الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de L'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITÉ MOHAMED EL BACHIR EL IBRAHIMI -BORDJ BOU ARRERIDJ

FACULTÉ DES SCIENCES ET DE LA TECHNOLOGIE

THÈSE

Présentée au Département d'Électronique

Pour l'obtention du diplôme de

DOCTORAT

Domaine : Sciences et Technologie

Filière : Électronique **Option : Systèmes Embarqués**

Par

YAHI Amina

THÈME

Développement d'Algorithmes de Cryptage d'Images à Base des Suites Chaotiques

Soutenue le 08/02/2023 devant le Jury:

LATOUI Abdelhakim	**M.C.A**	**Univ. Bordj Bou Arreridj**	**Président**
BEKKOUCHE Toufik	**M.C.A.**	**Univ. Bordj Bou Arreridj**	**Directeur de thèse**
DAACHI Mohamed El Hocine	**M.C.A.**	**Univ. Bordj Bou Arreridj**	**Co-Directeur**
ASBAI Nassim	**M.CA.**	**Univ. Bordj Bou Arreridj**	**Examinateur**
BOUGUZEL Saad	**Professeur**	**Univ. Ferhat Abbas Sétif 1**	**Examinateur**
ZIET Lahcene	**Professeur**	**Univ. Ferhat Abbas Sétif 1**	**Examinateur**

Résumé

Les systèmes de sécurité électronique actuels sont basés principalement sur la cryptographie, ceci est dû à la protection intégrale qu'elle offre aux données stockées ou transmises sur Internet, garantissant que les utilisateurs non autorisés ne peuvent pas comprendre ce qu'ils voient. Aussi, l'intégration des suites chaotiques dans les algorithmes de cryptage, augmente le niveau de sécurité de ces derniers, grâce au caractère aléatoire de leurs valeurs, et à leur sensibilité aux valeurs initiales, donc, l'impossibilité de prévoir l'évolution de leurs valeurs à long terme. Ces caractéristiques répondent parfaitement aux besoins des systèmes cryptographiques. À cet égard, nous proposons deux contributions dans le domaine du cryptage des images en couleur, et en niveaux de gris, car, d'une part, les images constituent un pourcentage important de nos messages transmis chaque jour et, d'autre part, la sensibilité et la confidentialité de leur contenu. Quant à la première contribution, elle consiste à développer les performances de l'une des suites chaotiques "cubique" et, à l'intégrer dans un nouvel algorithme de cryptage d'images. Les résultats des tests de performance appliqués sur la suite en particulier, et le système de cryptage en général, ont montré des résultats très satisfaisants. Surtout lorsqu'ils sont comparés aux travaux précédents. Quant à la seconde contribution, elle consiste à créer une nouvelle suite chaotique tridimensionnelle, basée sur les fonctions trigonométriques. Cette dernière fournit à l'algorithme de cryptage d'images dans lequel elle est utilisée, une grande taille de clé, et par conséquent, plus de protection. Les résultats de simulation et l'analyse de la sécurité ont démontré la rapidité et la haute performance jouée par le schéma de cryptage proposé.

Mots clés : Cryptographie, système de cryptage, cryptage des images, images, suite chaotique, protection, sécurité, analyse de la sécurité.

Abstract

Current electronic security systems are mainly based on cryptography, due to the full protection that it offers to data stored, or transmitted over the Internet, ensuring that unauthorized users cannot understand what they are seeing. Also, the integration of chaotic maps into encryption algorithms, raises the level of security in it, thanks to the randomness of their values, their sensitivity to initial values, and thus the inability to predict changes in their values over time. These features perfectly respond to cryptography systems' needs. In this regard, we propose two contributions in the field of color and grayscale image encryption. Since images constitute a large percentage of our daily-transmitted messages ; besides, the sensitivity and the confidentiality of their content. The first contribution is the development of the performance of one of the chaotic suits "cubic map", and the integration of it into a new image encryption algorithm. Performance test results, applied on the suit in particular and the encryption system in general, have shown very satisfactory results. Especially when compared to previous works. The second contribution consists in creating a new three-dimensional chaotic suit based on trigonometric functions. The latter provides the image encryption algorithm in which it is used, a large key size, and therefore, more protection. Simulation results and security analysis have demonstrated the speed and the high performance of the proposed encryption scheme.

Keywords : Cryptography, encryption system, images encryption, images, chaotic suits, protection, security, security analysis.

ملخص

ترتكز أنظمة الحماية الإلكترونية الحالية بصفة أساسية على التشفير، وهذا راجع لما يوفره من حماية شاملة للبيانات المخزنة أو المنتقلة عبر الإنترنت تضمن عدم تمكن الأشخاص الغير مصرح لهم بالوصول إليها من فهم ما يرونه. كما أن دمج المتتاليات العشوائية في خوارزميات التشفير، رفع من مستوى الأمن فيها، وهذا بفضل شكل قيمها العشوائي، وحساسيتها للقيم الابتدائية، بالتالي عدم القدرة على التنبؤ بتغيرات قيمها على المدى البعيد. هذه الخصائص تلبي بشكل مثالي احتياجات أنظمة التشفير. وفي هذا الصدد، نقدم مساهمتين في مجال تشفير الصور الملونة والرمادية. نظرًا لأن الصور تشكل نسبة كبيرة من رسائلنا المرسلة يوميًا من جهة، ولحساسية وخصوصية المحتوى من جهة ثانية. بالنسبة للمساهمة الأولى، فهي عبارة عن تطوير لأداء إحدى المتتاليات العشوائية "المتتالية التكعيبية"، ودمجها ضمن خوارزمية تشفير جديدة. أظهرت نتائج اختبارات فحص الأداء، المطبقة على الدالة بصفة خاصة ونظام التشفير بصفة عامة، نتائج جد مرضية. خاصة بمقارنتها مع أعمال سابقة. بالنسبة للمساهمة الثانية، فتتمثل في انشاء متتالية عشوائية جديدة ثلاثية الأبعاد انطلاقا من الدوال المثلثية، توفر هذه الأخيرة لخوارزمية التشفير التي تم استخدامها فيها، مفتاح بحجم كبير وبالتالي حماية أكبر. بينت نتائج المحاكاة والتحليل الأمني سرعة والأداء العالي لمخطط التشفير المقترح.

الكلمات المفتاحية : تشفير، أنظمة التشفير، تشفير الصور، الصور، المتتاليات العشوائية، الحماية، الأمن، فحص الأداء، التحليل الأمني.

Publications et Communications

Publication

A Color Image Encryption Scheme Based on 1D Cubic Map. Amina Yahi, Tewfik Bekkouche, Mohamed El Hossine Daachi, Nacira Diffellah. Optik, Volume 249, 2022, 168290, ISSN 0030-4026,
https ://doi.org/10.1016/j.ijleo.2021.168290.
(https ://www.sciencedirect.com/science/article/pii/S0030402621018076)

Communications

— Biometric Image Encryption Scheme based on Modified Double Random Phase Encoding System. Conference, Amina Yahi, Tewfik Bekkouche, Mohamed El Hossine Daachi, Nacira Diffellah, the 2nd International Conference on Computer Science's Complex Systems and their Applications ICCSA'2021 At Oum El Bouaghi Algerie.

— New Method for Image Encryption Using GOST And Chaotic Tent Map, Amina Yahi, Tewfik Bekkouche, Mohamed El Hossine Daachi, Nacira Diffellah, Artificial Intelligence and its Applications (AIAP2021) University of Eloued.
(http ://dspace.univ-eloued.dz/handle/123456789/10824)

Remerciement

Je tiens à remercier en premier lieu **ALLAH**, le tout puissant qui m'a donné la force la patience et la volonté pour terminer ce travail. Alhamdulillah.

Je me ferais un devoir agréable de remercier mon directeur de thèse **Dr T.Bekkouche**, pour sa disponibilité, son soutien, et sa patience. Je suis très reconnaissante de tous ce que vous avez fait.

Je remercie aussi mon co-directeur de thèse **Dr M. H. Daachi** de m'avoir honoré de collaborer à enrichir cette thèse par ces avis compétents, et sa disponibilité en cas de besoin.

Je tiens à citer dans ces remerciements les membres du jury qui ont bien voulu examiner et juger notre travail : **Dr N. Asbai**, **Pr S. Bouguezel**, **Pr L. Ziet** et **Dr A. Latoui** à leur tête.

Je remercie chaleureusement mes parents de fond de cœur pour m'avoir accompagné, aidé, soutenu moralement et financièrement.

J'adresse aussi par mes remerciements à tous les membres du laboratoire d'électronique et des télécommunications avancées **(ETA)**, sous la direction de **Dr M. A. Talbi**.

Mes remerciements vont également à tous celles et ceux qui ont contribué de près ou de loin à la réalisation de cette thèse.

Dédicace

Je dédié ce travail accompagné par un profond amour à mes très chers parents, la source de joie et mon bonheur, les mots me trompe de tous ce qu'il s'agit de vous. Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour l'encouragement, le soutien physique moral et financier, et les sacrifices que vous avez consenti depuis mon enfance pour mon instruction et mon bien être.

À ma jolie grande mère, la source de tendresse et bonté, je ne sais même pas par ou commencer, par ta motivation, par ton inspiration ou par les regards bienveillants et le doux sourire que tu me donne malgré la maladie qui t'a étouffé, tu essais toujours d'être forte pour nous, mais je te le dis mon cœurs, nous sommes fortes par toi.

À ma sœur et ma belle sœur, qui sont mes meilleures amies. Je sais que vous êtes là pour moi et que je peux compter sur vous pour me soutenir et me motiver. je vous suis infiniment reconnaissante. Je vous aime !

Mais surtout à mes frères, qu'ils se sont toujours montrés là pour moi. Vous êtes mon inspiration et ma source de force. Je suis si fière de vous. Je vous souhaite tout le bonheur et la réussite dans la vie. Avec tout mon amour.

À vous mes copines, et surtout à **Samira** et **Saloua**, **Hayat** et **Amina**. Merci pour tout ce que vous avez fait pour moi et pour votre présence, votre aide et votre amour inconditionnel.

Sans oublier mes petits neveux et nièces, je vous dédie également ce travail. Vous êtes des personnes très spéciales pour moi et je suis tellement contente de vous voir grandir. Je vous souhaite le meilleur et le succès dans tous ce que vous entreprenez.

Je dédie aussi ce travail à mes tantes et mes oncles, à mes cousins et mes cousines, et à tous ceux et celles que j'ai omis de citer.

Table des matières

Liste des Figures **VI**

Liste des Tableaux **X**

Liste des Abréviations **XII**

Introduction générale **1**

1 Théorie du Chaos et Systèmes Chaotiques **4**

1.1 Introduction 4

1.2 Théorie du Chaos 4

1.2.1 Travaux de Poincaré & Lorenz 5

1.2.2 Effet papillon 5

1.2.3 Attracteur de Lorenz 5

1.3 Terminologie 6

1.4 Applications du chaos 7

1.5 Théorie de bifurcation 9

1.6 Géométrie fractale 9

1.7 Systèmes dynamiques 10

1.8 Systèmes chaotiques 10

1.9 Générateurs de nombres aléatoires basés sur les systèmes chaotiques 11

1.10 Différents tests des systèmes chaotiques 11

1.10.1 Tests statistiques 11

1.10.1.1 Test de séquence de nombres pseudo-aléatoires (ENT) 12

1.10.1.2 Batterie de test DIEHARD 13

1.10.1.3 Batterie de test Nist (National Institute of Standards and Technology) . . . 16
1.10.1.4 Batterie TestU01 . . . 22
1.10.1.5 Practrand test . . . 22
1.10.2 Tests graphiques . . . 23
1.10.2.1 Diagramme de Lyapunov . . . 23
1.10.2.2 Diagramme de bifurcation . . . 23
1.10.2.3 Test de 0-1 (0-1 test) . . . 24
1.10.2.4 Analyse de trajectoire . . . 24
1.10.2.5 Test de la complexité linéaire . . . 25
1.10.3 Exemples de cartes chaotiques . . . 25
1.10.3.1 Cartes chaotiques classiques unidimensionnelles . . . 25
1.10.3.2 Cartes chaotiques classiques multidimensionnelles . . . 27
1.10.3.3 Cartes chaotiques modernes . . . 29
1.11 Conclusion . . . 34

2 Généralités sur le Cryptage d'Images 35

2.1 Introduction . . . 35
2.2 Historique . . . 35
2.3 Terminologie . . . 37
2.4 Méthodes crypto-analytiques . . . 38
2.5 Objectifs de la cryptographie . . . 39
2.6 Classification des systèmes de cryptage . . . 39
2.6.1 Selon l'architecture . . . 39
2.6.2 Selon la clé . . . 40
2.6.3 Selon le pourcentage de données cryptées . . . 40
2.6.4 Selon la plateforme . . . 40
2.7 Principes de Kirchhoff . . . 40
2.8 Théorème de Shannon . . . 41
2.9 Cryptographie visuelle . . . 43
2.10 Cryptographie quantique . . . 43
2.11 Multimédia et cryptographie . . . 44
2.12 Algorithmes cryptographiques . . . 45
2.12.1 DES, D-DES, T-DES . . . 45

2.12.1.1 Création des clés : . . . 45
2.12.1.2 Procédure de cryptage . . . 45
2.12.2 Advanced Encryption Standard(AES) . . . 46
2.12.3 GOST . . . 46
2.12.4 RC4 . . . 47
2.13 Chaos dans le cryptage . . . 47
2.14 Cryptage d'image . . . 48
2.14.1 Techniques de cryptage d'image . . . 49
2.14.2 Métriques d'évaluation . . . 49
2.14.2.1 Vision humaine . . . 49
2.14.2.2 Analyse des histogrammes . . . 50
2.14.2.3 Calcul de l'entropie . . . 50
2.14.2.4 Analyse de corrélation . . . 50
2.14.2.5 Sensibilité de l'image vis à vis des attaques différentielles . . . 51
2.14.2.6 PSNR, SSIM . . . 52
2.14.2.7 Erreur absolue moyenne (MAE) . . . 53
2.14.2.8 Autres tests quantitatifs . . . 53
2.14.2.9 Sensibilité de la clé . . . 53
2.14.2.10 Espace clé . . . 54
2.14.2.11 Rapidité, implémentation et coût . . . 54
2.14.2.12 Test de NIST . . . 55
2.14.2.13 Problèmes de transmission . . . 55
2.14.3 Méthodes de cryptage d'image . . . 56
2.14.3.1 Cryptage à base du chaos . . . 56
2.14.3.2 Cryptage à base de l'hyper chaos . . . 57
2.14.3.3 Cryptage avec la technique d'ADN . . . 58
2.14.3.4 Cryptage à base des transformées . . . 60
2.14.3.5 Cryptage à base des réseaux de neutrons . . . 62
2.14.3.6 Cryptage à base des courbes elliptiques . . . 63
2.15 Conclusion . . . 64

3 Schéma de Cryptage d'Images en Couleurs Basé sur une Carte Cubique 1D 65

3.1 Introduction . . . 65
3.2 Structure de la carte chaotique proposée . . . 66

3.2.1 Carte cubique proposée (1-DCE) 66
3.3 Test de performance 67
3.3.1 Comportement chaotique 67
3.3.2 Exposant de Lyapunov 68
3.3.3 Diagramme de bifurcation 69
3.3.4 Evaluation de la sensibilité 69
3.3.5 Analyse de tests statistiques 72
3.4 Système de cryptage proposé 74
3.4.1 Méthode de cryptage 74
3.4.1.1 Confusion 75
3.4.1.2 Diffusion 75
3.4.2 Méthode de décryptage 78
3.5 Résultats de la simulation et analyse de la sécurité 79
3.5.1 Analyse des histogrammes 79
3.5.2 Entropie 80
3.5.3 Analyse de corrélation 81
3.5.4 Sensibilité de l'image en clair 82
3.5.5 Analyse des espaces clés 83
3.5.6 Sensibilité de la clé 83
3.5.7 Perte de données 84
3.5.8 Analyse de la vitesse 85
3.6 Conclusion 86

4 Nouvelle Carte Chaotique Trigonométrique 3D et son Application dans un Système de Cryptage d'Image **87**
4.1 Introduction 87
4.2 Carte chaotique 3D proposée 87
4.3 Évaluation de la performance 88
4.3.1 Évaluation graphique 88
4.3.2 Évaluation statistique 92
4.4 Crypto-système proposé 93
4.4.1 Schéma de cryptage 94
4.4.2 Schéma de décryptage 98
4.5 Résultats expérimentaux et analyse de la sécurité 99

4.5.1 Analyse d'histogrammes 99
4.5.2 Analyse des coefficients de corrélation 100
4.5.3 Analyse de l'entropie de l'information 101
4.5.4 Attaque différentielle 101
4.5.5 Analyse de SSIM et PSNR 103
4.5.6 Analyse de l'espace clé 103
4.5.7 Sensibilité de la clé 104
4.5.8 Analyse de vitesse 107
4.5.9 Perte de données 107
4.6 Conclusion 108

Conclusion générale **110**

Bibliographie **112**

Table des figures

1.1 Attracteur étrange de Lorenz. 6
1.2 Ensemble de Mandelbrot. 9
1.3 Diagramme de Lyapunov de la carte logistique. 26
1.4 Diagramme de bifurcation de la carte Tent. 26
1.5 Comportement de la carte quadratique pour les 20 premières itérations. 27
1.6 Attracteurs hyper-chaotique de système de Chen. (a) sur le plane $(x - y)$, (b) et le plane $(y - z)$. 29
1.7 Diagramme de la toile d'araignée de la carte Tent à gauche, et de LTM à droite. . . 30
1.8 Complexité linéaire de PRNG : carte CLM vs carte logistique. 31
1.9 Sensibilité de 1-DCP : (a) Valeur initiale x_0 ; (b) Valeur du paramètre μ. 32

2.1 Schéma d'un crypto-système. 38
2.2 Schéma général d'un système de communication. 41
2.3 Exemple d'un crypto-système quantique. 44
2.4 Image claire et image cryptée. 49
2.5 Histogrammes de l'image en niveaux de gris et de son image chiffrée. 50
2.6 Exemple d'une image avec une perte partielle d'un quart dans deux endroits différents. 55
2.7 Organigramme de l'algorithme proposé par Zhang S et al 60
2.8 Implémentation opto-digitale d'un système de cryptage à base de DFRFTDRPE/ MPDFRFT-DRPE. 61
2.9 Schéma proposé de cryptage d'image. 62

3.1 Comparaison entre le comportement chaotique et le diagramme Cobweb de la carte cubique classique et 1-DCE. (a), (b) Comportement chaotique de la carte cubique classique. (c), (d) Comportement chaotique de 1-DCE. (e) Diagramme en toile d'araignée de la carte cubique classique. (f) Diagramme en toile d'araignée de 1-DCE. 67
3.2 Analyse et comparaison des exposants de Lyapunov. 68
3.3 Diagramme de bifurcation pour : (a) la carte cubique classique, (b) la carte 1-DCE. 69
3.4 Sensibilité des cartes chaotiques susmentionnées aux changements de $10^{(-17)}$ dans la valeur initiale, par rapport au graphique de comparaison entre les séries temporelles avec et sans changement dans le cas de (a) la carte cubique classique, (b) 1-DCP, (c) 1-DCE. 70
3.5 Sensibilité des cartes chaotiques susmentionnées aux changements de $10^{(-14)}$ dans r, par rapport au graphique de comparaison entre les séries temporelles avec et sans changement dans le cas de (a) la carte cubique classique, (b) 1-DCP, (c) 1-DCE. 71
3.6 Schéma fonctionnel du crypto-système proposé. 74
3.7 Exemple du processus de diffusion proposé pour une matrice de taille 4×4. (a) Matrice encadrée par un ruban. (b) Matrice après l'application de la première opération. (c) Décalage de la zone de calcul d'une seule colonne. (d) Matrice après la seconde opération. (e) Décalage de la zone de calcul d'une seule colonne. (f) Après avoir appliqué les opérations sur la zone encadrée. (g) Un autre décalage. (h) Matrice après avoir été entièrement balayée. (i) Masque. 76
3.8 Diagramme de bloc du processus de décryptage. 78
3.9 Quatre images claires et leurs images cryptées. 79
3.10 Histogrammes de l'image en niveau de gris brute et de son image chiffrée. 80
3.11 Histogrammes de l'image en couleur brute et de l'image chiffrée. 80
3.12 Analyse de corrélation. 81
3.13 Image décryptée de poivrons avec un léger changement de clés. (a) $k = k^{'}$. (b) $x'_{01} = x_{01} + 10^{-17}$. (c) $x'_{02} = x_{02} + 10^{-17}$. (d) $x'_{03} = x_{03} + 10^{-17}$. (e) $r'_1 = r_1 + 10^{-14}$. (f) $r'_2 = r_2 + 10^{-14}$. (g) $r'_3 = r_3 + 10^{-14}$ 84
3.14 Test par attaque en texte brut (a) Image brute. (b) Image chiffrée avec la clé k_2. (c) Image chiffrée avec la clé k_2. (d) Différence de pixel à pixel. 84
3.15 Test de perte de données : Images cryptées avec (a) 160×128 de perte de données, (b) 80×64 de perte de données, (c) 80×256 de perte de données. Les images décryptées correspondantes (d), (e), (f) respectivement. 85

4.1 Diagrammes de bifurcation pour chaque composante de 3D-CSC. 89
4.2 Exposant de Lyapunov pour chaque composante de 3D-CSC comparé au 3D-LTM. (a) Composante x. (b) Composante y. (c) composante z. 89
4.3 Attracteur chaotique de 3D-CSC. 90
4.4 Sensibilité de 3D-CSC à un changement minime de $10^{(-16)}$ aux valeurs initiales. (a) à x_0. (b) à y_0. (c) à z_0. 91
4.5 Sensibilité de 3D-CSC à un changement minime de $10^{(-16)}$ aux valeurs des paramètres de contrôle. (a) au r_1. (b) au r_2. (c) au r_3. 91
4.6 Représentation du dernier processus appliqué sur une matrice 4×4 en fonction des valeurs de vecteurs right/left et up/downs. (a) La matrice avec les vecteurs de position avant le début du processus de confusion. (b) La première étape du processus de confusion (inter-changer les deux colonnes de milieu). (c) La deuxième étape : échanger les deux lignes du milieu. (d) Étape 1 et étape 2 pour le deuxième vecteur horizontal/vertical à partir du milieu. (e) Étape 1, 2 pour le dernier vecteur horizontal/vertical à partir du milieu. 97
4.7 Résultat de l'application de la dernière étape directement sur l'image originale. . . 98
4.8 Système proposé de cryptage/décryptage. 98
4.9 Histogrammes d'une image couleur en clair et de son image chiffrée. 99
4.10 Histogrammes d'une image en niveau de gris en clair et de son image chiffrée. . . . 99
4.11 Distribution des pixels voisins dans différentes directions. La première ligne représente l'image claire, la deuxième ligne représente l'image chiffrée. 100
4.12 Image décryptée de "Peppers" avec un léger changement dans l'une des conditions initiales. (a)$x_{01}^{'} = x_{01} + 10^{-16}$. (b)$y_{01}^{'} = y_{01} + 10^{-16}$. (c)$z_{01}^{'} = z_{01} + 10^{-16}$. (d)$x_{02}^{'} = x_{02} + 10^{-16}$. (e)$y_{02}^{'} = y_{02} + 10^{-16}$. (f)$z_{02}^{'} = z_{02} + 10^{-16}$. (g)$x_{03}^{'} = x_{03} + 10^{-16}$. (h)$y_{03}^{'} = y_{03} + 10^{-16}$.(i)$z_{03}^{'} = z_{03} + 10^{-16}$.(j)$x_{04}^{'} = x_{04} + 10^{-16}(k)y_{04}^{'} = y_{04} + 10^{-16}$.(l)$z_{04}^{'} = z_{04} + 10^{-16}$ 105
4.13 Image décryptée de "Peppers" avec un léger changement dans l'un des paramètres de contrôle. (a) $r_{11}^{'} = r_{11} + 10^{-15}$. (b) $r_{12}^{'} = r_{12} + 10^{-15}$. (c) $r_{13}^{'} = r_{13} + 10^{-15}$. (d) $r_{21}^{'} = r_{21} + 10^{-15}$. (e) $r_{22}^{'} = r_{22} + 10^{-15}$. (f) $r_{23}^{'} = r_{23} + 10^{-15}$. (g) $r_{31}^{'} = r_{31} + 10^{-15}$. (h) $r_{32}^{'} = r_{32} + 10^{-15}$.(i) $r_{33}^{'} = r_{33} + 10^{-15}$.(j) $r_{41}^{'} = r_{41} + 10^{-15}$. (k) $r_{42}^{'} = r_{42} + 10^{-15}$.(l) $r_{43}^{'} = r_{43} + 10^{-15}$ 105
4.14 Test par attaque en texte clair (a) Image claire. (b) Image chiffrée par la clé k_1. (c) Image chiffrée par la clé k_2. (d) Image de différence entre C1 et C2. 106

4.15 Résultat d'analyse de la sensibilité de la clé secrète en employant NPCR et UACI dans la phase de décryptage. 107
4.16 Test de perte de données avec : (a) 96×128 de données perdues, (b) 96×512 de données perdues, de haut de l'image cryptée, (c) 96×512 de données perdues, de bas de l'image cryptée. images décryptées correspondantes(d), (e), (f) respectivement. . 108

Liste des Tableaux

1.1 Valeurs possibles de M en fonction de la taille de la série correspondante. 18
1.2 Valeurs de v_i conformément à la taille de M. 18
1.3 Valeurs de K et N selon les valeurs de M. 18
1.4 Comparaison des performances : carte logistique classique vs carte logistique améliorée. 31

2.1 Comparaison entre le chaos et les propriétés cryptographiques. 48
2.3 Règles de codage et de décodage de la séquence d'ADN. 59
2.4 Opérations logiques XOR pour les séquences d'ADN. 59

3.1 Comportement dynamique de la carte cubique classique. 66
3.2 Résultats du test Nist de 1-DCE. 72
3.3 Résultats du test DIEHARD de 1-DCE. 73
3.4 Suite de tests d'Ent appliquée sur 1-DCE. 73
3.6 Comparaison de différentes valeurs d'entropie. 80
3.7 Coefficients de corrélation des pixels adjacents dans l'image cryptée et l'image brute, et analyse comparative avec divers algorithmes de cryptage. 81
3.8 Résultats de l'analyse de NPCR. 82
3.9 Résultats de l'analyse de UACI. 82
3.10 Comparaison entre les tailles de l'espace clé. 83
3.11 Temps d'exécution. 85

4.1 Résultats de test de NIST de notre système et d'un autre système. 92
4.2 Résultats de test de DIEHARD de notre système en comparant avec un autre système. 93
4.3 Comparaison des résultats obtenus du coefficient de corrélation entre la méthode proposée et d'autres méthodes. 100

4.4 Analyse d'entropie de l'information dans le cas d'images en couleur. 101
4.5 Analyse d'entropie de l'information dans le cas d'images en niveaux de gris. 101
4.6 Résultats de l'analyse de NPCR. 102
4.7 Résultats de l'analyse d'UACI. 102
4.8 Résultats de NPCR pour les images en niveau de gris. 102
4.9 Résultats de UACI pour les images en niveau de gris. 102
4.10 Analyse de SSIM et PSNR pour une image en couleur. 103
4.11 Analyse de SSIM et PSNR pour une image en niveau de gris. 103
4.12 Analyse de l'espace clé. 104
4.13 Valeurs de PSNR après une petite modification dans un élément de la clé. 106
4.14 Temps d'exécution. 107

Liste des Abréviations

ODE45	*Ordinary Differential Equations Runge-Kutta(4,5)*
CD-ROM	*Compact Disc Read-Only Memory*
ENT	*A Pseudorandom Number Sequence Test Program*
GCD	*Greatest Common Divisor*
NIST	*National Institute of Standards and Technology*
RNG	*Random Number Generator*
LTM	*Logistic Tent Map*
CLM	*Cosinus Logistique Map*
1-DCP	*One-Dimensional Cosine Polynomial*
3D-PHM	*Tridimensionnel Piecewise-Hénon-Map*
DES	*Data Encryption Standard*
RSA	*Rivest Shamir Adleman*
AES	*Advanced Encryption Standard*
GOST	*Government Standard*
RC4	*Rivesr Cipher*
SEAL	*Software Optimized Encryption Algorithm*
BB84	*Bennett and rassard 1984*

NPCR *Number of Pixels Change Rate*

UACI *Unified Average Changing Intensity*

PSNR *Peak Signal to Noise Ratio*

SSIM *Structural Similarity Index Mesurement*

MSE *Mean Square Error*

MAE *Mean Absolate Error*

1-DCF *One Dimensionnel Cosinus Fractional*

SHA-256 *Secure Hash Algorithm*

CML *Coupled Map Lattice*

PWLCM *Piecewise Linear Chaotic Map*

DRPE *Double Random Phase Encoding*

MPDFRFT *Multiple-Parameter Discrete Fractional Fourier Transform*

PLCM *Piecewise Linear Chaotic Map*

FrFT *Fractional Fourier Transform*

SLM *spatial light modulator*

CCD *Charge Coupled Device*

R2C *Real To Complexe*

RPMPFRFT *Reality-Preserving Multiple-Parameter Fractional Fourier Transform*

FCNN *Fuzzy Cellular Neural Network*

ECDH *Elliptical Courbes of Diffie-Hellman*

1-DCE *One Dimensionnel Cubic Exponentiel*

1-DSP *Analyse en Composantes Principales*

MLE *Analyse en Composantes Principales*

3D-CSC *Analyse en Composantes Principales*

3D-LTM *Analyse en Composantes Principales*

Introduction générale

L'importance de la sécurité informatique est en perpétuelle croissance. Cela est dû à l'augmentation sans précédent de l'utilisation des technologies de l'information et de la communication. Le cryptage est l'une des méthodes de protection les plus efficaces [1]. Où, depuis la nuit des temps, il joue un rôle majeur dans la vie de l'être humain, il était la cause du meurtre de plusieurs personnalités historiques, comme il avait un impact considérable sur le cours des guerres à l'époque [2]. Le besoin de chiffrement est aujourd'hui en hausse, il n'est plus restreint aux applications politiques et militaires, mais il est plutôt utilisé par quiconque souhaite garder ses informations confidentielles. On peut le définir comme une ou plusieurs méthodes de transformation du message clair en un message complètement illisible par toute personne autre que le destinataire. Un système de cryptage sécurisé doit assurer les points suivants : la confidentialité, l'authenticité, l'intégrité et la non répudiation. L'ajout de la théorie du chaos à la cryptographie par Matthews en 1989 [3], a énormément dupliqué la sécurité. La théorie du chaos en elle-même est une percée mathématique, physique et même philosophique, elle a entravé l'homme et a révélé l'humilité de la science, ainsi que l'inexistence de sa globale domination prétendue. Malgré son élargissement, elle demeure limitée. Pour mieux comprendre de quoi s'agit-il cette théorie, revenons aux années soixante. Après plusieurs tâtonnements de prédire la météo, Lorenz conclut que c'est impossible d'anticiper la météo à long terme, vu sa dépendance sensitive aux conditions initiales, où la moindre brise d'air en Amérique pourra générer une tornade en Afrique, et cette même brise d'air peut empêcher une tornade qui pourrait frapper l'Afrique. Ce qui est étrange, c'est que malgré la puissance technologique actuelle, les satellites, les grandes centres météorologiques, les calculateurs puissants, les états météorologiques à long terme sont encore impossibles à prévoir. Lorenz conclut aussi que les phénomènes qui semblent aléatoires et incohérents sont en fait, ordonnés et en harmonie. L'usage des cartes chaotiques classiques, améliorées et toutes neuves dans la cryptographie, constitue un atout important, et c'est grâce à leur comportement chaotique, l'ergodicité, et surtout la dépendance sensitive aux paramètres de contrôle et aux valeurs initiales. Les paramètres de contrôle et

les valeurs initiales jouent le rôle de la clé secrète de tel schéma de cryptage, sachant que le succès d'un crypto-système repose essentiellement sur l'impossibilité de se procurer de la clé. Les cartes chaotiques peuvent être classées selon la dimension en deux catégories : les cartes chaotiques de faible dimension et de haute dimension. Chaque catégorie a ses avantages et ses inconvénients. Aussi, depuis environ deux décennies, l'humanité vit ce qu'on peut l'appeler l'ère de l'image, les images ont envahi notre quotidien, et elles font une partie intégrante des données stockées ou transmises sur les différents réseaux, leur contenu peut être secret et sensible auquel les propriétaires ne veulent pas que d'autres personnes accèdent sans autorisation, comme les images médicales et militaires. Il est donc nécessaire de mettre en place une méthode sécurisée de partage ou de stockage des images, en leur garantissant une protection globale contre les attaques numériques, telles que les vols, l'espionnage, les modifications et le dénigrement. Le cryptage d'image en utilisant le chaos représente l'ensemble magique qui permet d'atteindre cet objectif, étant donné les caractéristiques de l'image comme la grande taille, la forte corrélation entre les pixels, et le niveau élevé de densité de données [4]. Sur la même voie, nous proposons une nouvelle carte chaotique modulaire unidimensionnelle basée sur la composition de la carte cubique classique et de la fonction exponentielle. Ce système a amélioré considérablement le comportement chaotique de la carte cubique classique. Ensuite, nous l'investissons dans un nouveau schéma de cryptage d'image basé sur le modèle de confusion-diffusion. Une méthode de diffusion efficace est réalisée moyennant un masque et un ruban construit par ce système, elle permet de changer complètement les valeurs des pixels de manière aléatoire. Encore plus, nous proposons également un autre système chaotique tridimensionnel avec une simple structure, un faible temps de calcul et de performance chaotique élevée ; il est conçu à partir de fonctions trigonométriques, nous l'avons exploité bénéfiquement dans un nouveau schéma de cryptage d'image, qui s'articule auteur d'un double processus de permutation-diffusion pour augmenter davantage sa complexité.

La thèse est scindée en quatre chapitres et elle est organisée comme suit :

A. Le premier chapitre est consacré à introduire la théorie du chaos, les différentes notions liées à elle, un aperçu sur les systèmes chaotiques, y compris les tests de performance.

B. Le deuxième chapitre est destiné à la mise en évidence de la cryptographie et le cryptage d'images, ainsi que certains travaux effectués à cet égard.

C. le troisième chapitre est réservé à la présentation de notre première contribution. Il s'agit d'une amélioration de performance de la carte chaotique classique cubique, pour renforcer la sécurité du système de cryptage d'image, proposé dans ce même chapitre. Les résul-

tats d'un ensemble de tests appliqués, pour visualiser la qualité de la carte améliorée et la robustesse de notre système proposé, sont également exposés.

D. Nous allons présenter dans le quatrième chapitre notre deuxième contribution, elle se compose d'un nouveau système chaotique tridimensionnel simple et facile à implémenter, et d'un nouvel algorithme de cryptage/décryptage d'images, en utilisant ce système chaotique proposé. Comme nous allons communiquer les résultats issus de l'analyse sécuritaire.

Nous terminons par une conclusion et quelques perspectives.

