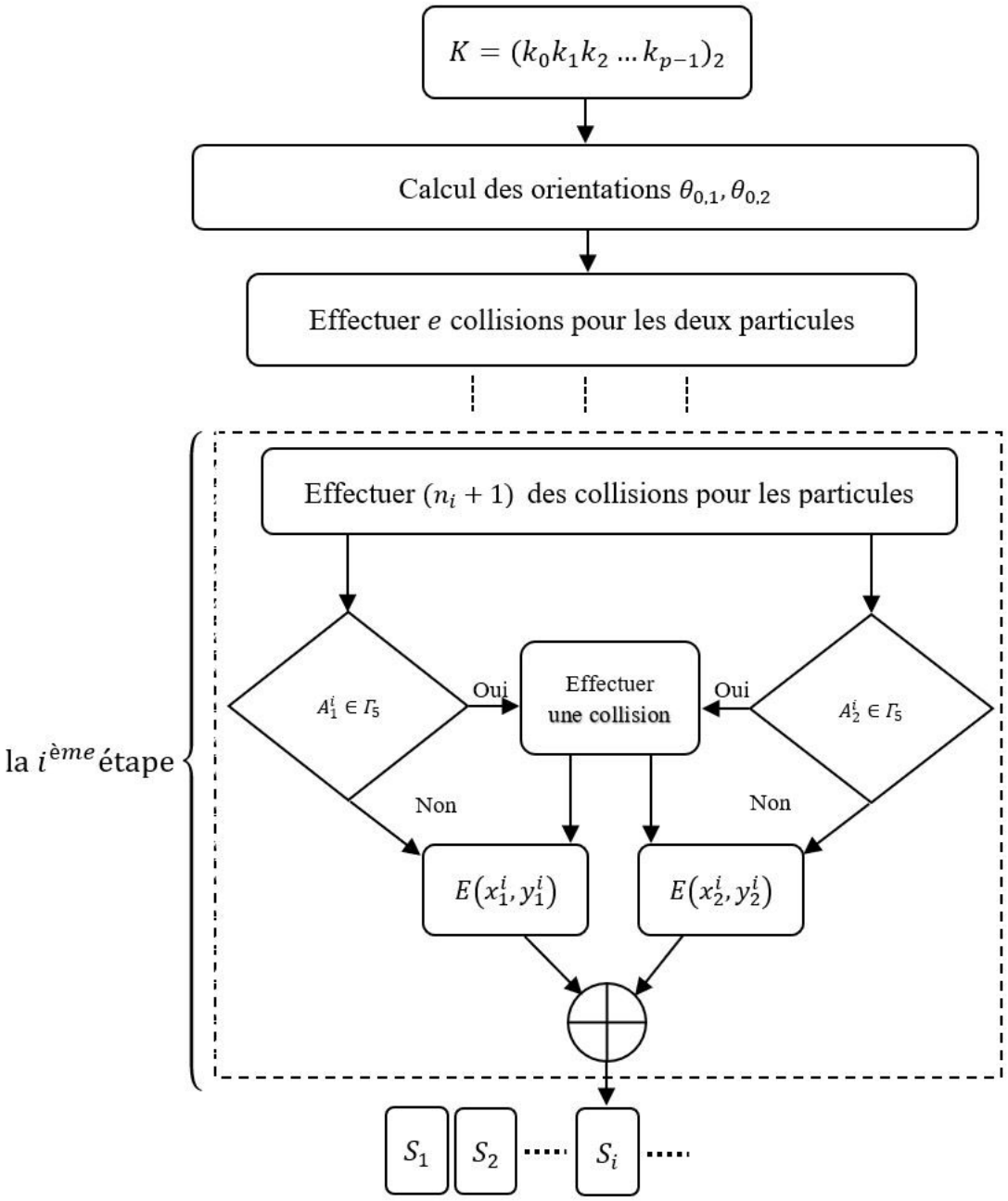

FIGURE 2.2 – L'architecture de notre GNPA

longueur de la séquence binaire demandée par l'utilisateur comme indiqué dans (Algorithm 2).

Algorithm 2 Génération du suite pseudo-aléatoire SF de longueur N

Début
$\theta_1, \theta_2 \leftarrow$ **initialiser**(K)
$(x_1, y_1) \leftarrow (0, \frac{3}{2}a)$
$(x_2, y_2) \leftarrow (0, -\frac{3}{2}a)$
$e \leftarrow p_0$
for $i = 1$ jusqu'à 7 **do**
    $e \leftarrow e + p_i \times 2^i$
end for
$(x_1, y_1, \theta_1) \leftarrow f^e(x_1, y_1, \theta_1)$
$(x_2, y_2, \theta_2) \leftarrow f^e(x_2, y_2, \theta_2)$
$I_{0,1} \leftarrow p_1$
$I_{0,2} \leftarrow p_{L-2}$
$i \leftarrow 1$
$j \leftarrow i \mod (L-1)$
$n \leftarrow 2 \times p_{j+1} + p_j$
$l \leftarrow 0$
while $l < \left\lceil \frac{N}{32} \right\rceil$ **do**
    $(x_1, y_1, \theta_1) \leftarrow f^{n+1}(x_1, y_1, \theta_1)$
    $(x_2, y_2, \theta_2) \leftarrow f^{n+1}(x_2, y_2, \theta_2)$
    if $(x_1)^2 + (y_1)^2 = r^2$ **then**
        $(x_1, y_1, \theta_1) \leftarrow f(x_1, y_1, \theta_1)$
    end if
    if $(x_2)^2 + (y_2)^2 = r^2$ **then**
        $(x_2, y_2, \theta_2) \leftarrow f(x_2, y_2, \theta_2)$
    end if
    $I_1 \leftarrow E(x_1, y_1)$
    $I_2 \leftarrow E(x_2, y_2)$
    $RS \leftarrow RS||(I_1 \bigoplus I_2)$
    $l \leftarrow l + 1$
    $i \leftarrow i + 1$
    $j \leftarrow 2 \times i \mod (L-1)$
end while
Fin

2.3 Analyse de sécurité

Un GNPA devrait vérifier les propriétés de sécurité pour résister aux attaques. L'analyse de sécurité doit être faite avec soin pour évaluer la qualité des séquences. Nous étudions dans les paragraphes suivants la taille de l'espace clé, la sensibilité aux conditions initiales et le niveau d'aléatoire des séquences. Dans l'étude suivante, nous avons fixé r à $\frac{a}{2}$.

2.3.1 L'espace des clé

La taille de la clé est l'un des critères selon lesquels un système cryptographique doit être robuste. Une taille importante rend les attaques par force brute impossibles. Notre algorithme a pour clé d'initialisation une chaîne binaire de toute taille mentionnée ci-dessus. Le billard à deux particules nécessite exactement 128 bits pour calculer ses orientations initiales. Ces 128 bits sont extraits via un pointeur qui traverse la clé. Cela nous amène à dire que la taille de la zone clé est suffisamment grande pour être attaquée de manière exhaustive.

2.3.2 La sensibilité à un bit de changement dans la clé

La sensibilité à un petit changement dans la clé est l'une des propriétés essentielles pour un GNPA. Autrement dit, une infime déférence dans les graines du système doit provoquer un grand changement dans les séquences pseudo-aléatoires. Cette propriété rend le générateur de haute sécurité contre les attaques statistiques et différentielles, et ainsi les séquences ne peuvent pas être cassées même s'il y a une petite différence entre les clés. En fait, pour l'analyse du comportement chaotique du générateur, nous utiliserons les critères suivantes : la distance de Hamming, UACI et NPCR.

La distance de Hamming

La distance de Hamming (DH) entre deux séquences $SF^i = x_{1,i}\, x_{2,i} \ldots x_{N,i}$ et $SF^j = x_{1,j}\, x_{2,j} \ldots x_{N,j}$ de longueur égale N est le nombre

$$DH(SF^i, SF^j) = card\left\{d / \; x_{d,i} \neq x_{d,j}\right\}.$$

La DH entre deux séquences binaires est donnée par :

$$DH(SF^i, SF^j) = \sum_{t=1}^{N} x_{t,i} \bigoplus y_{t,i}$$

Dans le cas où les deux séquences F^i et SF^j sont indépendant entre eux, cette distance est généralement presque égale à $\frac{N}{2}$, qui donne $\frac{DH(SF^i, SF^j)}{N}$ est d'environ $0,5$.

Pour quantifier la sensibilité à la clé dans le GNPA proposé, nous précédons comme suit :

1) Par la clé $K_0 = "GUENNOUN"$ de la représentation binaire en code $ASCII$ est $K_0 = (01010101\ 01001110\ 01000011\ 01001000\ 01000001\ 01010010\ 01001001\ 01000110\ )_2$, nous générons une séquence pseudo-aléatoire S^0 de taille $N = 10^6$;
2) Nous générons un ensemble de clés $\{K_i\}_{1 \leq i \leq 64}$ en modifiant un bit parmi les 64 bits de K_0. Le $i^{\text{ème}}$ de K_0 et modifié pour trouver K_i (nous remplaçons k_i par $\overline{k_i} = 1 - k_i$) ;
3) Nous générons 64 séquences $\left\{S^i\right\}_{1 \leq i \leq 64}$ de longueur $N = 10^6$ par les clés $\{K_i\}_{1 \leq i \leq 64}$;
4) nous calculons l'ensemble des valeurs $\left\{\frac{DH(SF^0, SF^i)}{N}\right\}_{1 \leq i \leq 64}$

les valeurs $\frac{DH(SF^0, SF^i)}{N}$ entre les séquences sont indiquées dans le graphe 2.3.

D'après les résultats obtenus, les différences de proportions entre les séquences sont d'environ 0.5, ce qui indique que le générateur proposé est extrêmement sensible aux conditions initiales.

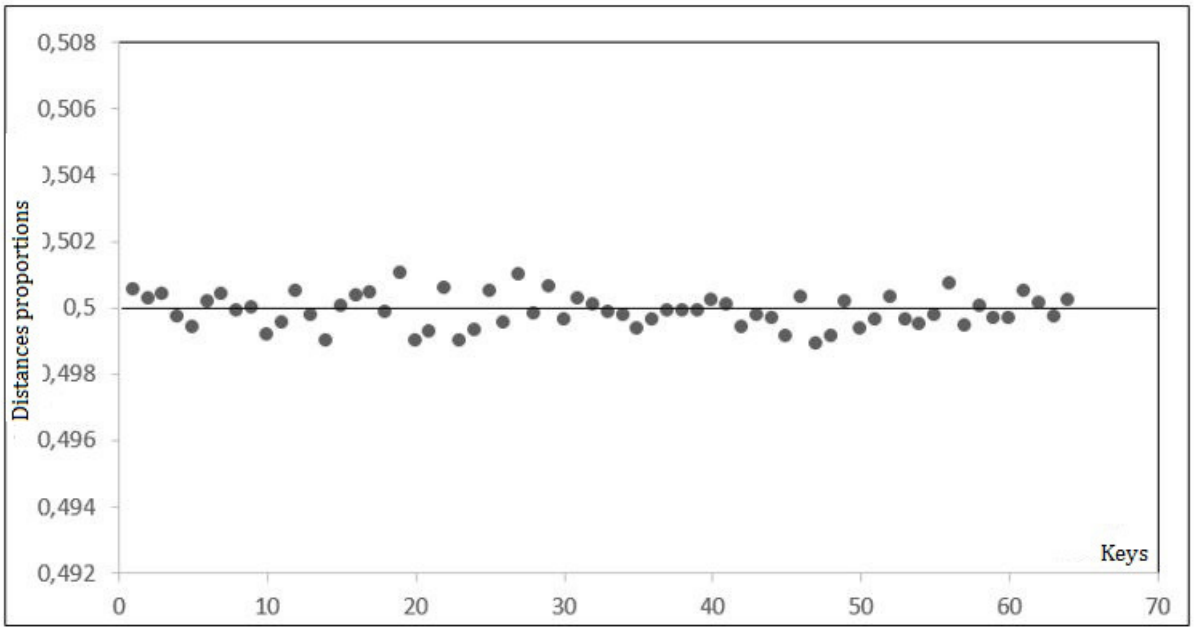

FIGURE 2.3 – La représentation des distances de Hamming

L'attaque différentielle

Dans cette partie, nous évaluons la sensibilité de notre générateur en utilisant autre critères parmi les plus courants pour mesurer la capacité d'attaque différentielle qui sont [109] : le nombre de taux de changement de pixels (NPCR) et l'intensité unifiée moyenne de changement (UACI). Pour analyser la différence entre S^0 générée par K_0 et l'ensemble de séquences $\{SF^i\}_{1\leq i\leq 64}$ généré par $\{K_i\}_{1\leq i\leq 64}$ de même taille $N = 10^6$, nous utilisons les deux mesures NPCR et UACI définis par :

$$NPCR(SF^0, SF^i) = \frac{\sum_{i=1}^{N} D_{(SF^0,SF^i)}(k)}{N} \times 100\%,$$

où $D_{(SF^0,SF^i)}(k)$ est une fonction définit comme suit :

$$D_{(SF^0,SF^i)}(k) = \begin{cases} 0 \text{ si } x_{k,0} = x_{0,i} \\ 1 \text{ si } x_{k,i} \neq x_{k,i} \end{cases}$$

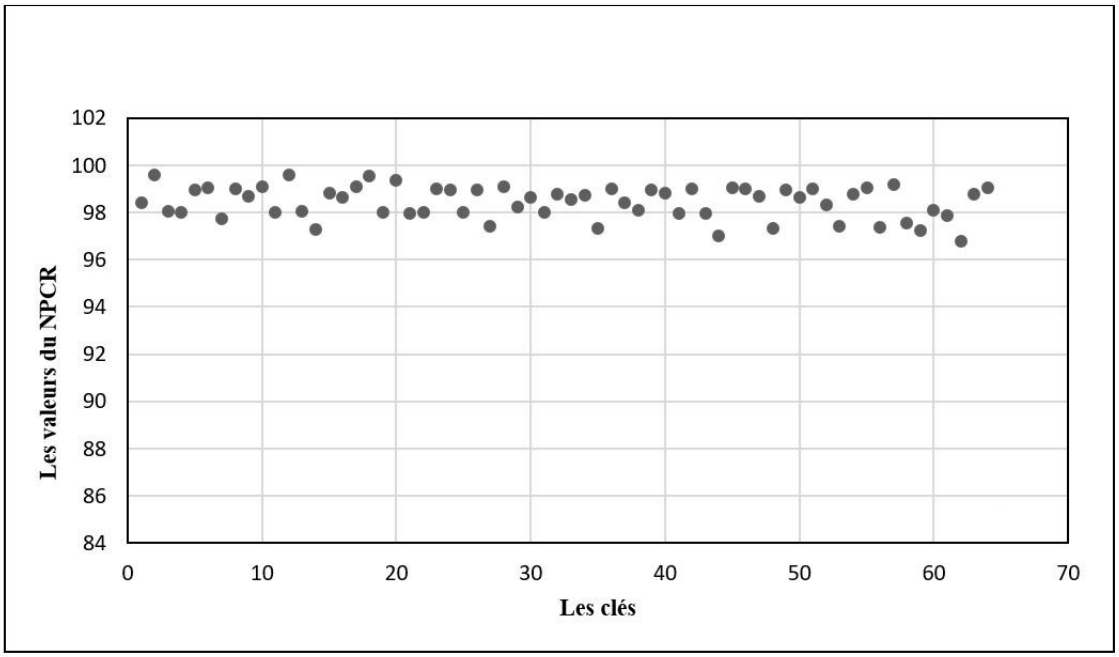

FIGURE 2.4 – Les résultats du NPCR entre la séquence SF^0 et les séquences $\{SF^i\}_{1\leq i\leq 64}$

$S^0(k)$ est la valeur du $k^{\text{éme}}$ bit de la séquence S^0 et

$$UACI_{(SF^0,SF^i)}(k) = \frac{\sum_{i=1}^{N} \frac{|SF^0(k) - SF^i(k)|}{255}}{N} \times 100\%.$$

Pour de séquences indépendants et réellement aléatoires, les valeurs attendus sont $NPCR_{attendu} = 99.609\%$ et $UACI_{attendu} = 33.464\%$. Les résultats de mesure du $NPCR(SF^0, \{SF^i\}_{1\leq i\leq 64})$ et $UACI(SF^0, \{SF^i\}_{1\leq i\leq 64})$ entre les séquences sont montrés respectivement dans le 2.4 et 2.5.

Les résultats obtenus sont tous aux voisinage des valeurs attendus, ce qui indique que le générateur proposé est purement sensible aux conditions initiales.

La sensibilité à une petite perturbation de la clé de notre générateur est due à deux raisons :

1. Le générateur est basé dans sa construction sur un système de billard chaotique, de sorte que les séquences générées héritent du chaos et de l'impré-

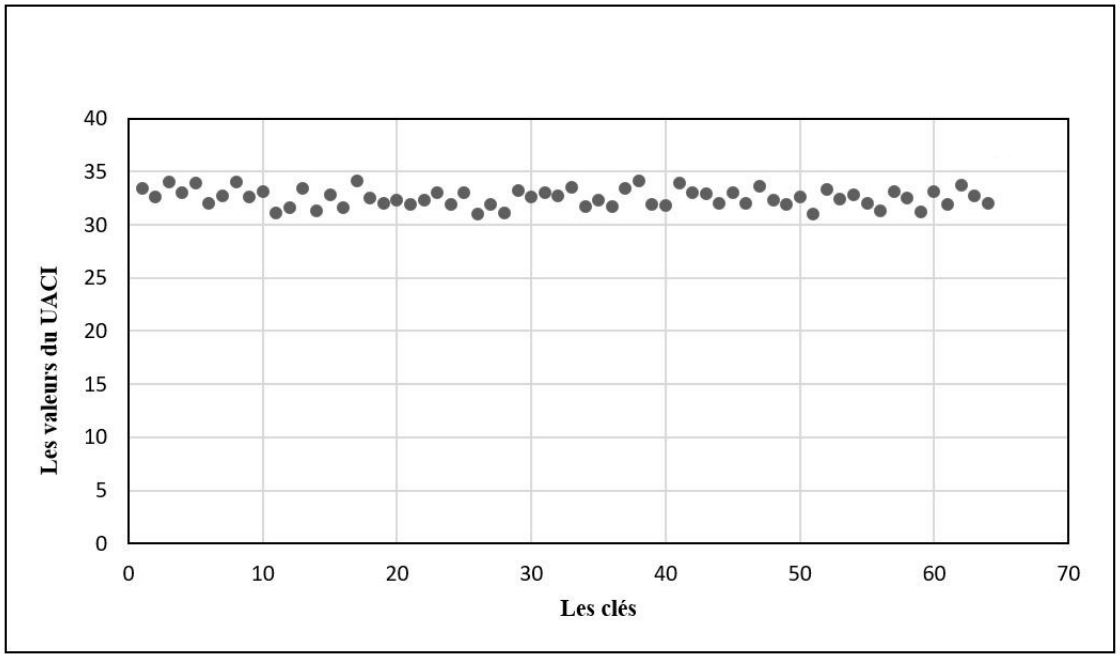

FIGURE 2.5 – Les résultats du UACI entre la séquence SF^0 et les séquences $\{SF^i\}_{1\leq i\leq 64}$

visibilité du billard. Un nombre supplémentaire d'itérations extraites directement du mot de passe permet au générateur de profiter au maximum du chaos offert par le billard ;

2. Les angles d'initialisation sont tirés du K, en utilisant un pointeur qui pointe vers différentes positions jusqu'à sa couverture totale. En effet, une différence de bit entre deux touches peut entraîner une orientation différente vers les particules et donc vers les séquences générées.

Le billard de Sinaï est chaotique pour toutes les valeurs du rayon r, mais il y a une différence dans le niveau de chaos pour chaque valeur de r comme indiqué dans [31] où l'EL est exprimé en fonction de r. Par conséquent, l'utilisateur peut contrôler le niveau du générateur de chaos par un paramètre d'entrée à l'algorithme (r, où $0 < r < \frac{3}{2}a$).

2.3.3 Coefficient de corrélation

Le test de corrélation est utilisé pour analyser la relation entre les les séquences générées. En effet, un GNPA robuste génère des séquences avec une faible corrélation. L'objectif de cette partie est de calculer le coefficient de corrélation des séquences générées par des clés distinctes choisies aléatoirement.

Le coefficient de corrélation de Pearson [67] $C(S^1, S^2)$ de deux séquences $SF^1 = x_1^1 x_2^1 \dots x_N^1$ et $SF^2 = x_1^2 x_2^2 \dots x_N^2$ de longueur $N = 10^6$ est défini par :

$$C(SF^1, SF^2) = \frac{\sum_{i=1}^{N}(x_{1,i} - \overline{x_1})(x_{2,i} - \overline{x_2})}{\sqrt{\sum_{i=1}^{N}(x_{1,i} - \overline{x_1})^2 \sum_{i=1}^{N}(x_{2,i} - \overline{x_2})^2}}$$

où $x_{1,i}$ et $x_{2,i}$ sont des entiers 32 bits, avec

$$\overline{x_1} = \frac{1}{N}\sum_{i=1}^{N} x_{1,i} \text{ et } \overline{x_2} = \frac{1}{N}\sum_{i=1}^{N} x_{2,i}$$

En général, la corrélation C varie dans l'intervalle $[-1; 1]$, s'il y a une forte corrélation entre les deux séquences alors $C \approx \pm 1$. Par contre si $C \approx 0$ alors les deux séquences sont indépendantes et il n'y a pas de corrélation entre elles. Dans notre étude, nous générons un ensemble de séquences $\{S_i\}_{1 \leq i \leq 10}$ de longueur $N = 10^6$ par des clés différentes $\{K_i\}_{1 \leq i \leq 10}$, puis nous effectuons une comparaison par (3) entre les séquences. Les résultats indiqués dans le tableau 2.1, les coefficients de corrélation sont tous près du zéro.

$C(SF^i, SF^j)$	SF^2	SF^3	SF^4	SF^5	SF^6	SF^7	SF^8	SF^9
SF^1	0.0028	0.0174	0.0083	0.0247	0.00536	0.0106	0.0304	0.0293
SF^2		0.0049	0.0021	0.0074	0.00625	0.0303	0.0067	0.0083
SF^3			0.0167	0.0013	0.03271	0.0171	0.0165	0.0341
SF^4				0.0261	0.00739	0.0215	0.0426	0.0057
SF^5					0.00853	0.0192	0.0091	0.0074
S^6						0.0089	0.0153	0.0049
SF^7							0.0613	0.0075
SF^8								0.0203

TABLE 2.1 – Les coefficients de corrélations entre plusieurs séquences binaires

2.3.4 Le test d'histogramme

L'histogramme et l'entropie sont les deux tests largement utilisés pour mesurer l'étendue du caractère aléatoire et évaluer la distribution statistique des séquences d'un GNPA. Les clés générées par un générateur acceptable doivent être uniformes au niveau d'histogramme. Pour ce test, nous générons deux séquences $SF^1 = x_{1,1}\, x_{2,1} \ldots x_{20\,000,1}$ et $SF^2 = x_{1,2}\, x_{2,2} \ldots x_{20\,000,2}$ de 640 000 bits par deux clés différentes, choisies aléatoirement, chaque séquence est composé par la concaténation des sous-séquences de 32 bits. L'histogramme de S^1 est une fonction $his_{SF^i}(x_{j,i})$, pour chaque $j \in [\![1, 20\,000]\!]$, $his_{S^i}(x_{j,i})$ est égal au nombre d'occurrences de $x_{j,i}$ dans S^i. Nous appelons $his_{SF^i}(x_{j,i})$, la fréquence de $x_{j,i}$ dans SF^i. Une séquence SF^i a un histogramme uniforme si tous les éléments de son le jeu de symboles a la même fréquence. Le diagramme d'histogramme de chaque séquence est présenté dans la figure 2.6.

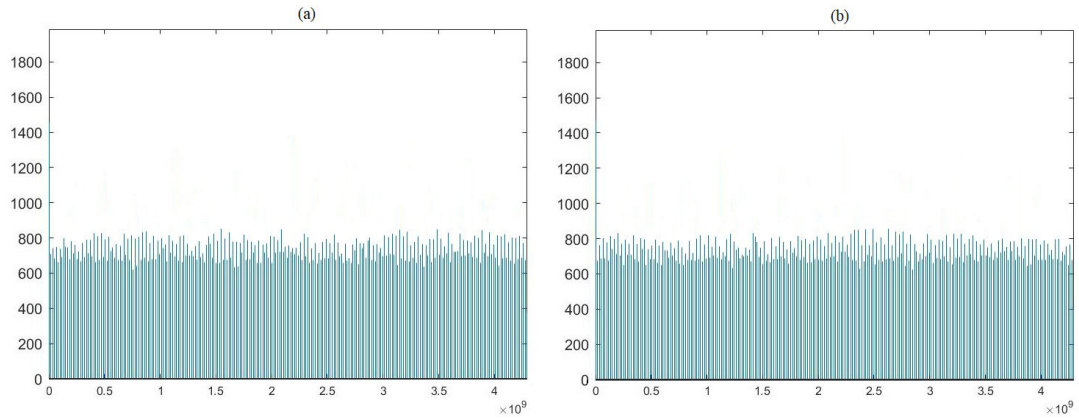

FIGURE 2.6 – (a) l'histogramme du séquence SF^1, (b) l'histogramme du séquence SF^2

Nous remarquons que la distribution des nombres séquences sont très près de l'uniformité. Nous pouvons dire que le générateur il a de bonne propriété statistique.

Dans la section suivante, nous examinons le caractère aléatoire du générateur par des tests statistiques NIST (National Institute of Standards and Technology), qui sont considérés comme les plus valorisés.

2.3.5 Les tests statistiques d'aléa

La suite des tests du NIST [91] est un progiciel statistique, le résultat de la collaboration entre la Division Ingénierie statistique au NIST (National Institute of Standards and Technology) et la division de la sécurité informatique. Cette suite comprend des tests 16, développés pour quantifier et évaluer le degré de séquences binaires aléatoires produites par des générateurs cryptographiques. Pour chaque test statistique, une P_{valeur} est calculée à partir de la séquence de bits. Ce P_{valeur} est comparé à un seuil prédéfini α, également appelé niveau de signification. Si P_{valeur} est supérieure α alors la séquence testée est considérée comme étant aléatoire avec un niveau de confiance de $1 - \alpha$, et la séquence passe le test statistique avec succès et ne parvient pas autrement. Généralement, comme suggéré par le NIST, α est mis à sa valeur par défaut de $0,01$, qui indique que l'on

Test Name	**The P_{value}**	**The proportion**	**Result**
Frequency	0.695200	992/1000	Success
Block-Frequency	0.861264	990/1000	Success
Cumulative Sums (1)	0.169981	995/1000	Success
Cumulative Sums (2)	0.978072	991/1000	Success
Runs	0.542228	985/1000	Success
Longest Run	0.709558	985/1000	Success
Rank	0.169981	995/1000	Success
FFT	0.080027	984/1000	Success
Non-Overlapping	0.505854	987/1000	Success
Overlapping	0.041169	991/1000	Success
Universal	0.334538	991/1000	Success
Approximate Entropy	0.851383	989/1000	Success
Random Excursions	0.478175	616/625	Success
Random Excursions Variant	0.470796	616/625	Success
Serial (1)	0.919131	986/1000	Success
Serial (2)	0.334538	980/1000	Success
Linear Complexity	0.948298	992/1000	Success

TABLE 2.2 – Les résultats des tests du NIST pour le GNPA proposé

pourrait s'attendre à une séquence parmi 100 séquences à être rejetée.

Pour tester notre GNPA et comme recommandé par le NIST, nous avons généré nous avons généré 1000 séquences, la taille de chaque séquence est 10^6 à partir de clés sélectionnées de manière aléatoire. Les résultats des tests sur les séquences sont présentés dans le tableau 2.2

Le taux minimal de réussite pour le test Random Excursions (Variant) est approximativement de 609 pour un échantillon de 625 séquences binaires. Le taux minimal de réussite pour les autres tests est approximativement de 980 pour un échantillon de 1000 séquences binaires. Nous pouvons voir que le nombre des séquences qui ont réussi à passer chaque test est plus grand que le taux minimal. Par conséquent, le générateur proposé a réussi tous les tests statistiques NIST. Nous pouvons conclure que les nombres générés par ce GNPA sont aléatoires.

2.4 Conclusion

Nous avons proposé un nouvel GNPA, à base de deux systèmes dynamiques d'un billard purement chaotique. Dans chaque étape, l'algorithme produit deux sous-séquences de 32 bits en reposant sur les systèmes de deux particules. Les deux séquences sont mélangées par l'opération de Xor avant d'entrer dans la construction de la séquence finale. Après une analyse rigoureuse, le GNPA a montré des résultats encourageants, il a passé tous les tests de sécurité avec succès. Le générateur a hérité de l'imprévisibilité du billard de Sinaï. Il peut être utilisé pour des applications cryptographiques critiques. On peut conclure que les systèmes de billard chaotique sont de bons candidats pour entrer dans une nouvelle conception de système cryptographique.

Chapitre 3

La conception d'un système de chiffrement des images basé sur le système de billard Sinaï

La sécurité des données multimédias transmises via une connexion non sécurisée est l'une des exigences fondamentales des réseaux de télécommunications. La demande pour les systèmes cryptographiques est devenue intense dans divers domaines, par exemple la communication Internet, l'armée, les soins de santé, les applications de messagerie photo sur les téléphones portables, les systèmes multimédias, l'imagerie médicale, la télémédecine, la confidentialité des documents gouvernementaux, etc. Cependant, les systèmes de chiffrement appropriés doivent garantir les deux exigences mentionnées par Shannon [94] : confusion et la diffusion. Les algorithmes de chiffrement traditionnels tels que DES, IDEA ou AES remplissent ces conditions pour les données de texte, mais ils ne conviennent pas à un type de données à grande échelle comme les images avec une capacité d'information importante avec une forte corrélation de pixels. À cette fin, de nouveaux algorithmes de chiffrement ont été proposés et basés sur différentes théories [24, 32, 53, 79, 81, 95, 110–112]. La théorie du chaos est celle qui a attiré plus d'attention, en raison de la similitude entre les propriétés offertes par les SCs et

les besoins cryptographiques. Dans le même contexte, Fridrich [37] a proposé le principe de confusion-diffusion du chiffrement d'images. Après cet article, divers systèmes de chiffrement d'images basés sur le chaos ont été conçus pour améliorer la sécurité des algorithmes de chiffrement d'images. Sauf que, la différence entre ces algorithmes réside dans les techniques utilisées pour la confusion et la diffusion des pixels, y compris les SCs appliqués à cette objectif. Par exemple, Liu et al. [73] a proposé un schéma de chiffrement basé sur le système logistique utilisant une permutation spatiale au niveau du bit. Wang et al. [107] ont conçu un algorithme de chiffrement d'images de blocs chaotiques basé sur des SCs et la technique de croissance aléatoire dynamique. Chen et al. [21] ont suggéré un schéma de chiffrement basé sur le système Baker unidimensionnel où un flux de clés de diffusion est obtenu à partir de la matrice de permutation, qui est produite et maintenue dans la phase de permutation. Zhu et al. [123] ont proposé un schéma de permutation au niveau du bit pour le chiffrement d'image basé sur une carte de discussion d'Arnold et la carte logistique pour la diffusion. Huang et al. [49] a proposé un algorithme de chiffrement d'image basé sur la fonction non linéaire de Chebyshev, où un flux de clés est généré, puis plusieurs permutations sont appliquées aux pixels.