Chapitre 1

Théorie du Chaos et Systèmes Chaotiques

Sommaire

1.1 Introduction
1.2 Théorie du Chaos
1.3 Terminologie
1.4 Applications du chaos
1.5 Théorie de bifurcation
1.6 Géométrie Fractale
1.7 Systèmes dynamiques
1.8 Systèmes chaotiques
1.9 Générateurs de nombres aléatoires basés sur les systèmes chaotiques
1.10 Différents tests des systèmes chaotiques
1.11 Conclusion

Chapitre 1

Théorie du Chaos et Systèmes Chaotiques

1.1 Introduction

Au fil du temps, l'homme a toujours essayé de sécuriser les informations transmises sous forme de messages. Pour cela, il a inventé des techniques de chiffrement allant des plus simples jusqu'aux plus complexes tels que l'emploi des suites chaotiques. En effet, la théorie du chaos est l'une des dernières théories physique-mathématique qui a ébranlé les quatre coins du monde. La dénomination du "chaos" n'est qu'un mot d'attraction, selon certains, c'est un mot exagéré, car le chaos étudié mêle l'ordre intrinsèque (un modèle mathématique) et le désordre extrinsèque (un comportement chaotique imprévisible). Pour qualifier un système de "chaotique", il faut qu'il ait une très haute sensibilité aux conditions initiales.

Dans ce chapitre, nous nous intéressons plus particulièrement à la théorie du chaos avec également les multiples tests qualitatifs et quantitatifs des systèmes chaotiques. Aussi, nous présentons quelques classiques et nouveaux systèmes chaotiques. À noter qu'on peut les renommer par cartes, suites, équations et même fonctions chaotiques.

1.2 Théorie du Chaos

À la première vue, le mot chaos fait allusion au désordre et confusion, mais dans les faits, c'est plus profond que ça. En effet, il s'agit d'une approche qui a met en question les lois de la physique classique de Newton, qui s'appuient principalement sur le déterminisme et la certitude. Cependant, le chaos en question ce n'est qu'un phénomène visiblement aléatoire, mais de nature déterministe. À travers lequel les savants cherchent à comprendre notre splendide univers. Dans cette partie du chapitre, nous donnons un historique concernant la genèse de la théorie du chaos.

1.2.1 Travaux de Poincaré & Lorenz

Les premières études servant de base pour la théorie du chaos reviennent au mathématicien français Henri Poincaré, à la fin du xix^e siècle [5]. Il travaillait sur la mécanique céleste lorsqu'il a remarqué qu'une petite erreur dans les conditions initiales peut engendrer une très grande différence dans l'état final, d'un tel phénomène *"Une cause très petite, qui nous échappe, détermine un effet considérable que nous ne pouvons pas ne pas voir"* [6]. Dans son livre "Calcul des probabilités", il expliqua que les phénomènes qui ont l'air aléatoires et fortuits ne le sont pas vraiment, et cela résulte plutôt d'une part de notre ignorance et d'autre part de causes infinitésimales, en disant *"Il faut donc bien que le hasard soit autre chose que le nom que nous donnons à notre ignorance"*. À son époque, l'idée ne fut pas valorisée par ses homologues. De même, la découverte du météorologue Américain Edward Lorenz s'agissant de la métrologie, a également subit aux moqueries de ses contemporaines. Aujourd'hui, cette découverte est considérée comme une troisième révolution scientifique. Lorenz, depuis tout petit, était passionné par l'apériodicité de la météo. Son histoire avec la théorie du chaos commença un jour d'hiver en 1961, étudiant la prédiction météorologique, il reprit l'exécution de son programme contenant les mêmes équations mathématiques, il entra une valeur arrondie des conditions initiales, elle était 0.506127. Par contre, cette fois-ci, il ne garda que trois chiffres après la virgule, supposant qu'une différence de un sur mille est insignifiante. Le résultat aurait dû être précisément l'ancien, mais Lorenz vit la divergence de ses prévisions des antécédents d'une manière très rapide, en seulement quelques mois, sachant qu'il n'y avait aucune anomalie au niveau de son ordinateur le Royal McBee. À ce moment-là, il comprit que le dysfonctionnement réside dans la valeur tronquée qu'il avait tapé comme condition initiale (0.506). De ce fait, il finit par conclure que c'est inévitablement impossible de prédire la météo à long terme, ce qui est vrai jusqu'à l'heure actuelle [5].

1.2.2 Effet papillon

Après la fameuse expérience de Lorenz, il conclut aussi que contrairement aux pré-suppositions de la science classique (un petit changement dans l'état initial donne un petit changement dans l'état final), les petits changements dans l'état initial peuvent se développer et s'accumuler pour avoir un effet considérable sur l'état final. Ainsi, la loi d'effet papillon a vu le jour, dont l'énoncé est : *"le battement des ailes d'un papillon au Brésil déclenche une tornade au Texas"*.

1.2.3 Attracteur de Lorenz

Par la suite, Lorenz introduisit un nouveau système mathématique de comportement complexe

pour représenter un écoulement d'un gaz ou d'un liquide sous l'influence de la chaleur. Ce phénomène est connu sous le nom de la convection thermique qui s'articule sur les trois équations différentielles suivantes :

$$\begin{cases} \frac{dx}{dt} = -10x + 10y \\ \frac{dy}{dt} = rx - y - xz \\ \frac{dz}{dt} = \frac{8}{3}z + xy \end{cases} \tag{1.1}$$

Avec x, y et z sont des variables d'état, dont x est proportionnelle à l'intensité du mouvement convectif, y est proportionnelle à la différence de température entre les courants ascendants et descendants, tandis que z est proportionnelle à la distorsion de la linéarité du profil vertical de la température. Quant à r, c'est le paramètre de contrôle. Pour r$\geq$24.73, ce système passe vers un régime chaotique. En 1963, Lorenz publia les résultats de ces recherches dans le journal "journal of the atmospheric sciences" sous le nom "Deterministic non periodic flow". L'article fut clôturé par une représentation graphique des solutions du modèle proposé sur l'espace des phases de trois dimensions, auquel on remarque une géométrie assez particulière : deux spirales en forme de papillon contenants des orbites denses. Cet étrange courbe porte le nom "attracteur de Lorenz". Pour son implémentation, il suffit juste de faire appel à la fonction prédéfinie "ODE45" issue de Matlab (Figure 1.1).

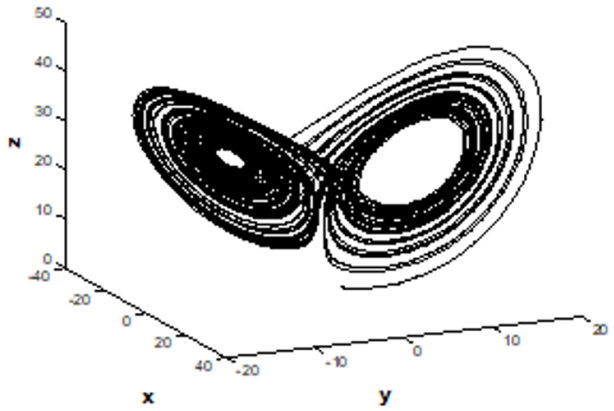


FIGURE 1.1 – Attracteur étrange de Lorenz.

1.3 Terminologie

Nous allons exposer les définitions des concepts qui nos semblent intéressantes :

— **Attracteur étrange :** Le vocabulaire de la théorie des systèmes dynamiques fut enrichi par un nouveau mot "attirer" et ses dérivés.

Définition 1.3.1. *[7, 8] Soit A un ensemble compact, fermé de l'espace des phases. On suppose que A est un ensemble invariant (i.e.* $\phi_t = (A)$ *pour tout t). on dit que A est*

stable si pour tout voisinage U de A, il existe un voisinage V de A tel que toute solution $x(x_0, t) = \phi_t(x_0)$ restera dans U si $x_0 \in V$ Si de plus :

$$\bigcap\nolimits_{t\geq 0} \phi_t(V) = A. \tag{1.2}$$

Et s'il existe une orbite dense dans A, alors A est un attracteur.

Quant à un attracteur étrange, c'est un objet géométrique atypique de l'attracteur. Il a été présenté par Ruelle et Takens [9] et il est caractérisé par :

- L'attracteur est de volume nul dans l'espace des phases ;
- Soit n la dimension de l'espace des phases de l'attracteur, sa dimension fractale d est : $2 < d < n$;
- Sensibilité aux conditions initiales.

— **Système aléatoire :** Un système est dit aléatoire lorsque la modélisation de son évolution par une équation mathématique ainsi que la prévision de ses futurs états s'avèrent impossible.

— **Système déterministe :** C'est un système dynamique, dont son évolution est prévisible, étant donné qu'il est régi par des lois mathématiques bien claires. Aussi, la succession des états est due au principe de causalité.

— **Espace des phases :** Cette terminologie a été introduite par Henri Poincaré, on peut l'appeler aussi espace des états. Il s'agit d'une représentation géométrique de l'état d'un système physique caractérisé par des équations différentielles. Chaque coordonnée de cet espace est une variable d'état du système considéré [10].

— **Imprédictibilité :** Dans son ouvrage "science et méthode", le fameux mathématicien Henri Poincaré avait évoqué le fait que le hasard et le déterminisme sont rendus compatibles par l'imprédictibilité à long terme, en exprimant par *"une cause très petite qui nous échappe, détermine un effet considérable que nous ne pouvons pas voir, et alors nous disons que cet effet est dû au hasard"*. Aussi, il rajouta qu'un tel phénomène est dû à la dépendance sensitive aux conditions initiales, ou si la valeur de départ est inconnue, la prochaine valeur de sortie dans la séquence devrait être imprévisible malgré la connaissance des valeurs aléatoires précédentes dans la séquence.

1.4 Applications du chaos

À l'instar de la physique et la cryptographie, la théorie du chaos s'est rapidement étendue à quasiment tous les domaines scientifiques :

La biologie : l'évolution de la population d'une espèce subite à une dynamique chaotique pour cause de plusieurs circonstances, comme la famine, l'épidémie, le climat ..., cette évolution a été modélisée par la fameuse carte logistique.

La médecine : le corps humain est très complexe car, en dépit de la considérable avancée en médecine, il est toujours ambigu, on entend toujours parler d'une apparition brusque des tumeurs dans le corps, l'épilepsie, l'Acouphène chez les jeunes gens et d'autres. Ces atteints sont parfois issus de plusieurs facteurs connus et inconnus, dont la capacité d'en prédire est presque impossible. La théorie du chaos considère le corps comme un univers dynamique, les chercheurs se servent de ses techniques pour comprendre mieux les disharmonies de corps, notamment les découvertes inédites concernant le cœur [11]. Une des irrégularités qui peuvent atteindre le cœur est la "fibrillation ventriculaire" où le cœur n'est jamais complètement contracté ou détendu. Après avoir fait de nombreuses expériences, les chercheurs ont suggéré que la fibrillation dans un corps humain peut être provoquée par l'apparition de foyers anormaux secondaires à l'intérieur du cœur sous forme d'impulsions, la collision entre cette dernière et le rythme cardiaque propre met le cœur dans un état chaotique, entrainant la fibrillation qui se considère comme un chaos stable. La décharge produite par un appareil de défibrillation peut ramener le cœur à son état stable, or que l'intensité, ainsi que la forme de ces charges sont encore indéfinies [11].

L'économie : est un système fluctuant, et ses fluctuations sont imprévisibles. Pour définir l'état économique futur, il faut s'intéresser par plusieurs variables de nature dynamique. C'est vrai qu'en économie, si le prix d'un produit augmente, le nombre d'achat diminue mais, ce n'est pas toujours le cas avec quelques produits comme l'or. Le prix de l'or soumis à des équations similaires aux celles qui décrivent l'augmentation d'une population donnée. Si le taux d'augmentation est très haut, son prix devient chaotique [11].

La psychologie : est aussi un domaine où la théorie du chaos a son mot à dire. Un petit changement dans les conditions initiales peut générer une grande diversité dans les résultats. Cette diversité se manifeste dans les décisions, les perspectives, les idées ou le comportement. En 1994, Thelen et Smith ont proposé une théorie de développement de la cognition et de la motrice, à base des systèmes dynamiques.

L'épidémiologie : pour prédire l'évolution d'une maladie, il faut une très bonne connaissance antérieure. Or, en ce qui concerne l'évolution d'une épidémie, la connaissance de la maladie et de sa prolifération est souvent très peu, ce qui pousse les épidémiologistes à recourir vers l'utilisation des modèles chaotiques pour déterminer le chemin.

1.5 Théorie de bifurcation

Le diagramme de bifurcation est un moyen utilisé pour obtenir un aperçu sur un changement de type topologique dans la trajectoire d'un système dynamique, lorsqu'un ou plusieurs paramètres sur lesquels elle dépend, varient [12]. La signification littérale de mot bifurcation se renvoie au fait qu'une zone se divise en deux branches, tandis que la signification conventionnelle, est un changement quelconque dans la forme qualitative de l'attracteur d'un système dynamique. Le terme bifurcation a été introduit pour la première fois par Henri Poincaré. On appelle une codimension d'une bifurcation, la plus petite dimension de l'espace des paramètres, qui permet d'aboutir de façon persistante à cette bifurcation [13].

1.6 Géométrie fractale

Le terme fractal est suggéré par le mathématicien Mandelbrot pour décrire des objets ayant une structure auto-similaire ou invariable par changement d'échelle, c-à-d à grande ou à petite échelle (dans des limites raisonnables), l'objet concerné présente la même structure malgré son irrégularité, ce qui se réfère au principe de chaos : l'ordre qui se cache derrière le désordre.

Un exemple de fractal est le fameux ensemble de Mandelbrot [12]. Sa géométrie est affichée sur la figure 1.2, et sa formule se définit comme :

$$\begin{cases} z_0 = 0 \\ z_{n+1} = z_n^2 + c \end{cases} \tag{1.3}$$

Avec z et c sont des nombres complexes. Si l'on effectue un zoom sur la figure, on se rend compte que le modèle porte copie de lui-même un peu partout.

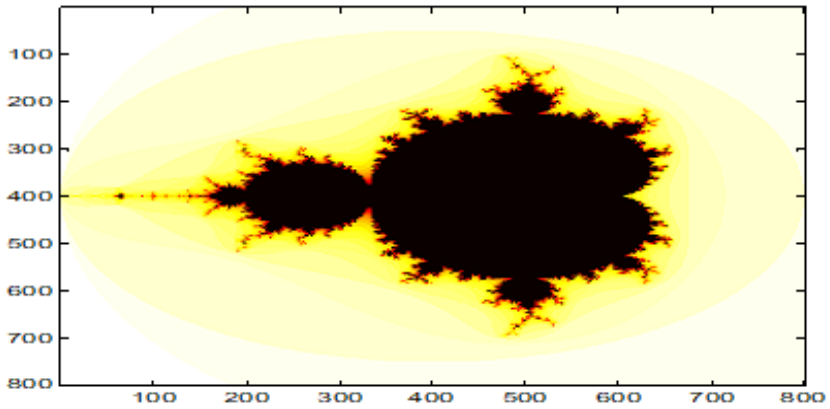


FIGURE 1.2 – Ensemble de Mandelbrot.

1.7 Systèmes dynamiques

Tout système dont l'état (les grandeurs nécessaires pour le caractériser) évolue en fonction du temps est un système dynamique [10], la connaissance de son état initial (t_0) ainsi que sa formule mathématique est requise pour pouvoir étudier son évolution. Un système dynamique à temps discret est donné par une formule mathématique itérative. Il peut être présenté sous la forme :

$$x_{n+1} = f\left(x_n, c\right),\ x_n \in \mathbb{R}^n\ ,\ c \in \mathbb{R}^p,\ n = 1, 2, \ldots \tag{1.4}$$

Concernant les systèmes dynamiques à temps continu, nous faisons recours aux équations différentielles :

$$\frac{dx}{dt} = \dot{x} = f\left(x, t, c\right),\ x \in \mathbb{R}^n\ ,\ c \in \mathbb{R}^p. \tag{1.5}$$

Où $\mathbb{R}^n$ et $\mathbb{R}^p$ sont respectivement l'espace des phases et l'espace des paramètres.

Exemple d'un système dynamique discret : le système de [14] :

$$F = 1 - \mu x^2, x \in [-1, 1], 0 \leq \mu \leq 2. \tag{1.6}$$

Le paramètre μ vérifie $0 \leq \mu \leq 2$, l'espace des phases est $\mathbb{R}^1$, et l'espace des paramètres est $\mathbb{R}^1$.

Exemple d'un système dynamique continu : l'oscillateur de Duffing [12] :

$$\begin{aligned} &dx/dt = y, \\ &dy/dt = x - x^3 - \delta y + \gamma \cos \omega t \end{aligned} \tag{1.7}$$

δ, γ et ω sont des paramètres réels. L'espace des phases est R^2 et l'espace des paramètres est R^3.

Un système dynamique peut être classé comme [10] :

Système dynamique autonome : la loi d'évolution de ce système dépend uniquement de son état actuel et pas du temps.

Système dynamique non autonome : Inversement au système dynamique autonome, l'évolution d'un système dynamique non autonome dépend du temps.

1.8 Systèmes chaotiques

Étymologiquement parlant, un système est dit chaotique si l'évolution présentée par ces grandeurs est en anarchie quasi-totale. Conventionnellement parlant, Le système chaotique est décrit par un ou plusieurs systèmes dynamiques non-linéaires, d'où sa nature déterministe. Pour qualifier

un système dynamique de chaotique, il faut qu'il soit imprévisible à long terme, pour cause de sa sensibilité aux conditions initiales. Géométriquement parlant (pour les systèmes à plus de deux dimensions), un système chaotique doit présenter une forme d'attracteur étrange dans l'espace des phases, c-à-d, un ensemble non errant, attractif, indécomposable et contenant des orbites denses [15].

1.9 Générateurs de nombres aléatoires basés sur les systèmes chaotiques

Un générateur de nombres aléatoires est un outil arithmétique ou physique qui génère une séquence de nombres désordonnés, il s'agit tout simplement de nombres aléatoires. En effet, il existe une forte relation entre les générateurs de nombres aléatoires et les systèmes chaotiques, étant donné que les systèmes chaotiques présentent un comportement irrégulier et imprévisible à long terme. Tout comme les générateurs de nombres aléatoires, les systèmes chaotiques sont aussi définis par des récusions. Certes, c'est difficile de définir des processus aléatoires efficaces basés sur des systèmes chaotiques, mais certains des résultats de la théorie du chaos peuvent être applicables pour la conception des générateurs pseudo-aléatoires [16]. C'est en 1994 qu'un générateur dit *subtract-with-borrow* et un système chaotique furent combinés, en vue d'interpréter les défauts corrélationnels du générateur, ce qui permet de les éviter ultérieurement [16].

1.10 Différents tests des systèmes chaotiques

Dans la littérature, il existe plusieurs tests de validation des systèmes chaotiques, certains sont statistiques, en utilisant une série de calculs sur la séquence générée par le système en question, d'autres sont graphiques, auquel cas, l'outil d'étude principal est l'œil. Nous résumons ci-après les principaux tests des cartes chaotiques.

1.10.1 Tests statistiques

En ce qui concerne ce type de tests, si les séquences générées par un tel système passent les tests statistiques, alors celui-ci peut être accepté comme un générateur de séquences aléatoires. À noter que nous allons garder les appellations par défaut pour certains tests.

1.10.1.1 Test de séquence de nombres pseudo-aléatoires (ENT)

Il s'agit d'un programme contenant un ensemble de six tests de séquences de nombres pseudo-aléatoires, sous la forme de flux d'octets dans un fichier. Les tests en question sont alors [17, 18] :

1.10.1.1.1 Entropie

L'entropie est la quantité d'information contenant dans un mot de 8 bits. Elle est exprimée par :

$$H(s) = \sum_{i=1}^{2N-1} P(s_i) \log \frac{1}{P(s_i)}. \tag{1.8}$$

s est l'échantillon étudié, H_s est la probabilité d'apparition de l'octet s_i.

1.10.1.1.2 Test de chi square

Ce test est très fréquent, il est utilisé afin de vérifier le caractère aléatoire des données. Son principe est d'effectuer une comparaison entre les variables aléatoires générées par un générateur aléatoire ou pseudo-aléatoire, avec les valeurs théoriques, car il est très sensible aux erreurs de ces générateurs. Chi square s'exprime à la fois par une valeur comme suit :

$$x^2 = -\sum_{i=0}^{2^n-1} \frac{N_i - Nt_i}{Nt_i}. \tag{1.9}$$

N_i est le nombre d'apparition de i de n bits, tandis que Nt_i représente son nombre d'apparition théorique.

En pourcentage :

$$P = 100 * \frac{1}{(\Gamma(v/2)2^{(v/2)})} \int_{x^2}^{\infty} x^{(v/2-1)} e^{(-x/2)} dx. \tag{1.10}$$

Où :

$$\Gamma(t) = \int_0^{\infty} x^{(t-1)} e^{(-x)} dx, t > 0. \tag{1.11}$$

v représente le nombre de mots possibles de la variable aléatoire moins de 1. Selon le pourcentage obtenu, on distingue plusieurs cas :

- Si $1\% < P < 99\%$: la série des données n'est presque pas aléatoire ;
- Si $95\% < P < 99\%$ ou $1\% < P < 5\%$: la séquence est incertaine ;
- Si $90\% < P < 95\%$ ou $5\% < P < 10\%$: la séquence est presque incertaine.

1.10.1.1.3 Moyenne arithmétique

La méthode Monté-Carlo est une méthode de calcul stochastique d'une valeur approchée à π. Son nom vient d'une ville française à Monaco, très connue par le jeux de hasard, le principe est de générer des points aléatoires de coordonnés (x, y) de 24 bit, dans un carré dont le coté égale à 2, à son intérieur (ils ont le même centre) un cercle de rayon 1. La fréquence d'apparition des points sur le quart du cercle divisé par la fréquence d'apparition des points sur le quart du carré donne :

$$\frac{Nin/N}{Nout/N} \approx \frac{\frac{1}{4} surface\ de\ cercle}{\frac{1}{4} surface\ de\ carré} = \frac{\pi}{4}. \tag{1.12}$$

Nin est le nombre de tirs à l'intérieur du cercle, $Nout$ est le nombre de tirage à l'extérieur du cercle, ainsi, à l'intérieur du carré, et N est le nombre total de tirage. Pi est donc le pourcentage de points à l'intérieur de cercle complet. On dit que notre système est chaotique si la valeur calculée tend vers Pi.

1.10.1.1.4 Coefficients de corrélation

Cette grandeur mesure la similarité entre les octets successifs. Son équation est la suivante :

$$corr = \sum_{i=0}^{N-2} \frac{(x_i - \mu)(x_{i+1} - \mu)}{\sigma^2}. \tag{1.13}$$

N est le nombre d'octets dans la séquence, x_i est l'octet numéro i, μ est la moyenne de toute la séquence d'octet, σ est la variance. Cette mesure devrait être proche à zéro.

1.10.1.2 Batterie de test DIEHARD

Dans le cadre de vérification de la qualité d'un générateur aléatoire donné, il y a le kit DIEHARD qui consiste en un ensemble de 12 tests statistiques développé par le professeur Goerge Marsgalia à l'université de Florida en 1995, et gravés sur un CD-ROM [19] . Ses versions sont toujours en amélioration, comme elles sont aussi disponibles sur internet. Les utilisations d'un tel test sont diverses et variées. Il s'emploie entre autres en cryptographie. Son implémentation est facile ; il requiert un fichier binaire d'environ 12 méga octets. Les valeurs de p (p-value) renvoyées par ces tests doivent être comprises dans l'intervalle [0.01,0.99] [20, 21, 22].

1.10.1.2.1 Test d'espacement des anniversaires

Ce test consiste à chercher le nombre d'intervalles égaux entre deux anniversaires, dont le nombre d'anniversaires est m dans une année de n jours, le nombre d'intervalles supérieur à un est

j, il est approché par la loi de poisson :

$$P(j) = e^{-\lambda}\frac{\lambda^j}{j!}. \tag{1.14}$$

$\lambda = \dfrac{m^3}{4n}$ est la moyenne, n doit être assez grande, vers 2^{24}, et le test du chi-square fournit la valeur p (p-value) pour chaque sous-séquence. Pour valider le résultat, les p-values résultantes vont être subies un autre test similaire au test de chi-square, c'est le test de Kolmogorov-Smirnov (KTEST), testant la distribution de cette dernière par rapport à un seuil donné.

1.10.1.2.2 Test de chevauchement des permutations (OPERM5)

OPERM5 (Overlapping 5-Permutation Test) consiste à tester une séquence d'un million d'entiers aléatoires de 32 bits, chaque cinq nombres successifs peuvent avoir 120 états (5 !), le nombre d'occurrence des transitions de chaque état sont calculés cumulativement, toutes les occurrences devaient avoir une probabilité statistiquement égale.

1.10.1.2.3 Test de rang binaire

Ce test est appliqué sur une matrice de 6×8 bits, 31×31 bits ou 32×32 bits, on calcule le rang de toutes les matrices formées, un test de Chi-square est effectué sur la fréquence des rangs obtenus, ce qui permet par la suite de comparer entre les valeurs calculées et les valeurs théoriques.

1.10.1.2.4 Test de flux binaire

La séquence binaire est composée de 20^{21} mots de 20 bits chevauchés, où le premier mot est b1b2 . . . b20, le deuxième mot est b2b3. . . b21, et ainsi de suite. L'intérêt de ce test est de compter le nombre de mots de 20 bits manquants dans un flux de 2^{21} bits. Ce nombre doit être normalement distribué.

1.10.1.2.5 Test d'occupation clairsemée de paires qui se chevauchent(Test Overlapping Pairs Sparse Occupancy (OPSO)) :

Le test OPSO s'articule sur l'analyse des mots de deux lettres. La lettre est constituée d'une séquence de dix bits, à partir d'un nombre entier de 32 bits dans la séquence à tester, et les bits qui définissent les lettres se chevauchent. Le test compte les combinaisons qui n'apparaissent pas dans la séquence entière testée. Le score a une distribution normale asymptotique, et c'est la base du test de qualité de l'ajustement, lorsque de nombreuses séquences sont testées [16].

1.10.1.2.6 Test d'occupation clairsemée de paires qui se chevauchent (OQSO)

Ce test est similaire au test OPSO précédent, sauf qu'il utilise des mots de quatre lettres, chaque lettre est de cinq bits, donc un alphabet de 2^5 lettres.

1.10.1.2.7 Test d'ADN

S'agissant de ce test, le mot utilisé est de 10 lettres de 2 bits, donc un alphabet de 2^2 lettres (selon les composants d'ADN : A, T, C et G).

1.10.1.2.8 Test de comptage des 1

Ce test consiste dans le calcul du nombre d'occurrence de 1's dans chaque octet, pour former par la suite une séquence de mots de cinq octets chevauchés. Conformément au nombre de bit 1 trouvé, les octets peuvent prendre cinq possibles lettres :A(0,2), B(3 bits), C(4 bit), D(5 bits), E(6 à 8 bits), on compte la fréquence de chaque mot, puis un test de chi deux est appliqué. Ce test peut être appliqué sur tous les octets réunis ou séparés.

1.10.1.2.9 Test des places de parking

Pour mieux expliquer le principe de ce test, nous admettons que des voitures rondes de rayon 1 essayent de garer correctement (sans chevauchement) dans un parking carré de longueur 100×100. 12000 essais sont admissibles, le nombre de tentatives qui ne sont pas planté devrait être normalement distribué avec ($\delta = 3523$,$\lambda = 21.9$), les valeurs résultantes sont converties en p-value moyennant KS test.

1.10.1.2.10 Test des distances minimales

On place 8000 points aléatoirement dans un carré de coté 10000, ensuite on calcule la distance d entre les $\frac{n^2 - n}{2}$ paires de points. Si d^2 est distribué exponentiellement avec une moyenne proche de 0,995, donc les points sont uniformes. Les résultats obtenus après cent répétitions de test, vont faire l'objet de KTEST.

1.10.1.2.11 Test des sphères aléatoires

Dans une cube de côté égale à 1000, et contenant 4000 points repérés par le biais de deux entiers de la séquence testée. À chaque point, une sphère se place de sorte que son rayon est la distance

entre ce point et le point le plus proche. On répète le processus 20 fois, tout en comptant le volume de la petite sphère qu'il doit être distribué exponentiellement, et KTEST leur est appliqué.

1.10.1.2.12 Test de compression

On multiplie les séquences aléatoires converties en valeurs dans l'intervalle]0,1] par 2^{31}, jusqu'à atteindre 1, j est un compteur de nombre d'obtention de la valeur 1. De tels j seront trouvés 100000. On répète le processus pour le nombre de fois où j appartient à $[6, 48]$. On calcule les valeurs de chi-square en vue de tirer p-value.

1.10.1.2.13 Test des sommes superposées

Son principe consiste à calculer une série des sommes, la somme de 100 consécutives valeurs de la séquence testées $(U(1), U(2), \dots)$ qui devraient être comprises entre 0 et 1. À chaque fois on décale la séquence par un seul élément, et on répète le calcul de sommation $(S(1) = U(1) + U(2) + \cdots + U(100)$,$S(2) = U(2) + U(3) + \cdots + U(101), \dots)$, et ainsi de suite. Ces S's devraient être distribués normalement.

1.10.1.2.14 Test de série (Runs test)

Ce test calcule le run-up et le run-down d'une séquence de longueur 10000, dont les valeurs appartiennent à l'intervalle [0,1[. Ce test est répété 10 fois et un test de chi-square est exécuté.

1.10.1.2.15 Test du craps

Ce test consiste en l'imitation du jeu de dé, où on compte le nombre de lancers réussis ainsi que le nombre de jets nécessaires pour finir une telle partie, sachant qu'il y a 200000 parties. Le nombre de lancers réussis devrait être très proche d'une distribution normale. Quant au nombre de lancers, il subira un test de chi-square. Diehard test a publié une version distillée avec uniquement trois tests GCD test, si un générateur de nombres aléatoires passe avec succès ces trois tests, il est probable qu'il passe tous les tests de Diehard.

1.10.1.3 Batterie de test Nist (National Institute of Standards and Technology)

C'est un ensemble de seize tests statistiques appliqués sur les séquences par un générateur aléatoire et pseudo-aléatoire, il s'agit d'un outil très important pour les qualifier d'aléatoire. Ce test est appliqué sur une séquence de nombres binaires, et il rend en retour une p-valeur qui teste l'hypothèse nulle H_0. Soit α une valeur de l'intervalle]0.001,0.01], si $p - valeur \geq \alpha$, la séquence

générée est aléatoire sinon, la séquence générée est non aléatoire. En cryptographie, elle est environ 0.01. Pour plus de détails, nous décrivons les différents tests dans ce qui suit [23].

1.10.1.3.1 Test de fréquence

L'hypothèse ici est que le nombre des 0 égale au nombre des 1, donc ce test étudie la proportion de "zéros" et de "uns" de toute la séquence testée. Pour ce faire on remplace 1 par $(+1)$ et 0 par (-1), et on les additionne pour avoir S_n, ensuite on calcule la valeur observée S_{obs} , $S_{obs} = \frac{S_n}{\sqrt{n}}$, pour trouver p-valeur on applique la fonction d'erreur complémentaire $erfc$ sur S_{obs}. Si p-valeur est supérieure à 0.01 la séquence est aléatoire. Il est recommandé que la séquence doive avoir au minimum 100 bits.

1.10.1.3.2 Test de fréquence par bloc

L'hypothèse est que la fréquence des uns dans un bloc de M bits $(M \neq 20)$ est d'environ $\frac{M}{2}$, ainsi, on calculera la proportion des uns dans des blocks de M bit. De ce fait, on partitionne la séquence en N blocs non chevauchés de même taille (M) avec $N \leq 100$, puis on calcule la proportion p_i des 1 dans chaque partition par :

$$p_i = \frac{\sum_{j=1}^{M} \varepsilon_{(i-1)M+j}}{M} \ \ pour\ 1 \leq i \leq N. \tag{1.15}$$

p_i vont être utilisées pour calculer la distribution chi χ^2 : $\chi^2 = 2M\sum_{i=1}^{N}(p_i - 1/2)^2$. Pour calculer p-value, on utilise la fonction de gamma incomplète $igamc$. La séquence testée doit avoir plus de 100 bits.

1.10.1.3.3 Test de série

Ici, série vaut une succession de bits identiques. On suppose que le nombre de séries de uns et de zéros dans la séquence (contenant plus de 100 bits) est égal au nombre de séries dans une séquence aléatoire. Ainsi, le principe de fonctionnement d'un tel test est donné comme suit : $V_n(obs) = \sum_{k=1}^{n-1} r(k) + 1$

$$\begin{cases} r(k) = 0\ si\ e_k = e_{k+1} \\ r(k) = 1 \qquad sinon \end{cases} \tag{1.16}$$

Ensuite, $P - value = erfc\left(\frac{|V_n(obs) - 2n\rho(1-\rho)|}{2\sqrt{2n}p(1-p)}\right)$, $\rho = \frac{\sum_k e_k}{n}$

1.10.1.3.4 Test de longue série de 1

L'objectif de ce test est de vérifier si la longueur de la plus longue suite des uns dans la séquence en question est similaire à la longueur de la plus longue suite des uns attendue dans une séquence aléatoire, dans lequel il examine la plus longue suite des uns dans des blocs de M bits. Comme le montre tableau 1.1, ce processus dépend beaucoup de la valeur de M.

TABLEAU 1.1 – Valeurs possibles de M en fonction de la taille de la série correspondante.

Minimum de n	M
128	8
6272	128
750,000	10^4

Aussi, l'organisation des fréquences par catégorie est donnée dans le tableau 1.2 suivant :

TABLEAU 1.2 – Valeurs de v_i conformément à la taille de M.

v_i	M=8	M=128	M=10^4
v_0	≥ 1	≥ 4	≥ 10
v_1	2	5	11
v_2	3	6	12
v_3	≤ 4	7	13
v_4		8	14
v_5			15
v_6			≤ 16

Après avoir divisé la séquence et déterminé les fréquences v_i, on calcule χ :

$$\chi^2 = \sum_{i=0}^{k} \frac{(v_i - N\rho_i)^2}{N\rho_i}. \tag{1.17}$$

Selon le tableau 1.3, k et N sont aussi conditionnées par la valeur de M.

TABLEAU 1.3 – Valeurs de K et N selon les valeurs de M.

M	K	N
8	3	16
128	5	49
10^4	6	75

La valeur p est donnée par : $P - value = igamc\left(\frac{k}{2}, \frac{\chi^2}{2}\right)$.

1.10.1.3.5 Test de rang

Ce test étudie la dépendance linéaire entre les sous séries de longueur fixe. Son principe est résumé comme suit :

— Former des matrices carrées $(M \times M)$ à partir de la division de la séquence en N blocs désunis :$N = \left\lfloor \frac{n}{M^2} \right\rfloor$.

— Calculer le rang de chacune de ces matrices.

— Effectuer le calcul de khi-deux (chi-square) comme l'illustre l'équation 1.18 :

$$\chi^2\,(obs) = \frac{(F_M - 0.2888N)^2}{0.2888N} + \frac{(F_{M-1} - 0.5776N)^2}{0.5776N} + \frac{(N - F_M - F_{M-1} - 0.1336N)^2}{0.1336N}. \tag{1.18}$$

Avec F_M et F_{M-1} sont le nombre des matrices de rang égale à M et $M-1$ respectivement.

— Calculer $p - valeur = e^{-\chi^2(obs)/2}$.