Dans ce chapitre, nous proposons un nouvel algorithme de chiffrement des images basé sur les systèmes des billard chaotiques [28] qui ont des propriétés chaotiques très intéressantes. Malgré ses bonnes propriétés, l'application de ces systèmes en cryptographie reste négligeable et n'a pas retenu l'attention des cryptographes. Parmi les raisons, est l'absence d'une formule explicite pour l'équation du mouvement d'une particule parcourant le billard. Dans notre concept, nous appliquons le billard de Sinaï [98] qui a de fortes propriétés chaotiques et a déjà prouvé ses performances dans la génération de séquences pseudo-aléatoires [19]. En effet, l'algorithme proposé est basé sur la marche aléatoire offerte par les systèmes de trois particules. Les coordonnées d'une particule sont utilisées pour générer les permutations pour la confusion des pixels de l'image, tandis que les co-

ordonnées des deux autres particules sont utilisées pour générer des séquences pseudo-aléatoires pour la diffusion.

Ce chapitre est organisé comme suit, dans la section 1 nous présentons l'image numérique, les schémas de chiffrement des images et nous avons effectué une discussion sur des données spatiales d'image et les propriétés des schéma de chiffrement idéal à ce type de données, la section 2 donne une description détaillée de notre schéma de chiffrement, la section 3 est consacrée à l'analyse de sécurité du schéma proposé et une conclusion est tirée dans la section 4.

3.1 Le chiffrement des images numérique

Dans cette section, nous pressentons quelques vocabulaires du traitement des images numériques et une discussion sur les systèmes chiffrement des images et ses particularités.

3.1.1 Présentation d'une image numérique

Une image numérique $I_{M\times N}$ est une représentation d'une image réelle comme une matrice $I_{M\times N}$ de taille $M \times N$. Afin de traduire l'image en nombres, elle est divisée en petites zones appelées pixels. Dans chaque pixel, le dispositif d'imagerie enregistre un nombre, ou un petit ensemble de nombres, qui décrivent certaines propriétés de ce pixel, telles que sa luminosité (l'intensité de la lumière) ou sa couleur. Le pixel peut être vu comme un point, il a deux caractéristiques : les coordonnées (i, j) et la valeur $I(i, j)$ avec $0 \leq i \leq M-1, 0 \leq J \leq N-1$. Les coordonnées identifient de manière univoque un seul pixel et ce sont les index du pixel à l'intérieur de la matrice. Le type de données de $I(i, j)$ change en fonction du type d'image numérique :

Image en niveaux de gris

Une image en niveaux de gris I_g est généralement considérée comme une matrice de pixels. La valeur de chaque pixel de I_g est généralement représentée par

un nombre de 0 à 255 qui représente l'intensité de la lumière à ce point. Une valeur de 255 signifie blanc, tandis qu'une valeur de 0 est noir. Certains systèmes utilisent une représentation en niveaux de gris plus précise, avec plus de valeurs pour représenter l'intensité lumineuse (par exemple, $[\![0, 4095]\!]$ (image de 12 bits par pixel) ou $[\![0, 65535]\!]$ (image de 16 bits par pixel).

Image couleur

Une image en couleurs I_{rgb} correspond à un mélange des trois couleurs primaires rouge, vert et bleu stockées dans trois matrices $I_{r\,M\times N}([\![0;255]\!])$, $I_{g\,M\times N}([\![0;255]\!])$ et $I_{b\,M\times N}([\![0;255]\!])$. Un pixel est généralement un triple de composantes de pixels qui représente l'intensité de chaque couleur.

3.1.2 Discussion

Le chiffrement d'image est de changer l'image en un format illisible afin qu'aucun utilisateur non autorisé ne puisse la décrypter. Cela peut être fait avec une modification au niveau des pixels de l'image, c'est-à-dire la valeur des pixels ou leur position dans la matrice d'origine, afin de protéger les informations. Les données d'image ont des propriétés spéciales telles que la redondance élevée et la forte corrélation entre les pixels adjacents qui impose des exigences particulières à toute technique de chiffrement. Hors, le chiffrement d'une image numérique en tant que flux binaire sans tenir compte à la relation qui existe entre les pixels, peut être assez faible et inefficaces. Par conséquent, compte tenu des propriétés spéciales des images numériques, de nombreux schémas de chiffrement d'images ont été proposés en utilisant différents types de méthodes [117, 121, 122]. Les SCs sont tout à fait adaptés à la cryptographie et ont été largement utilisés dans le chiffrement d'images depuis les années 90. Les niveaux de sécurité des schémas de chiffrement des images dépendent fortement des performances des SCs utilisés. Pour certains SCs unidimensionnels ($1D$), leurs orbites chaotiques sont assez

simples et peuvent être prédites facilement. Une fois certaines informations extraites [7, 48], leurs états initiaux peuvent être estimés à l'aide de certaines techniques [4, 22, 83, 108, 113]. En utilisant un SC avec un comportement chaotique simple, le schéma de chiffrement d'image correspondant peut être facilement attaqué [68, 69, 99]. Les SCs de grande dimension ont des comportements chaotiques complexes et leurs orbites chaotiques sont difficiles à prévoir. Cependant, ils présentent également certaines faiblesses, telles qu'une analyse complexe des performances et des coûts de mise en œuvre élevés [115]. En général, les algorithmes basés sur les SCs sont très populaires auprès des chercheurs comme une bonne solution pour le chiffrement de l'image au cours des dernières années. Cependant, puisque les cartes chaotiques deviennent plus familières au public et que l'espace clé est petit, il existe une certaine faiblesse dans la sécurité. Aussi longtemps qu'une petite information a priori, il est possible de prédire certains comportements des SCs traditionnels dans certaines circonstances. En d'autres termes, il peut fournir des services privilégiés à un attaquant en estimant les paramètres et les valeurs initiales dans un système d'image basé sur un système chaotique [4].

Dans ce chapitre, un algorithme de chiffrement des images est proposé en se basant sur trois systèmes d'un billards à ($2D$) fortement chaotique. Dans le processus de transformation des informations, nous avons proposé de nouvelle techniques de chiffrement ce qui les rendre illisibles pour quiconque, sauf ceux qui possèdent la clé de chiffrement. D'un autre côté, le déchiffrement de l'image récupère les informations exactes de l'image clair.

3.2 Proposition d'un algorithme de chiffrement des images

L'algorithme de chiffrement proposé a trois paramètres d'entrée : une image I de taille $M \times N$, une clé $K = (k_0 k_1 k_2 \dots k_{L-1})_2$ de longueur arbitraire L et un entier CD (défini par défaut à 1).

Avant le chiffrement, nous aimerions remodeler l'image I_x et la traiter comme une table TD_x^0 de taille $M \times N$ où $TD_x^0[i \times N + j] = I_x(i,j)$ avec $0 \leqslant i < M$ et $0 \leqslant j < N$.

Après le calcul de trois états d'initialisation $S_{0,1}$, $S_{0,2}$ et $S_{0,3}$ des trois particules p_1, p_2 et p_3 comme indiqué dans l'algorithme 3, les particules sont invitées à effectuer n_0 d'itérations (collisions) avec :

$$n_0 = \sum_{i=0}^{7} k_i$$

Notre système de chiffrement est basé sur CD tours de deux opérations traditionnellement utilisées dans les systèmes de chiffrement des images : confusion et diffusion (Figure 3.2). Maintenant, pour l'exécution de ces opérations, de nombreuses permutations selon l'algorithme 4 et séquences aléatoires comme indiqué dans l'algorithme 5 sont générées en s'appuyant sur les valeurs flottantes chaotiques prises par les coordonnées des particules lors de ses collisions avec le carré billard. Les coordonnées (x, y) des particules sont converties par une fonction EM_c, où $c \in \{256, M \times N\}$, pour générer l'entier $EM_c(x,y)$ $(0 \leqslant EM_c(x,y) < c)$.

La fonction EM_c est définie comme suit :

$$\begin{aligned} EM_c(x,y) &= \begin{cases} \left[\frac{x}{a}2^{[\log_2 c]+1}\right] \bmod c & \text{si } x \geq 0 \text{ et } |y| = a \\ \left[(1+\frac{x}{a})2^{[\log_2 c]+1}\right] \bmod c & \text{si } x < 0 \text{ et } |y| = a \\ \left[\frac{y}{a}2^{[\log_2 c]+1}\right] \bmod c & \text{si } y \geq 0 \text{ et } |x| = a \\ \left[(1+\frac{y}{a})2^{[\log_2 c]+1}\right] \bmod c & \text{si } y < 0 \text{ et } |x| = a \end{cases} \\ &= (b_0 b_1 \dots b_{c-1})_2, \end{aligned}$$

où $(b_0 b_1 \dots b_{c-1})_2$ est la présentation binaire de $EM_c(x,y)$. La fonction à valeur entière EM_c utilisé pour générer des permutations de confusion lorsque $c = M \times N$ et des séquences pseudo aléatoires pour la diffusion lorsque $c = 255$.

Maintenant, pour le chiffrement des images en niveaux de gris TD_g^{l-1} au $l^{\text{ième}}$ tours, où $1 \leq l \leq CD$, nous générons la permutation P_x^l en comptant sur les séquences chaotiques générées par $p_{(l-1)\%3+1}$. Ensuite, nous générons $M \times N$ séquences aléatoires $\{d^{l,i}\}_{i\in[\![1;M\times N]\!]}$ qui varient dans l'intervalle $[\![0;255]\!]$ par $p_{l\%3+1}$ et $p_{(l+1)\%3+1}$. Nous appliquons la permutation P_g^l sur TD_g^{l-1} pour obtenir TC_g^l, puis nous masquons ses valeurs de pixel par $\{d^{l,i}\}_{i\in[\![1;M\times N]\!]}$ selon le mode CFB (Cipher Feedback) pour obtenir TD_g^l. Notez que, nous passons d'une manière circulaire aux particules, l'une pour la confusion et les autres deux pour la diffusion. Enfin, nous obtenons un tableau, noté TD_g^{CD}, en le convertissant en une image chiffrée C_g de même dimension $M \times N$, avec la valeur de pixel $C_g(i,j) = TD_g^{CD}[i \times N + j]$.

En ce qui concerne l'image couleur, on procède de la même façon, décrite ci-dessus, pour les trois canaux T_r, T_v et T_b. Nous générons des permutations $P_{1,r}^1$ (respectivement $P_{2,v}^1$, $P_{3,b}^1$) par p_1 (respectivement p_2, p_3), et $M \times N$ séquences aléatoires $\{d_{2,3}^{1,i}\}_{i\in[\![1;M\times N]\!]}$ (respectivement $\{d_{1,3}^{1,i}\}_{i\in[\![1;M\times N]\!]}$, $\{d_{1,2}^{1,i}\}_{i\in[\![1;M\times N]\!]}$) par p_2 et p_3 (respectivement p_1 et p_3, p_1 et p_2). Nous appliquons $P_{1,r}^1$ (respectivement $P_{2,g}^1$, $P_{3,b}^1$) sur T_r (respectivement T_v, T_b) pour obtenir TC_r^1 (respectivement TC_g^1, TC_b^1). Ensuite, nous masquons les valeurs des pixels du tableau TC_r^1 (respective-

ment TC_g^1, TC_b^1) par un XOR avec $\{d_{2,3}^{1,i}\}_{i\in[\![1;M\times N]\!]}$ (respectivement $\{d_{1,3}^{1,i}\}_{i\in[\![1;M\times N]\!]}$, $\{d_{1,2}^{1,i}\}_{i\in[\![1;M\times N]\!]}$) pour obtenir TD_r^1 (respectivement TD_g^1, TD_b^1). Nous répétons ces processus CD fois pour chaque canal de couleur. Enfin, nous aurons TD_r^{CD}, TD_g^{CD} et TD_b^{CD}, nous le convertirons en une image chiffrée.

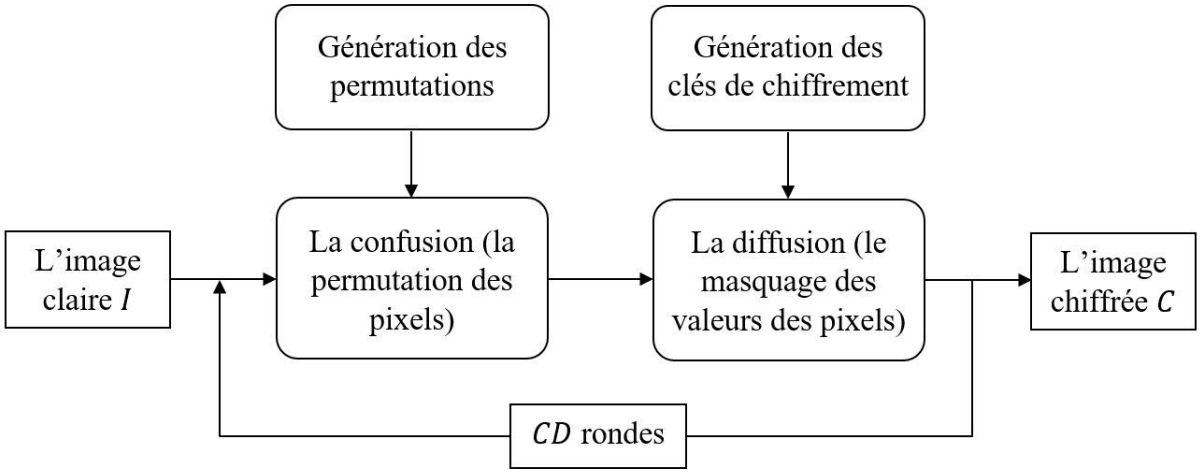

FIGURE 3.1 – L'architecture de chiffrement d'une image : confusion-diffusion

3.2.1 Le calcul des états initiaux

Nous générons trois états d'initialisation $\{S_{0,p}\}_{1\leq p\leq 3} = \{(x_{0,p}, y_{0,p}, \overrightarrow{v_{0,p}})\}_{1\leq p\leq 3}$ où $(x_{0,p}, y_{0,p})$ sont les positions de départ et $\overrightarrow{v_{0,p}}$ les orientations initiaux des particules avec :

$$r \leq |x_{0,p}, y_{0,p}| < a$$

et

$$\overrightarrow{v_{0,p}} = \big(\cos(\theta_{0,p}), \sin(\theta_{0,p})\big)$$

où

$$0 \leq \theta_{0,p} < 2\pi.$$

En effet, comme indiqué dans l'algorithme 3, pour générer les trois états, on utilise une clé secrète $K = (k_0k_1k_2\ldots k_{L-1})_2$ de longueur arbitraire L, où k_i sont les bits de sa représentation en mode ASCII. Les procédures de génération sont

les suivantes :

$$x_{0,p} = (-1)^{k_{L-p} \oplus k_p} \left(r + \frac{a-r}{2^s} \sum_{i=0}^{s-1} 2^i (\bar{k}_{L-6i-p-2} \oplus k_{6i+p-1} \right),$$

$$y_{0,p} = (-1)^{k_{L-p-3} \oplus k_{p+3}} \left(r + \frac{a-r}{2^s} \sum_{i=0}^{s-1} 2^i (\bar{k}_{L-6i-p-3} \oplus k_{6i+p+2} \right),$$

et

$$\theta_{0,p} = \frac{2\pi}{2^s} \sum_{i=0}^{s-1} 2^i (\bar{k}_{L-2ip-1} \oplus k_{2ip}),$$

avec

$$t = \left[\frac{L}{6} \right]$$

La nouvelle technique utilisée dans le calcul couvre presque tous les bits de la clé, ce qui permet aux états d'initialisation d'être sensibles à une petite perturbation de la clé. Par conséquent, un bit de différence entre deux clés donne des états complètement différents, ce qui a un effet important sur le reste du système.