1.10.1.3.6 Test de la transformée de Fourier discrète

Comme son nom l'indique, ce test applique la transformée de Fourier discrète à la séquence en question. Ensuite, il cherche si le nombre de pics au-dessus du seuil de 95% est très largement différent de 5%. Son objectif est la détection des propriétés périodiques. Le procédé se résume dans les points suivants :

— Appliquer la transformée de Fourier discrète sur la séquence à tester $S = DFT(\psi)$, dont les 0 deviennent des -1, et les 1 restent 1.

— Calculer le module de la première moitié de S.

— Compter le nombre de pics inférieurs à $\sqrt{3n}$.

— Calculer $d = \frac{(N_1 - N_0)}{\sqrt{n(0.95)(0.05)/2}}$ avec N_0 est le nombre théorique attendu (95%) de pics, et N_1 est le nombre réel observé de pics, qui sont inférieurs à $\sqrt{3n}$.

— Calculer $p - valeur = erfc\left(\frac{|d|}{\sqrt{2}}\right)$.

1.10.1.3.7 Test de correspondance de modèles sans chevauchement (Non overlapping template matching test)

L'hypothèse dans ce test est que la séquence testée ne présente pas trop d'occurrence d'une fenêtre de m bits apériodique. Si cette fenêtre correspond à un mot dans la séquence, le processus reprend la recherche à partir du bit après, sinon on avance seulement d'un bit. Pour aller loin,

on compte le nombre d'apparition de la fenêtre dans la séquence. Ensuite, on calcule χ^2 par : $\chi^2 = \sum_{j=1}^{N} \frac{(W_j - \mu)^2}{\sigma^2}$, avec W_j est le nombre d'apparition de la fenêtre dans un bloc de M bits, $\mu = \frac{M - m + 1}{2^m}$ et $\sigma^2 = M\left(\frac{1}{2^m} - \frac{2m-1}{2^m}\right)$.
À la fin, calculer p-value en utilisant la fonction $igamc$: $p - valeur = igamc\left(\frac{N}{2}, \frac{\chi^2(obs)}{2}\right)$.

1.10.1.3.8 Test de correspondance de modèles chevauchés (Overlapping template matching test)

Il s'agit du même test que le précédent, la seule différence est que lorsque le modèle est trouvé, la fenêtre ne glisse que d'un seul bit avant de poursuivre la procédure de recherche.

1.10.1.3.9 Test statistique universel de Maurer

Pour que la séquence soit aléatoire, il faut qu'elle soit non compressible. D'où l'importance de ce test, dont la procédure consiste à diviser la séquence en seulement deux segments : segment d'initialisation Q×L bits et segment de test k×L bits, ensuite on calcule la fonction statistique $f_n = \frac{1}{k} \sum_{i=Q+1}^{Q+k} log_2(i - T_i)$. T_i est la représentation décimale du contenu de i^{eme} L bloc.

$$p - valeur = erfc(|(f_n - expectedValue(L))/(\sqrt{2}s)|), s = c\sqrt{((variance(L))/k)}. \quad (1.19)$$

Où $expectedValue$ sont des valeurs théoriques pré-calculées et considérées comme aléatoires.

1.10.1.3.10 Test de compression de Lempel-Ziv

Ce test a le même objectif que le précédent test, dont il focalise sur le nombre de mots disjoints et distincts dans la séquence à évaluer (w_{obs}), et il calcule la valeur p moyennant l'équation : $p - value = 1/2erfc\left(\frac{m - w_{obs}}{\sqrt{2s^2}}\right)$, pour $n = 10^6$ m=69586.25 et $s = \sqrt{70.448718}$.

1.10.1.3.11 Test de complexité linéaire

L'objectif de ce test est de décider si la séquence testée est suffisamment complexe, pour être considérés comme aléatoire. La méthode utilisée pour cela, est de mesurer la longueur d'un registre à décalage à rétroaction linéaire, qui porte un bloc de M bits de la séquence concernée, au moyen de l'algorithme de Berlekamp Massey. Le principe de ce test nécessite le partitionnement de la séquence en N blocs de M bits. Après, le calcul de p-value $p - value = igamc\left(\frac{k}{2}, \frac{\chi^2}{2}\right)$

1.10.1.3.12 Test série

Ce test détermine la fréquence d'occurrence de tous les blocs de m bits qui se chevauchent dans la séquence entière. Ensuite, il la compare avec celle anticipée d'une séquence parfaitement aléatoire. Dans le cas où la séquence est aléatoire, la chance d'apparence de tous les mots de m bits est la même. Ainsi, le principe relatif à ce test est donné comme suit :

— Étendre la séquence par l'ajout de $i^{(m-1)}$ bit à la fin de la séquence testée ;

— Déterminer la fréquence v_i de tous les blocs chevauchés ;

— Mésurer $\psi_m^2 = \frac{2^m}{n} \sum_{i_1 \ldots\ i_m} \left(v_{i_1 \ldots\ i_m} - \frac{n}{2^m}\right)^2$;

— Calculer $\nabla\psi_m^2 = \psi_m^2 - \psi_{m-1}^2$, $\nabla^2\psi_m^2 = \psi_m^2 - 2\psi_{m-1}^2 - \psi_{m-2}^2$;

— Compter $p - valeur1 = igamc\,(2^{m-2}, \nabla\psi_m^2)$, $p - valeur2 = igamc\,(2^{m-3}, \nabla^2\psi_m^2)$.

1.10.1.3.13 Test de l'entropie approximative

Ce test a pour objectif de calculer la fréquence de tous les blocs de m bits en chevauchement dans la séquence testée, et de comparer celles des blocs consécutifs avec la fréquence attendue d'une séquence considérée comme aléatoire. Il est décrit comme suit :

— Soit l'exemple de la séquence suivante : $\varepsilon = 001001101$. Pour $m = 3$, on obtient : $\varepsilon = 00100110100$, on aura par la suite les blocs suivants : 001, 010, 100, 001, 011, 110, 101, 010, 100 ;

— La fréquence v_i des blocs : $v_{000} = 0, v_{001} = 2, v_{010} = 2, v_{011} = 1, v_{100} = 2, v_{101} = 1, v_{110} = 1, v_{111} = 0$;

— Calculer $c_i^m = \dfrac{\#_i}{n}$, $c_{000} = 0, c_{001} = 0.22, c_{010} = 0.22, c_{011} = 0.11, c_{100} = 0.22, c_{101} = 0.11, c_{110} = 0.11, c_{111} = 0$;

— Calculer φ : $\varphi^m = \sum_{i=0}^{2^m-1} c_i^m \; log \; c_i^m$;

— $p - valeur = igamc\left(2^{m-1}, \frac{\chi^2}{2}\right)$.

1.10.1.3.14 Test de la somme cumulée

En ce qui concerne le test de la somme cumulée, il est tout simplement une somme cumulée des bits de la séquence testée (en remplaçant 0 par -1) ; il a pour objectif de comparer le résultat trouvé avec la somme cumulative d'une séquence aléatoire. Soit $X_i = 2\mu\varepsilon_i - 1$:

— Calculer $S_k = S_{(k-1)} + X_k$ pour mode avant, $S_k = S_{(k-1)} + X_{n-k+1}$ pour mode arrière ;

— Calculer z la plus grande valeur absolue des sommes cumulatives partielles S_k : $z = max_{1 \leq k \leq n} |s_k|$;

— Calculer p-value :

$$1-\sum_{k=\left(-\frac{n}{z}+1\right)/4}^{\left(\frac{n}{z}-1\right)/4}\left[\phi\left(\frac{(4k+1)z}{\sqrt{n}}\right)-\phi\left(\frac{(4k-1)z}{\sqrt{n}}\right)\right]+\sum_{k=\left(-\frac{n}{z}-3\right)/4}^{\left(\frac{n}{z}-1\right)/4}\left[\phi\left(\frac{(4k+3)z}{\sqrt{n}}\right)-\phi\left(\frac{(4k+1)z}{\sqrt{n}}\right)\right].$$

(1.20)

1.10.1.3.15 Test d'excursions aléatoires

Le test d'excursions aléatoires vise à vérifier si le nombre de visites d'un état particulier x des sommes cumulées dans un cycle d'une marche aléatoire (excursion) est différent de ce qui est attendu. Un cycle commence par 0 et finit par 0, pour chacun, x pourra avoir 8 états possibles :$-1,-2,-3,-4,1,2,3,4$. Ce test requiert le calcul du nombre total de cycles $n_k(x)$ dans lesquels l'état x apparaît exactement k fois parmi tous les cycles (k=0,1,... 5). Le reste de ce test se déroule comme suit :

— Calculer le test statistique χ^2 :$\chi^2=\sum_{k=0}^{5}\frac{(n_k(x)-Jp_k(x))^2}{Jp_k(x)}$, $p_k(x)$ est la probabilité d'occurrence de x, k fois dans une distribution aléatoire. $J=\sum_{k=0}^{5}n_k\,(x)$;

— Calculer p-value : $p-value=igamc\left(\frac{5}{2},\frac{\chi^2}{2}\right)$.

1.10.1.3.16 Variante du test des excursions aléatoires

L'objectif et le principe de ce test sont les mêmes que ceux du test précédent, sauf que ce dernier a une série de 18 tests, et donc 18 états :$-9,-8,\cdots-1,9,8,\ldots,1$.

1.10.1.4 Batterie TestU01

Le package TestU01 est une large bibliothèque utilisée non seulement pour tester les générateurs de nombres aléatoires, mais aussi il dispose de divers types de générateurs. La bibliothèque TestU01 est flexible, efficace et elle implémente une plus grande variété de tests. Les tests peuvent être appliqués sur les chaînes de bits ainsi que sur les séquences de nombres réels dans l'intervalle (0, 1). Pour juger que la séquence est aléatoire, il faut que toutes les p-value soient à l'intérieur de l'intervalle $[10^{-10},1-10^{10}]$.

1.10.1.5 Practrand test

Practrand test (PracticallyRandom) est une bibliothèque des générateurs aléatoires et pseudo-aléatoires, comme il fournit des tests statistiques pour les RNG. Il permet de tester des séquences de longueurs illimitées, même si le nombre de séquences par défaut est de 32 TB. Ce test est plus développé que son homologue TestU01 [24, 25].

1.10.2 Tests graphiques

Les tests graphiques offrent la possibilité de visualiser le comportement du système en pleine évolution, en fonction, entre autres, des valeurs de l'état initial ou des paramètres de contrôle, ce qui permet de déterminer les valeurs pour lesquelles le système est chaotique.

1.10.2.1 Diagramme de Lyapunov

L'exposant de Lyapunov (LE) est l'une des importantes techniques d'évaluation du chaos, il assure une description qualitative et quantitative du comportement dynamique. Il peut être exprimé comme suit :

$$LE = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N} ln \left| f'\left(x_n\right)\right| . \tag{1.21}$$

$f(x_n)$ est la fonction du système en question, N est le nombre d'itération. D'où, LE quantifie le taux moyen de divergence exponentielle entre deux trajectoires partant de points initiaux très proches. En effet, L'exposant de Lyapunov peut être :

- Positif : il traduit la divergence des points initiaux très proches sur un nombre fini d'itérations. Par conséquent, le système est chaotique ;
- Négatif : ça signifie la convergence de ses trajectoires après un nombre fini d'itération, donc il s'agit d'un système stable ;
- Nul : il n'y a ni convergence ni divergence exponentielle de ses orbites dans l'espace des phases, la dynamique du système est stable.

Un système avec un ou plusieurs exposants de Lyapunov positifs est considéré comme chaotique. Plus LE est grand, plus la rapidité de divergence augmente, donc un système avec un comportement chaotique plus important.

1.10.2.2 Diagramme de bifurcation

La notion de bifurcation est un aspect indispensable dans l'étude de l'évolution des systèmes dynamiques. En principe, Le diagramme de bifurcation est une représentation qualitative de la transitions du comportement des systèmes dynamiques à long terme, en fonction du paramètre de contrôle [26]. Il révèle les zones de convergence, de bifurcation et du chaos en changeant le paramètre de contrôle.

Ce graphique ayant la forme d'un nuage de points donnant les "multiples" valeurs possibles du système, si on change le paramètre de contrôle.

1.10.2.3 Test de 0-1 (0-1 test)

Il s'agit d'un nouveau test révolutionnaire développé en 2009, il permet de distinguer la dynamique régulière de la dynamique chaotique dans les systèmes dynamiques. Parmi les avantages de ce test, et pour ne citer que cela, la simplicité de son implémentation numérique ainsi que la possibilité de son application directe sur la sortie du système, indépendamment des caractéristiques du système lui-même [27]. La mise en œuvre d'un tel test se résume dans les points suivants [28] :

— Calcul des variables de déplacement p_c et q_c :

$$p_c(n) = \sum_{(j=1)}^{n} \phi_j cos(jc),. \tag{1.22}$$

$$q_c(n) = \sum_{(j=1)}^{n} \phi_j sin(jc),. \tag{1.23}$$

n est le nombre de ronds, phi_j est la séquences de données avec $j = 1 \dots n.$ et la valeur réelle, c est constante $(c \in (0, \pi))$

— Calculer $M_c(n)$ (le déplacement quadratique moyen) :

$$M_c\,(n) = \lim_{N\to\infty} \frac{1}{N} \sum_{j=1}^{n} \left[p_c\,(j+n) - p_c(j)\right]^2 + \left[q_c\,(j+n) - q_c\,(j)\right]^2. \tag{1.24}$$

— Calculer le taux de croissance asymptotique k_c :

$$k_c = \frac{logM_c\,(n)}{\log n}. \tag{1.25}$$

Si on trouve $k_c \approx 1$, le système est chaotique, sinon $(k_c \approx 0)$ le système n'est pas chaotique.

1.10.2.4 Analyse de trajectoire

La représentation de trajectoire des systèmes dynamiques permet efficacement d'extraire pas mal d'informations sur ses caractéristiques, comme la région de convergence, de divergence et de périodicité, la précision et la sensibilité aux conditions initiales et aux paramètres de contrôle, et la visualisation de leurs comportement sous certaines conditions et dans certaines circonstances. Le diagramme en toile d'araignée est un modèle récursif pour l'étude visuelle des comportements des systèmes. Dans l'univers des systèmes dynamiques, les états passés et présents déterminent les états futurs ainsi, ce modèle expose cette liaison d'une manière explicite et simple. Au début, il a été beaucoup utilisé en économie, vu le besoin insistant de projections économiques à long terme et de prévision des effets probables des programmes et des politiques économiques [29].

1.10.2.5 Test de la complexité linéaire

La complexité est considérée comme un critère fondamental pour les PRNG. Cependant, un PRNG est dit complexe lorsqu'il requiert de nombreuses quantités d'informations pour caractériser son comportement, c-à-d, génère un comportement aléatoire avec peu, voire aucun motif mesurable dans sa séquence [30]. La complexité linéaire est l'une des méthodes utilisée pour étudier leur complexité linéaire qui consiste à compter les 0 et les 1 dans chaque séquence de nombres aléatoires générée. La complexité linéaire attendue d'une séquence de n variables aléatoires binaires indépendantes et uniformément distribuées est très proche de $N/2$. Pour tracer le graphique :

— Générer une séquences de nombres aléatoires de taille N pour tous les paramètres de contrôle de chaque PRNG ;

— Calculer la distribution binaire moyenne pour chaque N ;

— Tracer le nombre total de 1 (ou de 0) en fonction de paramètre de contrôle.

1.10.3 Exemples de cartes chaotiques

Dans cette partie du chapitre, nous donnons un aperçu sur les différents systèmes chaotiques, classiques et récents, unidimensionnels et multidimensionnels. Nous utilisons aussi quelques tests graphiques pour les représenter.

1.10.3.1 Cartes chaotiques classiques unidimensionnelles

Les cartes chaotiques classiques de dimension 1 sont des modèles très utilisés et plus simples que l'on peut imaginer. Elles ne montrent pas le caractère aléatoire que pour certaines valeurs du paramètre de contrôle. Elles sont généralement issues de la nature.

1.10.3.1.1 Carte Logistique

La carte chaotique logistique est une équation de différence polynomiale de degré 2, représentée par l'équation [31] :

$$X_{t+1} = aX_t \left(1 - X_t\right). \tag{1.26}$$

Vu sa simplicité ainsi que ses hautes performances chaotiques, elle a été exploitée dans de nombreuses applications cryptographiques comme PRNG, sans amélioration [32], et avec amélioration [33]. Cette équation de différence non-linéaire décrit la croissance purement exponentielle d'une population, certaines conditions initiales et valeurs de paramètres (par exemple, si $a > 4$) mènent à des croissances de population négatives pour autant. La Figure 1.3 représente son LE, et c'est à

partir de laquelle on peut savoir les valeurs de a faisant de la carte logistique classique, une carte chaotique.

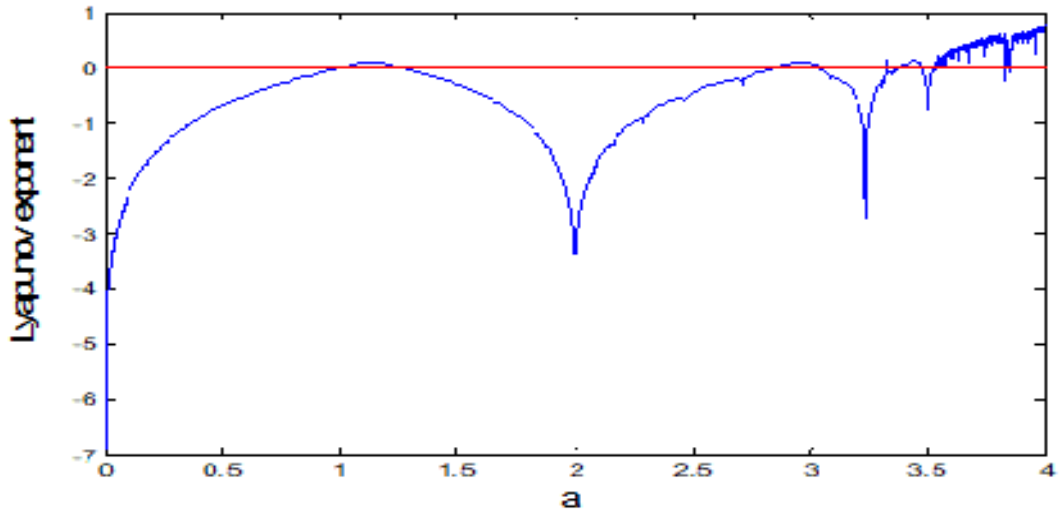


FIGURE 1.3 – Diagramme de Lyapunov de la carte logistique.

1.10.3.1.2 Carte Tent (Tent map)

La carte Tent est une fonction composée de deux parties continues et linéaires, elle est définie par :

$$x_{n+1} = \begin{cases} ax_n, & x \in \left[0,\ \frac{1}{2}\right] \\ a\left(1 - x_n\right), & x \in \left]\frac{1}{2}, 1\right] \end{cases} \tag{1.27}$$

a est le paramètre de contrôle, il varie entre 0 et 2. Pour $a = 2$, la carte devient chaotique [34], son diagramme de bifurcation est illustré dans la figure 1.4. La fonction Tent, que ce soit dans sa forme classique ou améliorée, a été le centre d'intérêt de plusieurs travaux de recherche [35, 36].

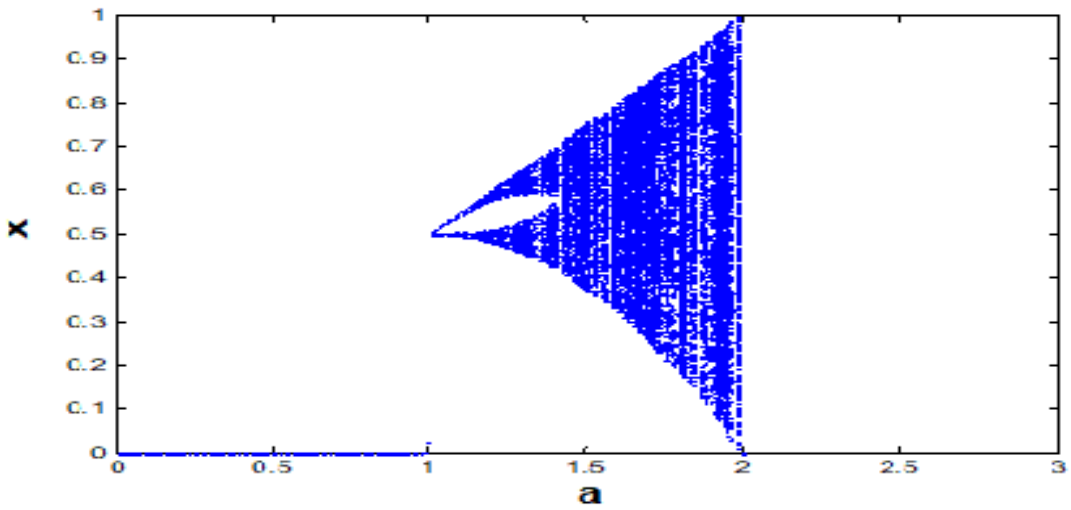


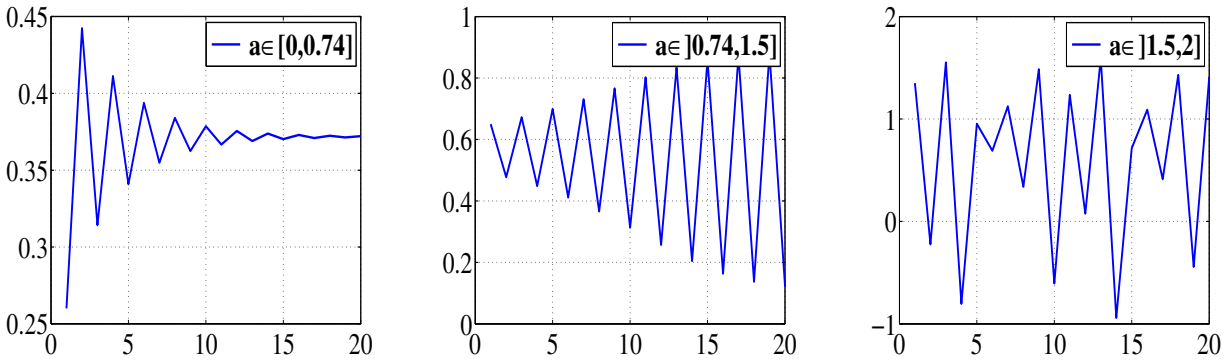
FIGURE 1.4 – Diagramme de bifurcation de la carte Tent.

1.10.3.1.3 Carte quadratique

Un autre modèle mathématique des cartes chaotiques est la fonction quadratique, il s'exprime comme suit :

$$x_{n+1} = a - x_n^2. \tag{1.28}$$

a est le paramètre de contrôle appartenant à l'intervalle $[0, 2]$, et les séquences x_n sont dans l'intervalle $[0, 1]$. Pour étudier les performances de cette carte, on trace le diagramme d'itération (Figure 1.5) pour trois régions de a.



(a) $a \in [0, 0.74]$. (b) $a \in]0.74, 1.5]$. (c) $a \in]1.5, 2]$

FIGURE 1.5 – Comportement de la carte quadratique pour les 20 premières itérations.

— Pour le cas où $a \in [0, 0.74]$: la fonction tend à se stabiliser, région de convergence ;

— Pour le cas où $a \in [0.74, 1.5]$: la fonction manifeste un comportement périodique ;

— Pour le cas où $a \in [1.5, 2]$: la fonction manifeste un comportement chaotique.

Nous pouvons donc conclure que la carte quadratique manifeste un comportement chaotique lorsque $a \in [1.5, 2]$.

1.10.3.2 Cartes chaotiques classiques multidimensionnelles

Il existe de nombreuses cartes chaotiques multidimensionnelles, la première était celle de Lorenz qui simulait un phénomène météorologique. L'utilisation des fonctions multidimensionnelles se justifie par le fait que la représentation d'un phénomène nécessite plus d'une variable.

1.10.3.2.1 Carte de boulanger (Baker map)

C'est un générateur des nombres pseudo aléatoires de deux dimensions, il peut être défini comme suit [37] :

$$B\,(x, y) = \left(2x, \frac{y}{2}\right) quand\ 0 \leq x < 1/2. \tag{1.29}$$

$$B\left(x,y\right)=\left(2x-1,\frac{y}{2}+1/2\right)\ quand\ 1/2\leq x\leq 1. \tag{1.30}$$

Son nom vient de l'opération du pétrissage. En fait, le boulanger coupe la pâte en deux, et les deux moitiés sont mises l'une sur l'autre, et compressée, et ainsi de suite.

1.10.3.2.2 Carte de chat 3D (Cat 3D map)

La carte cat par sa version bidimensionnelle, introduite par Arnold et Avez [38], est définie comme suit :

$$\begin{bmatrix}x_{n+1}\\y_{n+1}\end{bmatrix}=\begin{bmatrix}1 & a\\ b & ab+1\end{bmatrix}\times\begin{bmatrix}x_n\\y_n\end{bmatrix}\ mod\ 1. \tag{1.31}$$

Alors que sa version tridimensionnelle n'est qu'une extension de la version originale, elle se définie par [39] :

$$\begin{bmatrix}x_{n+1}\\y_{n+1}\\z_{n+1}\end{bmatrix}=A\begin{bmatrix}x_n\\y_n\\z_n\end{bmatrix}\ mod\ 1. \tag{1.32}$$

$$A=\begin{bmatrix}1+a_xa_zb_y\\ b_z+a_xb_y+a_xa_zb_yb_z\\ a_xb_xb_y+b_y\end{bmatrix}. \tag{1.33}$$

$$A=\begin{bmatrix}1+a_xa_zb_y & a_z & a_y+a_xa_y+a_xa_za_yb_y\\ b_z+a_xb_y+a_xa_zb_yb_z & 1+a_zb_z & a_ya_z+a_xa_za_yb_yb_z+a_xa_zb_y+a_xa_yb_y+a_x\\ a_xb_xb_y+b_y & b_x & a_xa_yb_xb_y+a_xb_x+1\end{bmatrix}. \tag{1.34}$$

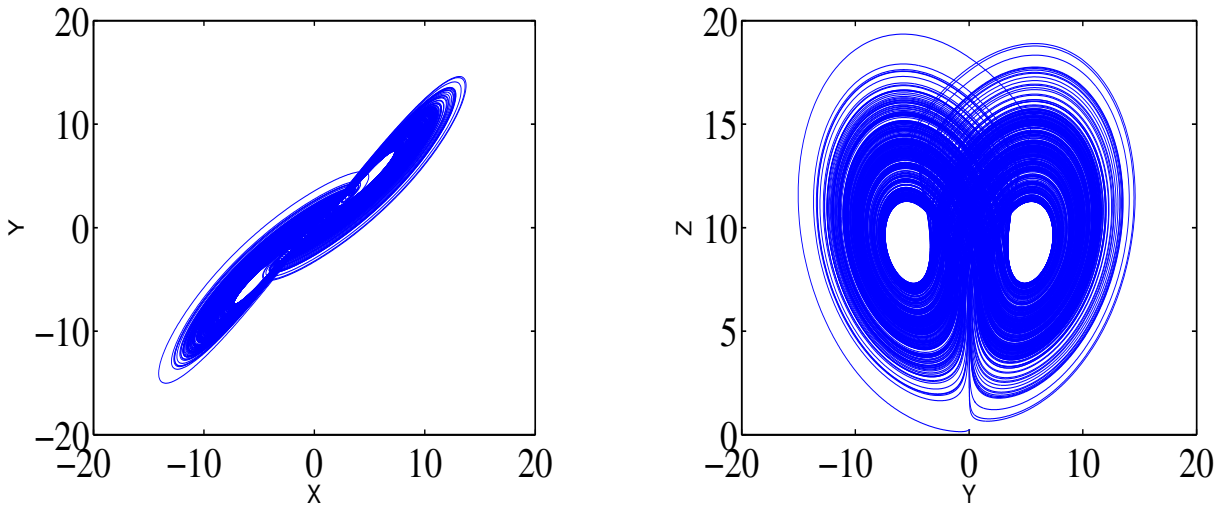
a_x, a_y, a_z, et b_x, b_y, b_z sont des entièrs positifs.

1.10.3.2.3 Système chaotique de Chen

Ce présent système proposé par Chen est un système tridimensionnel, il s'exprime comme suit [40] :

$$\begin{cases}\dot{x}=a\left(y-x\right)\\ \dot{y}=\left(c-a\right)x-xz+cy\\ \dot{z}=xy-bz\end{cases} \tag{1.35}$$

Pour $a=35$, $b=3$, et $c\in[20,28.4]$, le système est chaotique [40]. La figure 1.6 représente l'évolution de ce système dans le diagramme des phases, où l'on peut remarquer nettement les deux graphiques vérifiant les conditions d'un attracteur étrange.



(a) (b)

FIGURE 1.6 – Attracteurs hyper-chaotique de système de Chen. (a) sur le plane $(x-y)$, (b) et le plane $(y-z)$.

1.10.3.2.4 Carte Hénon

Les équations correspondantes au système chaotique 2D de la carte Hénon sont données ci-dessous [41] :

$$x_{n+1} = 1 - ax_n^2 + by_n \tag{1.36}$$

$$y_{n+1} = x_n \tag{1.37}$$

Un tel système chaotique ayant comme objectif de réduire la dimension de l'attracteur de Lorenz en deux dimensions. Il représente l'un des systèmes dynamiques les plus étudiés. Pour $a = 1.4$ et $b = 0.3$, le système diverge tout dépend des valeurs initiales (x_0, y_0).

1.10.3.3 Cartes chaotiques modernes

Un certain nombre de cartes chaotiques traditionnelles telles que la carte Pincher et la carte Zaslavskii ont des propriétés limitées et ne peuvent plus répondre à nos besoins. Sans leur amélioration, nos applications cryptographiques resteront inchangées et pourraient être soumises à de différentes attaques à l'avenir, Ce qui fait de l'amélioration de ces cartes chaotiques une nécessité absolue. Dans ce qui suit, nous présentons quelques nouveaux systèmes chaotiques.

1.10.3.3.1 Carte Tent améliorée

Dans le but d'améliorer la sécurité d'un système de cryptage et afin d'en améliorer les performances chaotiques de la carte Tent, dans [42], Congxu Zhunous a proposé un nouveau système

chaotique qui consiste dans la combinaison des deux cartes, logistique et Tent. Le système est exprimé comme suit :

$$\begin{cases} \mathrm{x_{i+1}} = \mathrm{f_1}\left(\mathrm{x_i}\right) = (4-\mu)\,\mathrm{x_i}\left(1-\mathrm{x_i}\right) + \frac{\mu}{2}\mathrm{x_i},\ if\ \mathrm{x_i} < 0.5, \\ \mathrm{x_{i+1}} = \mathrm{f_2}\left(\mathrm{x_i}\right) = (4-\mu)\,\mathrm{x_i}\left(1-\mathrm{x_i}\right) + \frac{\mu}{2}(1-\mathrm{x_i})\ ,\ if\ \mathrm{x_i} \geq 0.5. \end{cases} \tag{1.38}$$

Ce système présente un comportement chaotique pour $\mu \in [0, 4]$. Il est appelé "Logistique-Tent Map" (LTM). La Figure 1.7 illustre le diagramme de la toile d'araignée (Cobweb) de Tent map et de LTM pour $\mu = 1.2$. Les trajectoires présentées par Tent map convergent vers un point stable, à la différence de LTM qui représente un comportement aléatoire, où les carrés sont distribués un peu partout dans le plan.

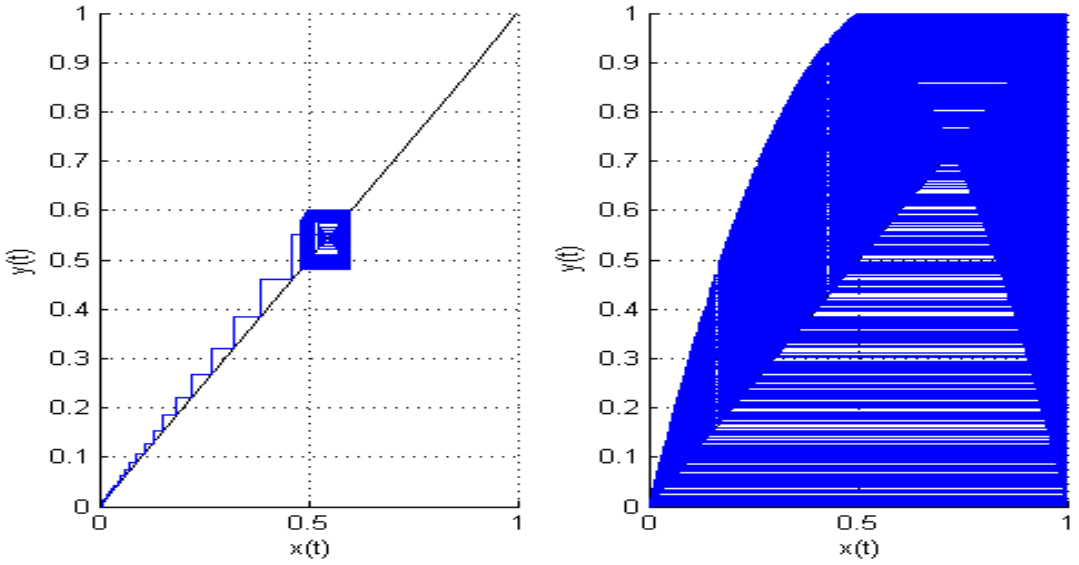


FIGURE 1.7 – Diagramme de la toile d'araignée de la carte Tent à gauche, et de LTM à droite.

1.10.3.3.2 Carte Logistique améliorée

Pour remédier aux lacunes des anciennes cartes chaotiques, de nombreux travaux de recherche ont suggéré des nouvelles fonctions chaotiques. Entre autres, la carte chaotique discrète unidimensionnelle, dont l'expression [43] :

$$x_{n+1} = \sin\left(\pi\left(\left(rx_n\left(1-\tan\left(x_n\right)\right)\right) mod\left(1\right) + \sin\left(\pi x_n\right)\right)\right). \tag{1.39}$$

Avec $x_n \in [-1, +1]$. Cette carte est basée sur la carte logistique, néanmoins la dynamique est plus chaotique. Dans le tableau 1.4 suivant, nous donnons les résultats de quelques tests réalisés sur la carte proposée et la carte logistique. Nous constatons une nette amélioration des performances de

la carte proposée.

TABLEAU 1.4 – Comparaison des performances : carte logistique classique vs carte logistique améliorée.

La carte chaotique	L'intervalle de x_n	Gamme chaotique	Le plus grand LE
La carte logistique	[0, 1]	3.57 à 4	0.6923
La carte proposée	[-1, 1]	1 à 4	3.5971

1.10.3.3.3 Carte chaotique numérique basée sur la fonction cosinus

Une autre variante de la carte logistique améliorée a été proposé dans [30]. Celle-ci repose sur l'emploi des propriétés de la fonction cosinus, pour illustrer le comportement chaotique. En effet, la combinaison du cosinus et de la carte logistique conduit à la création d'une autre carte chaotique améliorée dite carte cosinus-logistique CLM (Cosinus-logistic-map). Elle a pour expression :

$$x_{n+1} = \cos\left(2^{k+(r\times x_n\times(1-x_n))}\right). \tag{1.40}$$

Avec $k \in [10, 24]$ et $r \in [0, 4]$ sont les paramètres de contrôle. Les résultats de simulation confirment bel et bien la haute performance de CLM en termes de complexité chaotique par rapport à la carte logistique. L'un des tests appliqués pour étudier la complexité linéaire des fonctions CLM et la carte logistique est représenté par la figure 1.8. Le nombre total des 0 et 1 qu'on vient de représenter vaut 16000 bits, donc la valeur idéale des 0 ou 1 doit être à la limite de 8000. Comme on peut le vérifier, CLM a quasiment le même nombre des 0 et des 1, contrairement à la carte logistique. Ainsi, la nouvelle carte offre bien la possibilité d'améliorer la complexité linéaire par rapport à la carte de base.