3.2.2 Confusion et diffusion

Une description détaillée de la permutation pour confusion et du générateur de séquence pseudo-aléatoire pour la diffusion est présentée dans cette section.

Confusion : Concevoir une permutation

Nous considérons une particule dans le BS dont l'état de départ est S^0. Nous comptons sur les coordonnées des points de collision, en tirant parti d'un nombre supplémentaire d'itérations n_i calculées à partir de la clé K. Nous commençons par générer un tableau Q de taille $M \times N$:

$$Q[M \times N] = \{Q_1, Q_2, \ldots, Q_{M \times N}\}.$$

Algorithm 3 Calcul des états initiaux $S_{0,1}$, $S_{0,2}$ et $S_{0,3}$ (initialiser)

Début
Entrée : une clé $K = (k_0 k_1 k_2 \dots k_{L-1})_2$.
Sortie : un triplet d'états $S_{0,1}$, $S_{0,2}$, $S_{0,3}$.
$(x_{0,1}, y_{0,1}, \theta_{0,1}) \leftarrow (0,0,0)$
$(x_{0,2}, y_{0,2}, \theta_{0,2}) \leftarrow (0,0,0)$
$(x_{0,3}, y_{0,3}, \theta_{0,3}) \leftarrow (0,0,0)$
$s \leftarrow \left[\frac{L}{6}\right] - 1$
for $i = 0$ to s **do**
    $x_{0,1} \leftarrow x_{0,1} + 2^i \left(\overline{k}_{L-6i-1} \oplus k_{6i}\right)$
    $y_{0,1} \leftarrow y_{0,1} + 2^i \left(\overline{k}_{L-6i-4} \oplus k_{6i+3}\right)$
    $x_{0,2} \leftarrow x_{0,2} + 2^i \left(\overline{k}_{L-6i-2} \oplus k_{6i+1}\right)$
    $y_{0,2} \leftarrow y_{0,2} + 2^i \left(\overline{k}_{L-6i-5} \oplus k_{6i+4}\right)$
    $x_{0,3} \leftarrow x_{0,3} + 2^i \left(\overline{k}_{L-6i-3} \oplus k_{6i+2}\right)$
    $y_{0,3} \leftarrow y_{0,3} + 2^i \left(\overline{k}_{L-6i-6} \oplus k_{6i+5}\right)$
    $\theta_{0,1} \leftarrow \theta_{0,1} + 2^i \left(\overline{k}_{L-2i-1} \oplus k_{2i}\right)$
    $\theta_{0,2} \leftarrow \theta_{0,2} + 2^i \left(\overline{k}_{L-3i-1} \oplus k_{3i}\right)$
    $\theta_{0,3} \leftarrow \theta_{0,3} + 2^i \left(\overline{k}_{L-6i-1} \oplus k_{6i}\right)$
end for
$x_{0,1} \leftarrow (-1)^{k_{L-1} \oplus k_1} \left(r + x_{0,1} \times \frac{a-r}{2^s}\right)$
$y_{0,1} \leftarrow (-1)^{k_{L-4} \oplus k_4} \left(r + y_{0,1} \times \frac{a-r}{2^s}\right)$
$x_{0,2} \leftarrow (-1)^{k_{L-2} \oplus k_2} \left(r + x_{0,2} \times \frac{a-r}{2^s}\right)$
$y_{0,2} \leftarrow (-1)^{2k_{L-5} \oplus k_5} \left(r + y_{0,2} \times \frac{a-r}{2^s}\right)$
$x_{0,3} \leftarrow (-1)^{k_{L-3} \oplus k_3} \left(r + x_{0,3} \times \frac{a-r}{2^s}\right)$
$y_{0,3} \leftarrow (-1)^{2k_{L-6} \oplus k_6} \left(r + y_{0,3} \times \frac{a-r}{2^s}\right)$
$\theta_{0,1} \leftarrow \frac{2\pi}{2^s} \times \theta_{0,1}$
$\theta_{0,2} \leftarrow \frac{2\pi}{2^s} \times \theta_{0,2}$
$\theta_{0,3} \leftarrow \frac{2\pi}{2^s} \times \theta_{0,3}$
$S_{0,1} \leftarrow (x_{0,1}, y_{0,1}, \theta_{0,1})$
$S_{0,2} \leftarrow (x_{0,2}, y_{0,2}, \theta_{0,2})$
$S_{0,3} \leftarrow (x_{0,3}, y_{0,3}, \theta_{0,3})$
Fin

où

$$Q_i = EM_{M\times N}\left(proj_{(x,y)}\left(S^i\right)\right) + 1 \text{ for } 1 \leq i \leq M \times N$$

avec

$$S^i = f^{n_i+1}(S^{i-1}),$$

et

$$n_i = \begin{cases} 2k_{j+1} + k_j + 1 & \text{si } f^{2k_{j+1}+k_j+1}(S^{i-1}) \in \Gamma_5 \times V^1 \\ 2k_{j+1} + k_j & \text{sinon} \end{cases}, \tag{3.1}$$

où

$$j = 2 \times i \, mod(L-1),$$

et $proj_{(x,y)}\left(S^i\right)$ est la projection de l'état S^i sur le plan (Oxy).
Soit P la permutation d'identité

$$P = \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ 1 & 2 & 3 & \dots & n \end{pmatrix}.$$

P sera la permutation générée après une modification sur elle, comme indiqué dans la boucle suivante :

pour $i = 1$ jusqu'à $M \times N$ faire :

$$\begin{cases} P(i) & \leftarrow P(i) + P\left(Q_i\right) \\ P\left(Q_i\right) & \leftarrow P(i) - P\left(Q_i\right) \\ P(i) & \leftarrow P(i) - P\left(Q_i\right) \end{cases}.$$

Nous nous intéressons aux valeurs prises par les particules sur la bordure carrée du billard (c'est-à-dire $\cup_{i=1}^{4}\Gamma_i$), pour cela nous ajoutons une collision pour la par-

ticule si elle se trouve dans la bordure circulaire (c'est-à-dire Γ_5).

L'algorithme proposé pour générer la permutation a deux paramètres d'entrée, une clé et un état. Le nombre supplémentaire de collisions n_i extraites directement de la clé permet à la permutation de tirer le meilleur parti de l'imprévisibilité chaotique offerte par le billard. En effet, même s'il y a le même état dans l'entrée, un peu de différence dans la clé peut donner une permutation totalement différente.

Algorithm 4 Algorithme de permutation (permutation)

Début
Entrée : une clé $K = (k_0 k_1 k_2 \dots k_{L-1})_2$, un état S^0 et un entier $M \times N$
Sortie : une permutation P de taille $M \times N$, et un état $S^{M \times N}$.
$n \leftarrow 2k_1 + k_0$
for $i = 1$ to $M \times N$ **do**
  $S^i \leftarrow f^{n+1}\left(S^{i-1}\right)$
  if $(S^i \in \Gamma_5 \times V^1)$ **then**
    $S^i \leftarrow f\left(S^i\right)$
  end if
  $Q_i \leftarrow \left(E_{M \times N}\left(proj_{(x,y)}(S^i)\right)\right)$
  $P(i) \leftarrow i$
  $j \leftarrow (2 \times i) mod(L-1)$
  $n \leftarrow 2k_{j+1} + k_j$
end for
for $i = 1$ to $M \times N$ **do**
  $P(i) \leftarrow P(i) + P\left(Q_i\right)$
  $P\left(Q_i\right) \leftarrow P(i) - P\left(Q_i\right)$
  $P(i) \leftarrow P(i) - P\left(Q_i\right)$
end for
Fin

Diffusion : pseudo-random generator

Dans la partie diffusion, nous utilisons un générateur de nombres pseudo-aléatoires déjà proposé dans [19], il possède de bonnes propriétés statistiques et des performances chaotiques. Nous considérons deux particules p_j et p_k qui ont pour état de départ S_j^0 et S_k^0. A chaque étape i, nous effectuons des itérations $n_i + 1$

pour les deux particules, si la particule p_j (respectivement p_k) se situe au dernier point de collision situé sur la bordure circulaire du billard, nous en ajoutons un autre. collision pour p_j (respectivement p_k). Comme décrit dans l'algorithme 3, d^i $(0 \leqslant d^i < 2^8)$ une séquence pseudo-aléatoire de 8 bit est générée comme suit :

$$d^i = E_{256}\Big(proj_{(x,y)}(S_j^i)\Big) \oplus E_{256}\left(proj_{(x,y)}(S_k^i)\right),$$

où

$$\left(S_j^i, S_k^i\right) = \left(f^{n_i+1}(S_j^{i-1}), f^{n_i+1}(S_k^{i-1})\right).$$

Les pixels de l'image seront masqués par les séquences aléatoires $\{d^i\}_{1\leq i\leq M\times N}$ selon le mode CFB. Après cette opération, les propriétés statistiques de l'image peuvent être considérablement modifiées, ce qui rend les attaques à texte clair choisi ou connu difficiles.

Algorithm 5 Algorithme du générateur de nombres pseudo aléatoires (GNPA)

Début
Entrée : Une clé $K = (k_0k_1k_2\ldots k_{L-1})_2$, deux états S_1^0, S_2^0 et un entier $M \times N$
Sortie : $M \times N$ séquences aléatoire $\{d^i\}_{i\in[\![1;M\times N]\!]}$ où d^i est un entier non signé de 8 bits et deux états $S_1^{M\times N}$, $S_2^{M\times N}$.
$n \leftarrow 2k_1 + k_0$
for $i = 1$ jusqu'à $M \times N$ **do**
    $S_1^i \leftarrow f^{n+1}(S_1^{i-1})$
    if $(S_1^i \in \Gamma_5 \times V^1)$ **then**
        $S_1^i \leftarrow f(S_1^i)$
    end if
    $S_2^i \leftarrow f^{n+1}(S_2^{i-1})$
    if $(S_2^i \in \Gamma_5 \times V^1)$ **then**
        $S_2^i \leftarrow f(S_2^i)$
    end if
    $d^i \leftarrow E_{256}\Big(proj_{(x,y)}(S_1^i)\Big) \oplus E_{256}\Big(proj_{(x,y)}(S_2^i)\Big)$
    $j \leftarrow (2 \times i) mod(L-1)$
    $j \leftarrow 2 \times k_{j+1} + k_j$
end for
Fin

3.2.3 Chiffrement des images en niveaux de gris

Dans cette partie, nous décrivons en détail un schéma de chiffrement d'image en niveaux de gris. L'algorithme a pour entrée : une image I de taille $M \times N$, une clé secrète K et un entier CD. La description des étapes suivantes :

Étape 1 : A partir de la clé K, nous calculons les états initiaux du système $S_{0,p}$, pour $p = 1, 2, 3$.

Étape 2 : Nous effectuons n_0 itérations pour les trois particules pour obtenir :

$$S_p^0 = f^{n_0}(S_{0,p}) \text{ pour } p = 1, 2, 3 \text{ où } n_0 = \sum_{i=0}^{7} k_i \times 2^i.$$

Étape 3 : Nous remodelons l'image I sous la forme d'une table TD^0 de taille $M \times N$ où $TD^0[i \times N + j] = I(i, j)$ avec $0 \leq i < M$ et $0 \leq j < N$.

Étape 4 : Nous effectuons des tours de confusion-diffusion CD comme indiqué dans la boucle suivante :

pour $l = 1$ jusqu'à CD, nous effectuons les opérations suivantes :

— Nous générons la permutation P^l :

$$\left(P^l, S^l_{(l-1)\%3+1}\right) \leftarrow permutation\left(K, S^{l-1}_{(l-1)\%3+1}, M \times N\right).$$

— Nous appliquons P^l à TD^{l-1} pour avoir TC^l. $for \quad i = 1$ jusqu'à $M \times N \quad do$:

$$TC^l[i-1] \leftarrow TD^{l-1}[P^l(i) - 1].$$

— Nous générons les séquences pseudo-aléatoires $\{d^{l,i}\}_{1 \leq i \leq M \times N}$

$$\left(\{d^{l,i}\}_{1 \leq i \leq M \times N}, S^l_{l\%3+1}, S^l_{(l+1)\%3+1}\right) \leftarrow GNPA\left(K, S^{l-1}_{l\%3+1}, S^{l-1}_{(l+1)\%3+1}, M \times N\right).$$

— Nous chiffrons les données TC^l selon le mode CFB :

— $TD^l[0] \leftarrow d^{l,1} \oplus TC^l[0]$

— pour $i = 2$ jusqu'à $M \times N$ faire :

$$TD^l[i-1] \leftarrow TD^l[i-2] \oplus d^{l,i} \oplus TC^l[i-1].$$

Étape 5 : Nous conversons TD^{CD} en une image chiffrée C de taille $M \times N$, où la valeur du pixel $C(i,j) = TD^{CD}[i \times N + j]$.

Algorithm 6 L'algorithme de chiffrement

1: Début
2: Entrée : une clé $K = (k_0 k_1 k_2 \dots k_{L-1})_2$, une image grise I de taille $M \times N$ et un entier CD
3: Sortie : une image chiffrée C de taille $M \times N$.
4: $(S_1, S_2, S_3) \leftarrow$ **initialise**(K)
5: $n \leftarrow \sum_{i=0}^{7} k_i \times 2^i$
6: $S_1^0 \leftarrow f^n(S_1)$
7: $S_2^0 \leftarrow f^n(S_2)$
8: $S_3^0 \leftarrow f^n(S_3)$
9: **for** $i = 0$ jusqu'à $M-1$ **do**
10: **for** $j = 0$ jusqu'à $N-1$ **do**
11: $T[i \times N + j] \leftarrow I(i,j)$
12: **end for**
13: **end for**
14: **for** $l = 1$ jusqu'à CD **do**
15: $\left(P^l, S^l_{l-1\%3+1}\right) \leftarrow$ **permutation**$\left(K, S^{l-1}_{l-1\%3+1}, M \times N\right)$
16: **for** $i = 1$ jusqu'à $M \times N$ **do**
17: $TC^l[i-1] \leftarrow TD^{l-1}[P(i)-1]$
18: **end for**
19: $\left(\{d^{l,i}_{l\%3+1,(l+1)\%3+1}\}_{1 \le i \le M \times N}, S^l_{l\%3+1}, S^l_{(l+1)\%3+1}\right) \leftarrow$ **GNPA**$\left(S^{l-1}_{l\%3+1}, S^{l-1}_{(l+1)\%3+1}, M \times N\right)$
20: $TD^l[0] \leftarrow TC^l[0] \oplus d^{l,1}_{l\%3+1,(l+1)\%3+1}$
21: **for** $i = 2$ to $M \times N$ **do**
22: $TD^l[i-1] \leftarrow TC^l[i-1] \oplus d^{l,i}_{l\%3+1,(l+1)\%3+1} \oplus TD^l[i-2]$
23: **end for**
24: **end for**
25: **for** $i = 0$ to $M-1$ **do**
26: **for** $j = 0$ to $N-1$ **do**
27: $C(i,j) \leftarrow TD^{CD}[i \times N + j]$
28: **end for**
29: **end for**
30: Fin

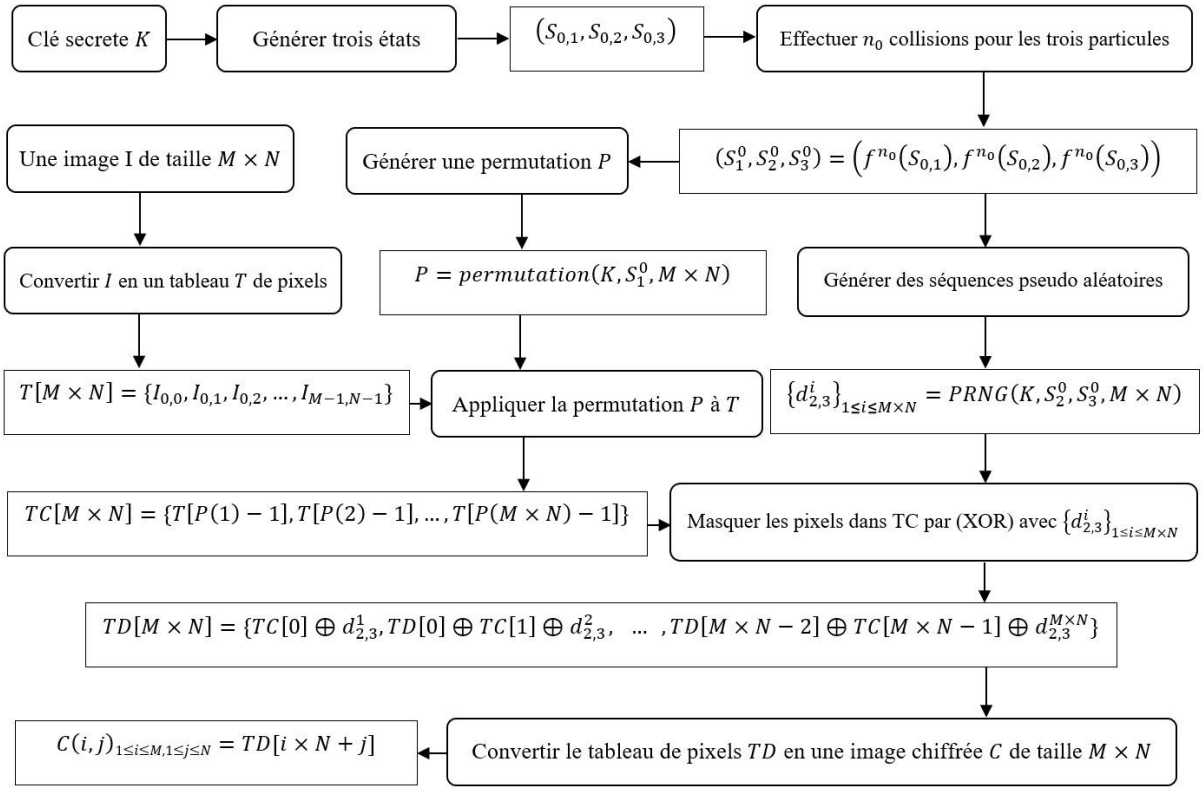

FIGURE 3.2 – Une ronde du schéma de chiffrement proposé

3.2.4 Déchiffrement de l'image en niveaux de gris

Le déchiffrement est le processus inverse du chiffrement. L'algorithme a en entrée une image chiffrée C, une clé secrète K et un entier CD. La sortie de l'algorithme est l'image d'origine I. La description détaillée est la suivante :

Étape 1 : De la clé K, nous générons les états initiaux du système $S_{0,p}$, pour $p = 1, 2, 3$.

Étape 2 : Nous effectuons n_0 itérations pour les trois particules et obtenons :

$$S_p^0 = f^{n_0}(S_{0,p}) \text{ pour } p = 1, 2, 3 \text{ où } n_0 = \sum_{i=0}^{7} k_i \times 2^i.$$

Étape 3 : Nous remodelons l'image chiffrée C sous la forme d'une table TD^{CD} de taille $M \times N$ où $TD^{CD}[i \times N + j] = C(i, j)$ avec $0 \leq i < M$ et $0 \leq j < N$.

Étape 4 : nous générons CD permutations $\{P^l\}_{1 \leq l \leq CD}$ et des séqunces peudo-aléatoires $\{d^{l,i}\}_{1 \leq i \leq M \times N, 1 \leq l \leq CD}$ comme indiqué dans la boucle suivante :

pour $l = 1$ *to* CD, nous effectuons les opérations suivantes :

- $\left(P^l, S^l_{(l-1)\%3+1}\right) \leftarrow permutation\left(K, S^{l-1}_{(l-1)\%3+1}, M \times N\right)$
- Nous déterminons $\{R^l\}_{1 \leq l \leq CD}$ où R^l l'inverse du permutation P^l.

- pour $i = 1$ jusqu'à $M \times N$ faire :

$$\left(\{d^{l,i}\}_{1 \leq l \leq M \times N}, S^l_{l\%3+1}, S^l_{(l+1)\%3+1}\right) \leftarrow GNPA\left(K, S^{l-1}_{l\%3+1}, S^{l-1}_{(l+1)\%3+1}\right)$$

Étape 5 :pour $l = CD$ jusqu'à 1 (décrément l par 1), nous effectuons les opérations suivantes :

- pour $i = M \times N$ jusqu'à 2 (décrément i par 1), nous effectuons :

$$TC^l[i-1] \leftarrow TD^l[i-1] \oplus d^{l,i} \oplus TD^l[i-2].$$

- $TC^l[0] \leftarrow TD^l[0] \oplus d^{l,1}$.

— Nous appliquons la permutation R^l sur TC^l pour trouver le tableau TD^{l-1}.

Étape 6 : Nous convertissons le tableau TD^0 à l'image d'origine I de taille $M \times N$, où la valeur du pixel $I(i,j) = TD^0[i \times N + j]$, avec $0 \leq i < M$ et $0 \leq j < N$.

Algorithm 7 Algorithme de déchiffrement

Début
Entrée : une clé $K = (k_0 k_1 k_2 \dots k_{L-1})_2$, une image chiffré en niveau de gris C de taille $M \times N$ et un entier CD
Sortie : Image original I de taille $M \times N$.
$(S_1, S_2, S_3) \leftarrow$ **initialise**(K)
$n \leftarrow \sum_{i=0}^{7} k_i \times 2^i$
$S_1^0 \leftarrow f^n(S_1)$
$S_2^0 \leftarrow f^n(S_2)$
$S_3^0 \leftarrow f^n(S_3)$
for $i = 0$ jusqu'à $M - 1$ **do**
  for $j = 0$ jusqu'à $N - 1$ **do**
    $T[i \times N + j] \leftarrow C(i,j)$
  end for
end for
for $l = 1$ jusqu'à CD **do**
  $\left(P^l, S^l_{l\%3+1}\right) \leftarrow$ **permutation**$\left(K, S^{l-1}_{l-1\%3+1}, M \times N\right)$
  for $i = 1$ jusqu'à $M \times N$ **do**
    $R^l\left(P^l(i)\right) = i$
  end for
  $\left(\{d^{l,i}_{l\%3+1,(l+1)\%3+1}\}_{1 \leq i \leq M \times N}, S^l_{l\%3+1}, S^l_{(l+1)\%3+1}\right) \leftarrow$ **GNPA**$\left(S^{l-1}_{l\%3+1}, S^{l-1}_{(l+1)\%3+1}, M \times N\right)$
end for
for $l = CD$ jusqu'à 1 **do**
  for $i = M \times N$ jusqu'à 2 **do**
    $TC^l[i-1] \leftarrow TD^l[i-1] \oplus d^{l,i}_{(CD-1)\%3+1,CD\%3+1} \oplus TD^l[i-2]$
  end for
  $TC^l[0] \leftarrow TD^l[0] \oplus d^{l,1}_{(CD-1)\%3+1,CD\%3+1}$
  for $i = 1$ jusqu'à $M \times N$ **do**
    $TD^{l-1}[i-1] \leftarrow TC^l[R^l(i) - 1]$
  end for
end for
for $i = 0$ jusqu'à $M - 1$ **do**
  for $j = 0$ jusqu'à $N - 1$ **do**
    $I(i,j) \leftarrow TD^0[i \times N + j]$
  end for
end for
Fin

3.3 L'analyse de sécurité

Dans cette section, la sécurité du système proposé est analysée. L'algorithme est implémenté dans $MATLAB$ et les dimensions du billard sont définies sur : $a = 2000$ et $r = 500$. L'image simulée de "chats", de taille 200×200 et d'un niveau de gris de $Gl = 256$, est montrée à la figure 3.3 (a). Le résultat de la simulation après un ronde de confusion-diffusion ($CD = 1$) est à la figure 3.3 (b). Les différents aspects de la sécurité sont décrits en détail dans les sou-sections suivantes.

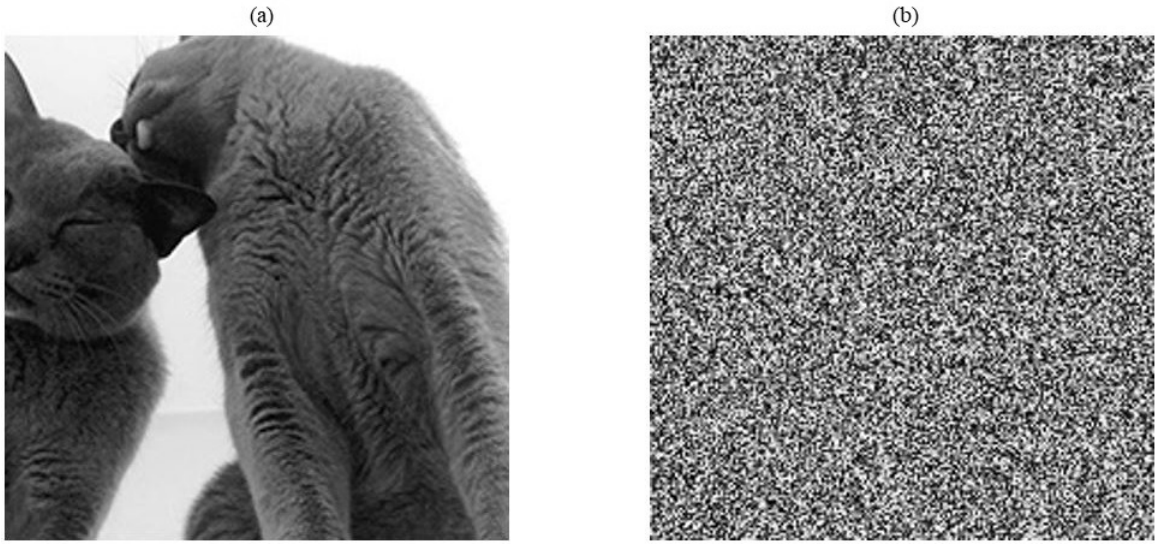

FIGURE 3.3 – (a) l'image originale et (b) l'image chiffrée

3.3.1 L'espace des clés

Notre cryptosystème se caractérise par une propriété bien meilleure que les autres systèmes. Il a pour clé une chaîne binaire de longueur arbitraire. Les paramètres d'initialisation sont dérivés de manière à couvrir toutes les parties de la clé. De plus, les processus sont exécutés en ajoutant un nombre d'itérations supplémentaires en fonction des bits de la représentation ASCII du clé. En conséquence, les attaques exhaustives sont impossible contre le cryptosystème proposé.

3.3.2 Analyse d'histogramme

L'analyse par histogramme est un paramètre de sécurité nécessaire pour évaluer les caractéristiques statistiques de l'image chiffrée. Elle affiche la fréquence des valeurs de pixels à chaque niveau de gris. Cependant, l'histogramme d'une image chiffrée doit être uniformément distribué et différent de celui de l'image d'origine, comme indiqué aux Fig. 3.4 (b) pour éviter les attaques statistiques.

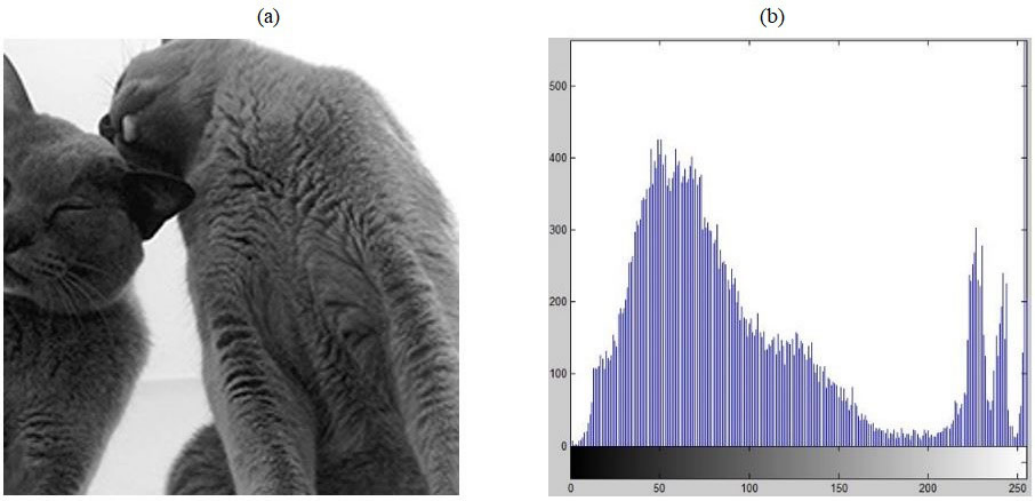

FIGURE 3.4 – (a) l'image originale et (b) l'histogramme de (a)

L'histogramme correspondant de l'image chiffrée est présenté en Figs. 3.5(b).

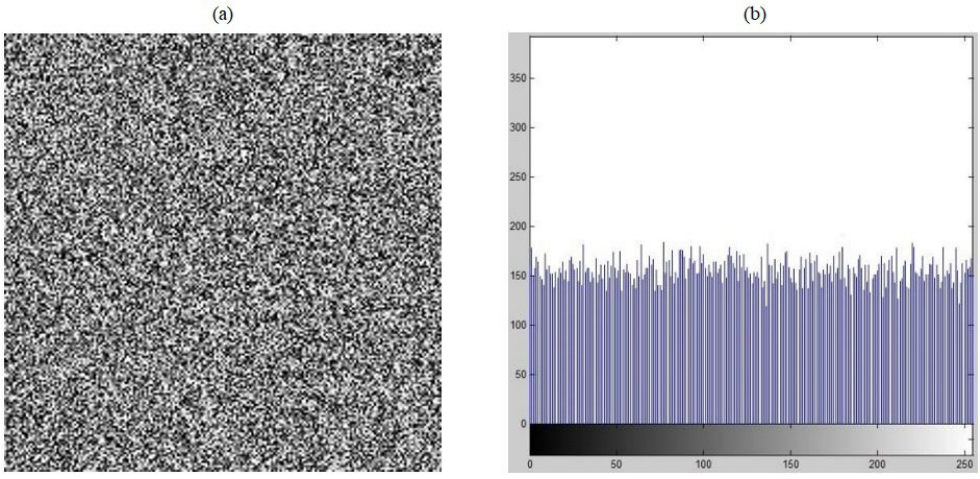

FIGURE 3.5 – (a) l'image chiffrée et (b) l'histogramme de (a)

La forme de l'histogramme de l'image chiffrée est clairement différente de celle d'origine. De plus, les fréquences des pixels sont uniformément réparties. Par conséquent, l'image chiffrée ne révèle aucune information statistique sur l'image d'origine et les attaques statistiques sont donc irréalisables contre ce système.