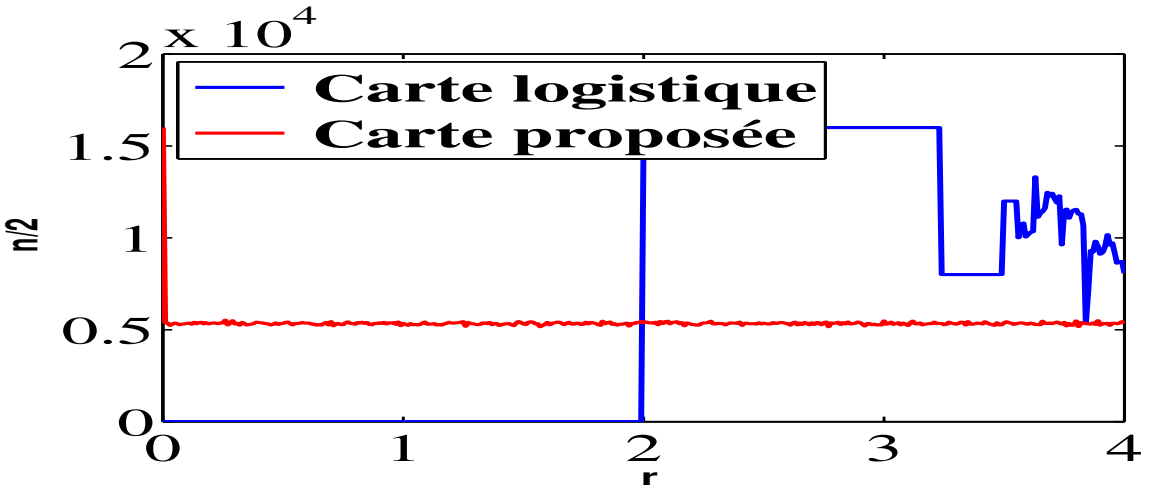


FIGURE 1.8 – Complexité linéaire de PRNG : carte CLM vs carte logistique.

1.10.3.3.4 Carte chaotique à polynôme cosinus unidimensionnelle (1-DCP)

Dans [44], les auteurs ont développé une carte chaotique unidimensionnelle en raison de l'employer dans un système de cryptage d'image. Elle est exprimé par l'équation suivante :

$$x_{n+1} = \cos\left(2^{k+(r\times x_n\times(1-x_n))}\right). \tag{1.41}$$

La 1-DCP manifeste un comportement chaotique pour toute valeur réelle positive μ (paramètre de contrôle). En outre, elle est très sensible aux minimes changements effectués sur les conditions initiales et le paramètre de contrôle. La Figure 1.9, quant à elle, illustre bien la sensibilité de la nouvelle fonction aux changements effectués sur les conditions initiales et le paramètre de contrôle. Elle est estimée par 10^{-16} et 10^{-12} respectivement. Pour les deux courbes a et b, nous remarquons que les trajectoires divergent après seulement quelques itérations, ce qui confirme la haute sensibilité du système.

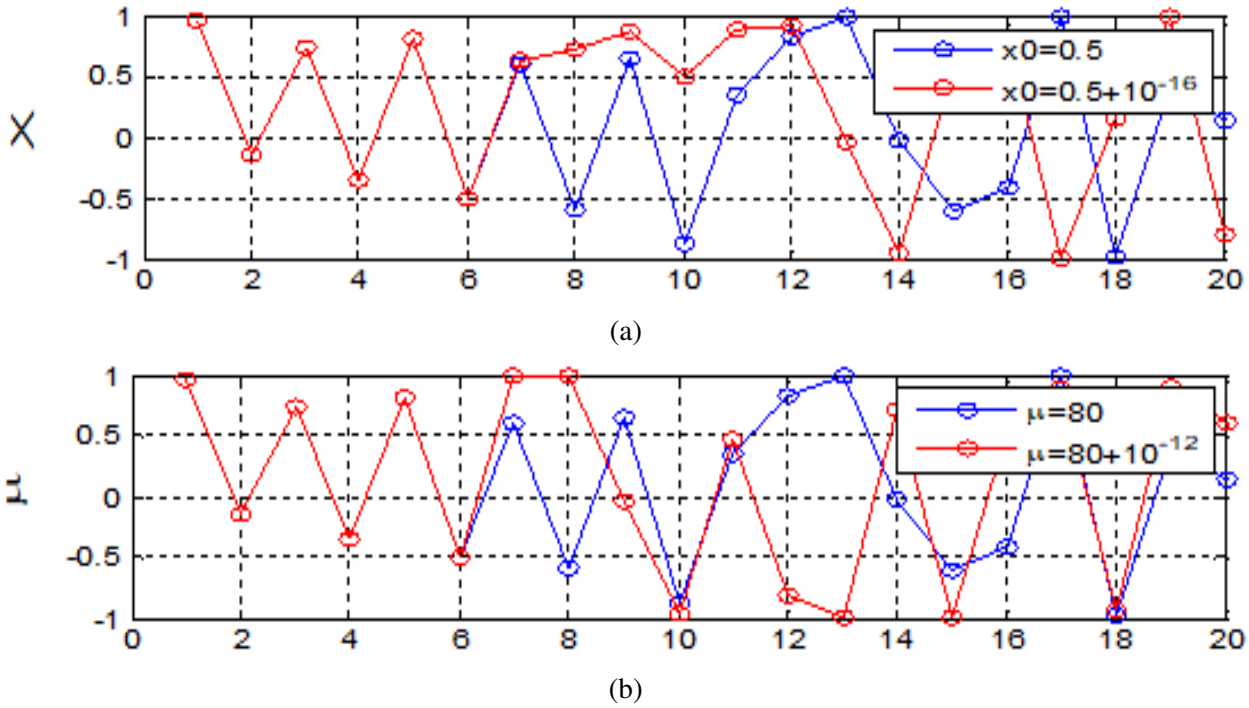


FIGURE 1.9 – Sensibilité de 1-DCP : (a) Valeur initiale x_0 ; (b) Valeur du paramètre μ.

1.10.3.3.5 Carte de Piecewise-Hénon

Il s'agit d'un système tridimensionnel issu de la fonction chaotique linéaire par moceaux (piecewise), et de la formule de la composante y de la carte Hénon. Ce système est défini sur [0,1]

comme suit [41] :

$$T\left(x_n,\ y_n,\ z_n\right) = \begin{cases} x_{n+1} = \psi_{c1}\left(x_n\right) + \Lambda_{c2}\left(y_n, z_n\right) mod 1 \\ y_{n+1} = \psi_{c1}\left(y_n\right) + \Lambda_{c2}\left(z_n, x_n\right) mod 1 \\ z_{n+1} = \psi_{c1}\left(z_n\right) + \Lambda_{c2}\left(x_n, y_n\right) mod 1 \end{cases} \tag{1.42}$$

Avec

$$\psi_{c1}\left(x\right) = \left|1 - c_1 x\right|, \tag{1.43}$$

$$\Lambda_{c2}\left(x, y\right) = y + 1 - c_2 x^2. \tag{1.44}$$

Où c_1, et c_2 sont des paramètres de contrôle de type réel. Elle est noté $3D - PHM$ pour Tridimensionnel Piecewise-Henon-Map. Elle est caractérisée par de bonnes performances, en termes de complexité et de sensibilité quant aux conditions initiales et au paramètre de contrôle.

1.10.3.3.6 Nouveau Système chaotique 5D

Dans [45], Hayder Najm propose un nouveau système hyper-chaotique pour des fins cryptographiques. Le système proposé est exprimé comme suit :

$$\begin{cases} x_{i+1} = -rx_i + by_i k_i - 2.5sp_i \\ y_{i+1} = -qy_i - sx_i z_i + rx_i - up_i \\ z_{i+1} = 2z_i x_i y_i - 1.1rp_i - qk_i \\ k_{i+1} = rx_i + sy_i - uk_i \\ p_{i+1} = b\left(\dfrac{x_i + k_i}{z_i}\right) + ry_i \end{cases} \tag{1.45}$$

Pour que le système ait un comportement chaotique, il faut que les valeurs des paramètres de contrôle et celles de l'état initial soient respectivement : $b = 0.001$, $r = 0.7$, $s = 0.5$, $u = 1.9$, et $q = 0.2$. et $x = 2.1$, $y_0 = 0.5$, $z_0 = 1.1$, $k_0 = 1$, $p_0 = 0.1$. Pour ce système, les exposants de Lyapunov sont aussi positifs selon l'auteur.

1.10.3.3.7 Système hyper-chaotique 6D

Dans [46], un système hyper chaotique a été proposé, il est décrit par l'équation 1.46 :

$$\begin{cases} \dot{x_1} = d_1(1 - \beta\,|x_6|\,x_2 - ax_1 \\ \dot{x_2} = cx_1 + dx_2 - x_1x_3 + x_5 \\ \dot{x_3} = -bx_3 + x_1^2 \\ \dot{x_4} = ex_2 + fx_4 \\ \dot{x_5} = -rx_1 - kx_5 \\ \dot{x_6} = -x_2 \end{cases} \tag{1.46}$$

$x_1, x_2, x_3, x_4, x_5, x_6$ sont les variables d'état, $a, b, c, d, e, f, d_1, r, k$ sont ses paramètres. Ce système est issu de la combinaison d'un memristance (memristor) à contrôle de flux, qui consiste en une relation entre les bornes de tension et le courant d'entrée (équation 1.47), et un système 5D relevant de la littérature [47] (équation 1.48) :

$$\begin{cases} i = \omega\,(\varphi)\,v \\ \dot{\varphi} = v \end{cases} \tag{1.47}$$

$$\begin{cases} \dot{x} = a\,(y - x) \\ \dot{y} = cx + dy - xz + w \\ \dot{z} = -bz + x^2 \\ \dot{u} = ey + fu \\ \dot{w} = -rx_1 - kx_5 \end{cases} \tag{1.48}$$

1.11 Conclusion

Dans ce chapitre, nous avons introduit la théorie du chaos avec également les différents concepts qui lui sont relatifs. Aussi, nous avons passé en revue les différents tests numériques et graphiques dédiés aux systèmes chaotiques relevant de la littérature. Dans la dernière partie du présent chapitre, nous avons présenté un ensemble de cartes chaotiques tout en mettant en relief leurs caractéristiques. Par ailleurs, ce premier chapitre servira comme étant une base pour la proposition de nos propres systèmes chaotiques. Dans le prochain chapitre, nous allons nous intéresser à la cryptographie et plus particulièrement au cryptage d'image.

Chapitre 2

Généralités sur le Cryptage d'Images

Sommaire

2.1 Introduction
2.2 Historique
2.3 Terminologie
2.4 Méthodes crypto-analytiques
2.5 Objectifs de la cryptographie
2.6 Classification des systèmes de cryptage
2.7 Principes de Kirchhoff
2.8 Théorème de Shannon
2.9 Cryptographie visuelle
2.10 Cryptographie quantique
2.11 Multimédia et cryptographie
2.12 Algorithmes cryptographiques
2.13 Chaos dans le cryptage
2.14 Cryptage d'image
2.15 Conclusion

Chapitre 2

Généralités sur le Cryptage d'Images

2.1 Introduction

Le monde est témoin d'un pas de géant dans l'utilisation de l'internet, malgré toutes les facilités et les améliorations qu'il nous a apporté, il reste comme même dangereux, vu ce qui pourrait se passer avec nos messages d'espionnage, vols ou modifications de contenu. Pour faire face aux ces éventuelles cyber-attaques, les chercheurs recoururent à trouver des solutions pertinentes et efficaces. Par intuition, ils suivirent l'approche utilisée par nos ancêtres pour protéger leurs lettres sensibles, où ils se sont appuyés sur des astuces pour les dissimuler ou en brouiller le contenu, afin qu'il devienne incompréhensible, s'il tombait entre les mains de l'ennemi. De mêmes, les efforts déployés actuellement tournent principalement auteur de ces deux axes, la dissimulation et le chiffrement de l'information à base de multiple techniques sophistiquées et surtout rapide. Au sujet de chiffrement, il s'agit d'un ensemble de procédures à être appliquer sur le message pour le rendre illisible pour les utilisateurs non autorisés, l'opération inverse s'appelle le déchiffrement. Dans ce chapitre, nous allons discuter la cryptographie en général et le cryptage d'image en particulier, à savoir l'historique de cet art et comment il s'est évolué avec le temps, les concepts les plus fréquents et leurs principaux théorèmes, nous allons citer aussi quelques algorithmes cryptographiques célèbres, les différents paradigmes de cryptage d'image, et les différents outils de test de performance d'un système de cryptage donné. Nous clôturons le chapitre en énumérant quelques-unes des contributions contemporaines des chercheurs, et une conclusion.

2.2 Historique

La cryptographie est non seulement une science du secret, mais aussi un instinct humain depuis l'antiquité, son histoire se représente en un combat entre les codeurs et les briseurs de code.

Bien avant l'apparition de la cryptographie, la stéganographie ou la science de dissimulation de l'information qui dominait, il s'agit d'une méthode obsolète utilisée pour protéger les messages envoyés. L'un des forme de la stéganographie remonte au 5è siècle av. J- C, c'est l'écriture sur le crâne rasé de messager, après le repoussement de ses cheveux, le messager se fut envoyé. À l'arrivé, le destinataire rasa de nouveau le crâne de messager pour pouvoir lire le message [48]. Cette méthode paraît très utile, sauf que dans le cas où l'intercepteur découvre l'endroit de message, il pourra immédiatement révéler son contenu, c'est la raison pour laquelle l'importance de la cryptographie se manifeste davantage. Malgré tout, la force de cette dernière est requise. L'un des événements historiques qui montrent le rôle du cryptage est la mort de la reine écosse Marie Stewart en 1587 [2], après que sa vie fut hypothéqué à la force de son chiffre, alors que l'ennemi britannique réussit à briser le code de son message qui contenait un attentat de meurtre de la reine d'Angleterre, donc le chiffrement est souvent une question de vie ou de mort [48]. L'histoire est témoin de plusieurs astuces de cryptage ; à ce moment-là, le cryptage n'était que des astuces ingénieuses, comme le cryptage par minuscule piqure d'épingle, sous certaines lettres de message. Ces lettres pointées représentent le message clair [48]. Une autre astuce inventée par Sherlock Holmes l'un des grands experts en cryptographie, elle est sous forme d'un dessin des hommes dansants, chaque position représente une lettre. Au 5^{me} siècle avant J -C, il y'avait l'ingéniosité d'un bâton en bois sur lequel un ruban de parchemin est enroulé, le message est écrit sur ce ruban de gauche à droite, et une lettre sur chaque circonvolution[48]. En 750, la contribution des arabes avait la part du lion, c'est eux qu'ils ont inventé la cryptanalyse, dont ils réussirent à rompre le chiffre de substitution, grâce à une technique d'étude de la fréquence d'apparition de chaque lettre dans le texte crypté. Elle a été créée par Al-kindi, connu comme le philosophe des arabes. A cet époque, la substituions alphabétique avait assouvi les besoins de communication en Europe [48]. Tel est le cas de chiffre de Jule César, son principe de fonctionnement est à base de décalage, ce décalage est fixe par rapport à l'ordre alphabétique, par exemple : on fixe le décalage à 4, A est remplacé par E, F remplace B, jusqu'à W qui devient A. Un autre chiffre de substitution alphabétique est le chiffre de Viginere, il ressemble au son précèdent, mais avec un décalage déterminé par un mot, ce chiffre perdura inattaquable pour deux siècles. A l'instar de la substitution alphabétique, une autre efficace méthode de substitution fut introduit, est la substitution homophonique, auquel cas, la substitution de chaque lettre est par plusieurs lettres selon sa fréquence d'apparition proportionnelle, comme la lettre V en français correspond environ à 2% de l'ensemble des lettres, pour le chiffrer on lui assigne deux symboles [48].

On ne peut pas parler sur l'histoire de code sans citer le fameux Grand Chiffre, c'est un chiffrement destiné aux messages méga-secrets, il utilise des chiffres pour coder des syllabes, exemple :

le mot les-en-ne-mi est codé par 22-125-46-345 chaque chiffre correspond à un syllabe, ou par fois une lettre. Ce chiffrement a résisté pendant des siècles [48]. Au cours de la première et de la deuxième guerre mondiale, le cryptage prit sa part de ces guerres-ci, notamment, avec l'invention des communications sans fils "télégraphie sans fil", et les machines de cryptage comme Enigma. À l'ère de l'informatique, DES (Data Encryption Standard) a fait une révolution, c'est un chiffrement par bloc itératif, il utilise une clé de 56 bits [48, 49]. Un an après, une autre avancée dans la cryptographie fut construite par les trois célèbres ingénieux Rivest, Shamir et Adleman, c'était le système RSA, il était le premier chiffrement à clé publique. Le cryptage de l'époque se limitait aux gouvernements et à l'armé. Aujourd'hui, la cryptographie électronique est au service de toute personne. Tout individu a le droit de protéger son contenu et sa vie privée.

2.3 Terminologie

Ici, on va introduire les principaux termes relevant de ce domaine, comme :

A. **Cryptage :** ou chiffrement, c'est un procédé cryptographique qui consiste à rendre un message clair en un message incompréhensif pour toute personne n'a pas la clé de déchiffrement.

B. **Décryptage :** ou déchiffrement, c'est l'opposé de cryptage, qui consiste à rendre un message crypté en un message clair, par la personne qui possède la clé de déchiffrement.

C. **Crypto-système :** c'est l'ensemble des techniques de chiffrement et de déchiffrement, y compris les clés utilisées. Son schéma est représenté sur la figure 2.1.

D. **Cryptographie :** c'est la discipline d'étude et de développement des méthodes de chiffrement et de déchiffrement des messages envoyés [49].

E. **Cryptanalyse :** il s'agit d'une technique permettant de détecter, soit de contenu de message chiffré sans avoir la clé de chiffrement, soit de la clé elle-même [39].

F. **Cryptologie :** étymologiquement, ce mot vient de mot grec kryptos "caché" et logos "science", ce qui signifie "science de secret". Conventionnellement, c'est la science qui étudie les méthodes de sécurisation "cryptographie et stéganographie", et les méthodes d'analyse des informations cryptées "cryptanalyse" [50]. Cette discipline a attiré l'attention de beaucoup de chercheurs dans le monde, il repose essentiellement sur les mathématiques.

G. **Clé :** il s'agit d'un paramètre essentiel dans l'opération de cryptage et de décryptage, il sert à déterminer la sortie d'un système de cryptage. Il se peut qu'elle soit indépendante de l'algorithme, comme il se peut qu'elle soit l'algorithme elle-même.

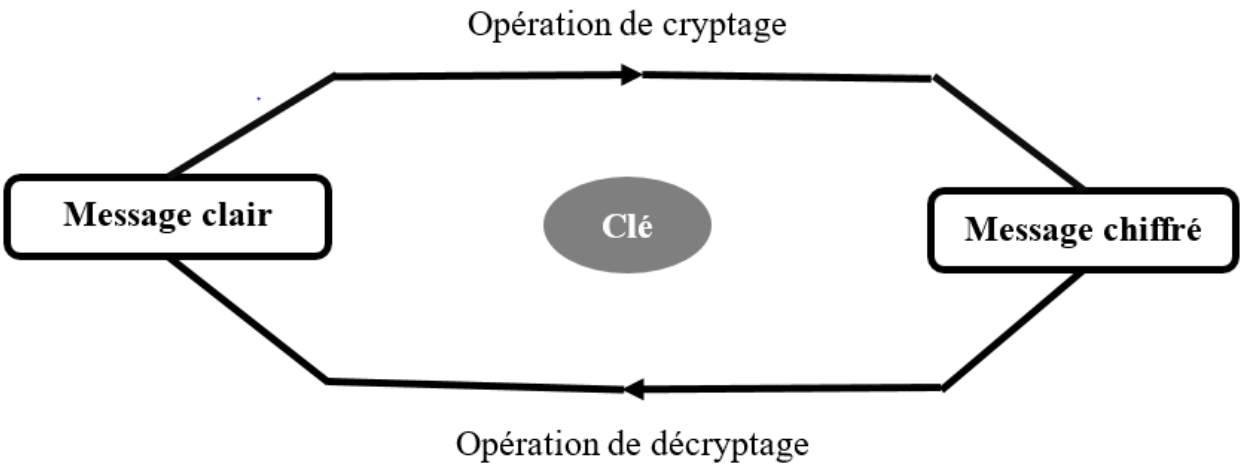


FIGURE 2.1 – Schéma d'un crypto-système.

H. **Image digitale :** c'est un tableau (matrice) de petits éléments (pixels), associés chacun à une valeur spécifique, varient entre 0 et 255. Le nombre de colonne représente la largeur de l'image (M), et le nombre de lignes représente la longueur de l'image (N). D'où, la taille de l'image est (M×N). À titre indicatif et non exhaustif, on distingue trois types d'images :
   - **Image binaire :** elle utilise un seul bit, donc deux états 0 ou 1. chaque pixel peut prendre l'une des deux couleurs, généralement le noir ou le blanc.
   - **Image en niveau de gris :** chaque pixel de cette image prend une valeur unique correspond au niveau de gris, ce qui signifie 256 niveaux de gris possibles.
   - **Image RGB :** comme son nom l'indique, cette image est composée de trois matrices, chaque matrice correspond à l'une des couleurs primaires (rouge, vert et bleu).

2.4 Méthodes crypto-analytiques

Un système de chiffrement fort doit lutter contre quatre principales attaques probables :

A. **Attaque à texte chiffré seul :** c'est le cas le moins fréquent, vu sa difficulté surtout pour le chiffrement moderne, l'attaquant essaie de trouver des informations sur le message clair ou sur la/les clé(s) utilisée(s) à partir du message crypté uniquement.

B. **Attaque à texte clair connu :** c'est le type d'attaque le plus fréquent, dans lequel l'attaquant dispose de différents couples de type message clair/message chiffré.

C. **Attaque à texte clair choisi :** c'est le type d'attaque le plus puissant, il permet à l'attaquant de choisir des messages clairs et de trouver leur messages chiffrés correspondants [50], à travers une boite noir de déchiffrement. Dans le but d'extraire la/les clé(s) utilisée(s).

D. **Attaque à texte chiffré choisi :** L'attaquant introduit un message chiffré choisi dans une boîte noire de déchiffrement. Le pair message clair produit/ message chiffré introduit, lui permet d'obtenir toute ou une partie de la clé utilisée [50].

2.5 Objectifs de la cryptographie

La cryptographie vise à assurer la sécurité des messages envoyés dans un canal de transmission peu sûr, et à protéger la vie privée des internautes. Cette phrase peut se résumer en quatre mots, confidentialité, intégrité, authentification et non-répudiation [51]. Détaillons chacun d'entre eux :

A. **Confidentialité :** il s'agit de la protection des informations contre tout accès non autorisé. Seul le destinataire peut lire le message envoyé.

B. **Intégrité :** il s'agit d'un mécanisme visant à assurer l'intégrité de contenu du message, contre toute modification volontaire ou involontaire.

C. **Authentification :** il s'agit d'une méthode permettant au destinataire d'authentifier l'origine du message.

D. **Non répudiation :** cela signifie que l'expéditeur ne peut pas nier l'envoi du message.

2.6 Classification des systèmes de cryptage

On peut catégoriser les systèmes de cryptage conformément aux leurs architectures, leurs clés, leur plateforme, ou au pourcentage des données cryptées.

2.6.1 Selon l'architecture

On distingue deux méthodes, cryptage par blocs et cryptage par flot.

Cryptage par blocs : Le principe de cryptage par blocs est de couper le message clair en plusieurs blocs de taille fixe, et puis les transformer en blocs chiffrés. La taille de bloc varie entre 32 et 512 [51], plus la taille est grande, meilleur est le système de cryptage. L'uns des systèmes de cryptage par blocs les plus connus est : AES *(Advanced Encryption Standard)* et GOST *(Government Standard)*.

Cryptage par flot : il s'agit d'un chiffrement par de petites unités de message, il dispose une sécurité parfaite et une grande rapidité. Un exemple de chiffrement par flot est le RC4 *(Rivest Cipher)* et SEAL *(Software-optimized-Encryption-Al-gorithm)* [51].

2.6.2 Selon la clé

On peut classer les systèmes de cryptage selon la clé en deux classes, cryptage à clé privée ou cryptage symétrique, dont la clé de cryptage est le même que la clé de décryptage $k_e = k_d$. Auquel cas, la clé devra passer de l'émetteur au récepteur par un canal secret distinct. Tandis que le cryptage dont la clé de cryptage diffère de la clé de décryptage $k_e \neq k_d$ s'appelle cryptage à clé publique (cryptage asymétrique). Auquel cas, la clé de cryptage k_e est diffusée, et la clé de décryptage k_e est gardée privée.

2.6.3 Selon le pourcentage de données cryptées

On a deux types, cryptage total, où le message sera complètement crypté, et on a le cryptage partiel, où une partie de message qui représente la zone d'intérêt sera cryptée. Donc il dépend de quantité d'informations cryptées dans le message.

2.6.4 Selon la plateforme

Selon la plateforme, on trouve Le cryptage matériel et le cryptage logiciel. Pour la première technique, s'agissant de petites puces constituées d'un processeur qui gère le processus de cryptage/décryptage, ainsi qu'une mémoire qui sert à stocker la clé, ce type est caractérisé par sa vitesse et sa haute sécurité [51]. Contrairement au cryptage matériel, le cryptage logiciel n'a ni processeur ni mémoire dédiée, mais plutôt il utilise le système général afin d'effectuer les opérations de cryptage et de décryptage. L'un de ses avantages réside dans sa flexibilité, sa portabilité et sa facilité d'utilisation [51].

2.7 Principes de Kirchhoff

En 1883, Auguste Kirckhoff publia son article "la cryptographie militaire" ; il y présenta quelques lois intéressantes, dans le but de concevoir un bon système cryptographique militaire, car à cet époque, la cryptographie était une occupation politique et militaire. Ces principes qui constituent même la cryptographie moderne, sont les suivants [52] :

- Le système doit être matériellement, sinon mathématiquement indéchiffrable.
- Il faut qu'il n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi
- La clé doit pouvoir être communiquée et conservée sans l'aide de notes écrites, et être changée ou modérée au gré des correspondants.

— Il faut qu'il soit applicable à la correspondance télégraphique.

— Il faut qu'il soit portatif, et que son maniement ou son fonctionnement n'exige pas le concours de plusieurs personnes.

— Vu les circonstances qui en commandent l'application, il est nécessaire que le système soit d'un usage facile, ne demandant ni tension d'esprit, ni la connaissance d'une longue série de règles à observer.

Comme il a lié la sécurité d'un message par la sécurité de sa clé, et non par l'algorithme lui-même, étant donné que l'ennemi connait déjà l'algorithme. Cette règle est encore valable.

2.8 Théorème de Shannon

Le théorème de Shannon ou théorie d'information qu'est aujourd'hui tout une discipline socle dans le domaine de la communication, et le fruit de ces années de travail opiniâtre dans la cryptographie de la défense américaine. Il publia en 1948 un article portant le nom "A mathematical theory of communications", dans ce précieux papier, Shannon a traité les problèmes mathématiques de base, liés aux systèmes de communication (figure 2.2 [53]). Si un tel schéma peut nous apparaître évident à présent, il était une véritable révolution à l'époque, où il définit clairement la source, l'émetteur et le récepteur, le signal et le reste des composantes d'un système de communication. Il introduit également 17 théorèmes auteur de codage/décodage d'information, de bruit des canaux de transmission ainsi que son capacité sur les systèmes de communication discrets, continus et mixtes.

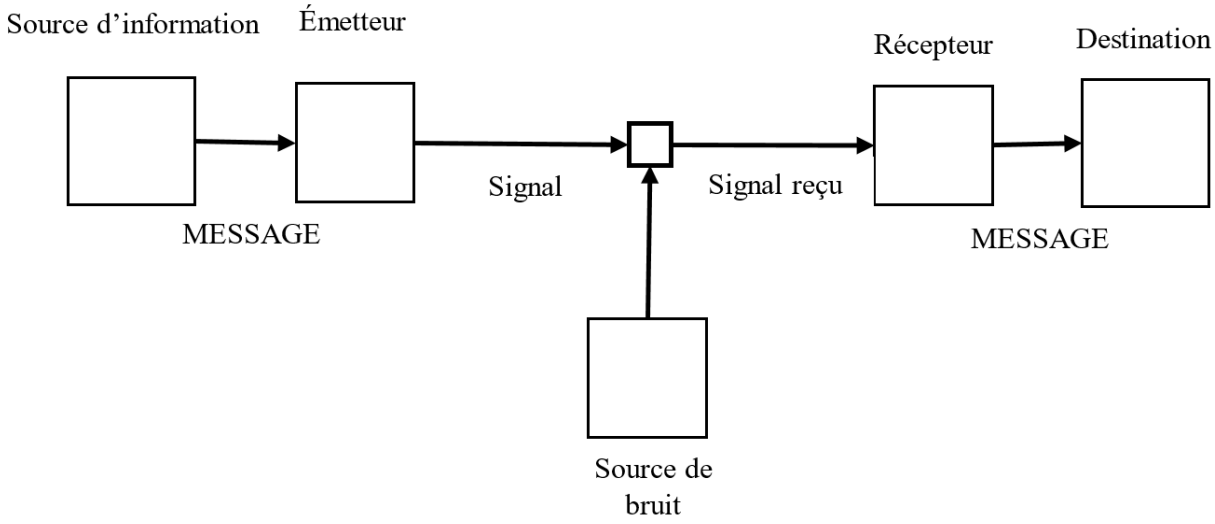


FIGURE 2.2 – Schéma général d'un système de communication.

En 1949, Claude Shannon fit apparaitre un autre article phare dans les systèmes de secret, c-

à-d, une étude théorique basée sur les structures mathématiques et leurs propriétés, focalisée sur les systèmes de communication. Dans cet article, Nous avons constaté qu'il mettait en évidence différentes méthodes de cryptage et les moyens de les casser. D'après lui, un système de sécurité est une famille de transformations uniquement réversibles d'un ensemble de messages possibles, en un ensemble de cryptogrammes, la transformation ayant une probabilité associée. En outre, n'importe quel ensemble d'entités de ce type sera appelé "système de secret". L'ensemble des messages possibles sera appelé, par commodité, "espace des messages" et l'ensemble des cryptogrammes possibles "espace des cryptogrammes". D'après Shannon, les systèmes de secret se divisent en trois principaux systèmes, systèmes de dissimulation qu'est un problème psychique, systèmes de privauté qu'est un problème technique, et "a true securecy systemes" qu'est selon lui, le vrai système cryptographique. Pour l'évaluer il faut assurer de cinq principaux critères :

- Le degré de secret, il s'agit d'un critère qui se diffère d'un code à un autre, il correspond à la quantité de travail nécessaire pour le casser par l'interceptant.
- La taille de la clé. Les chercheurs se mettent d'accord sur l'importance de la clé, elle doit être d'une taille supérieure ou égale à la taille de message, comme elle doit être transmise par un canal sécurisé.
- La complexité des opérations de cryptage et de décryptage, la complexité entraîne des pertes de temps, des erreurs, ou même elle exige des machines couteuses, c'est la raison pour laquelle, un système de cryptage doit être le plus simple possible.
- Propagation des erreurs, les erreurs qui peuvent attendre un message durant sa transmission mènent à une grande perte dans le message décrypté.
- Expansion de message par l'ajout de plusieurs substituts, ce procédé est indésirable.

Et pour qu'un système de sécurité soit parfait, Shannon annonce :

Théorème. ***6.****Une condition nécessaire et efficiente pour un secret parfait est que :*

$$P_M(E) = P(E). \tag{2.1}$$

Pour tous M et E, $P_M(E)$ *doit être indépendant de M, où :*

M : est le message clair.

E : est le message crypté.

$P_M(E)$ *: Probabilité conditionnelle de E si M est réalisé, c'est-à-dire la somme des probabilités de toutes les clés qui produisent E à partir de M.*

$P(E)$ *: Probabilité d'obtenir le message crypté E.*

Ce qu'on peut conclure de ci-dessus, dans les systèmes de sécurité parfaits, le nombre de chiffre

possible est plus grand que le nombre des messages, d'où le nombre de k est supérieur au nombre de M, et aucune clé ne peut être réutilisée [51].

2.9 Cryptographie visuelle

Ou le début du cryptage d'image, il est proposé en 1994 par Moni Naor et Adi Shamir [54], ce terme relie entre la cryptographie et le traitement d'image. Sauf que cette technique n'utilise aucun calcul mathématique dans la plupart de ses cas. Pour décrypter le message, c'est à travers l'œil humain, cette méthode est vachement sécurisée car, elle base essentiellement sur le cryptage de Vernam [55], son principe est fondé sur l'opération XOR entre l'image secrète (S) et un masque jetable (M) constitué de pixels blanc et noir $(C = SxorM)$, le masque devait être inscrit sur une matière transparente, pour lui en servir dans l'opération de décryptage, en lui superposant avec le message secret. De plus, ce masque n'est utilisé qu'une seul fois.

2.10 Cryptographie quantique

La nature était toujours notre source d'inspiration. L'histoire de quantique remonte au 1801, ou YOUNG thomas publia un travail sur la nature de la lumière intitulé "The Bakerian Lecture. On the Theory of Light and Colors" [56] dans lequel, il démontra la nature ondulaire de la lumières par sa fameuse expérience fente de Young. Dans cette expérience, il fit passer un faisceau lumineux entre deux fentes, les ondes diffractées à la sortie de ses deux fentes se superposent, en lui faisant former une interférence sur l'autre face de l'écran, et se dessinent des rubans lumineuses. D'où le principe de la superposition de la physique quantique, et c'est ici que le q-bit a vu le jour. Cet élément peut prendre la valeur $|0 > |1 >$ et $|0 > |1 >$ au même temps, ce qui permet aux ordinateurs quantiques d'augmenter leur vitesse de calcul, ainsi que la construction d'un canal quantique pour faire la distribution quantique des clés sous forme de $|0 > |1 >$ et $|0 > |1 >$, à une sécurité absolue, aucune tentative d'attaque aura la possibilité d'intercepter le message. Les recherches dans la cryptographie quantique ont été mené avant même le développement des ordinateurs quantiques. En 1970, Stephen Wiesner présenta son idée ingénieuse de l'utilisation de la mécanique quantique, tantôt pour coder les billets de banque, tantôt pour établir un canal de type multiplexeur capable de confondre deux messages de manière à ce qu'un seul message pourra être lu. À ce moment-là, son idée n'était applicable qu'en 1989, où Bennett & Brassard firent la première expérience de leur premier protocole de distribution des clefs secrètes BB84 [57]. L'application de la cryptographie quantique nécessite un calculateur quantique, [58]. Dans cette branche physique, les qubits utilisés

sont les photons, ils sont décrit par trois polarisations linéaires, horizontal $(+)$, vertical $(\times)$, ou une superposition des deux. Selon le protocole BB84 qui se sert de la cryptographie asymétrique, le destinataire envoie des photons sur un canal quantique avec différentes polarisations, il choisit de son part une base de mesure complètement aléatoire. Après, ces deux interlocuteurs discutent sur le canal public, le destinateur révèle sur la base des mesures de chaque q-bit et non sur le bit lui-même, si le destinataire trouve qu'il s'agit de bonne base, donc il est sûr de résultat, sinon il rejette l'information [59].