3.3.3 Analyse d'entropie

L'entropie de l'information a été introduite par Shannon dans 1949 [93]. Il est défini pour exprimer le degré d'incertitude ou d'aléatoire dans un système de chiffrement. L'entropie des informations sur l'image est un critère de mesure de l'effet d'un algorithme de chiffrement d'image. Cela indique si la distribution des valeurs de pixels est aléatoire ou non. L'entropie H d'une image en niveaux de gris est calculée à l'aide de la formule suivante :

$$H(s) = \sum_{i=0}^{Gl-1} P(s_i) log_2 \left(\frac{1}{P(s_i)} \right)$$

avec Gl est le niveau de gris de l'image et $P(s_i)$ la probabilité d'occurrence du niveau s_i dans l'image. Pour une image de taille $M \times N$, probabilité d'apparition d'un pixel de niveau de gris d'une valeur de s_i égale à :

$$P(s_i) = \frac{\text{nombre de pixels au niveau } s_i}{M \times N}$$

Pour une image chiffrée $Gl = 256$ de niveau de gris, la valeur d'entropie idéale des informations devrait être 8 ($\log_2 256 = 8$). En d'autres termes, plus la valeur d'entropie de l'image chiffrée est proche de 8, et la distribution des niveaux de gris de l'image chiffrée est homogène et il n'y a aucune divulgation d'informations sur l'image claire.

Dans notre simulation, nous avons calculé l'entropie des informations de l'image chiffrée. La table 3.1 résume les valeurs d'entropie d'information pour les images chiffrées sur plusieurs rondes CD du processus de confusion-diffusion.

CD	1	2	3	4	5	6	7
Entropie	7.99987	7.99997	7.99993	7.99986	7.99998	7.99998	7.99999

TABLE 3.1 – Les résultats d'entropie de l'image chiffrée sur plusieurs rondes CD

Toutes les valeurs sont très proches de 8, nous pouvons donc en conclure que cet algorithme a un très bon effet de chiffrement.

3.3.4 Analyse du coefficient de corrélation

Le coefficient de corrélation joue un rôle important dans l'étude statistique des images. Nous allons traiter la corrélation pixels adjacents dans trois directions. Pour continuer, nous allons choisir un échantillon aléatoire de NPI paires de pixels de l'image traitée, après avoir calculé le coefficient de corrélation, dans la direction horizontale (r,s) / $(r+1,s)$, vertical (r,s) / $(r,s+1)$ ou diagonale $(r,s)/((r+1,s+1)$ de l'image. Nous calculons les coefficients de corrélation en utilisant la formule suivante :

$$r(u,v) = \frac{cov(u,v)}{\sqrt{Var(u)} \times \sqrt{Var(v)}}$$

où u et v sont les valeurs respectives des deux pixels adjacents, avec :

$$cov(u,v) = E\Big(\Big(u - E(u)\Big) \times \Big(v - E(v)\Big)\Big),$$

où

$$E(u) = \frac{1}{NPI} \sum_{i=1}^{NPI} u_i,$$

et

$$V(u) = \frac{1}{NPI} \sum_{i=1}^{NPI} \Big(u_i - E(u)\Big).$$

Les pixels adjacents dans une image naturelle peuvent avoir un coefficient de corrélation proche de 1 (c'est-à-dire que les pixels adjacents ont des valeurs simi-

laires). Alors que les pixels d'une image chiffrée doivent être proches de 0 (c'est-à-dire que les pixels doivent être distribués de manière aléatoire). Par conséquent, un algorithme de chiffrement d'image doit effectivement réduire la corrélation entre les pixels adjacents. Dans notre étude, nous avons choisi $NPI = 2000$, un nombre assez représentatif pour une image de taille 200×200. Nous avons comparé les corrélations des pixels de l'image d'origine Figs. 3.3 (a) et celles de l'image chiffrée dans les Figs. 3.3 (b) sur l'échantillon choisi.

Il est clair que les points sont situés le long de la diagonale (Figs. 3.6), ce qui in-

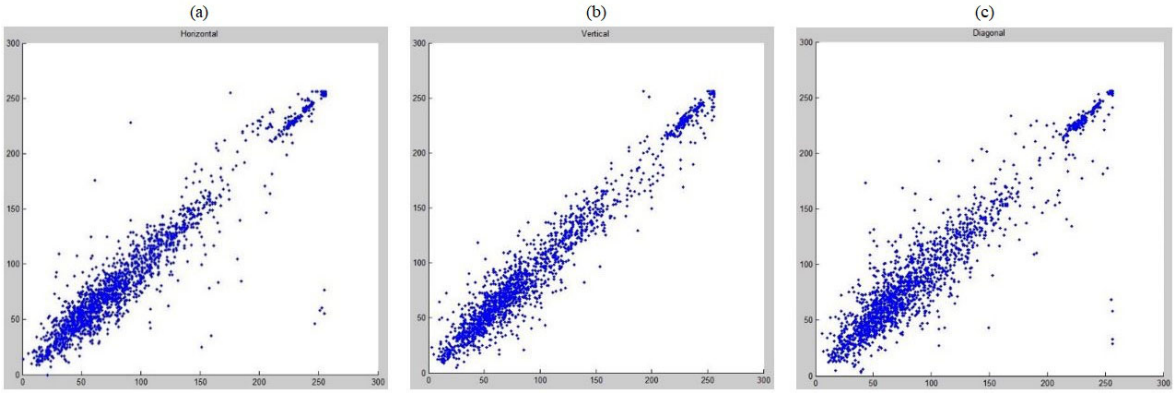

FIGURE 3.6 – Distribution de corrélation des pixels adjacents de l'image d'original. (a) horizontal, (b) vertical, (c) diagonale

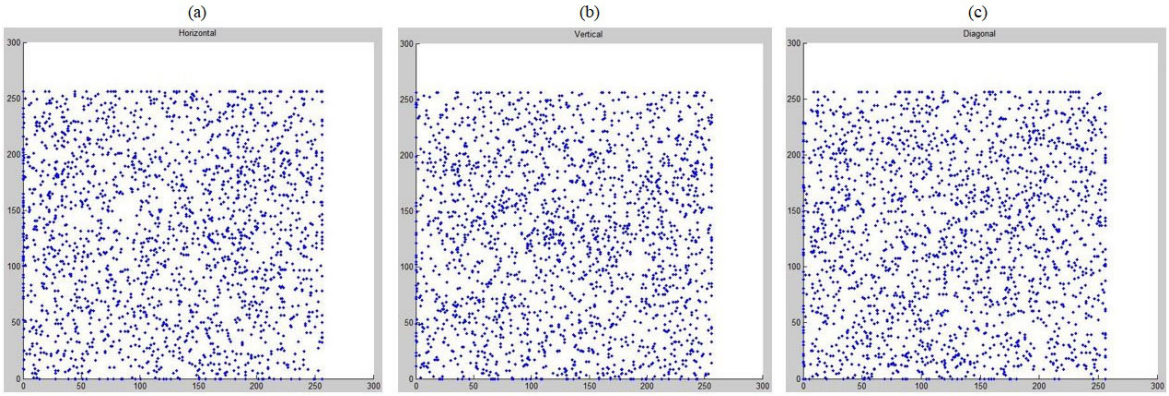

FIGURE 3.7 – Distribution de corrélation de pixels adjacents de l'image chiffrée. (a) horizontal, (b) vertical, (c) diagonale

dique une forte corrélation de pixels adjacents dans les trois directions de l'image d'origine, tandis que ceux de l'image chiffrée sont dispersés sur tout le plan (Figs

3.7), ce qui indique que la corrélation est grandement réduite. L'analyse de corrélation prouve que notre algorithme satisfait une corrélation nulle.

3.3.5 L'attaque différentielle

Une attaque différentielle consiste à modifier l'image d'origine, généralement en pixels, avant de la chiffrer pour obtenir une image chiffrée modifiée. Pendant ce temps, l'image d'origine est également chiffrée avec la même clé. enfin, les deux images chiffrées sont comparées pour déterminer s'il existe une relation entre l'image simple et l'image chiffrée, ce qui facilite encore la détermination de la clé secrète. Si une modification mineure de l'image d'origine entraîne des modifications importantes de l'image chiffrée, l'attaque différentielle peut devenir inefficace. En d'autres termes, si un schéma de chiffrement est invulnérable aux attaques différentielles, et seulement si il est sensible au changement de pixel de l'image originale.

Les deux critères les plus courants pour mesurer la capacité d'attaque différentielle sont : le nombre de taux de changement de pixels ($NPCR$) et l'intensité de changement moyenne unifiée ($UACI$). Le NPCR et le UACI entre deux images chiffrées C_0 et C_1 de la même taille $M \times N$ sont définis par :

$$NPCR(C_0, C_1) = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} D_{(C_0,C_1)}(i,j)}{M \times N} \times 100\%,$$

et

$$UACI(C_0, C_1) = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \frac{|C_0(j,k) - C_1(j,k)|}{L-1}}{M \times N} \times 100\%.$$

où $C_0(i,j)$ est le pixel correspondant à la rangée i -th et à la colonne j -th de l'image C_0 et $D_{(C_0,C_1)}(i,j)$ est une fonction définie comme suit :

$$D_{(C_0,C_1)}(i,j) = \begin{cases} 0 \text{ if } C_0(i,j) = C_1(i,j) \\ 1 \text{ if } C_0(i,j) \neq C_1(i,j) \end{cases}$$

et Gl est le niveau de gris de l'image.

Récemment, les valeurs $UACI$ et $NPCR$ de deux images aléatoires, estimation attendue pour un bon système cryptographique, ont été prouvées dans [39] et données par :

$$NPCR_{expected} = \left(1 - \frac{1}{2^{\log_2 L}}\right) \times 100\%,$$

et

$$UACI_{expected} = \frac{1}{L^2}\left(1 - \frac{\sum_{i=1}^{L-1} i(i+1)}{L-1}\right) \times 100\%.$$

Donc, pour un niveau de gris $Gl = 256$, nous pouvons avoir $NPCR_{attendu} = 99.609\%$ et $UACI_{attendu} = 33.464\%$.

Dans notre analyse, nous modifions le premier pixel à la valeur opposée dans l'image "cats" I_0, pour obtenir I_1, puis nous chiffrons les deux images avec la même clé en utilisant un cycle de chiffrement ($CD = 1$) pour avoir C_0 et C_1.

Nous obtenons $UACI(C_0, C_1) = 33.078\%$ et $NPCR(C_0, C_1) = 99.356\%$ sont tous deux proches des valeurs attendues, ce qui indique que l'algorithme proposé est robuste contre les attaques différentielles.

3.3.6 Analyse de sensibilité à la clé

La sensibilité à une petite modification de la clé est une propriété essentielle des systèmes de chiffrement d'image. En d'autres termes, une infime modification de la clé secrète doit produire des images totalement différentes et le processus de déchiffrement ne doit pas aboutir. Cette propriété rend les systèmes de

chiffrement plus robustes et sécurisés contre les attaques statistiques et différentielles. En fait, pour l'analyse de la sensibilité clé du système, nous avons effectué les étapes suivantes :

— L'image originale I_0 de la Fig. 3.3 (a) est chiffrée à l'aide de la clé K_0 = "GUENNOUNCHARIF" de la représentation binaire en code ASCII (01000111 01010101 01000101 01001110 01001110 01001111 01010101 01001110 01000011 01001000 01000001 01010010 01001001 01000110)$_2$, pour obtenir une image chiffrée C_0.

— Nous générons un groupe de clés $\{K_i\}_{1\leq i\leq 112)}$ de K_0. Pour trouver K_i, nous modifions le i-iéme bit (nous remplaçons k_i par $\overline{k}_i = 1 - k_i$) parmi les bits 112 de la représentation binaire en code ASCII de K_0.

— Nous déchiffrent l'image C_0 à l'aide des clés $\{K_i\}_{1\leq i\leq 112)}$ pour obtenir les images. $\{D_i\}_{1\leq i\leq 112}$.

— Nous effectuons une comparaison pixel par pixel entre les images $\{D_i\}_{1\leq i\leq 112)}$ et l'image I_0 en utilisant le $NPCR$ et $UACI$.

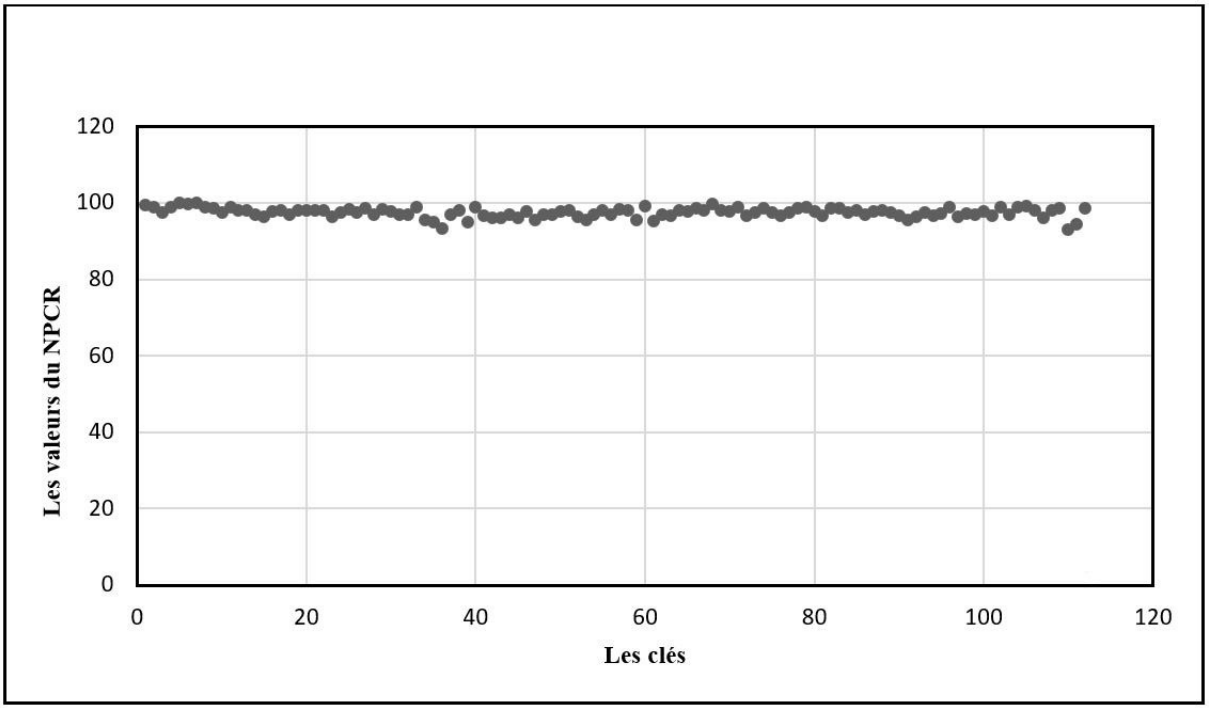

FIGURE 3.8 – Les résultats du $NPCR$ entre l'images I_0 et les images $\{D_i\}_{1\leq i\leq 112)}$