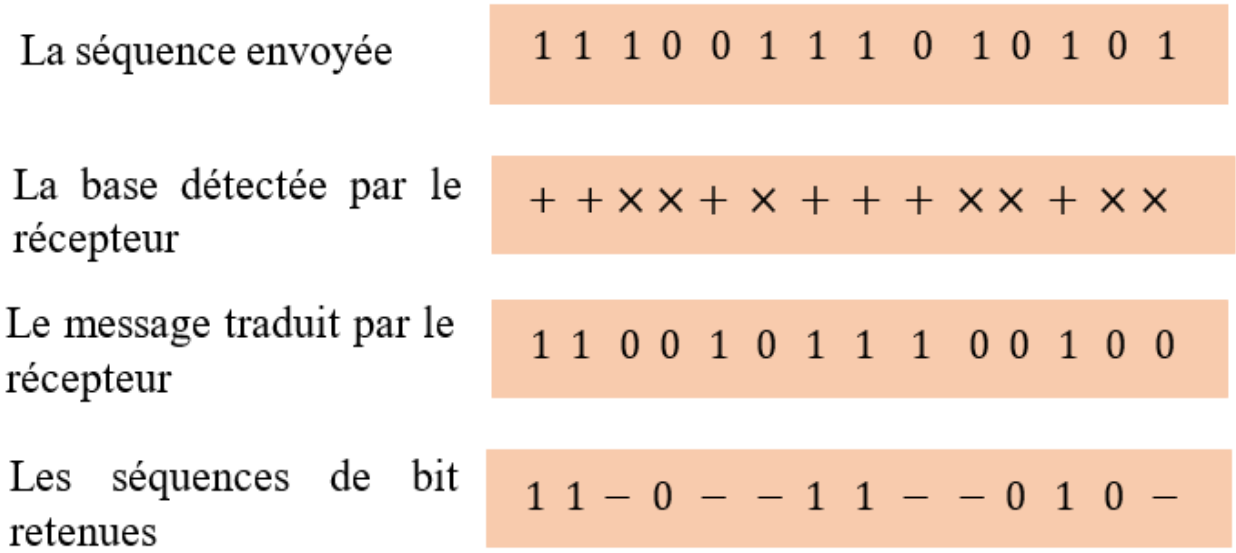


FIGURE 2.3 – Exemple d'un crypto-système quantique.

La cryptographie quantique peut porter le coup fatal aux attaques en ligne, cela dit, elle a des limites, c'est qu'elle ne peut utiliser que des liaisons optiques, ce qui explique l'apparition de la cryptographie post-quantique, qui base sur les algorithmes classiques pour lutter contre les attaques issues des ordinateurs quantiques.

2.11 Multimédia et cryptographie

De nos jours, la technologie de la communication est en horrible progression. Grâce aux développements dans presque tous les domaines scientifiques, comme le web, les multimédias, les réseaux sociaux, et par l'apparition de nouvelles applications pertinentes qu'elles ont participé à son évolution. Ce qu'il fait de l'individu en besoin permanent d'outils de la communication [39]. Malgré les avantages incomparables du développement multimédias, elles restent peu sûre, car elles disposent des réseaux de communication publics et ouverts. Leur contenu doit être protégé, vu la confidentialité et la valeur des informations qu'elle porte. Mais, et grâce à la cryptographie,

on peut fort heureusement protéger leurs contenus [39]. Un bon algorithme de cryptage multimédia devait garantir une livraison sécurisée en temps réel de différents types de messages, de haute qualité, et du haut débit sur un réseau peu sûr. Ce qui représente un vrai défi pour les chercheurs [39].

2.12 Algorithmes cryptographiques

Cette section présente un petit aperçu sur certains algorithmes de chiffrement les plus populaires. Malgré leur ancienneté, ils sont toujours en usage, et en développement conformément aux conditions de l'époque.

2.12.1 DES, D-DES, T-DES

Data Encryption Standard (DES) est l'une des techniques cryptographiques les plus forte et efficace, il se considère comme un algorithme de chiffrement symétrique par blocs , chaque bloc est de 64 bits. La clé utilisée pour crypter et décrypter est de 56 bits. Le DES se déroule selon deux processus de base [50, 39] :

2.12.1.1 Création des clés :

Les 64 bits de la clé vont passer par 3 opérations :

— Une permutation initiale : à l'aide d'une matrice de permutation, on change les positions des bits et on élimine les bits de parité dans chaque octet. La taille résultante est de 56 bits.

— Décalage circulaire : ici, on divise la clé en deux sous-blocs, ces deux sous blocs subissent au un ensemble de 16 décalages circulaires, de sorte qu'il en résulte 16 clés $(k_1, \ldots k_{16})$ (après la concaténation).

— Une seconde permutation : une seconde permutation est appliquée sur les seize clés, toujours les bits de parité sont ignorés, ce qui rend la taille de la clé vaut 48 bits.

2.12.1.2 Procédure de cryptage

Tout d'abord, les bits du bloc à chiffrer sont subi la permutation initiale, via une matrice de permutation initiale. Ensuite, le bloc de 64 bits est scindé en deux sous blocs de 32 bits (G_0, D_0). Le sous bloc droit va s'étendre à 48 bits grâce à une table appelée table d'expansion, pour que l'algorithme puisse procéder à ou exclusif entre k_1 et le sous bloc droit étendu. La nouvelle valeur de D_0 passe par des tables de substitution (S-box) et de permutation pour former de nouveau

un sous bloc de 32 bits D_1, G_0 et D_1 soumettent à une opération XOR, le résultat est reçu par D_1, tandis que G_1 possèdera D_0. L'algorithme DES effectue 16 tours. À la fin des itérations, les deux sous-blocs (G_{16} , D_{16}) seront re-concaténés et puis subiront la permutation initiale mais inversement. Pour l'algorithme D-DES (double DES) et T-DES (triple DES), il s'agit de deux naïves méthodes d'amélioration de la sécurité ; chaque bloc est crypté deux fois avec deux clés, et trois fois avec trois clés respectivement.

2.12.2 Advanced Encryption Standard(AES)

AES est pour Advanced Encryption Standard, un algorithme de chiffrement itératif par blocs, il vient pour remplacer DES, car il n'est plus sûre, et il résiste mieux aux attaques. Les clés supportées par AES sont de taille 128, 192 ou 256 bits, tandis que la taille d'un bloc de message est 128 bits. Comme DES, AES s'articule sur des ronds mais le nombre de ronds est relatif à la taille de la clé (10 ronds pour 128 bits, 12 ronds pour 192 bits, 14 ronds pour 256 bits). Nous détaillerons AES-128 qui utilise 128 bits de données repartis en 16 blocs (*state*) et organisés dans un tableau (*state*) [50].

- **SubBytes :** ici, on utilise le boites de substitution (S-box) pour permuter les éléments du bloc *state*.
- **ShiftRows :** on effectue un décalage circulaire sur chaque ligne de *state*, sauf la première, on la laisse inchangée. Ce qu'il fait à chaque colonne de *state* de sortie porte des octets de chaque colonne de *state* de l'entrée.
- **MixColumns :** les transformations de cette étape se font au niveau des colonnes, étant donné que toute les colonnes de *state* vont être transformées en matrices constantes utilisées pour crypter et pour décrypter. Le principe est que chaque colonne de la matrice *state* est multipliée par la matrice constante de gauche Dans Glois Field 2^8.
- **AddRoundKey :** on effectue l'opération XOR entre *state* et la matrice clé, élément par élément.

Quant à la clé secrète, elle comprend plusieurs sous-clés dont le nombre est égal au nombre de tours plus un, qui est elle-même dépend de la taille de la clé utilisée.

2.12.3 GOST

GOST est l'abréviation de "Gosudarstvennyi Standard" ou "norme gouvernementale", est un algorithme de chiffrement par bloc, il utilise un bloc de 64 bits, et une clé de 256 bits. GOST adopte le réseau feistel de 32 tours. Son principe est de diviser le bloc en deux moitiés égales,

moitié droite et moitié gauche. Puis d'ajouter une sous-clé de 32 bits au module 2^{32}. Les résultats passent à travers des couches s-box afin de faire une confusion dans les valeurs d'entrée, ce qui lui fait jouer le rôle d'une autre clé, et renforce la sécurité de cette technique. Après la substitution, un décalage circulaire de 11 bits est appliqué sur les résultats de la sortie. Enfin, les résultats sont couplés à la moitié gauche par ou exclusif, puis la moitié gauche devient la moitié droite, tandis que la moitié droite devient la nouvelle moitié gauche. Ces étapes sont répétées 32 fois [51].

2.12.4 RC4

RC4 *(Rivest Cipher)* est un algorithme de chiffrement par flot avec une clé de taille variable de 1 à 256 octets utilisées pour initialiser un vecteur d'état S de 256 octets, S consiste en une permutation de tous les nombres de 8 bits de 0 à 255. Un octet k est généré à partir de S en sélectionnant l'une de 255 entrées, d'une manière systématique. À chaque génération de la valeur de k, les entrées de S sont à nouveau permutées. Une opération xor est appliquée entre le flot aléatoire de k et le texte en clair. Il est à noter que cette technique repose essentiellement sur la sécurité de la clé [51].

2.13 Chaos dans le cryptage

Aux années 60, le météorologiste Edward Lorenz a fait sortir au monde une nouvelle théorie, ce qu'on appelle la théorie de chaos, connue par "l'effet de papillon". Cette théorie étudie la conduite des systèmes dynamiques, dont Lorenz a confirmé qu'il est impossible de prédire l'évolution de ces systèmes à long terme, sans connaitre les conditions initiales avec minutie. On trouve ses applications dans de nombreux domaines, comme la physique, les mathématiques, l'ingénierie, la biologie, etc.[50]. Les systèmes chaotiques sont généralement caractérisés par l'ergodicité, la sensibilité aux conditions initiales et aux paramètres de contrôle, et un comportement aléatoire [60, 61]. Ces caractéristiques ont attiré l'attention des chercheurs en cryptographie, dont le premier qui a introduit le chaos en cryptographie était Matthews en 1989 [3]. le tableau ci-dessous récapitule les points communs entre le chaos et la cryptographie [62].

TABLEAU 2.1 – Comparaison entre le chaos et les propriétés cryptographiques.

Caractéristiques chaotiques	Propriété cryptographique	Description
Ergodicité	Confusion	La sortie a la même distribution pour toute entrée.
Sensibilité aux conditions initiales et aux paramètres de contrôle	Diffusion avec un petit changement dans le texte en clair/clé secrète	Une petite déviation dans l'entrée peut provoquer un grand changement dans l'espace entier.
Propriété de mélange	Diffusion avec un petit changement dans un bloc de texte en clair de l'ensemble du texte en clair.	Une petite déviation dans l'entrée peut provoquer un grand changement dans la zone locale.
Dynamique déterministe	Pseudo-aléatoire déterministe	Un processus déterministe peut provoquer un comportement de type aléatoire (pseudo-aléatoire)
Complexité de la structure	Complexité des algorithmes	Un processus simple a une complexité très élevée

2.14 Cryptage d'image

Le cryptage d'image consiste essentiellement à modifier l'image et à rendre son contenu complètement méconnaissable, de telle sorte que seul un destinataire légal puisse le reconstituer. Il est caractérisé par :

P : l'image en clair à crypter (plain image).

C : l'image cryptée (cipher image).

K : la ou les clés de cryptage/décryptage (key).

E : un système de transformations de cryptage (Encryption).

D : un système de transformations de décryptage (Decryption).

Le processus de cryptage et de décryptage est décrit par 2.2 et 2.3 respectivement.

$$C = E_{k_e}(P) \tag{2.2}$$

$$P = D_{k_d}(C). \tag{2.3}$$

2.14.1 Techniques de cryptage d'image

Selon Shannon, La confusion et la diffusion sont des techniques de suppression de redondance dans le texte original, du fait que les images en clair ont une forte redondance, l'application de cette technique devrait largement augmenter le brouillage et l'ambiguïté dans l'image cryptée [51]. Pour une image, ça se définit comme suit :

Confusion : c'est une ou un ensemble de transformations des valeurs des pixels de l'image à crypter. Par exemple, l'application de l'opération XOR entre le pixel en question, et les pixels voisins.

Diffusion : il s'agit d'une opération de répartition des positions de pixels sans changer leurs valeurs. Exemple : utilisation d'un générateur pseudo-aléatoire, et redistribution des pixels de l'image clair dans l'ordre des valeurs générées.

2.14.2 Métriques d'évaluation

Pour évaluer la qualité d'un système de cryptage d'image donné, il faut le faire passer par plusieurs tests de performance, à travers eux, on peut savoir s'il est capable de résister aux différents types d'attaque ou il échoue devant la moindre attaque, Nous présenterons ci-après quelques importants tests.

2.14.2.1 Vision humaine

Après le processus de cryptage, l'utilisateur peut remarquer avec seulement son œil si l'image cryptée est lisible ou illisible, la figure 2.4 donne un exemple d'une image en couleur avant et après un processus de cryptage. À noter que cette méthode ne suffit plus, surtout à l'heure actuelle.

FIGURE 2.4 – Image claire et image cryptée.

2.14.2.2 Analyse des histogrammes

L'histogramme représente la distribution de l'intensité des pixels dans une image, est un outil de test très utile dans le cryptage d'images [63]. Sur l'axe horizontal on trouve les valeurs des intensités des pixels, varie entre 0 et 255, alors que sur l'axe vertical on trouve les fréquences d'apparition de chaque pixel dans l'image. Chaque image simple a son propre histogramme qui présente beaucoup d'informations à son sujet. Cependant, les images bien cryptées n'ont aucune différence dans leurs histogrammes, la distribution de leurs pixels doit être soit Gaussienne, uniforme, exponentielle décroissante ou une autre forme aléatoire, mais différente de l'image originale.

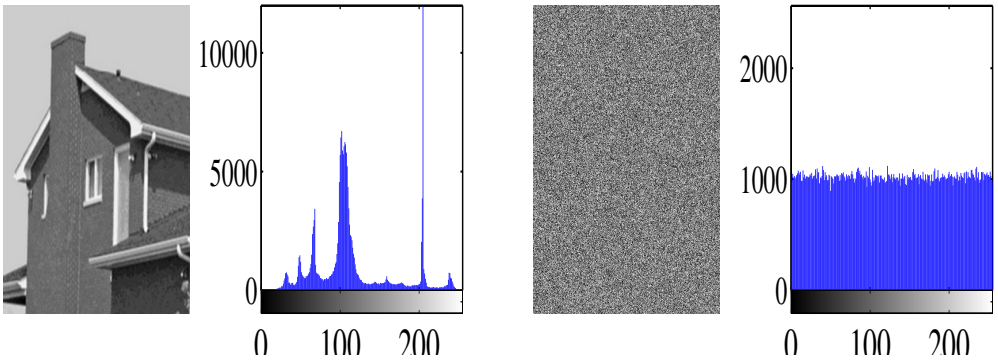


FIGURE 2.5 – Histogrammes de l'image en niveaux de gris et de son image chiffrée.

2.14.2.3 Calcul de l'entropie

L'information de l'entropie est une des théories mathématiques les plus célèbres, elle a été proposée par Claude E. Shannon en 1949 [53]. Contrairement aux systèmes de compression, plus que l'entropie est grande meilleur est le message crypté. Il s'agit à la fois d'une mesure quantitative de l'information contenue dans le message, et qualitative de l'information cryptée (ou compressée). Dans de cryptage d'image, l'entropie de Shannon calcule le degré de l'aspect aléatoire dans l'image cryptée. Elle peut être exprimée par l'équation 1.8. Où $P(s_i)$ représente la probabilité d'apparition de l'intensité de la valeur de pixel. La valeur théorique idéale de H(s) est égale à 8 [63], plus l'entropie est proche de 8, mieux est la distribution aléatoire dans l'image cryptée.

2.14.2.4 Analyse de corrélation

Dans la même image, la mesure de corrélation entre pixels voisins dans l'image cryptée est une métrique très importante. Les pixels d'une image cryptée ne doivent pas être corrélés, et la valeur de corrélation doit être proche de zéro. Pour la calculer, on prend un échantillon aléatoire de N pixels adjacents de l'image en question, puis on calcule les coefficients de corrélation entre pixels adjacents dans les trois directions (horizontale, verticale et diagonale). Pour quantifier la

corrélation, on peut calculer la déviation entre les pixels dans l'image cryptée et les pixels dans leur image originale, en utilisant l'équation 2.4 :

$$corr\,(x,y) = \frac{cov\,(x,y)}{\sqrt{D\,(x)} \times \sqrt{D\,(y)}}. \tag{2.4}$$

$$cov\,(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E\,(x))(y_i - E\,(y)). \tag{2.5}$$

$$D\,(x) = \sum_{i=1}^{N}(x_i - E\,(x))^2\,,\; E\,(x) = \frac{1}{N}\sum_{i=1}^{N} x_i. \tag{2.6}$$

N est le nombre total de pixels de l'image, E(x) et E(y) sont les moyennes des pixels x_i, y_i respectivement.

2.14.2.5 Sensibilité de l'image vis à vis des attaques différentielles

La sensibilité de l'image en clair à l'attaque du texte en clair constitue un critère très important, car, les systèmes de cryptage dont la sensibilité est limitée sont très vulnérables à ce type d'attaques, qui sont à leur tour très répondus [64, 65]. Afin de quantifier cette sensibilité, nous mesurons l'impact d'un seul pixel sur l'image correspondante, nous avons deux mesures quantitatives peuvent être utilisées, le NPCR (Number of Pixels Change Rate) ou bien le taux de changement du nombre de pixels et l'UACI (Unified Average Changing Intensity) ou bien la moyenne unifiée du changement d'intensité, leurs formules sont données comme suit :

$$NPCR = \frac{\sum_{i,j} D\,(i,j)}{M \times N} \times 100. \tag{2.7}$$

$$UACI = \frac{1}{M \times N}\left[\sum_{i,j}\frac{C_1\,(i,j) - C_2\,(i,j)}{255}\right] \times 100. \tag{2.8}$$

On crypte l'image originale, on obtient C_1, puis on re-crypte la même image avec un petit changement dans l'un de ses pixels, on obtient C_2. D est également défini par l'équation ci-dessous :

$$D\,(i,j) = \begin{cases} 0 & C_1\,(i,j) = C_2\,(i,j) \\ 1 & else \end{cases} \tag{2.9}$$

Pour que notre système de cryptage soit idéal, c.-à-d qu'il soit capable de résister à une attaque différentielle, NPCR et UACI devraient valoir vers 33.46% [66] et 100% respectivement.

2.14.2.6 PSNR, SSIM

PSNR et SSIM sont deux mesures qualitatives de l'image, dont PSNR (Peak signal to noise ratio) calcule la qualité de l'image après un traitement, il est donné par [67] :

$$PSNR = 10log_{10}\left(\frac{L^2}{MSE}\right) \tag{2.10}$$

Où L est la valeur maximale possible d'un pixel dans l'image.

$$MSE = \frac{1}{mn}\sum_{i,j}(X_{i,j} - Y_{i,j})^2. \tag{2.11}$$

X et Y sont l'image avant et après le traitement, m et n sont le nombre de lignes et de colonnes dans l'image. MSE (the mean square error) est l'erreur quadratique moyenne de la même image, si le MSE tend vers 0, la valeur de PSNR tend vers l'infini, ce qui montre qu'une valeur PSNR plus élevée fournit une meilleure qualité d'image. Quant à SSIM (structural similarity index measurement), elle mesure la similarité entre deux images, elle a été développé par Wang et al [68]. SSIM base sur la modélisation de toute distorsion d'image sous la forme d'une association de trois facteurs, notamment la perte de corrélation et la perte d'image [69]. elle est défini par :

$$SSIM(X,Y) = \frac{(2u_x u_y + C_1)(2\sigma_{xy} + C_2)}{(u_x^2 + u_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \tag{2.12}$$

u_x et u_y représentent les moyennes des pixels en niveaux de gris dans deux images X, Y, δ_x et δ_y sont leurs écarts types, tandis que δ_{xy} est la covariance croisée de X et Y. C_1 et C_2 sont deux constantes prédéfinies. On peut utiliser une relation généralisée entre le SSIM et PSNR pour toute forme de détérioration de l'image. Elle est donnée comme suit [69] :

$$PSNR = 10log_{10}\left[\frac{2\sigma_{xy}\left(L\left(X,Y\right) - SSIM\right)}{255^2 SSIM} + \left(\frac{u_x - u_y}{255}\right)^2\right]. \tag{2.13}$$

On utilise souvent ses deux métriques dans le cryptage d'image, pour évaluer la dégradation de la qualité des images décryptées après avoir été submergées dans le bruit, où elles ont subi une perte de donnés pendant leur transmission, donc l'influence de bruit sur la résistance de crypto-système. Auquel cas, les valeurs PSNR et SSIM doivent être en dessous de 10 et près de 1 respectivement. Ou bien, pour tester le dégrée de lucidité de l'image cryptée, auquel cas, la valeur de PSNR devrait être inférieur à 10, tandis que le SSIM devrait s'approcher de 0.

2.14.2.7 Erreur absolue moyenne (MAE)

MAE (Mean Absolute Error) est une mesure standard fréquemment utilisée pour étudier la robustesse d'un schéma de cryptage proposé, elle permet d'évaluer l'erreur entre l'image brute P et l'image chiffrée C. Une MAE suffisamment grande signifie un chiffrement plus sûr. La définition de MAE est donnée par [70] :

$$MAE = \frac{1}{M \times N} \sum_{j=1}^{M} \sum_{i=1}^{N} |C(i,j) - P(i,j)| . \tag{2.14}$$

Avec $M \times N$ est la taille de l'image.

2.14.2.8 Autres tests quantitatifs

La qualité d'un système de cryptage peut être testée en analysant la texture de l'image cryptée, en utilisant trois autres mesures : L'homogénéité, le contraste et l'énergie. Un cryptage de haute qualité doit générer des images chiffrées avec un contraste, une homogénéité et une énergie proches de la valeur optimale, elles sont citées de même ordre comme suit : 10.5, 0.38940 et 0.015625. Leurs formules sont [71] :

$$contrast(C) = \sum_{i,j \in (1,..,8)} |i-j|^2 pr(i,j) . \tag{2.15}$$

$$Homogeneity(C) = \sum_{i,j \in (1,..,8)} \frac{pr(i,j)}{1+|i-j|}. \tag{2.16}$$

$$Energy(C) = \sum_{i,j \in (1,..,8)} pr(i,j)^2. \tag{2.17}$$

Avec C est l'image cryptée divisée en 8 rangs, pr est une matrice de probabilité de co-occurrence en niveaux de gris 8×8, elle est définie par :

$$pr(i,j) = \frac{1}{r(c-1)}. \tag{2.18}$$

r est pour le nombre de lignes, c est pour le nombre de colonnes.

2.14.2.9 Sensibilité de la clé

Une sensibilité élevée des clés est requise pour un crypto-système mieux sécurisé. Autrement dit, une légère différence au niveau de l'une des clés de cryptage ou de décryptage, rend le déchif-

frement impossible. Une forte sensibilité de la clé permet au crypto-système de faire face contre les attaques par force brute, où l'attaquant essaie de trouver la clé du cryptogramme. Il existe plusieurs méthodes pour tester la sensibilité de la clé, à titre d'exemple :

— On induit un tout petit changement à un de composants de la clé de décryptage, l'image décryptée résultante c'est elle qui va affirmer la précision.

— On crypte l'image originale une fois par la clé normalement, et une autre fois par la même clé mais, avec une petite modification. Ensuite, on compare les deux images cryptées résultantes.

Pour le premier exemple, et après avoir balayé toutes les composantes de la clé, si les résultats visuels ou quantitatifs (via PSNR, SSIM, NPCR, UACI) montrent que l'image résultante est complètement endommagée et qu'elle ne peut pas révéler sur la moindre information, donc notre clé est de haute sensibilité. Pour le deuxième exemple, la clé est sensible, si les pixels de l'image résultante de comparaison ne sont pas nuls, ou au moins loin de l'être.

2.14.2.10 Espace clé

Selon Kirchhoff, la sécurité d'un crypto-système ne tient qu'à sa clé. Si la clé n'est pas à la hauteur ou très petite, le crypto-systèmes sera facilement cassable, quel que soit la force de son algorithme de cryptage. Avec une taille égale à k, un attaquant devait faire 2^k opérations pour trouver la clé. Dans [72], Wu et al ont annoncé que pour garantir un niveau de sécurité élevé, le k devait être supérieur ou égal à 100.

2.14.2.11 Rapidité, implémentation et coût

Le temps de traitement d'un algorithme de cryptage/ décryptage est un facteur crucial, il dépend de plusieurs paramètres, par exemple : les caractéristiques de l'ordinateur utilisé, l'algorithme elle-même, la taille de l'image.... Bien évidemment, plus le temps de traitement est faible, plus le message est sécurisé, dont l'attaquant ne peut pas révéler sur son contenu. Quant à l'implémentation, beaucoup de travail sur la cryptographie et l'amélioration de la sécurité ne prend pas en considération la mise en œuvre, en dépit de son importance. Un crypto-système doit être simple dans sa mise en œuvre matérielle ou logicielle, comme il doit être à bas coût. Par conséquent, un crypto-système doit non seulement garantir une haute sécurité mais aussi, une rapidité, un coût raisonnable et une implémentation facile.

2.14.2.12 Test de NIST

NIST ((National Institute of Standards and Technology) est un ensemble de quinze tests utilisés pour évaluer l'aspect aléatoire chez les générateurs de nombre aléatoire (RNG) et pseudo-aléatoires (PRNG), il peut être utilisé pour de multiples fins, y compris la cryptographie [23]. plus que les pixels dans l'image cryptée sont en désordre plus que l'image est méconnaissable, d'où le crypto-système est sécurisé. On applique ce test sur l'image cryptée pour tester d'une manière profonde (en termes de bit) le comportement chaotique de ses pixels constituants. Tous ces tests calculent la valeur de p P_{value}, si pour chaque test, cette valeur est supérieure à 0.01, on dit que la séquence testée est aléatoire, sinon elle n'est pas aléatoire. Pour faire usage de ces tests. NIST propose un logiciel de plusieurs versions, comme on peut reprogrammer chaque test seul, car leurs algorithmes sont open source. Les détails ont été exposés dans le premier chapitre.

2.14.2.13 Problèmes de transmission

Un message durant sa transmission peut confronter plusieurs obstacles, comme la perte de données (à cause d'un vol ou d'une attaque active, ou autre), ou l'immersion dans un bruit (bruit blanc, de fond...). Un bon système de cryptage doit être robuste, et il doit surmonter tous les sorts d'interception volontaire ou involontaire. Pour tester sa robustesse, on peut appliquer une perte partielle sur l'image cryptée (figure 2.6), ou ajouter un pourcentage de bruit (gaussien, sel et poivre,...). Ensuite, on décrypte l'image. Si l'image décryptée est encore reconnaissable, le crypto-système est solide et donc l'algorithme proposé peut résister aux pertes, ou aux bruits potentiels.

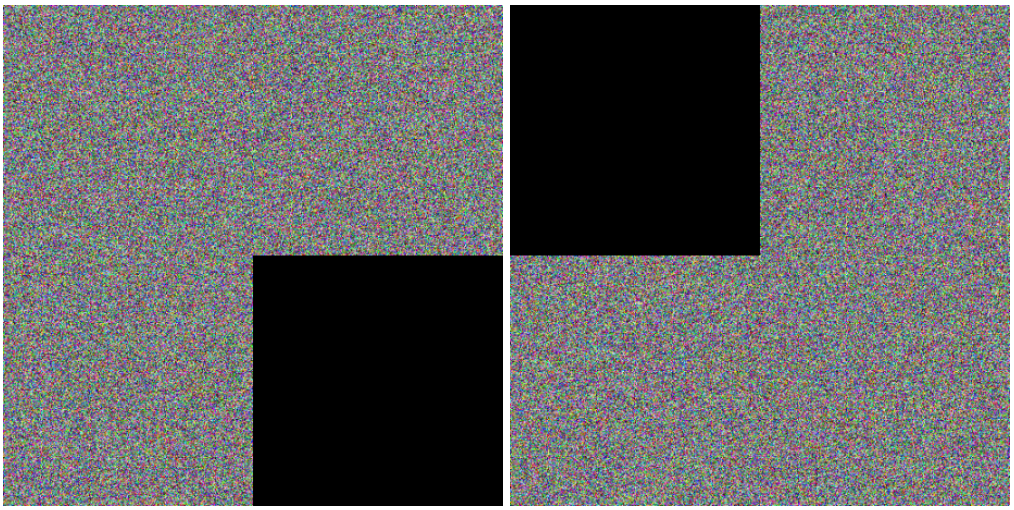


FIGURE 2.6 – Exemple d'une image avec une perte partielle d'un quart dans deux endroits différents.

2.14.3 Méthodes de cryptage d'image

La cryptographie joue un grand rôle dans la protection des images partagées et la sûreté de sa transmission, étant donné le nombre astronomique des internautes, de nombreux algorithmes ont été proposés, entre autres : ceux à base du chaos, de technique d'ADN, de réseaux de neurones.... Les sous-sections à venir décrivent brièvement diverses méthodes de cryptage.

2.14.3.1 Cryptage à base du chaos

On peut affirmer que l'utilisation du chaos est devenue très populaire dans les algorithmes de cryptage en général, et le cryptage d'images précisément. Vu leurs multiples avantages, en termes de la haute sensibilité aux conditions initiales, et le comportement aléatoire. Ces avantages ont été exploités par plusieurs chercheurs en cryptographie, comme la célèbre fonction chaotique "logistique", elle a été utilisée par Wang et al en [73]. l'idée principale de ce travail est de faire la combinaison entre un système chaotique et la technologie de décalage cyclique, chaque étape dans sa méthode, influence sur la prochaine étape, et ceci à travers l'exploitation de la valeur initiale issue de l'image brouillée, ce qui a pour effet d'augmenter la sensibilité des images claires du schéma. [74] Herbadji et al ont amélioré les performances de l'un la carte logistique. Leur amélioration consiste en un prolongement dans l'intervalle du paramètre de contrôle et dans la valeur maximale de Lyapunov, ce qui a augmenté le désordre dans les séquences générées. Puis, ils l'ont intégré dans un algorithme de cryptage d'image en niveau de gris. Malgré la simplicité de leur algorithme, il peut résister à plusieurs types d'attaques.

La mise en pratique des cartes chaotiques existantes dans le cryptage et même de les développer n'empêche pas la mise en place de nouvelles cartes chaotiques, comme il a fait Mansouri dans son article [75], dans lequel il a introduit une fonction chaotique inédite à base de cosinus et la fonction logarithmique, sa définition mathématique est la suivante :

$$x_{n+1} = \left| \cos\left(\frac{\alpha \pi g\left(x_i, \beta\right)}{\log\left(3 - g\left(x_i, \beta\right)\right)} \right) \right|. \tag{2.19}$$

α est le paramètre de contrôle, g pourra être la carte logistique, Sine ou Tent. Elle a été exploitée pour générer les clés secrètes ainsi que dans les deux processus de confusion/diffusion proposés par Mansouri. Les résultats obtenus présentent un niveau d'aléatoire plus satisfaisant et une capacité à résister aux attaques différentielles. Sur la même voie, M. Z. Talhaoui, X. Wang et M. Midoun ont utilisé une nouvelle carte chaotique unidimensionnelle et fractionnaire en cosinus (1-DCF) dans

un nouveau système de cryptage d'image, cette fonction est définie comme suit [76] :

$$\begin{cases} f : I \to I \\ x_{n+1} = f\left(x_n\right) = \cos\left(\dfrac{\alpha}{x_n^{\beta}}\right),\ \alpha > 0,\ \beta \in \mathbb{N}^* \end{cases} \tag{2.20}$$

L'intervalle I est égal à $[-1, 0[\cup]0, 1]$. Plusieurs tests d'analyse de comportement démontrent que la carte proposée possède un caractère fortement chaotique, une grande plage chaotique de paramètre de contrôle, et une large sensibilité aux conditions initiales. Pour le schéma de cryptage suggéré, ils ont adopté une architecture sans permutation pour augmenter la rapidité de son exécution. Ces travaux approuvent la mise en œuvre de la théorie du chaos dans la cryptographie.

2.14.3.2 Cryptage à base de l'hyper chaos

On dit qu'un système est hyper chaotique, s'il a plus qu'un seul positif exposant de Lyapunov , et plus que deux variables. De nombreux articles ont été publiés sur l'exploitation de l'hyper chaos dans la génération de séquences aléatoires, surtout pour la sécurité et le cryptage des images, en raison de sa grande complexité. En 2016 [77] Zhengchao Ni, Xuejing Kang, Lei Wang ont proposé une méthode à base de deux fameux hyper chaotiques systèmes, celui de lorenz et celui de Rossler. Leur principe de base était le suivant : la transformation des valeurs de pixels en binaire de huit bits, l'utilisation de ses deux systèmes chaotiques pour brouiller l'image en question, et l'application de la transformée d'Arlond. Cette méthode montre une faible corrélation entre pixels, une forte résistance contre les attaques par force brute et une faible corrélation entre les deux images. Toujours dans le but de résoudre le problème de la transmission et de la distribution sécurisée des clés complexes dans un système de cryptage, une toute autre empreinte de l'hyper chaos dans le cryptage d'image a été introduite par Yujia Liu et al, basé sur la transformation optique [78], en utilisant le système hyper-chaotique à quatre ailes avec le système de Chen 4D. Ils ont construit deux masques de phase, ces deux masques vont être utilisés par un double codage de phase aléatoire dans le domaine de Fresnel.

Quant aux nouveaux systèmes hyper chaotiques, Li-Hua Gong, Hui-Xin Luo, Rou-Qing Wu, Nan-Run Zhou ont proposé un nouveau système chaotique 4D avec des attracteurs auto-excités ou des attracteurs cachés [79], basé sur le système de Rucklidge [80], il est décrit par :

$$\begin{cases} \dot{x} = -2x + 10y - yz \\ \dot{y} = x + 0.1w^2 + e \\ \dot{z} = y^2 - z \\ \dot{w} = 0.1y \end{cases} \tag{2.21}$$

Le système 2.21 donne l'attracteur caché si $e \geq 0$, et génère l'attracteur auto-excité si $e < 0$, leurs performances ont été testées par un circuit électronique. Ensuite, il a été utilisé dans un algorithme de cryptage d'image à base de la méthode SHA-256 pour générer des nombres aléatoires, utilisables dans les transformées d'Arnold et l'opération XOR. Il est démontré que l'algorithme de cryptage d'images qui convient aux images grises peut résister aux attaques statistiques et différentielles. Dans [81], un autre chiffrement d'images à base d'un système chaotique de sept dimensions a été proposé. Ce système est le fruit d'une composition efficace entre, d'une part, un système chaotique 6D, d'autre part, un autre système chaotique linéaire et unidimensionnel. Sa formule est la suivante [82] :

$$\begin{cases} u_1^{'} = s\left(u_2 - u_1\right) + u_4 + ru_6 \\ u_2^{'} = pu_1 - u_2 - u_1u_3 + u_5 \\ u_3^{'} = -tu_3 + u_1u_2 \\ u_4^{'} = eu_4 - u_1u_3 \\ u_5^{'} = -iu_2 + u_6 \\ u_6^{'} = q_1u_1 + q_2u_2 \\ u_7^{'} = gu_7 + nu_4 \end{cases} \tag{2.22}$$

Le crypto-système proposé dans [83] utilise l'équation 2.22 pour générer les clés, étant donné que ce système a dix paramètres de contrôle et 7 valeurs initiales. Ainsi, L'évolution différentielle minimax est en effet appliquée pour obtenir les paramètres optimaux de la carte chaotique. D'où, une image cryptée optimale.

2.14.3.3 Cryptage avec la technique d'ADN

Immergeant dans le domaine de la bio-informatique, les chercheurs en cryptographie ont exploités les caractéristiques de l'ADN pour crypter leurs messages envoyés. Au premier abord, ça pourrait être bizarre, mais Le nombre de travaux fructueux réalisés à cet égard démontre la possibilité [84, 85].

Ces mécanismes de cryptage basés sur l'ADN se fondent sur deux étapes : le codage et la

confusion/diffusion. Les règles de codage et de décodage de la séquence d'ADN sont présentées dans [86].

TABLEAU 2.3 – Règles de codage et de décodage de la séquence d'ADN.

RULE	1	2	3	4	5	6	7	8
00	A	A	T	T	G	G	C	C
01	G	C	G	C	A	T	A	T
10	C	G	C	G	T	A	T	A
11	T	T	A	A	C	C	G	G

Plusieurs schémas de cryptage d'images basés sur l'ADN ont été proposés, parmi eux, le schéma proposé par Shadi Yoosefian Dezfuli Nezhad, Naser Safdarian, Seyed Ali Hoseini Zadeh, pour crypter les images des empreintes digitales. Son principe de fonctionnement est de crypter l'image originale et les sorties d'un système chaotique (ici ils ont utilisé la carte Tent) indépendamment, ensuit, l'application de l'opération xor (tableau 2.4 [87]) entre eux, pour avoir l'image cryptée. Les résultats obtenus sont satisfaisants surtout contre les attaques communes.

TABLEAU 2.4 – Opérations logiques XOR pour les séquences d'ADN.

XOR	A	T	C	G
A	A	T	C	G
T	T	A	G	C
C	C	G	A	T
G	G	C	T	A

Dans [88], Zhang S et Gao T ont utilisé une autre procédure de haute sécurité, une méthode de cryptage dynamique de l'ADN, dont les règles de codage/décodage de chaque pixel changent en fonction de l'image à crypter et les clés utilisées. Pour améliorer l'aspect aléatoire et pallier aux ses failles, un autre système chaotique CML basé sur la carte PWLCM a été proposé, les valeurs initiales sont calculées en combinant une valeur de hachage de 64 bits générée par SHA-256 avec des clés externes. Deux étapes de diffusion ont été appliquées au niveau de pixel, tout en utilisant la matrice ADN. Son schéma est illustré ci-dessous :

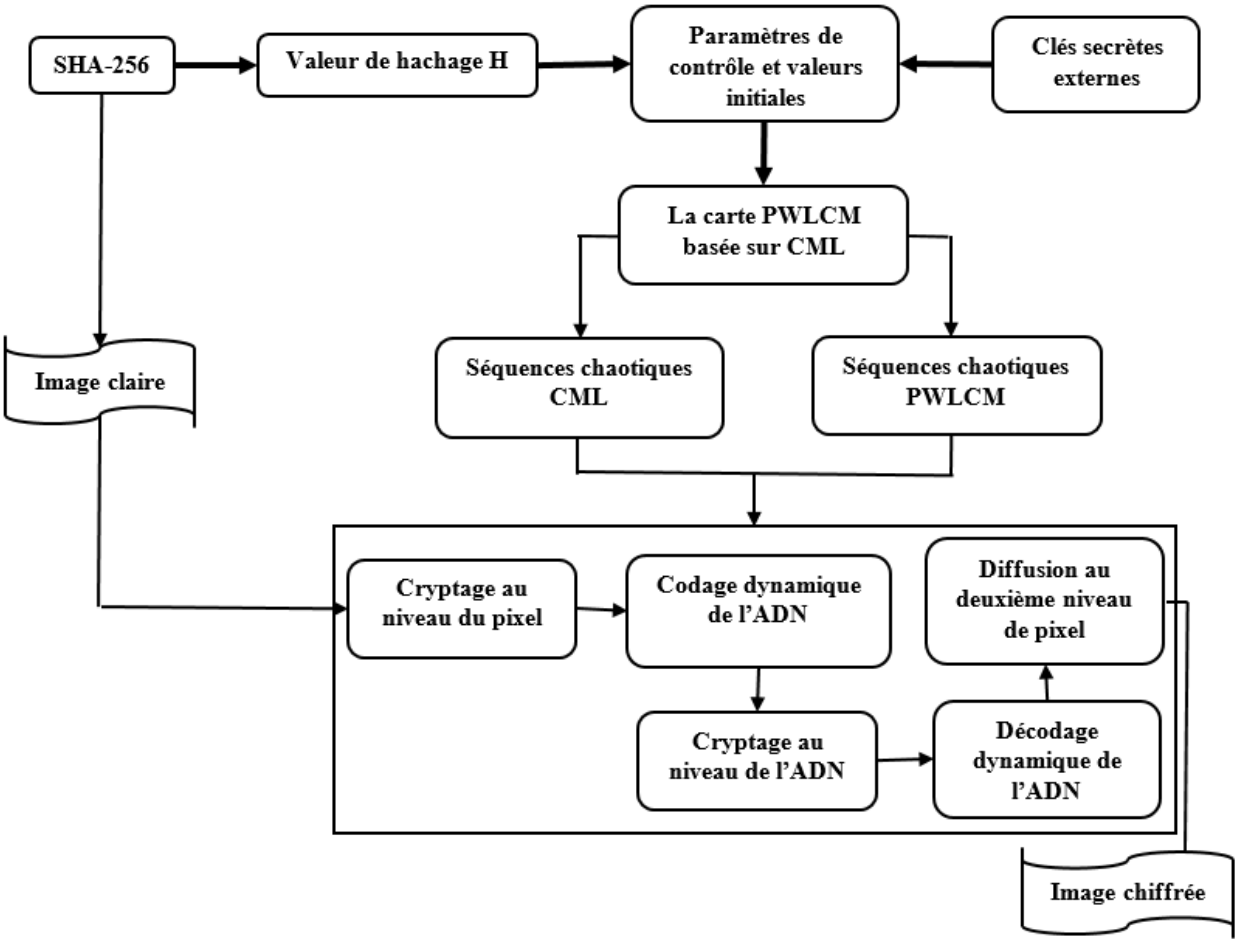


FIGURE 2.7 – Organigramme de l'algorithme proposé par Zhang S et al

Un nouvel algorithme combinant le cryptage chaotique et le cryptage ADN est proposé en [89]. Il se sert d'une clé externe et d'une clé interne, la clé interne vient de l'image originale tandis que la clé externe tiendra le rôle de paramètre d'un générateur chaotique. Ce travail utilise ces paramètres pour générer de multiples séquences chaotiques, qui se mélangent pour produire les séquences de substitution. De plus, les séquences de permutation changent en fonction de phase de cryptage. Par la suite, la phase de cryptage par ADN, où on utilisera les 24 règles de codage de l'ADN, ainsi que les 16 opérations de jointure de l'ADN proposées dans [90], dans le but de permuter l'image résultante ligne par ligne. Cet algorithme est très sensible à l'image brute et à une légère modification dans la clé de cryptage.

2.14.3.4 Cryptage à base des transformées

Afin de remédier au problème de la linéarité des systèmes DRPE, bekkouche et al ont proposé une technique MPDFRFT-DRPE (Multiple Parameter discret Fractional Fourier Transform-Double Random Phase Encoding) de cryptage basé sur un nouveau pré-traitement non linéaire récursif, le pré-traitement s'effectue dans le domaine spatial, dont il utilise le couple substitution-

diffusion, la carte chaotique PLCM et l'opération XOR. Cette étape se répète plusieurs fois. Ensuite, le résultat va être utilisé à l'entrée de FrFT- ou MPDFrFT-DRPE. La technique DRPE (Double Random Phase Encoding) s'effectue dans le domaine spatial et fréquentiel ensemble, figure 2.8 est plus parlante. Le but de l'ajout de cette technique est de renforcer la sécurité de leur système de cryptage.

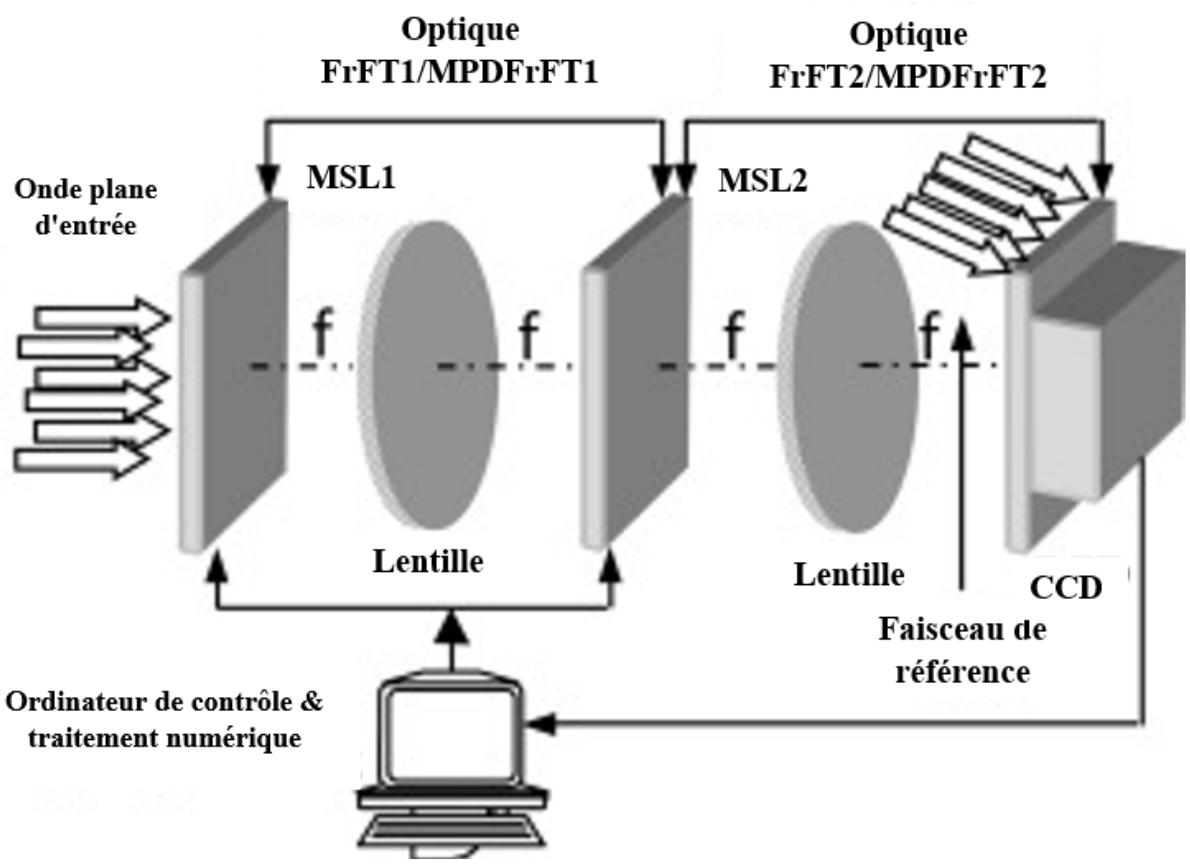


FIGURE 2.8 – Implémentation opto-digitale d'un système de cryptage à base de DFRFTDRPE/ MPDFRFT-DRPE.

SLM1 et SLM2 sont les modulateurs spatiaux de la lumière utilisés pour afficher le signal à valeur complexe pendant les étapes de cryptage/décryptage. Tandis que La caméra CCD est utilisée pour enregistrer numériquement le signal à valeur complexe [91]. Dans un autre travail [92], le schéma de cryptage d'image optique proposé est toujours à base de DRPE, mais cette fois-ci, le pré-cryptage non linéaire d'image est couplé à une conversion réel-complexe (R2C). Le principe de cette dernière est de diviser l'image sur deux parties égales, partie imaginaire et partie réel, puis on les ré-fusionne pour avoir l'image complexe. Pour obtenir l'image chiffrée, il consiste à multiplier les pixels de l'image par un masque de phase aléatoire $exp(j\phi)$, puis à transformer le résultat obtenu par la transformée de Fourier fractionnaire discrète (TFDF). Les résultats montrent que les dimensions de l'image cryptée sont réduites de moitié, et que la méthode proposée est plus efficace que les méthodes existantes en termes de sensibilité et d'espace de clé secrète. Une autre approche

de cryptage d'image [93] assure la sécurité des données de trois images en couleurs multiples dans le domaine temporel, le domaine fréquentiel et le domaine des coordonnées. Se servant de la carte chaotique Baker et 2D MPFrDFT. Ce schéma suggeré comprend trois principaux points : la substitution des pixels dans le domaine spatial à l'aide de la carte chaotique Baker, la MPFrDFT 2D dans le domaine de la transformation, puis la permutation à l'aide de l'AT 3D. Ce crypto système présente une robustesse et une transmission de haute sécurité de plusieurs images par un seul algorithme. Un nouveau schéma de cryptage d'images en couleur est présenté dans [94]. Un mélange de couleurs de pixels est utilisé pour cacher l'information que chaque pixel contient. La sortie va passer vers le domaine fréquentiel via RPMPFRFT (Reality-Preserving Multiple-Parameter Fractional Fourier Transform). Par la suite, un brouillage a été appliqué par un système chaotique 3D pour avoir l'image cryptée, le processus est bien clair sur la figure 2.9.

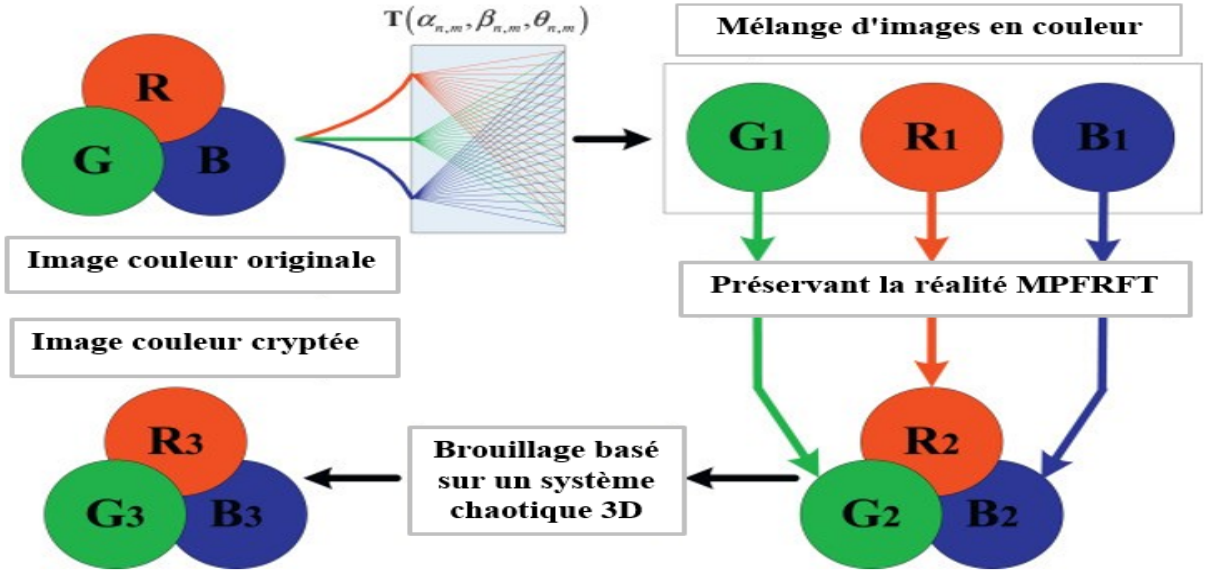


FIGURE 2.9 – Schéma proposé de cryptage d'image.

Cette méthode garantit la sécurité des images transmisses et améliore sa robustesse contre plusieurs types d'attaques.

2.14.3.5 Cryptage à base des réseaux de neutrons

La méthode proposée par K Ratnavelu et al en [95] s'articule sur FCNN (Fuzzy Cellular Neural Network) modifié, l'idée est d'intégrer le flou dans le paradigme CNN, de sorte que chaque signal chaotique généré par FCNN est combiné avec chaque pixel de l'image à crypter, ça permet d'obtenir un pixel crypté indépendant du pixel de l'image originale. Les résultats de multitude de tests, montrent une grande efficacité de cette méthode. Les avantages apportés par le couplage systèmes chaotiques-réseaux neuronaux avaient fait de Uğur Erkan al [96] proposer une technique

de cryptage d'image à base de cette combinaison, le modèle proposé dépend d'une nouvelle suite chaotique la carte log (équation 2.23) pour générer une série de valeurs chaotiques, pour les utiliser dans l'un des processus d'un système de cryptage (telles que la permutation, le codage ADN, la diffusion et la réversion de bits), un réseau neuronal convolutionnel est certes intégré pour produire la clé publique. Les résultats absolus et comparatifs démontrent la réussite de ce schéma.

$$v_{i+1} = mod\left((u+e)\, ln\, v_i, 1\right),\ v_i \in (0,1)\,. \tag{2.23}$$

$u \in [0, \infty)$ est le paramètre de contrôle, v_i la valeur initiale.

2.14.3.6 Cryptage à base des courbes elliptiques

Les schémas de cryptage d'image peuvent utiliser les courbes elliptiques à plusieurs fins, telles que : (a) la génération de nombres pseudo-aléatoires (PRNG), (b) pour l'échange de clés, et le cryptage de pixels. Le système de cryptage proposé par [97] utilise les courbes elliptiques pour empêcher la dépendance séquentielle des systèmes chaotiques traditionnels. Dans lequel, les additions de points récurrents sont remplacées par des additions parallèles, dans le but d'élever la vitesse de cryptage/décryptage, et de profiter aussi des capacités de traitement parallèle, qu'est aujourd'hui disponible sur la plupart des plateformes informatiques. Les résultats obtenus prouvent que l'objectif a été atteint. Dans [98], un schéma de cryptage d'image et d'authentification à clé publique a été proposé, à base du système chaotique Cat 3D et 4D, le chiffre ElGamal amélioré, et les courbes elliptiques, spécifiquement la méthode d'échange de clés ECDH (Diffie-Hellman à courbes elliptiques). Cette méthode garantit un échange sécurisé de la clé partagée entre deux utilisateurs. Son principe est comme suit :

— L'utilisateur X génère sa clé secrète aléatoire, n_x. Ensuite, X calcule la clé publique P_x en utilisant sa clé privée n_x et le point générateur G sur la courbe, comme suit :

$$P_x = n_x * G. \tag{2.24}$$

Et il l'envoie à Y.

— L'utilisateur Y génère sa clé secrète aléatoire, n_y. Ensuite, Y calcule la clé publique P_y en utilisant sa clé privée n_y et le point générateur G sur la courbe, comme suit :

$$P_y = n_y * G. \tag{2.25}$$

Et il l'envoie à X.

— Les deux utilisateurs X et Y partagent la même clé ECDH, notée par K_{xy} dans l'équation suivante :

X compte K_{xy} comme suit :

$$K_{xy} = n_x * P_y = n_x * n_y * G. \tag{2.26}$$

Y compte K_{xy} comme suit :

$$K_{xy} = n_y * P_x = n_y * n_x * G. \tag{2.27}$$

Le modèle proposé a montré sa robustesse, sa compétence, et sa résistance contre les attaques statistiques et crypt-analytiques. Un autre travail suggéré par M. Ramzan et al dans [99]. Toujours dans le cadre de l'exploitation des courbes elliptiques, les auteurs s'en servent pour concevoir un algorithme de cryptage d'images plus fort et plus rapide, également, pour garantir la confidentialité de toute donnée numérique pendant la transmission en ligne. Cet algorithme utilise les boites de substitution à base de EC avec de bonnes propriétés cryptographiques. Il utilise un des s-box obtenus pour substituer une matrice brouillée de taille 16×16 de l'image. Ensuite, un autre s-box obtenu est intervenu pour soumettre à une opération d'addition avec la matrice substituée modulo 256.

2.15 Conclusion

Le besoin de protéger nos données transmises est en constante augmentation. De nombreux chercheurs ont proposé des algorithmes cryptographiques pour surmonter les risques qui peuvent intercepter nos communications sur des supports non sécurisés. Dans ce chapitre, nous avons vu l'histoire de la cryptographie qui est un ancien art en progression permanente, et les différents concepts liés à cette discipline. Nous avons présenté aussi la branche cryptage d'image, vu le nombre important des images qui circulent en ligne, et la différence entre le cryptage d'un texte et le cryptage d'une image, les multiples mesures d'évaluation d'un système de cryptage, et la multitude de schémas de cryptage d'image proposés. Tout ça va nous aider à mieux maitriser le sujet, et à bien comprendre nos deux contributions introduites dans les prochains chapitres.

Chapitre 3

Schéma de Cryptage d'Images en Couleurs Basé sur une Carte Cubique 1D

Sommaire

3.1 Introduction
3.2 Structure de la carte chaotique proposée
3.3 Test de performance
3.4 Système de cryptage proposé
3.5 Résultats de la simulation et analyse de la sécurité
3.6 Conclusion

Chapitre 3

Schéma de Cryptage d'Images en Couleurs Basé sur une Carte Cubique 1D

3.1 Introduction

L'introduction du chaos dans les systèmes de cryptage a permis de réaliser un saut considérable dans leur efficacité, en raison de leurs propriétés favorables à la communication sécurisée, telles que l'ergodicité, la sensibilité aux conditions initiales et aux paramètres de contrôle. Cependant, les cartes chaotiques ont quelques handicaps, comme le comportement chaotique seulement pour certaines valeurs du paramètre de contrôle, ce qui peut rendre les systèmes de cryptage vulnérables, et cela a conduit les chercheurs à développer des nouveaux systèmes chaotiques efficaces, ou à améliorer le comportement de certains d'entre eux, puis les intégrer dans des crypto-systèmes. De la même façon, nous introduisons un nouveau système chaotique modulaire unidimensionnel inspiré de la carte classique cubique, où nous remédions aux défauts de cette dernière. En outre, nous étudierons ses applications dans un nouveau système de cryptage d'images, celui-ci utilise le modèle de confusion-diffusion, dans lequel le changement des valeurs des pixels sera aléatoire. Cette technique permet d'obtenir une version cryptée unique à chaque fois qu'on crypte, et le rend imprévisible et sensible à toute modification infinitésimale, dans le but de minimiser tous les types de cryptanalyse et d'augmenter le niveau de sécurité. Les excellents résultats ont été démontrés à l'aide de plusieurs tests graphiques, tels que les histogrammes, et numériques, tels que le test de NIST. La structure de ce chapitre est la suivante : La deuxième partie présente la carte chaotique améliorée et son évaluation. La troisième partie donne les détails du schéma de cryptage proposé. Les résultats expérimentaux sont présentés dans la quatrième partie. Nous terminerons par une conclusion.

3.2 Structure de la carte chaotique proposée

Dans le cadre d'étude de bifurcation dans les systèmes écologiques, Robert M May a introduit au monde un exemple canonique d'une carte à deux points critiques, c'est la carte cubique [100] :

$$\begin{cases} f \ : \ ]-1,1[\rightarrow [-1,1[\\ x_{n+1} = f\left(x_n\right) = r x_n^3 + \left(1-r\right) x_n \end{cases} \tag{3.1}$$

Où r est le paramètre de contrôle, n est le nombre d'itérations, et x est la variable d'état. Cet équation a un comportement non trivial pour $r \in]0,4[$, dont les valeurs plausibles de x pour cet intervalle de r varient entre -1 et 1. Tableau 3.1 montre le comportement de l'équation 3.1 pour chaque intervalle de r. L'équation prouve un comportement chaotique uniquement dans l'intervalle $]4, 1+\sqrt{5}[$.

TABLEAU 3.1 – Comportement dynamique de la carte cubique classique.

Valeur de r	Comportement dynamique
$2 > r > 0$	Point stable.
$3 > r > 2$	Cycle stable de la période 2.
$1+\sqrt{5} > r > 3$	Deux cycles distincts, chacun ayant une période de 2.
$4 > r > 1+\sqrt{5}$	La dynamique est apparemment chaotique.

Tableau 3.1 [101] montre le comportement de l'équation 3.1 pour chaque intervalle de r, dont l'équation prouve un comportement chaotique uniquement sur l'intervalle $]4, 1+\sqrt{5}[$.

3.2.1 Carte cubique proposée (1-DCE)

Dans le but d'améliorer le comportement chaotique de la carte cubique classique, nous présentons une nouvelle carte chaotique modulaire unidimensionnelle basée sur la composition de la carte cubique et la fonction exponentielle, nommée 1-DCE. Elle est définie par l'équation suivante :

$$\begin{cases} f \ : \ ]0,1[\rightarrow [0,1[\\ x_{n+1} = f\left(x_n\right) = r e^{{x_n}^3} + \left(1-r\right) e^{x_n} \ mod \ 1 \end{cases} \tag{3.2}$$

La carte 1-DCE présente un comportement chaotique élevé pour quasiment toutes les valeur positive de r, ainsi que la distribution de ses séquences est aléatoire dans $[0,1[$. Par conséquent, la carte 1-DCE est bien adaptée pour satisfaire les besoins cryptographiques, en matière de l'espace

clés, complexité et de sensibilité. Pour cela, dans ce qui suit, nous allons confirmer son efficacité en étudiant ses performances à travers plusieurs tests de systèmes dynamiques.

3.3 Test de performance

3.3.1 Comportement chaotique

Pour visualiser le comportement de notre carte proposée, en comparant avec l'ancienne carte cubique. nous avons tracé la trajectoire de 1-DCE et la carte cubique classique dans l'espace des phases 2D et 3D (figure 3.1), pour environ 3000 itérations, ainsi que le diagramme de toile d'araignée (cob-web) est appliqué directement aux données de la série x(n), pour i=1 à 300.

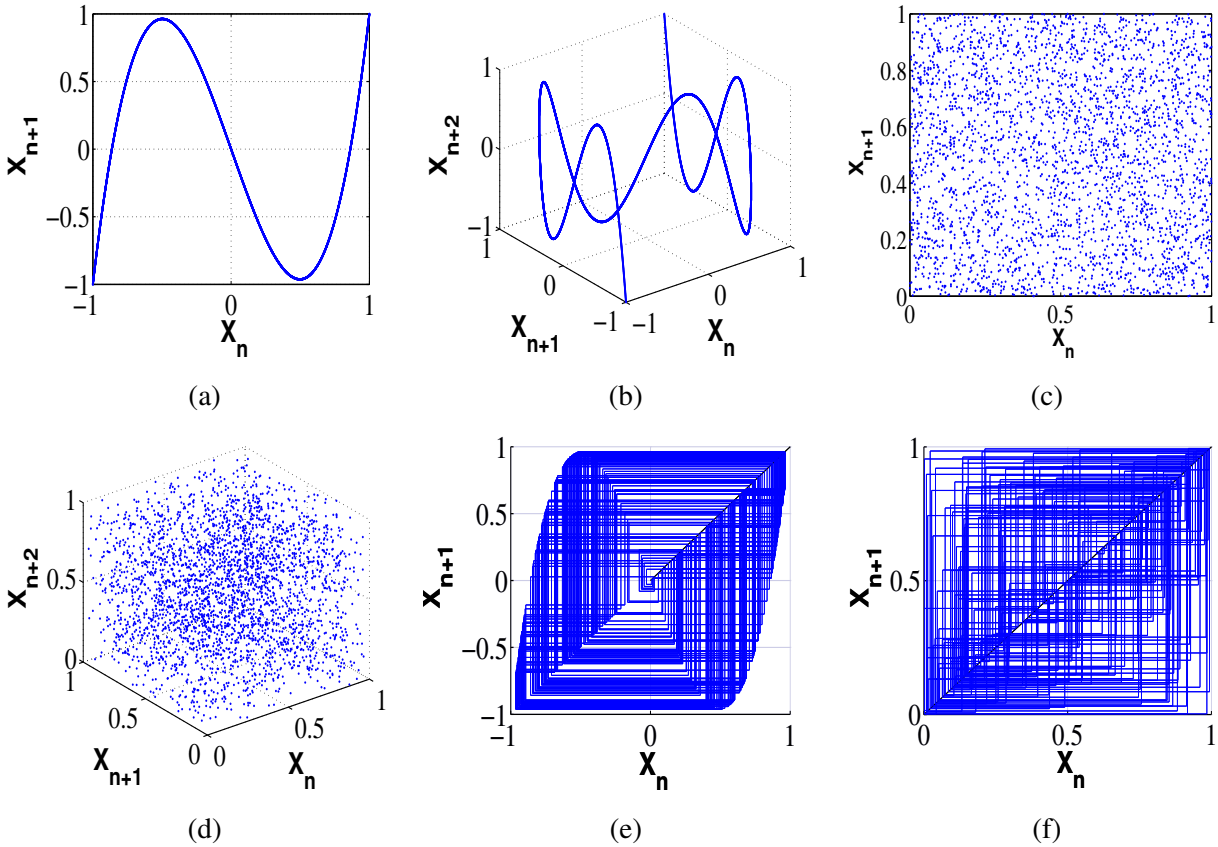


FIGURE 3.1 – Comparaison entre le comportement chaotique et le diagramme Cobweb de la carte cubique classique et 1-DCE. (a), (b) Comportement chaotique de la carte cubique classique. (c), (d) Comportement chaotique de 1-DCE. (e) Diagramme en toile d'araignée de la carte cubique classique. (f) Diagramme en toile d'araignée de 1-DCE.

Dans le diagramme des phases, on peut visualiser les différents états de système, ainsi que leurs transitions. Il peut également illustrer le comportement de système dans les zones ou il peut se trouver. Il ressort du diagramme des phases que la trajectoire de la nouvelle carte a une distribution uniforme et tend à remplir la totalité de l'espace des phases 2D et 3D, tandis que la carte

cubique présente une distribution plus ondulée que désordonnée. Afin de confirmer davantage le comportement chaotique de 1-DCE, on se sert aussi de diagrammes en toile d'araignée pour dessiner le tracé de la carte classique et 1-DCE. Il est constitué d'une ligne diagonale et la courbe de la fonction en question, leur intersection représentent la valeur produite par itérations de la fonction. Si les suites obtenues sont complexes et forment des spirales externes, on dit que cette fonction est chaotique. Comme on peut le voir, 1-DCE que nous avons développé présente une distribution uniforme sur la plage de sortie, contrairement à la carte cubique classique qui ne le fait pas. Il est clair que le système classique oscille sur plusieurs points et que son comportement est chaotique. Mais, l'orbite chaotique ne remplit pas la plage de sortie avec des rectangles. En revanche, l'orbite chaotique de 1-DCE remplit entièrement la plage de sortie, par un nombre infini de trajectoires non répétables formant le plan.

3.3.2 Exposant de Lyapunov

L'exposant de Lyapunov est le principal critère d'évaluation du chaos. Elle représente la mesure de la prévisibilité et de la sensibilité de système aux conditions initiales [102]. Pour les systèmes chaotiques, λ doit être positif au moins une seule fois. De la figure 3.2, nous pouvons remarquer que notre carte proposée a le plus grand exposant positif (λ) pour toutes les valeurs de r ($r \geq 0$) par rapport aux courbes de la carte cubique, 1-DCP [44] et 1-DSP [103], cela est dû à la fonction exponentielle qui la compose, qui ne cesse pas d'augmenter. (L'exposant de Lyapunov maximal MLE est 3.85, quand r=20). De ce qui précède, nous pouvons conclure d'un côté que l'utilisation de LE a confirmé la nature non linéaire de 1-DCE pour toute valeur positive du paramètre de contrôle, d'autre côté, notre carte est très sensible aux conditions initiales.

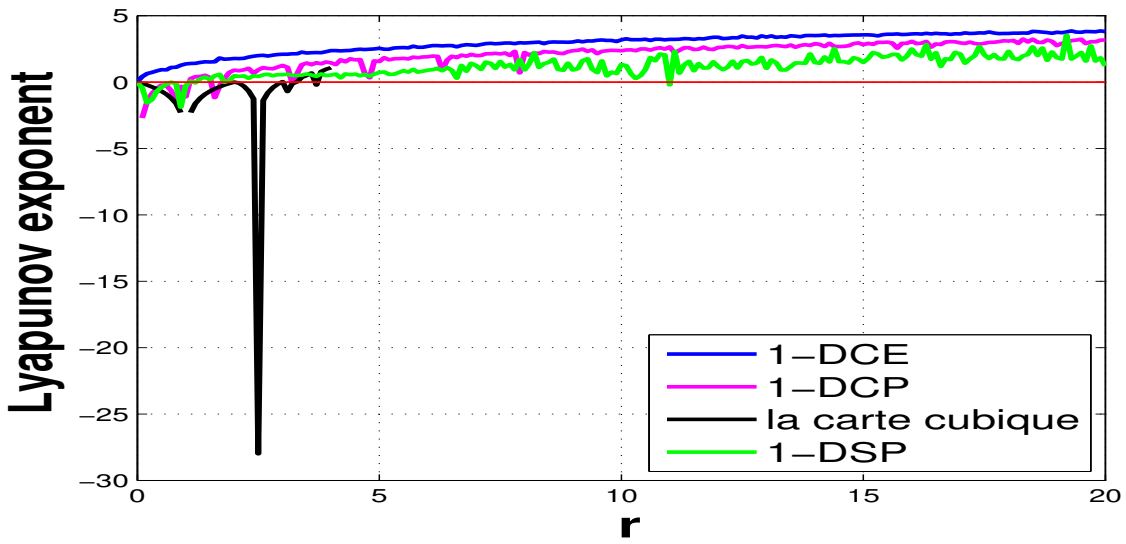
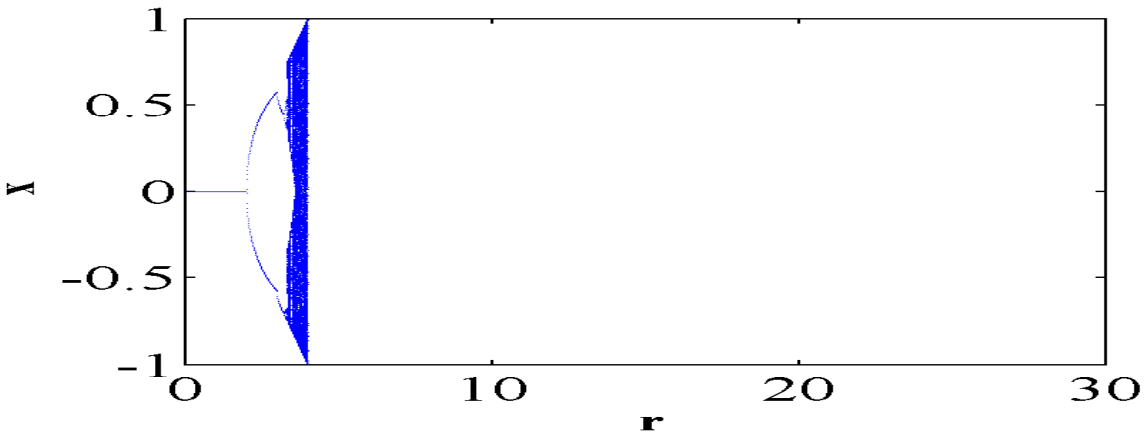


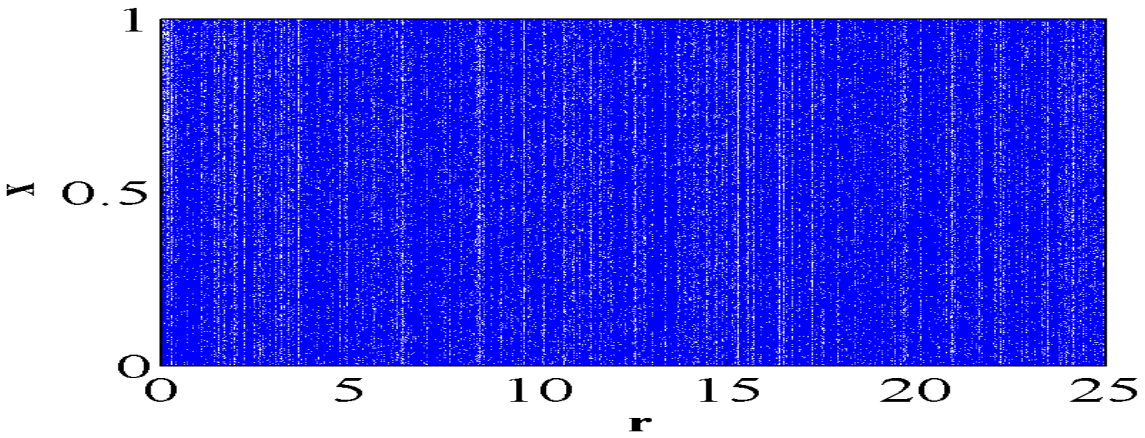
FIGURE 3.2 – Analyse et comparaison des exposants de Lyapunov.