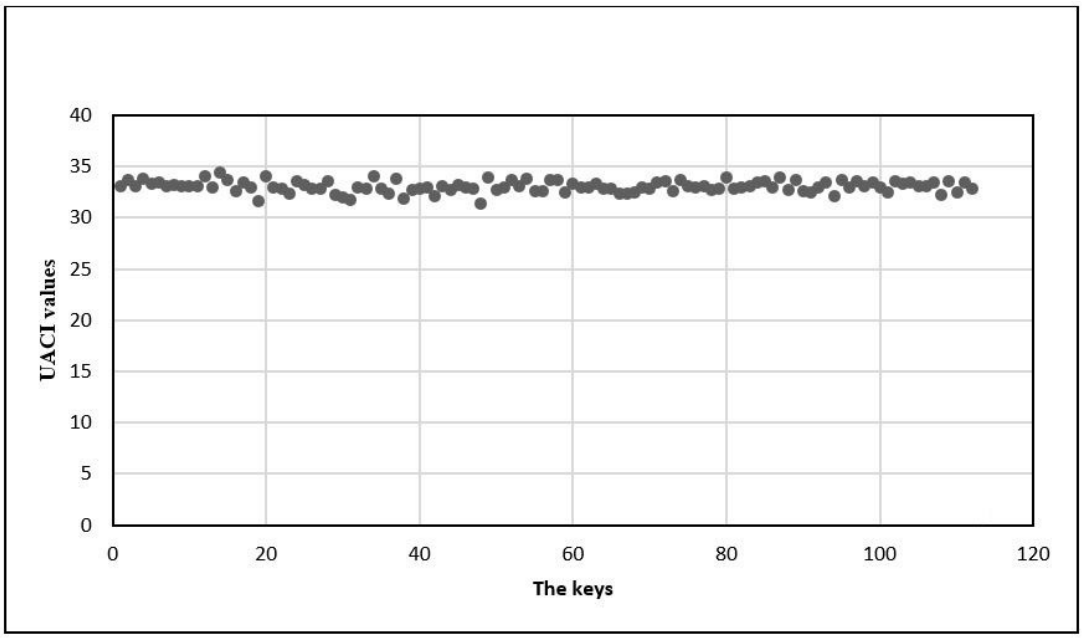

FIGURE 3.9 – Les résultats du $UACI$ entre l'image I_0 et et les images $\{D_i\}_{1\leq i\leq 112)}$

Comme indiqué dans les figures 3.8 et 3.9, nous remarquons que $NPCR$ et $UACI$ varient respectivement au voisinage de $NPCR_{attendu}$ et de $UACI_{attendu}$. Ainsi, nous pouvons voir que l'algorithme proposé est très sensible à une petite modification de la clé.

3.4 Conclusion

Nous avons profité des bonnes caractéristiques offertes par le billard du Sinaï, telles que le chaos et l'imprévisibilité, pour concevoir un nouvel algorithme de chiffrement d'image. Le système de chiffrement développé est basé sur la structure Fridrich. Une nouvelle permutation a été conçue pour la confusion. La diffusion réalisée par un générateur de séquences pseudo-aléatoires qui possède de bonnes propriétés statistiques et chaotiques. Pour l'analyse de la sécurité, plusieurs tests ont été effectués, y compris l'espace clé, les analyses statistiques, différentielles et la sensibilité à un petit changement dans la clé. Les résultats ont montré que le cryptosystème proposé a un niveau de sécurité bien élevé. Une

fois de plus, le billard chaotique a montré des performances intéressantes qui peuvent renforcer la sécurité des systèmes de communication.

Conclusion et perspective

Dans cette thèse, nous avons conçu et mis en œuvre de nouvelles schémas de chiffrement utilisant les systèmes d'un billard chaotique pour enrichir les schémas de communication pour protéger les contenus des données. À cette fin, les propriétés chaotiques intéressantes du billard de Sinaï sont bien exploitées et utilisées pour concevoir des cryptosysèmes.

Nous avons conçu et implémenté de manière efficace et sécurisée un générateur de nombres pseudo chaotiques. Ce générateur est basé sur l'implémentation de deux systèmes du billard de Sinaï, en utilisant de nouvelles techniques. Après l'analyse de sécurité, les résultats obtenus sont prometteurs. En effet, notre générateur est sensible à un bit de changement dans la clé et a passé tous les tests statistiques avec succès. Ensuite, nous avons proposé un nouveau système de chiffrement pour protéger la confidentialité des images. Le schéma proposé repose sur l'implémentation de trois systèmes de billard de Sinaï. La confusion des pixels de l'image est effectué par une permutation générée par l'une des systèmes, tandis que le confusion est basé sur le générateur basé sur le chaos réalisé au chapitre 2. Les résultats obtenus après l'analyse du cryptage d'image montrent que le schéma proposé a de bonnes propriétés cryptographiques.

Les résultats obtenus montrent que les cryptosystèmes proposés présentent de bonnes propriétés cryptographiques comparables à d'autres travaux. En effet, le système de billard de Sinaï est utilisé pour concevoir un nouveaux schéma

de chiffrement plus sécurisés, plus rapides et plus efficaces. Ainsi, cela peut être une telle motivation pour explorer plus d'options et impliquer les systèmes des billards chaotiques dans la conception de nouveaux protocoles cryptographiques. Sur la base de cette thèse, plusieurs explorations et perspectives pourront être réalisées dans d'autre travail futur :

1. Une proposition d'un générateur de bit pseudo-aléatoire basé sur le même système chaotique en introduisant de nouvelle technique cryptographique où la clé secrète joue un rôle plus important dans la génération des sous-suites.
2. La conception d'un cryptosystème à base sur le même système pour le chiffrement des vidéos.
3. La proposition d'une fonction de hachage à base de plusieurs système du billard de Sinaï où la longueur du sortie de fonction est contrôlée par l'utilisateur.
4. L'application d'une autre type de billards chaotiques à 3 dimensions à titre d'exemple avec des propriétés chaotiques intéressantes, pour la conception nouveaux protocoles cryptographiques.

Bibliographie

[1] AHMAD, M., ALAM, B., AND FAROOQ, O. Chaos based mixed keystream generation for voice data encryption. *arXiv preprint arXiv :1403.4782* (2014).

[2] ALVAREZ, G., AMIGÓ, J. M., ARROYO, D., AND LI, S. Lessons learnt from the cryptanalysis of chaos-based ciphers. In *Chaos-Based Cryptography*. Springer, 2011, pp. 257–295.

[3] ALVAREZ, G., AND LI, S. Some basic cryptographic requirements for chaos-based cryptosystems. *International Journal of Bifurcation and Chaos 16*, 08 (2006), 2129–2151.

[4] ALVAREZ, G., MONTOYA, F., ROMERA, M., AND PASTOR, G. Cryptanalysis of an ergodic chaotic cipher. *Physics letters a 311*, 2-3 (2003), 172–179.

[5] ALVAREZ, G., MONTOYA, P., PASTOR, G., AND ROMERA, M. Chaotic cryptosystems. In *Proceedings IEEE 33rd Annual 1999 International Carnahan Conference on Security Technology (Cat. No. 99CH36303)* (1999), IEEE, pp. 332–338.

[6] ANDRECUT, M. Logistic map as a random number generator. *International Journal of Modern Physics B 12*, 09 (1998), 921–930.

[7] ARROYO, D., RHOUMA, R., ALVAREZ, G., LI, S., AND FERNANDEZ, V. On the security of a new image encryption scheme based on chaotic map lattices. *Chaos : An Interdisciplinary Journal of Nonlinear Science 18*, 3 (2008), 033112.

[8] BECK, C., AND SCHÖGL, F. *Thermodynamics of chaotic systems : an introduction*. No. 4. Cambridge University Press, 1995.

[9] BERNSTEIN, G. M., AND LIEBERMAN, M. A. Method and apparatus for generating secure random numbers using chaos, Apr. 9 1991. US Patent 5,007,087.

[10] BERRY, M. V. Quantizing a classically ergodic system : Sinai's billiard and the kkr method. *Annals of Physics 131*, 1 (1981), 163–216.

[11] BIHAM, E., AND SHAMIR, A. *Differential cryptanalysis of the data encryption standard*. Springer Science & Business Media, 2012.

[12] BOCCALETTI, S., KURTHS, J., OSIPOV, G., VALLADARES, D., AND ZHOU, C. The synchronization of chaotic systems. *Physics reports 366*, 1-2 (2002), 1–101.

[13] BRENT, R. P., ET AL. Uniform random number generators for supercomputers.

[14] BROER, H., AND TAKENS, F. *Dynamical systems and chaos*, vol. 172. Springer Science & Business Media, 2010.

[15] BROWN, T. A. Measuring chaos using the lyapunov exponent. *Chaos Theory in the Social Science* (1996), 53–66.

[16] BUCHMANN, J. A., KARATSIOLIS, E., AND WIESMAIER, A. *Introduction to public key infrastructures*. Springer Science & Business Media, 2013.

[17] BUNIMOVICH, L. A. On billiards close to dispersing. *Matematicheskii Sbornik 136*, 1 (1974), 49–73.

[18] BUNIMOVICH, L. A., SINAI, Y. G., AND CHERNOV, N. I. Statistical properties of two-dimensional hyperbolic billiards. *Russian Mathematical Surveys 46*, 4 (1991), 47–106.

[19] CHARIF, K., DRISSI, A., AND GUENNOUN, Z. E. A. A pseudo random number generator based on chaotic billiards. *International Journal of Network Security 19*, 3 (2017), 479–486.

[20] CHARIF, K., AND GUENNOUN, Z. E. A. A novel image encryption algorithm based on chaotic billiards. *Journal of Discrete Mathematical Sciences and Cryptography* (2019), 1–26.

[21] CHEN, J.-X., ZHU, Z.-L., FU, C., YU, H., AND ZHANG, Y. Reusing the permutation matrix dynamically for efficient image cryptographic algorithm. *Signal Processing 111* (2015), 294–307.

[22] CHEN, S., AND LÜ, J. Parameters identification and synchronization of chaotic systems based upon adaptive control. *Physics Letters A 299*, 4 (2002), 353–358.

[23] CHEN, S., ZHONG, X., AND WU, Z. Chaos block cipher for wireless sensor network. *Science in China Series F : Information Sciences 51*, 8 (2008), 1055.

[24] CHEN, T.-H., TSAO, K.-H., AND LEE, Y.-S. Yet another multiple-image encryption by rotating random grids. *Signal Processing 92*, 9 (2012), 2229–2237.

[25] CHERNOV, N. Entropy, lyapunov exponents, and mean free path for billiards. *Journal of statistical physics 88*, 1-2 (1997), 1–29.

[26] CHERNOV, N. Decay of correlations and dispersing billiards. *Journal of Statistical Physics 94*, 3-4 (1999), 513–556.

[27] CHERNOV, N., AND HASKELL, C. Nonuniformly hyperbolic k-systems are bernoulli. *Ergodic theory and dynamical systems 16*, 01 (1996), 19–44.

[28] CHERNOV, N., AND MARKARIAN, R. *Chaotic billiards*. No. 127. American Mathematical Soc., 2006.

[29] CHERNOV, N., AND YOUNG, L.-S. Decay of correlations for lorentz gases and hard balls. In *Hard ball systems and the Lorentz gas*. Springer, 2000, pp. 89–120.

[30] CRISTEA, B., CHARGÉ, P., FOURNIER-PRUNARET, D., PEYRARD, F., AND MERCIER, J.-J. Behavior of chaotic sequences under a finite representation

and its cryptographic applications. In *IEEE Workshop on nonlinear maps and applications (NOMA'07)* (2007).

[31] DAHLQVIST, P. The lyapunov exponent in the sinai billiard in the small scatterer limit. *Nonlinearity 10*, 1 (1997), 159.

[32] DEEPTHI, P., NITHIN, V., AND SATHIDEVI, P. Implementation and analysis of stream ciphers based on the elliptic curves. *Computers & Electrical Engineering 35*, 2 (2009), 300–314.

[33] DITTO, W., AND MUNAKATA, T. Principles and applications of chaotic systems. *Communications of the ACM 38*, 11 (1995), 96–102.

[34] ECKMANN, J. J.-p. eckmann and d. ruelle, rev. mod. phys. 57, 617 (1985). *Rev. Mod. Phys. 57* (1985), 617.

[35] ELGAMAL, T. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory 31*, 4 (1985), 469–472.

[36] FELDMAN, P. A practical scheme for non-interactive verifiable secret sharing. In *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)* (1987), IEEE, pp. 427–438.

[37] FRIDRICH, J. Symmetric ciphers based on two-dimensional chaotic maps. *International Journal of Bifurcation and chaos 8*, 06 (1998), 1259–1284.

[38] FRIEDMAN, B., OONO, Y., AND KUBO, I. Universal behavior of sinai billiard systems in the small-scatterer limit. *Physical review letters 52*, 9 (1984), 709.

[39] FU, C., CHEN, J.-J., ZOU, H., MENG, W.-H., ZHAN, Y.-F., AND YU, Y.-W. A chaos-based digital image encryption scheme with an improved diffusion strategy. *Optics Express 20*, 3 (2012), 2363–2378.

[40] GALLAVOTTI, G., AND ORNSTEIN, D. S. Billiards and bernoulli schemes. *Communications in Mathematical Physics 38*, 2 (1974), 83–101.

[41] GLENDINNING, P. *Stability, instability and chaos : an introduction to the theory of nonlinear differential equations*, vol. 11. Cambridge university press, 1994.

[42] GUYEUX, C., WANG, Q., AND BAHI, J. M. A pseudo random numbers generator based on chaotic iterations : application to watermarking. In *Web Information Systems and Mining*. Springer, 2010, pp. 202–211.

[43] HABUTSU, T., NISHIO, Y., SASASE, I., AND MORI, S. A secret key cryptosystem by iterating a chaotic map. In *Workshop on the Theory and Application of of Cryptographic Techniques* (1991), Springer, pp. 127–140.

[44] HASLER, M. Synchronization of chaotic systems and transmission of information. *International Journal of Bifurcation and Chaos 8*, 04 (1998), 647–659.

[45] HERON, S. Advanced encryption standard (aes). *Network Security 2009*, 12 (2009), 8–12.

[46] HILBORN, R. C., ET AL. *Chaos and nonlinear dynamics : an introduction for scientists and engineers*. Oxford University Press on Demand, 2000.

[47] HONG, Z., AND XIETING, L. Generating chaotic secure sequences with desired statistical properties and high security. *International Journal of Bifurcation and Chaos 7*, 01 (1997), 205–213.

[48] HUA, Z., ZHOU, Y., AND CHEN, C. P. A new series-wound framework for generating 1d chaotic maps. In *2013 IEEE Digital Signal Processing and Signal Processing Education Meeting (DSP/SPE)* (2013), IEEE, pp. 118–123.

[49] HUANG, X. Image encryption algorithm using chaotic chebyshev generator. *Nonlinear Dynamics 67*, 4 (2012), 2411–2417.

[50] JAKIMOSKI, G., AND KOCAREV, L. Analysis of some recently proposed chaos-based encryption algorithms. *Physics Letters A 291*, 6 (2001), 381–384.

[51] JAMES, F. A review of pseudorandom number generators. *Computer physics communications 60*, 3 (1990), 329–344.

[52] JOLFAEI, A., AND MIRGHADRI, A. Image encryption using chaos and block cipher. *Computer and Information Science 4*, 1 (2010), 172.

[53] JOSHI, M., SHAKHER, C., AND SINGH, K. Image encryption and decryption using fractional fourier transform and radial hilbert transform. *Optics and Lasers in Engineering 46*, 7 (2008), 522–526.

[54] JUNOD, P. Cryptographic secure pseudo-random bits generation : The blum-blum-shub generator, 1999.

[55] KERCKHOFFS, A. La cryptographic militaire. *Journal des sciences militaires* (1883), 5–38.

[56] KOCAREV, L. Chaos-based cryptography : a brief overview. *IEEE Circuits and Systems Magazine 1*, 3 (2001), 6–21.

[57] KOCAREV, L., JAKIMOSKI, G., STOJANOVSKI, T., AND PARLITZ, U. From chaotic maps to encryption schemes. In *Circuits and Systems, 1998. ISCAS'98. Proceedings of the 1998 IEEE International Symposium on* (1998), vol. 4, IEEE, pp. 514–517.

[58] KOCAREV, L., AND LIAN, S. *Chaos-based cryptography : Theory, algorithms and applications*, vol. 354. Springer Science & Business Media, 2011.

[59] KOLMOGOROV, A. New metric invariant of transitive dynamical systems and endomorphisms of lebesgue spaces. *Doklady of Russian Academy of Sciences 119*, 5 (1958), 861–864.

[60] KYRTSOU, C., AND VORLOW, C. E. Complex dynamics in macroeconomics : A novel approach. In *New Trends in Macroeconomics*. Springer, 2005, pp. 223–238.

[61] LASOTA, A., AND MACKEY, M. C. *Chaos, Fractals, and Noise : Stochastic Aspects of Dynamics*, vol. 97. Springer Science & Business Media, 1998.

[62] LASOTA, A., AND MACKEY, M. C. *Chaos, fractals, and noise : stochastic aspects of dynamics*, vol. 97. Springer Science & Business Media, 2013.

[63] L'ECUYER, P. Uniform random number generation. *Annals of Operations Research 53*, 1 (1994), 77–120.

[64] L'ECUYER, P. Tables of maximally equidistributed combined lfsr generators. *Mathematics of Computation of the American Mathematical Society 68*, 225 (1999), 261–269.

[65] L'ECUYER, P., AND BLOUIN, F. Linear congruential generators of order k> 1. In *1988 Winter Simulation Conference Proceedings* (1998), IEEE, pp. 432–439.

[66] LEE, P.-H., PEI, S.-C., AND CHEN, Y.-Y. Generating chaotic stream ciphers using chaotic systems. *Chinese Journal of physics 41*, 6 (2003), 559–581.

[67] LEE RODGERS, J., AND NICEWANDER, W. A. Thirteen ways to look at the correlation coefficient. *The American Statistician 42*, 1 (1988), 59–66.

[68] LI, C., ARROYO, D., AND LO, K.-T. Breaking a chaotic cryptographic scheme based on composition maps. *International Journal of Bifurcation and Chaos 20*, 08 (2010), 2561–2568.

[69] LI, C., LIU, Y., ZHANG, L. Y., AND CHEN, M. Z. Breaking a chaotic image encryption algorithm based on modulo addition and xor operation. *International Journal of Bifurcation and Chaos 23*, 04 (2013), 1350075.

[70] LI, S., LI, Q., LI, W., MOU, X., AND CAI, Y. Statistical properties of digital piecewise linear chaotic maps and their roles in cryptography and pseudo-random coding. In *Cryptography and Coding*. Springer, 2001, pp. 205–221.

[71] LI, S.-J. *Analyses and new designs of digital chaotic ciphers*. PhD thesis, Xi'an Jiaotong University, 2003.

[72] LIAN, S., SUN, J., WANG, J., AND WANG, Z. A chaotic stream cipher and the usage in video protection. *Chaos, Solitons & Fractals 34*, 3 (2007), 851–859.

[73] LIU, H., AND WANG, X. Color image encryption using spatial bit-level permutation and high-dimension chaotic system. *Optics Communications 284*, 16-17 (2011), 3895–3903.

[74] MARSAGLIA, G., AND ZAMAN, A. A new class of random number generators. *The Annals of Applied Probability* (1991), 462–480.

[75] MARTIN, B. *Codage, cryptologie et applications*. PPUR presses polytechniques, 2004.

[76] MASUDA, N., AND AIHARA, K. Cryptosystems with discretized chaotic maps. *Ieee transactions on circuits and systems i : fundamental theory and applications 49*, 1 (2002), 28–40.

[77] MAY, R. M. Simple mathematical models with very complicated dynamics. *Nature 261*, 5560 (1976), 459–467.

[78] OISHI, S., AND INOUE, H. Pseudo-random number generators and chaos. *IEICE TRANSACTIONS (1976-1990) 65*, 9 (1982), 534–541.

[79] OMARY, F. Application des algorithmes évolutionnistes à la cryptographie.

[80] OTT, E. *Chaos in dynamical systems*. Cambridge university press, 2002.

[81] PANDEY, D., RAWAT, U. S., AND KUMAR, A. Robust progressive block based visual cryptography with chaotic map. *Journal of Discrete Mathematical Sciences and Cryptography 19*, 5-6 (2016), 1025–1040.

[82] PAPADIMITRIOU, S., BOUNTIS, T., MAVROUDI, S., AND BEZERIANOS, A. A probabilistic symmetric encryption scheme for very fast secure communication based on chaotic systems of difference equations. *International Journal of Bifurcation and Chaos 11*, 12 (2001), 3107–3115.

[83] PAPADOPOULOS, H., AND WORNELL, G. W. Maximum-likelihood estimation of a class of chaotic signals. *IEEE Transactions on Information Theory 41*, 1 (1995), 312–317.

[84] PATIDAR, V., AND SUD, K. A novel pseudo random bit generator based on chaotic standard map and its testing. *Electronic Journal of Theoretical Physics 6*, 20 (2009), 327–344.

[85] PATIDAR, V., SUD, K. K., AND PAREEK, N. K. A pseudo random bit generator based on chaotic logistic map and its statistical testing. *Informatica 33*, 4 (2009).

[86] PECORA, L. M., AND CARROLL, T. L. Synchronization in chaotic systems. *Physical review letters 64*, 8 (1990), 821.

[87] PESIN, Y. B. Characteristic lyapunov exponents and smooth ergodic theory. *Uspekhi Matematicheskikh Nauk 32*, 4 (1977), 55–112.

[88] PROTOPOPESCU, V. A., SANTORO, R. T., AND TOLLIVER, J. S. Fast and secure encryption-decryption method based on chaotic dynamics, Dec. 26 1995. US Patent 5,479,513.

[89] RABIN, M. O. Digitalized signatures and public-key functions as intractable as factorization. Tech. rep., Massachusetts Inst of Tech Cambridge Lab for Computer Science, 1979.

[90] RIVEST, R. L., SHAMIR, A., AND ADLEMAN, L. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM 21*, 2 (1978), 120–126.

[91] RUKHIN, A., SOTO, J., NECHVATAL, J., SMID, M., AND BARKER, E. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Tech. rep., DTIC Document, 2001.

[92] SHANG, F., SUN, K., AND CAI, Y. An efficient mpeg video encryption scheme based on chaotic cipher. In *Image and Signal Processing, 2008. CISP'08. Congress on* (2008), vol. 3, IEEE, pp. 12–16.

[93] SHANNON, C. E. A mathematical theory of communication. *Bell system technical journal 27*, 3 (1948), 379–423.

[94] SHANNON, C. E. Communication theory of secrecy systems. *Bell system technical journal 28*, 4 (1949), 656–715.

[95] SHARMA, M., KOWAR, M., AND SHARMA, M. An improved evolutionary algorithm for secured image using adaptive genetic algorithm. *Journal of Discrete Mathematical Sciences and Cryptography 11*, 6 (2008), 673–683.

[96] SHUJUN, L., XUANQIN, M., AND YUANLONG, C. Pseudo-random bit generator based on couple chaotic systems and its applications in stream-cipher

cryptography. In *International Conference on Cryptology in India* (2001), Springer, pp. 316–329.

[97] SILVA, C. P., AND YOUNG, A. M. Introduction to chaos-based communications and signal processing. In *2000 IEEE Aerospace Conference. Proceedings (Cat. No. 00TH8484)* (2000), vol. 1, IEEE, pp. 279–299.

[98] SINAI, Y. G. Dynamical systems with elastic reflections. *Russian Mathematical Surveys 25*, 2 (1970), 137–189.

[99] SKROBEK, A. Cryptanalysis of chaotic stream cipher. *Physics Letters A 363*, 1-2 (2007), 84–90.

[100] SKROBEK, A. Approximation of a chaotic orbit as a cryptanalytical method on baptista's cipher. *Physics Letters A 372*, 6 (2008), 849–859.

[101] SNEYERS, R. Climate chaotic instability : statistical determination and theoretical background. *Environmetrics : The official journal of the International Environmetrics Society 8*, 5 (1997), 517–532.

[102] SOLAK, E. Cryptanalysis of chaotic ciphers. In *Chaos-Based Cryptography*. Springer, 2011, pp. 227–256.

[103] STINSON, D. R. *Cryptography : theory and practice*. Chapman and Hall/CRC, 2005.

[104] STOJANOVSKI, T., AND KOCAREV, L. Chaos-based random number generators-part i : analysis [cryptography]. *IEEE Transactions on Circuits and Systems I : Fundamental Theory and Applications 48*, 3 (2001), 281–288.

[105] STROGATZ, S. H. *Nonlinear dynamics and chaos : with applications to physics, biology, chemistry, and engineering*. CRC Press, 2018.

[106] SUNEEL, M. Cryptographic pseudo-random sequences from the chaotic hénon map. *Sadhana 34*, 5 (2009), 689–701.

[107] WANG, X., LIU, L., AND ZHANG, Y. A novel chaotic block image encryption algorithm based on dynamic random growth technique. *Optics and Lasers in Engineering 66* (2015), 10–18.

[108] WU, X., HU, H., AND ZHANG, B. Parameter estimation only from the symbolic sequences generated by chaos system. *Chaos, Solitons & Fractals 22*, 2 (2004), 359–366.

[109] WU, Y., NOONAN, J. P., AGAIAN, S., ET AL. Npcr and uaci randomness tests for image encryption. *Cyber journals : multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT) 1*, 2 (2011), 31–38.

[110] WU, Y., ZHOU, Y., AGAIAN, S., AND NOONAN, J. P. A symmetric image cipher using wave perturbations. *Signal Processing 102* (2014), 122–131.

[111] WU, Y., ZHOU, Y., NOONAN, J. P., AND AGAIAN, S. Design of image cipher using latin squares. *Information Sciences 264* (2014), 317–339.

[112] XIAO, G., LU, M., QIN, L., AND LAI, X. New field of cryptography : Dna cryptography. *Chinese Science Bulletin 51*, 12 (2006), 1413–1420.

[113] XIAOFU, W., SONGGENG, S., ET AL. A general efficient method for chaotic signal estimation. *IEEE Transactions on signal processing 47*, 5 (1999), 1424–1428.

[114] YANG, T. A survey of chaotic secure communication systems. *International journal of computational cognition 2*, 2 (2004), 81–130.

[115] YE, G. Image scrambling encryption algorithm of pixel bit based on chaos map. *Pattern Recognition Letters 31*, 5 (2010), 347–354.

[116] YOUNG, L.-S. Statistical properties of dynamical systems with some hyperbolicity. *Annals of Mathematics 147*, 3 (1998), 585–650.

[117] ZHANG, G., AND LIU, Q. A novel image encryption method based on total shuffling scheme. *Optics communications 284*, 12 (2011), 2775–2780.

[118] ZHENG, F., TIAN, X. J., SONG, J. Y., AND LI, X. Y. Pseudo-random sequence generator based on the generalized henon map. *The Journal of China Universities of Posts and Telecommunications 15*, 3 (2008), 64–68.

[119] ZHOU, H., AND LING, X.-T. Problems with the chaotic inverse system encryption approach. *IEEE Transactions on Circuits and Systems I : Fundamental Theory and Applications 44*, 3 (1997), 268–271.

[120] ZHOU, L.-H., AND FENG, Z.-J. A new idea of using one-dimensional pwl map in digital secure communications-dual-resolution approach. *IEEE Transactions on Circuits and Systems II : Analog and Digital Signal Processing 47*, 10 (2000), 1107–1111.

[121] ZHOU, Y., PANETTA, K., AND AGAIAN, S. Image encryption based on edge information. In *Multimedia on Mobile Devices 2009* (2009), vol. 7256, International Society for Optics and Photonics, p. 725603.

[122] ZHU, W. T. A cost-efficient secure multimedia proxy system. *IEEE transactions on multimedia 10*, 6 (2008), 1214–1220.

[123] ZHU, Z.-L., ZHANG, W., WONG, K.-W., AND YU, H. A chaos-based symmetric image encryption scheme using a bit-level permutation. *Information Sciences 181*, 6 (2011), 1171–1186.

Résumé

Dans cette thèse, nous avons profité de la marche aléatoire et de l'imprévisibilité du mouvement des particules dans un billard chaotique (Billard de Sinaï), pour concevoir de nouveaux algorithmes cryptographiques à clé secrète de longueur arbitraire. Malgré ses propriétés chaotiques bien développées, les systèmes des billards n'ont pas pris plus d'attention par les cryptographes, parmi les raisons est la forme complexe de ces systèmes. Notre première contribution propose un nouveau générateur de nombres pseudo-aléatoires, en se reposant sur les systèmes de deux particules. Tandis que la deuxième, est la conception d'un nouveau schéma de chiffrement des images selon l'architecture de confusion-diffusion, en utilisant les systèmes de trois particules. Dans notre concept, nous avons proposé de nouvelles techniques pour résoudre des problèmes de sécurité et profité le maximum possible du chaos offert par le billard de Sinaï.
Les résultats de simulation montrent que les algorithmes proposés, sont simples à mettre en œuvre, rapides et hautement sécurisés. Ils sont très sensibles à un changement de bit dans la clé, robuste contre les attaques différentielles et passe tous les tests statistiques de validation.

Mots-clés : Générateur de nombres pseudo-aléatoires, NIST, Billard de Sinaï, Confusion, Diffusion, Permutation, Chiffrement des images.

Abstract

In this thesis, we took advantage of the random walk and the unpredictability of the movement of particles in a chaotic billiards (Billiards of Sinai), to design new cryptographic algorithms with secret key of arbitrary length. Despite its well-developed chaotic properties, billiard systems have not been given more attention by cryptographers, among the reasons is the complex form of these systems. Our first contribution proposes a new pseudo-random number generator, based on systems of two particles. While the second is the design of a new image encryption scheme according to the confusion-diffusion architecture, using three-particle systems. In our concept, we have proposed new techniques to solve security problems and took full advantage of the chaos offered by the Sinai billiards.
The simulation results show that the proposed algorithms are simple to implement, fast and highly secure. They are very sensitive to a bit change in the key, robust against differential attacks, and pass all statistical validation tests.

Keywords : Pseudo-random number generator, NIST, Sinaï Billiard, Confusion, Diffusion, Permutation, Image encryption.