3.3.3 Diagramme de bifurcation

Le diagramme de bifurcation trace l'évolution d'un système dynamique en fonction de paramètre de contrôle, tout en montrant son comportement (stabilité, périodicité, chaos) [104] en fonction de ce paramètre. Comme le montre la figure 3.3, notre carte améliorée couvre une région chaotique infinie sans phases périodiques, alors que la carte cubique classique ne couvre qu'une petite plage de l'intervalle de r, comme nous pouvons clairement remarquer l'émergence de phases périodiques. Par conséquent, le 1-DCE a un bon comportement chaotique pour une valeur infinie du paramètre de contrôle.



(a)

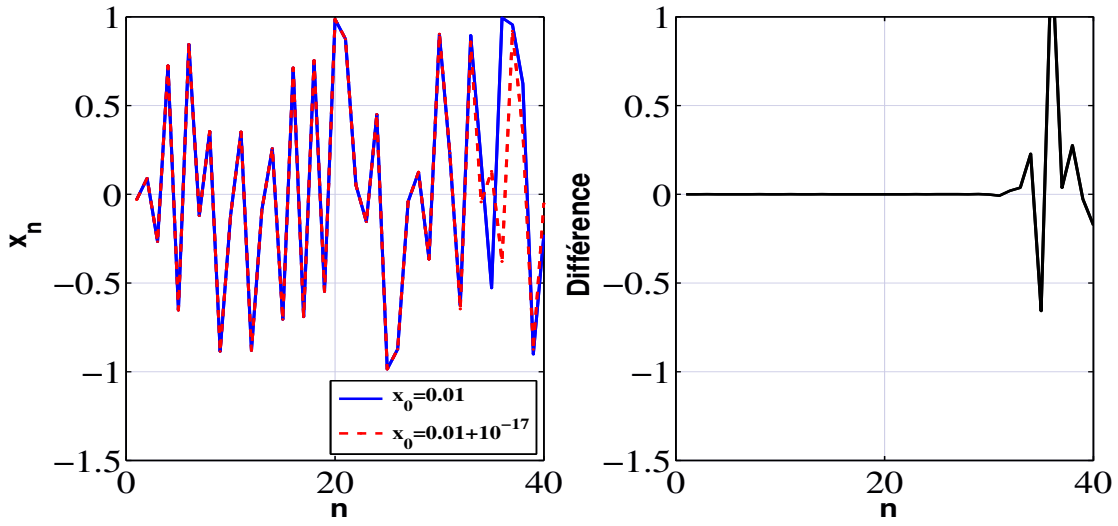


(b)

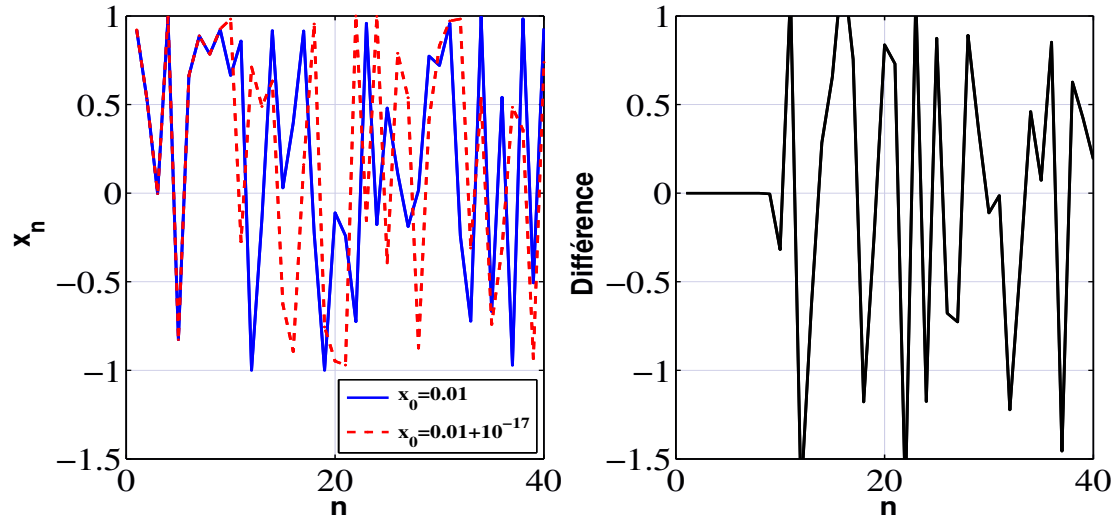
FIGURE 3.3 – Diagramme de bifurcation pour : (a) la carte cubique classique, (b) la carte 1-DCE.

3.3.4 Evaluation de la sensibilité

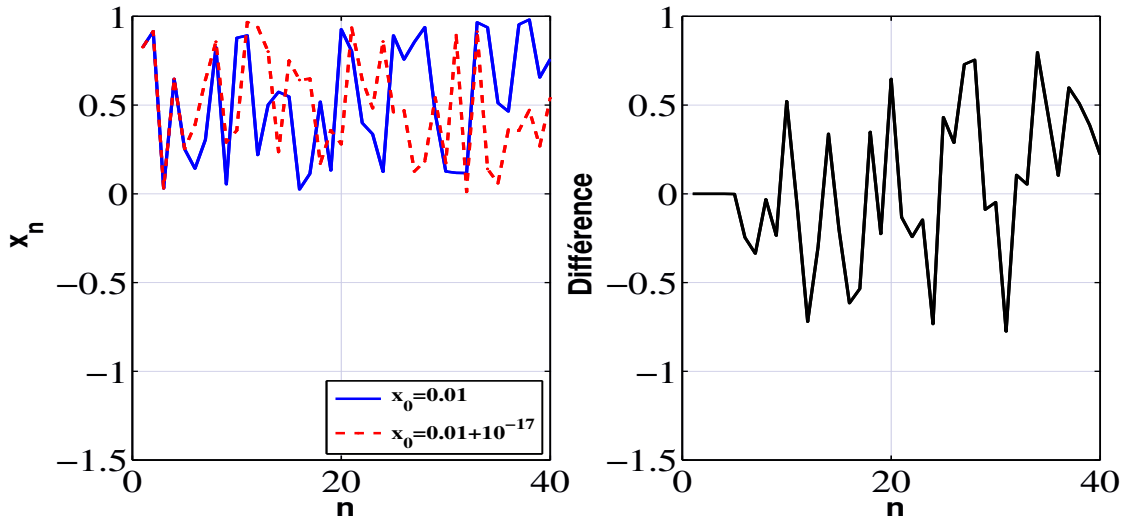
Afin d'évaluer visuellement les résultats du test de sensibilité aux conditions initiales, nous traçons les trajectoires de la carte avec un léger changement dans les conditions initiales et les paramètres de contrôle.



(a)

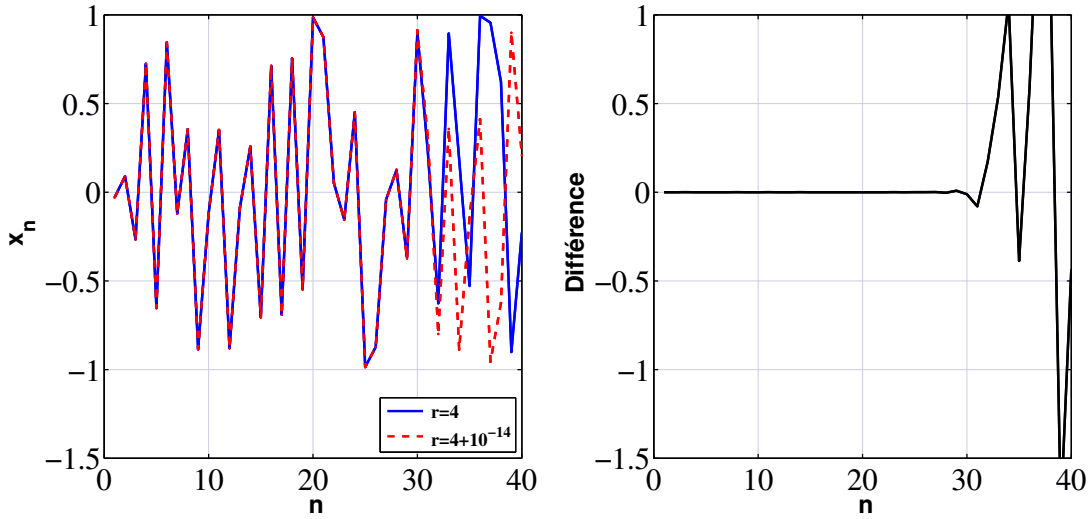


(b)

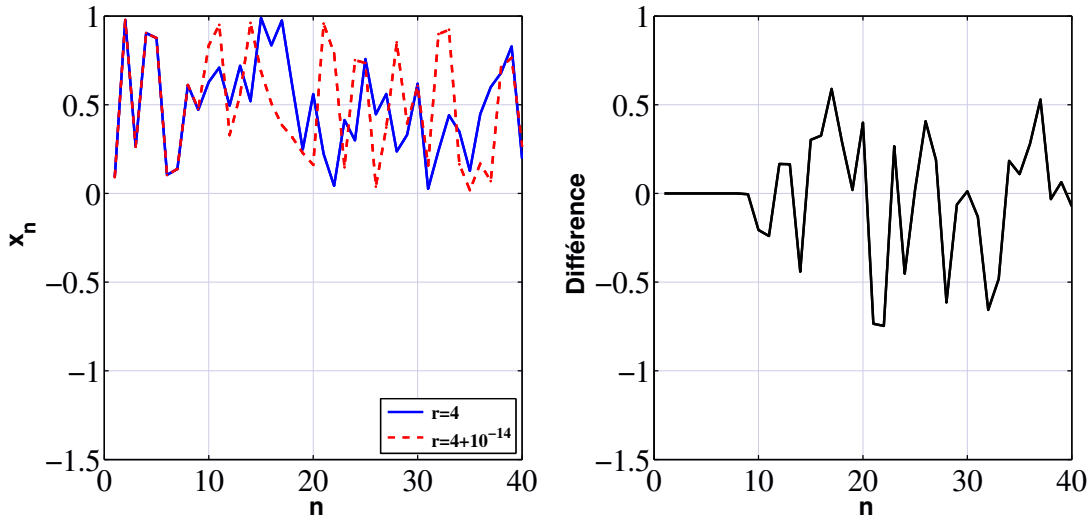


(c)

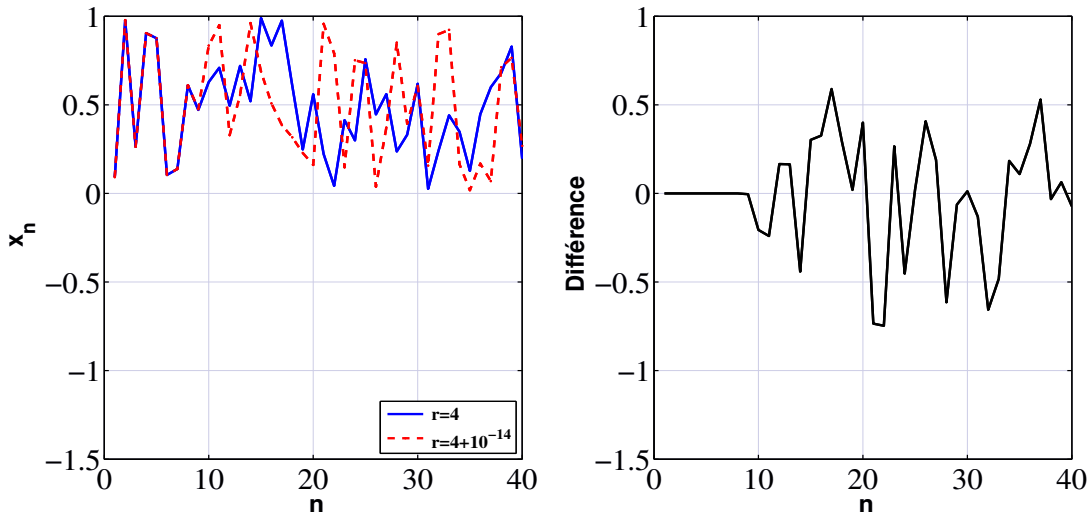
FIGURE 3.4 – Sensibilité des cartes chaotiques susmentionnées aux changements de $10^{(-17)}$ dans la valeur initiale, par rapport au graphique de comparaison entre les séries temporelles avec et sans changement dans le cas de (a) la carte cubique classique, (b) 1-DCP, (c) 1-DCE.



(a)



(b)



(c)

FIGURE 3.5 – Sensibilité des cartes chaotiques susmentionnées aux changements de $10^{(-14)}$ dans r, par rapport au graphique de comparaison entre les séries temporelles avec et sans changement dans le cas de (a) la carte cubique classique, (b) 1-DCP, (c) 1-DCE.

Comme illustré sur la figure 3.4, le comportement de 1-DCE après un changement de $10^{(-17)}$ dans les conditions initiales diverge après 4 itérations, tandis que la carte cubique et 1-DCP divergent après 34 et 8 itérations respectivement. Le diagramme à droite représente le traçage de comparaison entre les deux trajectoires. Il en va de même pour le paramètre de contrôle, ici nous faisons un changement de $10^{(-14)}$ dans le paramètre de contrôle r, et nous traçons les trajectoires avec et sans changement, en plus de l'écart entre leurs trajectoires. Nous pouvons remarquer clairement sur la Figure 3.5 que 1-DCE diverge après avoir fait huit itérations, meilleure que 1-DCP et la carte cubique. L'étude ci-dessus indique que la carte que nous proposons a amélioré la sensibilité de la carte cubique. En addition, sa sensibilité est aussi meilleure que celle des autres cartes proposées dans l'état de l'art.

3.3.5 Analyse de tests statistiques

Dans cette thèse, nous essayons d'évaluer le caractère aléatoire de la carte 1-DCE car elle se considère aussi comme un PRNG, en utilisant les tests statistiques Nist [23], Ent [17] et Diehard [105].

TABLEAU 3.2 – Résultats du test Nist de 1-DCE.

Tests	P-value	Résultats
Frequency	0.95	Passed
Block Frequency	0.97	Passed
Runs	1.00	Passed
Longest Runs Of Ones	0.98	Passed
Rank	0.96	Passed
Spectral	1.00	Passed
Non Overlapping Template Matching	1.00	Passed
Overlapping Template Matching	0.87	Passed
Universal	0.93	Passed
Linear Complexity	1.00	Passed
Serial	0.93	Passed
Approximate Entropy	0.97	Passed
Cumulative Sums	0.99	Passed
Random Excursions	1.00	Passed
Random Excursions Variant	1.00	Passed

TABLEAU 3.3 – Résultats du test DIEHARD de 1-DCE.

Tests	P-value	Résultats
BIRTHDAY SPACINGS	0.5523	Passed
THE OVERLAPPING 5-PERMUTATION	0.3269	Passed
BINARY RANK	0.5920	Passed
THE BITSTREAM	0.7431	Passed
OPSO	0.7149	Passed
OQSO	0.5144	Passed
DNA	0.6885	Passed
COUNT-THE-1's TEST on a stream of bytes	0.5224	Passed
COUNT-THE-1's TEST for specific bytes	0.3047	Passed
PARKING LOT	0.1168	Passed
MINIMUM DISTANCE	0.1778	Passed
3D SPHERES	0.1142	Passed
SQEEZE	0.4525	Passed
OVERLAPPING SUMS test	0.8791	Passed
RUNS test	0.0628	Passed
CRAPS test	0.4966	Passed

TABLEAU 3.4 – Suite de tests d'Ent appliquée sur 1-DCE.

Test statistique ENT	Résultats
Entropie	7.998397.
Réduction de la dimension par compression	0%
χ^2 Distribution	<0.01%
Moyenne arithmétique	128.0135 (si =127.5, donc la séquence est aléatoire).
Pi par la méthode de Monte Carlo	3.125816450.
Autocorrélation	-0.000689.

Tous les tests de Nist produisent une $p-value \in [0, 1]$. Lorsque la p-value > 0.01, les séquences testées sont tout à fait aléatoires [106], tandis que la plage acceptable de toutes les p-values dans le cas de Diehard est $[0, 1[$. D'après les tableaux 3.2, 3.3 et 3.4, la nouvelle carte a réussi tous les

tests d'Ent, Nist et de Diehard.

3.4 Système de cryptage proposé

Dans cette section, nous avons proposé un schéma de cryptage/décryptage efficace. Le processus de cryptage est représenté en deux opérations principales, la fameuse architecture de confusion/diffusion. En outre, nous avons exploité une nouvelle technique de diffusion. Cette technique a prouvé son efficacité à travers les valeurs du taux de changement du nombre de pixels (NPCR), de l'intensité moyenne unifiée de changement (UACI), de l'entropie et d'autres tests précieux.

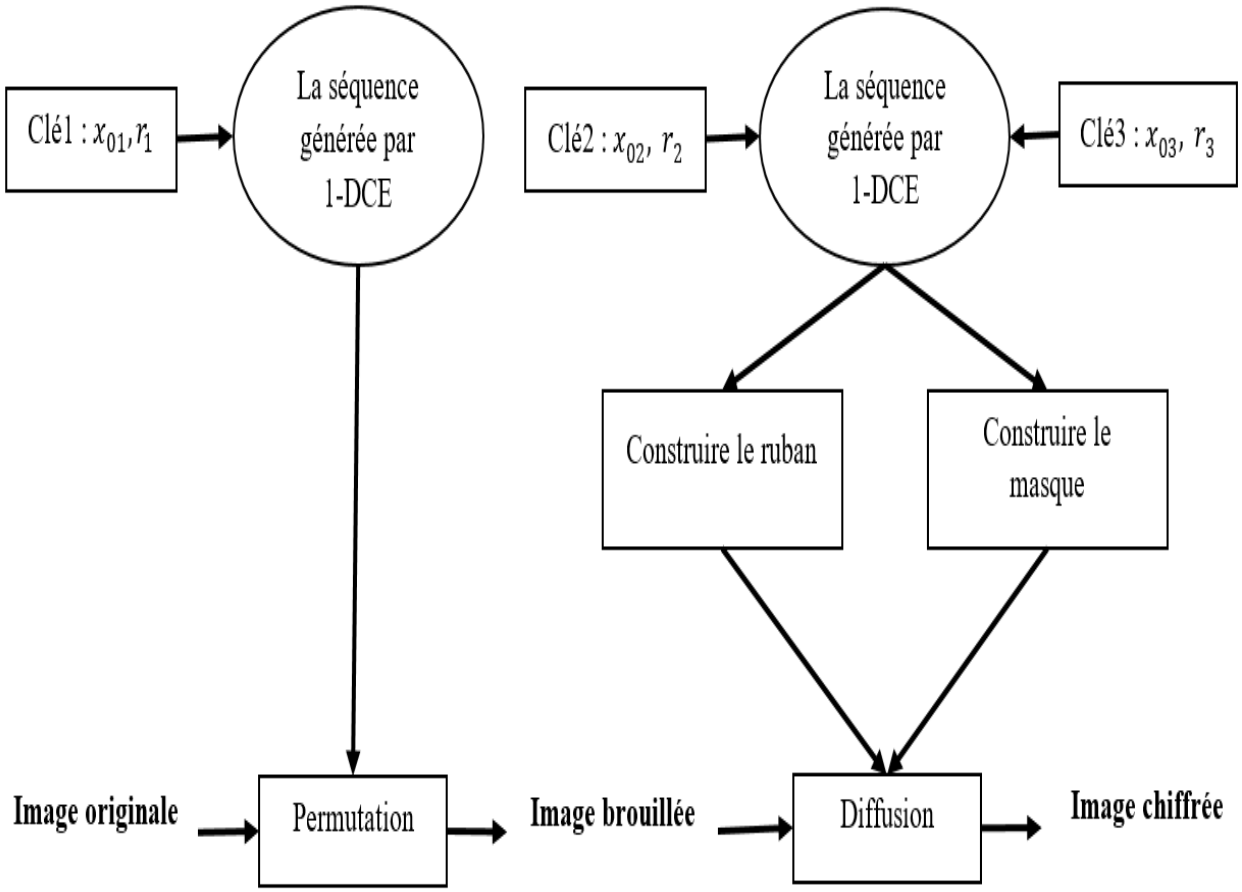


FIGURE 3.6 – Schéma fonctionnel du crypto-système proposé.

3.4.1 Méthode de cryptage

Dans cette présente sous-section, nous allons détailler petit à petit notre mécanisme de cryptage ; une illustration du système de cryptage est présentée sur la figure 3.6, sur laquelle on peut constater que notre schéma proposé repose sur deux processus essentiels, ainsi que les séquences générés par 1-DCE. Les détails sont les suivants :

3.4.1.1 Confusion

Le déroulement de processus de confusion est décrit en détail dans les points ci-dessous :

A. Soit RGB l'image claire de taille $(3 \times m \times n)$ constituée de trois matrices R, G et B.

B. Reformer les trois matrices en trois vecteurs $R^{'}$,$G^{'}$ et $B^{'}$, puis les concaténer afin d'obtenir un vecteur $RGB^{'}(1 \times 3 \times m \times n)$. Dans le cas d'une image en niveau de gris, la matrice est reformée en un seul vecteur $(1 \times m \times n)$ directement.

C. À partir du 1-DCE, nous Réordonnons de manière chaotique les pixels de $RGB^{'}$.

D. Après avoir permuté les positions des pixels, nous transformons le vecteur en un tableau de taille $(m \times 3 \times n)$ ($(m \times n)$ en cas d'images grises), pour l'utiliser dans une nouvelle opération de diffusion.

3.4.1.2 Diffusion

Afin de simplifier la description du processus de diffusion, nous avons présenté un modèle matriciel miniaturisé. Les étapes de la diffusion sont représentées sur la figure 3.7 par une matrice (4×4). Supposons que les pixels en rouge sont les pixels d'un ruban construit, tandis que les pixels en blanc sont les pixels propres de la matrice. La direction du processus est de bas en haut et de haut en bas mutuellement. Les opérations de diffusion sont appliquées régulièrement sur la zone encadrée en bleu.

Remarque : $a_{(i,j)}$ sont les valeurs originales, $b_{(i,j)}$ les secondes valeurs calculées, $c_{(i,j)}$ les troisièmes valeurs calculées, $d_{(i,j)}$les dernières valeurs calculées. Concernant la première et la dernière colonne de la matrice, elles ne subiront que deux opérations. Nous présentons ci-après les étapes détaillées du processus de diffusion :

A. En utilisant 1-DCE, générer un vecteur chaotique $x2$ de longueur $(6 \times n + 2 \times m + 4)$, ce vecteur joue le rôle d'un ruban tout autour de l'image (figure 3.7). En employant l'équation 3.3, les valeurs obtenues sont comprises entre 0 et 255 :

$$Y = Arrondi(10^{15} \times x2 \; mod \; 256). \tag{3.3}$$

Le ruban a pour utilité d'inclure les pixels situés dans les limites de l'image dans le processus de diffusion.

B. Placer le ruban conçu tout autour de l'image $RGB^{'}$ comme illustré sur la figure 3.7. La nouvelle dimension de l'image est (mm×3nn), où $mm = m + 2$; et $nn = n + 2$.

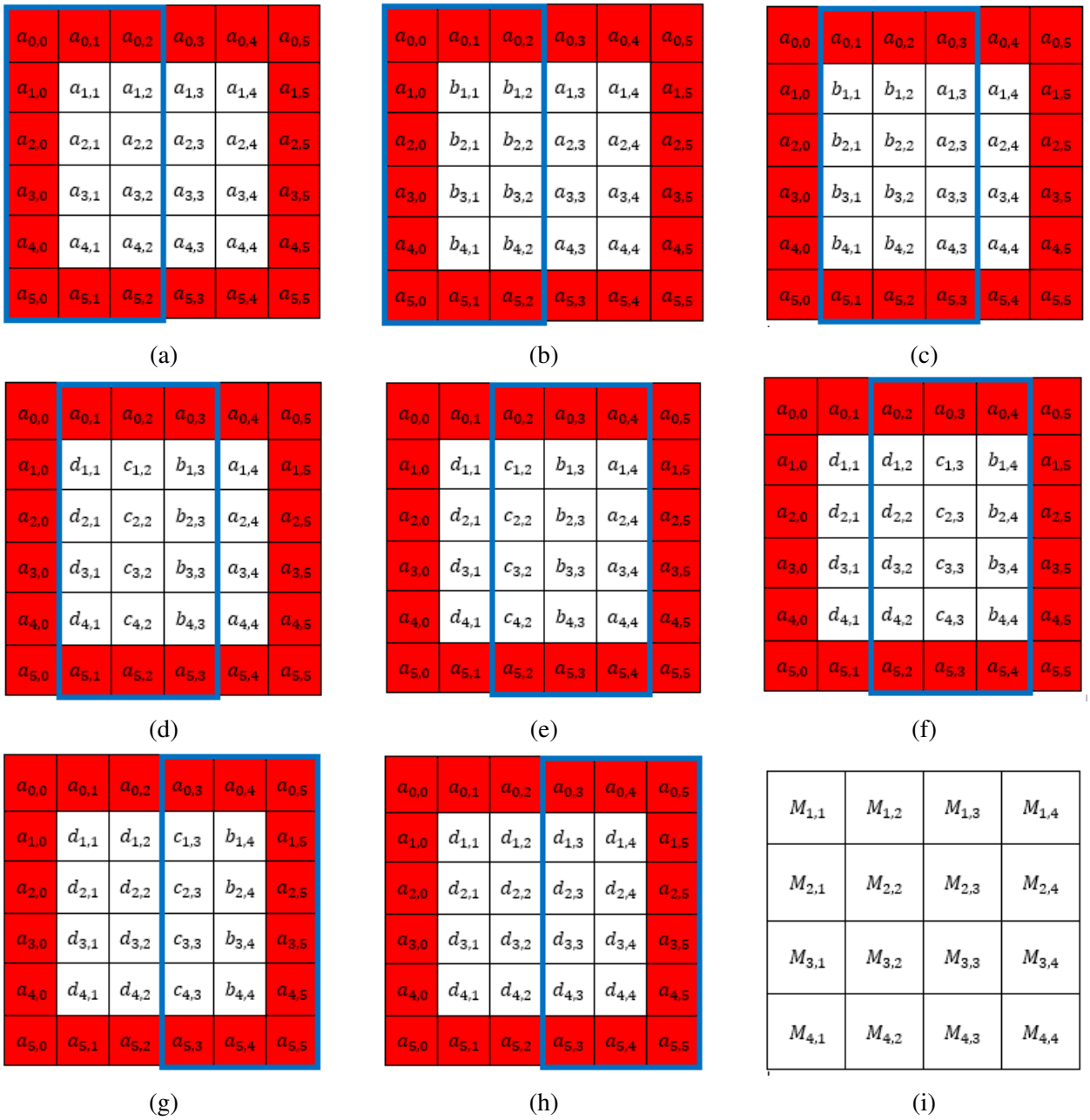


FIGURE 3.7 – Exemple du processus de diffusion proposé pour une matrice de taille 4×4. (a) Matrice encadrée par un ruban. (b) Matrice après l'application de la première opération. (c) Décalage de la zone de calcul d'une seule colonne. (d) Matrice après la seconde opération. (e) Décalage de la zone de calcul d'une seule colonne. (f) Après avoir appliqué les opérations sur la zone encadrée. (g) Un autre décalage. (h) Matrice après avoir été entièrement balayée. (i) Masque.

C. Pour construire le masque, créer une matrice vide avec la même dimension de RGB', ses éléments sont complétés par 1-DCE via la formule 3.4. Les valeurs obtenues sont évidemment comprises entre 0 et 255 :

$$Z(i) = Arrondi(x(i) \times 255). \tag{3.4}$$

D. En se basant toujours sur l'exemple de la figure 3.7, l'opération de diffusion est effectuée

par 3 colonnes. Quant au premier trio (Figure 3.7a) :

$$RGB^{'}{}_{i+1,2} = RGB^{'}{}_{i,2} \oplus RGB^{'}{}_{i+1,2}. \tag{3.5}$$

$$RGB^{'}{}_{i+1,3} = RGB^{'}{}_{i,1} \oplus RGB^{'}{}_{i+1,3}. \tag{3.6}$$

Où i=$\{1, 2, 3, \ldots mm-2\}$.

$$RGB^{'}{}_{i+1,2} = RGB^{'}{}_{i,2} \oplus RGB^{'}{}_{i+1,2}. \tag{3.7}$$

$$RGB^{'}{}_{i+1,3} = RGB^{'}{}_{i,1} \oplus RGB^{'}{}_{i+1,3}. \tag{3.8}$$

E. Quant au dernier trio de la figure 3.7g, si nn est pair, on commence de bas en haut :

$$RGB^{'}{}_{i-1,nn-2} = RGB^{'}{}_{i,nn} \oplus RGB^{'}{}_{i-1,nn-2} \oplus Mask_{i-2,n-1}. \tag{3.9}$$

$$RGB^{'}{}_{i-1,nn-1} = RGB^{'}{}_{i,nn-1} \oplus RGB^{'}{}_{i-1,nn-1} \oplus Mask_{i-2,n}. \tag{3.10}$$

Où i=$\{mm, mm-1, mm-2, \ldots 3\}$ Si nn est impair, vice versa :

$$RGB^{'}{}_{i+1,nn-2} = RGB^{'}{}_{i,nn} \oplus RGB^{'}{}_{i+1,nn-2} \oplus Mask_{j,n-1}. \tag{3.11}$$

$$RGB^{'}{}_{i+1,nn-1} = RGB^{'}{}_{i,nn-1} \oplus RGB^{'}{}_{i+1,nn-1} \oplus Mask_{j,n}. \tag{3.12}$$

Où $i = \{1, 2, 3, \ldots mm-2\}$, et $j = \{1, 2, 3, \ldots m\}$.

F. Pour le reste des trios intermédiaires, nous effectuons les étapes suivantes, dans la direction ascendante puis dans la direction descendante.

En descendant :

$$RGB^{'}{}_{i+1,j} = RGB^{'}{}_{i,j+2} \oplus RGB^{'}{}_{i+1,j} \oplus Mask_{ii,jj}. \tag{3.13}$$

$$RGB^{'}{}_{i+1,j+1} = RGB^{'}{}_{i+1,j+1} \oplus RGB^{'}{}_{i,j+1}. \tag{3.14}$$

$$RGB^{'}{}_{i+1,j+2} = RGB^{'}{}_{i+1,j+2} \oplus RGB^{'}{}_{i,j+2}. \tag{3.15}$$

En ascendant :

$$RGB^{'}{}_{i-1,j} = RGB^{'}{}_{i,j+2} \oplus RGB^{'}{}_{i-1,j} \oplus Mask_{ii,jj}. \tag{3.16}$$

$$RGB^{'}_{i-1,j+1} = RGB^{'}_{i-1,j+1} \oplus RGB^{'}_{i,j+1}. \tag{3.17}$$

$$RGB^{'}_{i-1,j+2} = RGB^{'}_{i-1,j+2} \oplus RGB^{'}_{i,j+2}. \tag{3.18}$$

$i = \{1, 2, 3, \ldots mm - 1\}$, et $j = \{1, 2, 3, \ldots mm\}$ La lecture du masque suit le même sens que le processus.

Note : Nous allons appliquer ces instructions pour balayer toute la matrice $RGB^{'}$ sauf la première et la dernière colonne et ligne, qui appartiennent au ruban.

3.4.2 Méthode de décryptage

Dans le processus de décryptage (figure 3.8), la méthode utilisée est la même que celle de cryptage mais de manière inversée. On commence par le processus de diffusion, à ce stade-là, si l'image dont le nombre de colonnes est impair, le processus se fait de bas en haut, puis de haut en bas. Sinon, de haut en bas, puis de bas en haut, ensuite on remet les pixels dans leurs positions initiales.

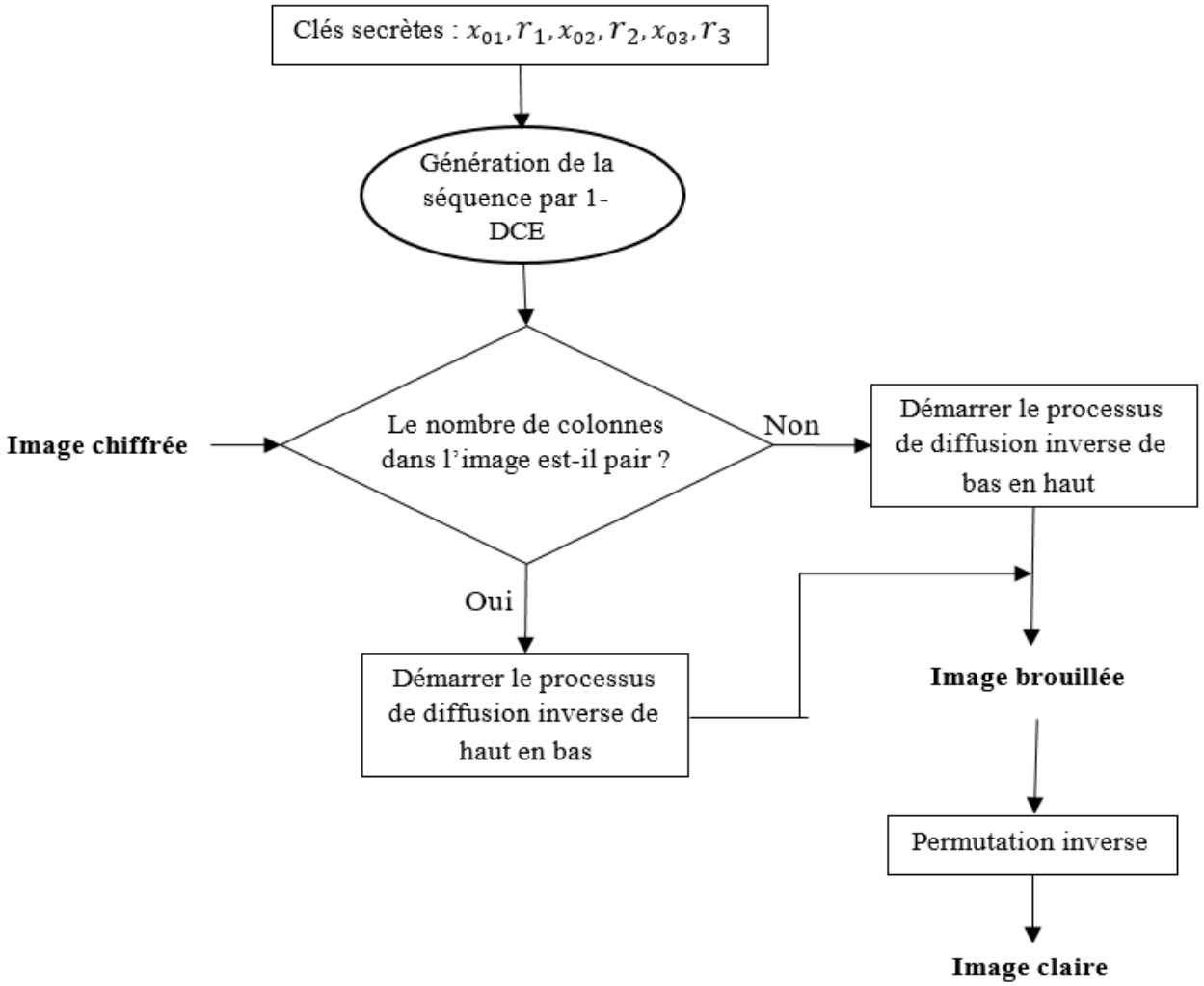


FIGURE 3.8 – Diagramme de bloc du processus de décryptage.

3.5 Résultats de la simulation et analyse de la sécurité

Afin d'évaluer la performance du schéma proposé, nous avons appliqué un grand nombre de tests de sécurité sur un ensemble d'images, en utilisant MATLAB 2013 sur un ordinateur équipé d'un processeur Intel i5-7200U, et de 8 Go de mémoire. Commençons l'évaluation par l'œil nu, la figure 3.9 illustre 4 différentes images claires et leurs images cryptées correspondantes. À titre de comparaison, il n'y a aucune information visuelle dans toutes les images cryptées, et il y a un grand écart entre chacune d'entre elles. Dans la partie suivante de cette section, nous allons démontrer l'efficacité et la robustesse de ce schéma numériquement et graphiquement.

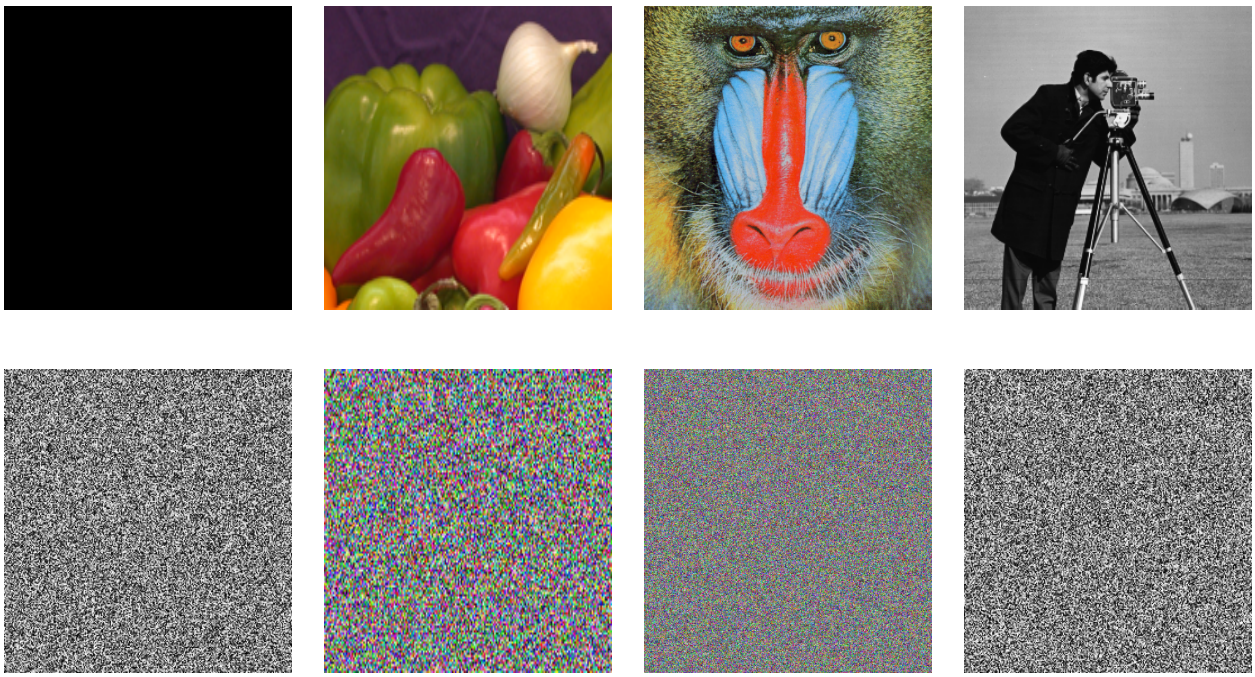


FIGURE 3.9 – Quatre images claires et leurs images cryptées.

3.5.1 Analyse des histogrammes

L'histogramme représente la distribution de l'intensité des pixels dans une image, est un outil de test très utile pour tester la qualité d'un système de cryptage d'images donné. Les figures 3.10 et 3.11 montrent les histogrammes de deux images simples (en niveau de gris et en couleur) et de leurs images cryptées. Chaque image simple a son propre histogramme qui fournit beaucoup d'informations sur elle. Cependant, les images cryptées ne présentent aucune différence dans leurs histogrammes qui sont plats et uniformément distribués. Cela signifie que les images cryptées ne fournissent aucune information statistique à l'attaquant.

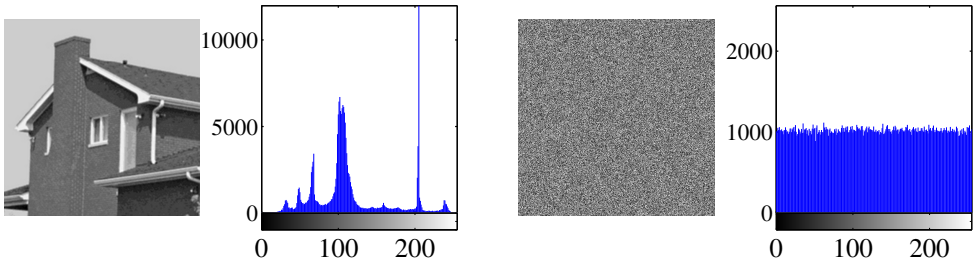


FIGURE 3.10 – Histogrammes de l'image en niveau de gris brute et de son image chiffrée.

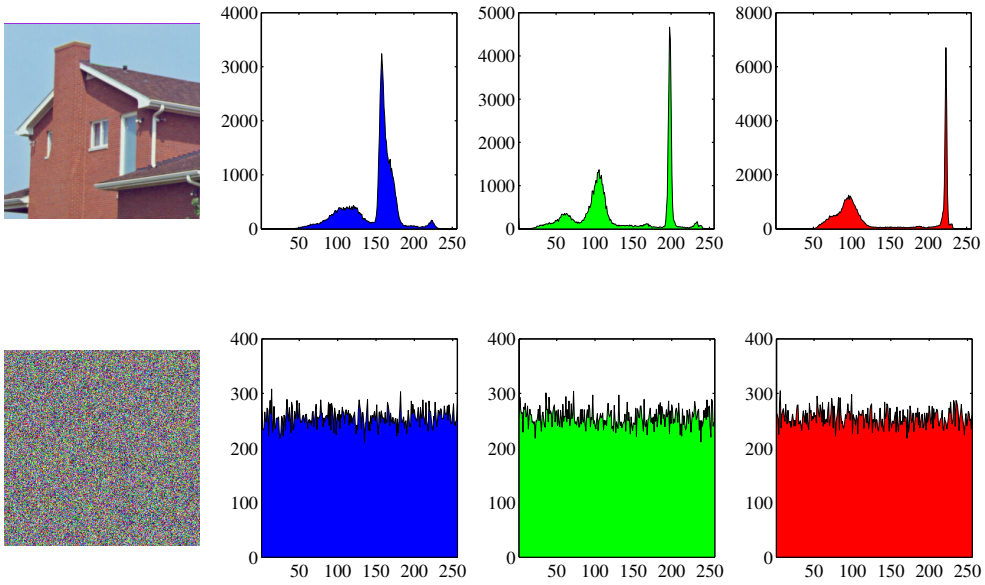


FIGURE 3.11 – Histogrammes de l'image en couleur brute et de l'image chiffrée.

3.5.2 Entropie

Dans notre cas, l'entropie de Shannon calcule le degré de désordre dans l'image cryptée. Comme sa valeur théorique idéale est égale à 8 [63]. Car dans le cas d'une image, les bits utilisés pour représenter les pixels sont 8, l'entropie peut aller de 0 à 8 bits. plus ses résultats sont proches de 8, meilleure est la distribution aléatoire dans l'image cryptée, les résultats de l'entropie du schéma proposé sont proches de la valeur idéale, ainsi que la comparaison avec les schémas des précédents travaux prouve la haute performance de notre schéma.

TABLEAU 3.6 – Comparaison de différentes valeurs d'entropie.

Images	Image claire	Notre algorithme	Réf [107]	Réf [108]	Réf [109]
Football	7.1796	7.9992	7.9992	7.9993	7.9992
Airelane	7.7622	7.9998	7.9998	7.9997	7.9997
House	7.0686	7.9992	7.9992	7.9992	7.9992

3.5.3 Analyse de corrélation

Les pixels d'une image cryptée ne doivent pas être corrélés, de même, la valeur de corrélation doit éventuellement être proche de zéro. Pour la calculer, nous avons utilisé l'équation présentée dans le chapitre précédent. Le tableau 3.7 montre la corrélation horizontale, verticale et diagonale entre les pixels voisins dans l'image de "poivrons", et son image cryptée par notre schéma et trois autres œuvres. Nous pouvons observer clairement que la corrélation dans toutes les directions de l'image cryptée est proche de zéro, contrairement à la corrélation dans le cas de l'image brute qui est proche de un. De plus, les valeurs de corrélation de nos coefficients sont meilleures que les trois autres références. Et cela montre notamment la résistance de notre schéma aux attaques par cassage de corrélation.

TABLEAU 3.7 – Coefficients de corrélation des pixels adjacents dans l'image cryptée et l'image brute, et analyse comparative avec divers algorithmes de cryptage.

Images	Verticale	Horizontale	Diagonale
Image claire "peppers"	0.9886	0.9933	0.9884
Notre méthode	0.0008	-0.0006	0.0002
Réf [107]	0.0012	-0.0021	-0.0015
Réf [108]	-0.0015	-0.0009	-0.0024
Réf [109]	0.0015	-0.0016	-0.0015

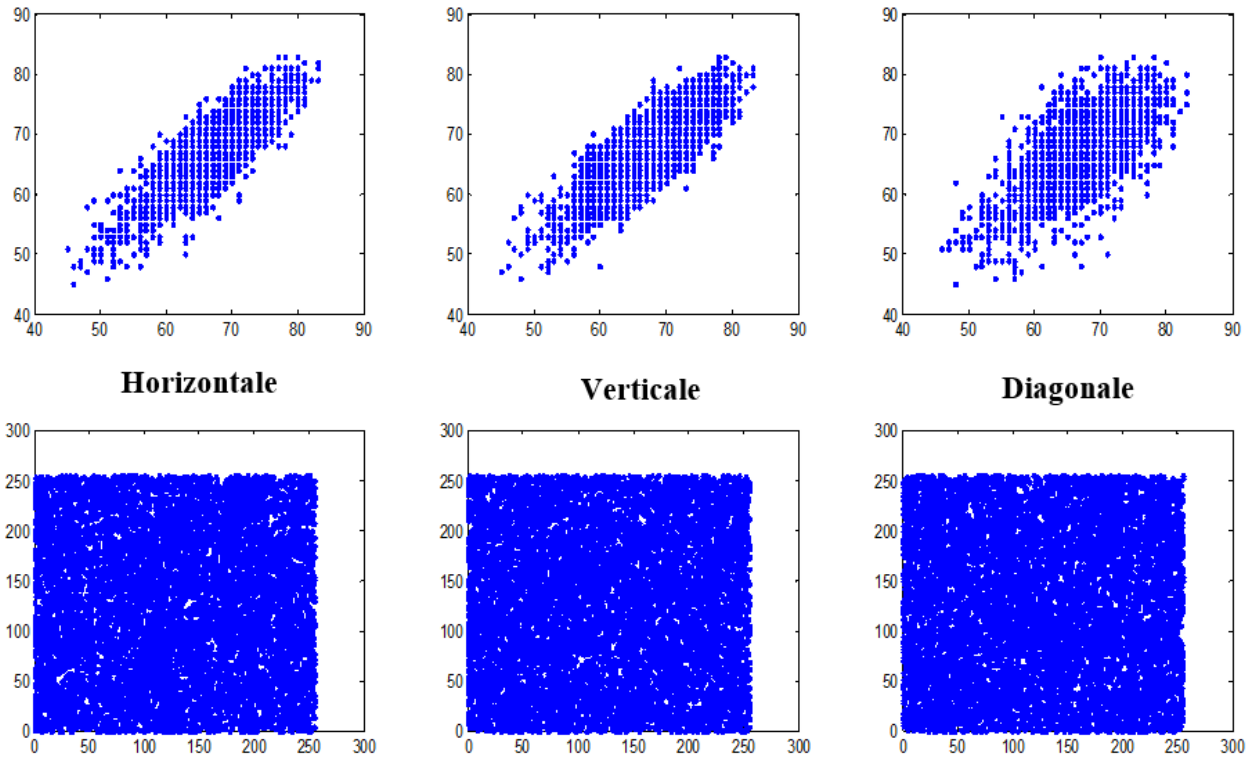


FIGURE 3.12 – Analyse de corrélation.

Ces résultats sont soutenus par les graphiques de corrélation affichés sur la figure 3.12 ci-

dessus, dont on peut voir que les valeurs des pixels adjacents dans l'image cryptée sont placées de façon aléatoire, ce qui indique qu'il n'y a pas de relation entre eux, tandis que dans l'image originale, les points sont presque sur la droite linéaire, donc les pixels ont pratiquement les même valeurs que leurs pixels voisins.

3.5.4 Sensibilité de l'image en clair

Nous allons maintenant quantifier la sensibilité vis-à-vis d'une attaque à texte clair, en mesurant l'impact d'un seul pixel sur l'image cryptée correspondante, moyennant le NPCR et l'UACI.

TABLEAU 3.8 – Résultats de l'analyse de NPCR.

Images	Notre méthode	Réf [107]	Réf [108]	Réf [109]
Lena (512×512)	R :99.62	R :99.60	R :99.60	R :99.62
	G :99.60	G :99.61	G :99.56	G :99.62
	B :99.60	B :99.61	B :99.61	B :99.58
Pears	R :99.60	R :99.61	R :99.60	R :99.61
	G :99.62	G :99.59	G :99.60	G :99.60
	B :99.62	B :99.62	B :99.62	B :99.60
Tree	R :99.63	R :99.59	R :99.57	R :99.60
	G :99.62	G :99.62	G :99.60	G :99.60
	B :99.61	B :99.58	B :99.61	B :99.59

TABLEAU 3.9 – Résultats de l'analyse de UACI.

Images	Notre méthode	Réf [107]	Réf [108]	Réf [109]
Lena (512×512)	R :33.45	R :33.41	R :33.54	R :33.35
	G :33.38	G :33.47	G :33.44	G :33.51
	B :33.42	B :33.43	B :33.50	B :33.33
Pears	R :33.39	R :33.51	R :33.47	R :33.40
	G :33.43	G :33.53	G :33.44	G :33.44
	B :33.48	B :33.45	B :33.42	B :33.47
Tree	R :33.37	R :33.53	R :33.46	R :33.43
	G :33.48	G :33.47	G :33.48	G :33.51
	B :33.37	B :33.40	B :33.62	B :33.53

D'après le tableau 3.8, les valeurs de NPCR sont plus proches de 100 %, ce qui signifie que notre algorithme de cryptage est sensible au moindre changement et qu'il peut donc résister aux attaques en clair. En ce qui concerne les valeurs de l'UACI illustrées dans le tableau 3.9, nous pouvons remarquer qu'elles sont proches de 33.33%, donc, cet algorithme peut résister à une attaque

différentielle. Comparativement, les résultats du schéma proposé sont plus ou moins similaires à ceux des autres approches.

3.5.5 Analyse des espaces clés

la taille de l'espace clé doit être $> 2^{100}$, un algorithme de cryptage avec un grand espace clé contrecarre le processus d'attaque par force brute, dans notre schéma proposé il y a trois clés, chacune d'entre elles a deux paramètres x_{0i}, et r_i, où i=1,2,3. Conformément à la section 3.3.4 , la précision de x_{0i}, est $10^{(17)}$ et la précision de r_i est $10^{(14)}$, ainsi, l'espace total des clés est $10^{(17\times3)} \times 10^{(14\times3)} = 10^{(51+42)} \approx 2^{3.322\times93} \approx 2^{308}$, ce qui est largement suffisant pour faire face à toute tentative d'attaque. De même, le tableau 3.10 illustre que la taille de notre espace clé est plus grande que celle de certains travaux.

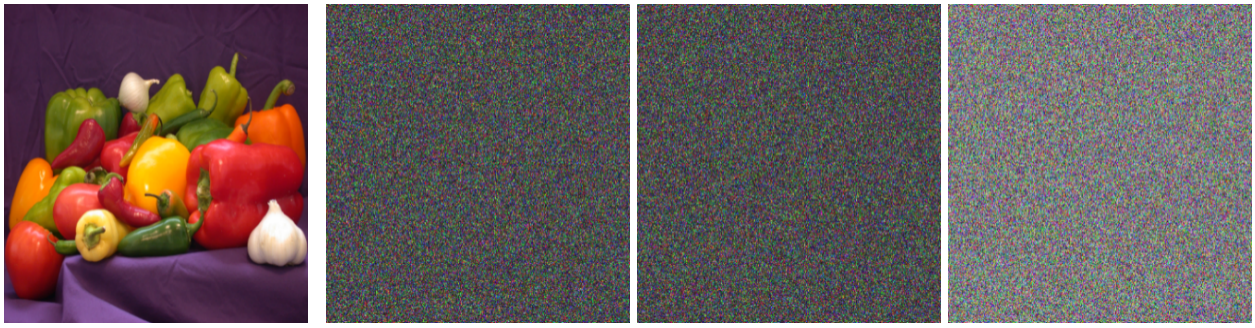
TABLEAU 3.10 – Comparaison entre les tailles de l'espace clé.

Images	Notre méthode	Réf [109]	Réf [110]	Réf [111]
Taille de l'espace clé	2^{308}	2^{138}	2^{195}	2^{130}

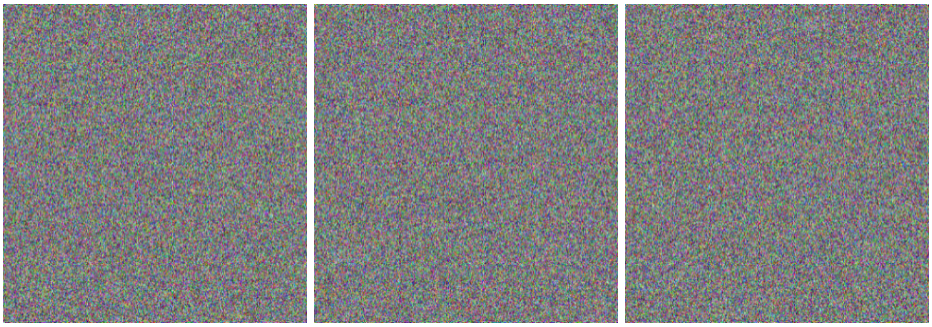
3.5.6 Sensibilité de la clé

Afin d'étudier la sensibilité de notre schéma proposé, admettons que $k(x_{01}, r_1, x_{02}, r_2, x_{03}, r_3)$ sont la clé de cryptage et k' $(x'_{01}, r'_1, x'_{02}, r'_2, x'_{03}, r'_3)$ sont la clé de décryptage correspondante. Si la clé de cryptage est exactement celle de décryptage, c'est-à-dire $k = k^{'}$, l'image décryptée est identique à l'image originale (Figure 3.13). Nous faisons un changement mineur dans les paramètres de la clé, un changement de $10^{(-17)}$ au niveau de la valeur initiale x_{0i} $\{1, 2, 3\}$ et un changement de $10^{(-14)}$ au niveau du paramètre de contrôle r_i $\{1, 2, 3\}$, ces valeurs ont été choisis comme déjà mentionné car la sensibilité de 1-DCE atteint $10^{(-17)}$ pour x_{0i}, et $10^{(-14)}$ pour r_i. À Chaque fois nous effectuons le changement sur un seul de ces six paramètres. Comme nous pouvons le constater sur la figure 3.13, toutes les images décryptées résultantes sont complètement inconnues, bien que le changement apporté aux conditions initiales soit très minime. Cela confirme non seulement la grande sensibilité des clés, mais aussi celle du 1-DCE. Un autre test de sensibilité des clés a été établi et affiché sur la Figure 3.14, où nous chiffrons l'image en clair avec une clé de chiffrement k_1 (figure 3.14b), puis nous apportons un changement mineur à un élément de la clé de chiffrement k_1 pour obtenir k_2. Nous chiffrons la même image par k_2 (figure 3.14c). Après cela, nous comparons les deux images chiffrées obtenues C1 et C2 (figure 3.14d), où nous pouvons remarquer la différence entre C1 et C2, même si la différence est minime, c.à.d au niveau d'une

seule valeur de la clé. Cela rend l'algorithme que nous proposons robuste contre plusieurs types d'attaques en clair.



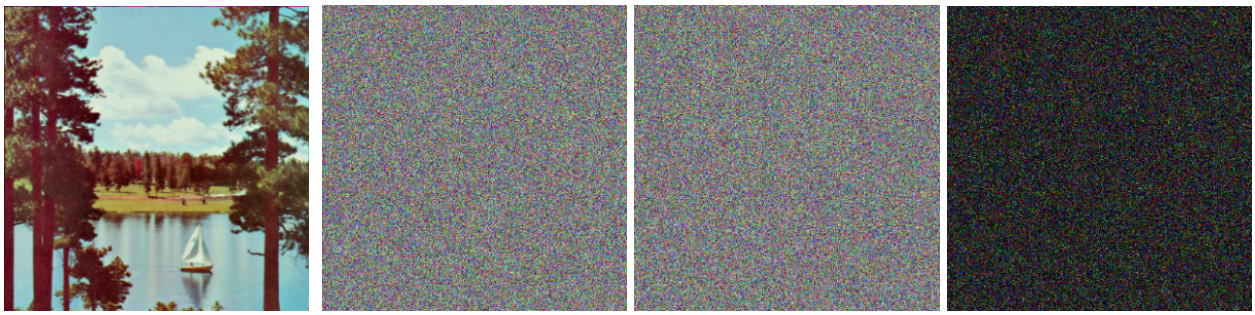
(a) (b) (c) (d)



(e) (f) (g)

FIGURE 3.13 – Image décryptée de poivrons avec un léger changement de clés. (a) $k = k^{'}$. (b) $x'_{01} = x_{01} + 10^{-17}$. (c) $x'_{02} = x_{02} + 10^{-17}$. (d) $x'_{03} = x_{03} + 10^{-17}$. (e) $r'_1 = r_1 + 10^{-14}$. (f) $r'_2 = r_2 + 10^{-14}$. (g) $r'_3 = r_3 + 10^{-14}$

.



(a) (b) (c) (d)

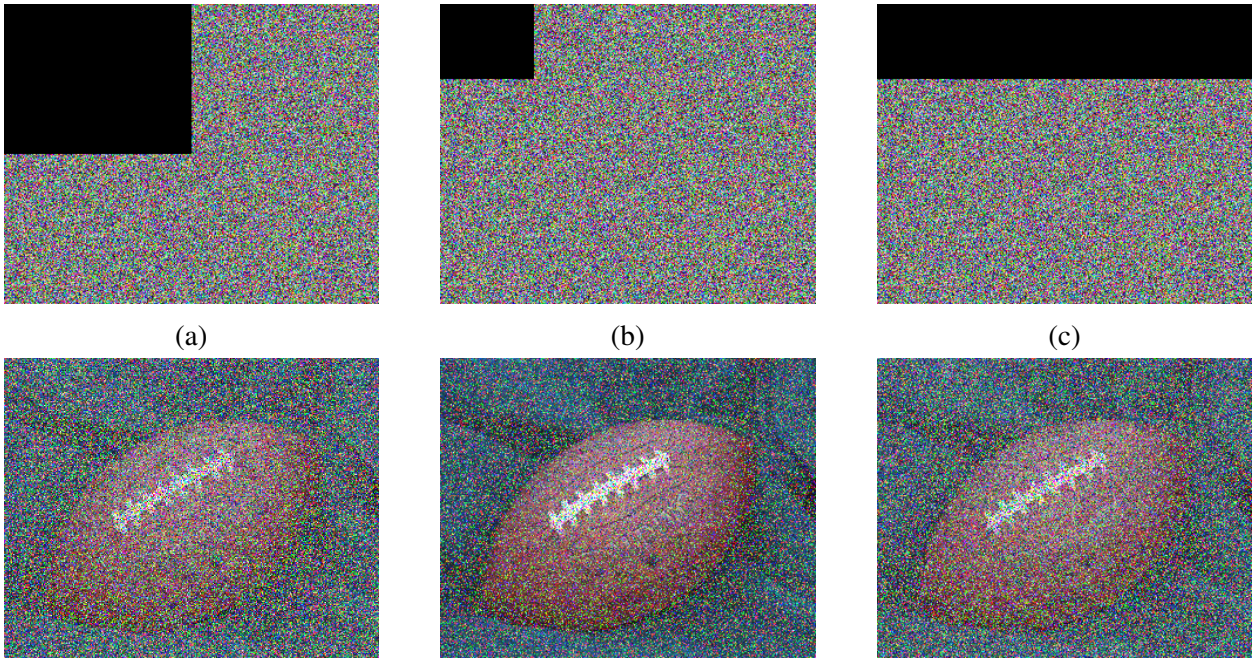
FIGURE 3.14 – Test par attaque en texte brut (a) Image brute. (b) Image chiffrée avec la clé k_2. (c) Image chiffrée avec la clé k_2. (d) Différence de pixel à pixel.

.

3.5.7 Perte de données

Dans cette sous-section, nous allons submerger notre image dans un autre problème qui peut se produire pendant sa transmission, c'est la perte de données. Où l'image chiffrée perd certaines de

ses informations comme illustré sur la figure 3.15a, 3.15b et 3.15c. Nous pouvons observer dans les images décryptées (figure 3.15d, 3.15e, et 3.15f), que cette perte n'a pas affecté les caractéristiques de l'image qui sont encore reconnaissables. D'après ces résultats visuels, nous concluons que la méthode proposée peut résister à une éventuelle perte de données.



(a) (b) (c)

(d) (e) (f)

FIGURE 3.15 – Test de perte de données : Images cryptées avec (a) 160×128 de perte de données, (b) 80×64 de perte de données, (c) 80×256 de perte de données. Les images décryptées correspondantes (d), (e), (f) respectivement.

3.5.8 Analyse de la vitesse

Le facteur le plus significatif de la méthode proposée est sans doute sa rapidité, étant donné que la vitesse joue un rôle important dans les systèmes cryptographiques en particulier, et dans les applications en temps réel en général. De plus, c'est un critère crucial pour la sécurité des clés. Nous remarquons dans le tableau 3.11 que le modèle proposé est rapide. Encore loin, il est plus rapide que les autres algorithmes proposés.

TABLEAU 3.11 – Temps d'exécution.

Images	Notre méthode	Réf [107]	Réf [108]
Image (256×256)	0.040	0.31	0.051
Image (512×512)	0.15	0.83	0.17

3.6 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle carte chaotique, appelée 1-DCE, construite à partir de la carte cubique classique, du modulaire arithmétique et de la fonction exponentielle. L'évaluation de 1-DCE à l'aide de nombreux tests prouvent sa haute performance, surtout lorsqu'elle est comparée à la carte cubique classique, ainsi qu'à d'autres cartes chaotiques. Ensuite, nous avons intégré 1-DCE dans un système de cryptage des images en présentant un schéma efficace basé sur la confusion et une nouvelle méthode de diffusion. Nous avons montré que le schéma de cryptage proposé peut résister à de nombreuses attaques statistiques, différentielles et en texte clair. Il peut être appliqué aux images en couleurs et en niveaux de gris.

Le dernier chapitre contiendra une autre contribution, une nouvelle carte chaotique tridimensionnelle et son utilisation dans un nouveau schéma de cryptage d'images sera discuté.

Chapitre 4

Nouvelle Carte Chaotique Trigonométrique 3D et son Application dans un Système de Cryptage d'Image

Sommaire

4.1 Introduction
4.2 Carte chaotique 3D proposée
4.3 Évaluation de la performance
4.4 Crypto-système proposé
4.5 Résultats expérimentaux et analyse de la sécurité
4.6 Conclusion

Chapitre 4

Nouvelle Carte Chaotique Trigonométrique 3D et son Application dans un Système de Cryptage d'Image

4.1 Introduction

Le cryptage d'images par le chaos fait l'objet de nombreux schémas de cryptage avancés, vu ses hautes performances, comme un comportement chaotique, la sensibilité aux conditions initiales et aux le paramètres de contrôle, et l'ergodicité. Malgré les avantages des cartes chaotiques de faible dimension dans les systèmes cryptographiques, à cause de la simplicité de leur structure et le temps de calcul, les cartes chaotiques de plusieurs dimensions possèdent également un grand espace des clés, un attracteur complexe et de fortes caractéristiques dynamiques. Le but de ce chapitre est l'exploitation d'un nouveau système chaotique tridimensionnel basé sur les fonctions trigonométriques sinus et cosinus (3D-CSC), dans un algorithme de cryptage d'image, en utilisant une nouvelle architecture de permutation-diffusion. Le présent chapitre se scinde en trois principales parties. Dans la première partie, nous allons introduire notre nouvelle carte chaotique (3D-CSC) et les résultats des tests qui lui ont été appliqués. Dans la seconde partie, nous allons mettre en place la technique de cryptage d'image utilisée et même le schéma inverse. La troisième partie contiendra l'évaluation de la performance et de robustesse de schéma proposé. Nous terminerons le chapitre par une conclusion.

4.2 Carte chaotique 3D proposée

Les cartes chaotiques multidimensionnelles (MD) sont actuellement largement utilisées dans de nombreux systèmes cryptographiques, en raison de la croissance de la vitesse de calcul, des

performances élevées des équipements informatiques modernes et de la volumineuse taille des données en général, et les images en particulier. L'avantage de ces dernières est qu'elles sont plus complexes et possèdent un grand espace de clé. Dans notre cas, après plusieurs expériences, nous avons constaté qu'une combinaison de fonctions trigonométriques successives (sinus, cosinus) donne un excellent résultat concernant les tests de performance chaotique. Parmi celles-ci, nous avons proposé 3D-CSC, qui base sur trois fonctions trigonométriques cosinus et sinus exprimées par :

$$\begin{cases} f :]-1,1[^3 = [-1,1]^3 \\ x_{n+1} = \cos\left((r_1+1)\times 8\times\pi\times(x_n - y_n)\right) \\ y_{n+1} = \sin\left((r_2+1)\times 8\times\pi\times(y_n - z_n)\right) \\ z_{n+1} = \cos\left((r_3+1)\times 8\times\pi\times(z_n + x_n)\right) \end{cases} \tag{4.1}$$

Avec x_n, y_n, z_n sont les variables d'état, r_1, r_2, r_3 sont les paramètres de contrôle. De surcroît, ce système peut être étendu à N dimensions (ND-CSC), de sorte que nous aurons :

$$\begin{cases} x_{n+1}^{(1)} = \cos\left((r_{11}+1)\times 8\times\pi\times\left(x_n^{(1)} - y_n^{(1)}\right)\right) \\ y_{n+1}^{(1)} = \sin\left((r_{12}+1)\times 8\times\pi\times\left(y_n^{(1)} - z_n^{(1)}\right)\right) \\ z_{n+1}^{(1)} = \cos\left((r_{13}+1)\times 8\times\pi\times\left(z_n^{(1)} + x_n^{(2)}\right)\right) \\ x_{n+1}^{(2)} = \cos\left((r_{21}+1)\times 8\times\pi\times\left(x_n^{(2)} - y_n^{(2)}\right)\right) \\ y_{n+1}^{(2)} = \sin\left((r_{22}+1)\times 8\times\pi\times\left(y_n^{(2)} - z_n^{(2)}\right)\right) \\ z_{n+1}^{(2)} = \cos\left((r_{23}+1)\times 8\times\pi\times\left(z_n^{(2)} + x_n^{(3)}\right)\right) \\ . \\ . \\ . \\ x_{n+1}^{(N)} = \cos\left((r_{N1}+1)\times 8\times\pi\times\left(x_n^{(N)} - y_n^{(N)}\right)\right) \\ y_{n+1}^{(N)} = \sin\left((r_{N2}+1)\times 8\times\pi\times\left(y_n^{(N)} - z_n^{(N)}\right)\right) \\ z_{n+1}^{(N)} = \cos\left((r_{N3}+1)\times 8\times\pi\times\left(z_n^{(N)} + x_n^{(1)}\right)\right) \end{cases} \tag{4.2}$$

4.3 Évaluation de la performance

4.3.1 Évaluation graphique

Pour étudier les performances de 3D-CSC, nous effectuerons plusieurs tests mathématiques. En ce qui concerne les diagrammes de bifurcation, nous allons le tracer pour toutes les compo-

santes x, y, z. La principale observation qu'on peut tirer de la figure 4.1 est que les diagrammes de bifurcation couvrent la totalité de la plage affichée de r.

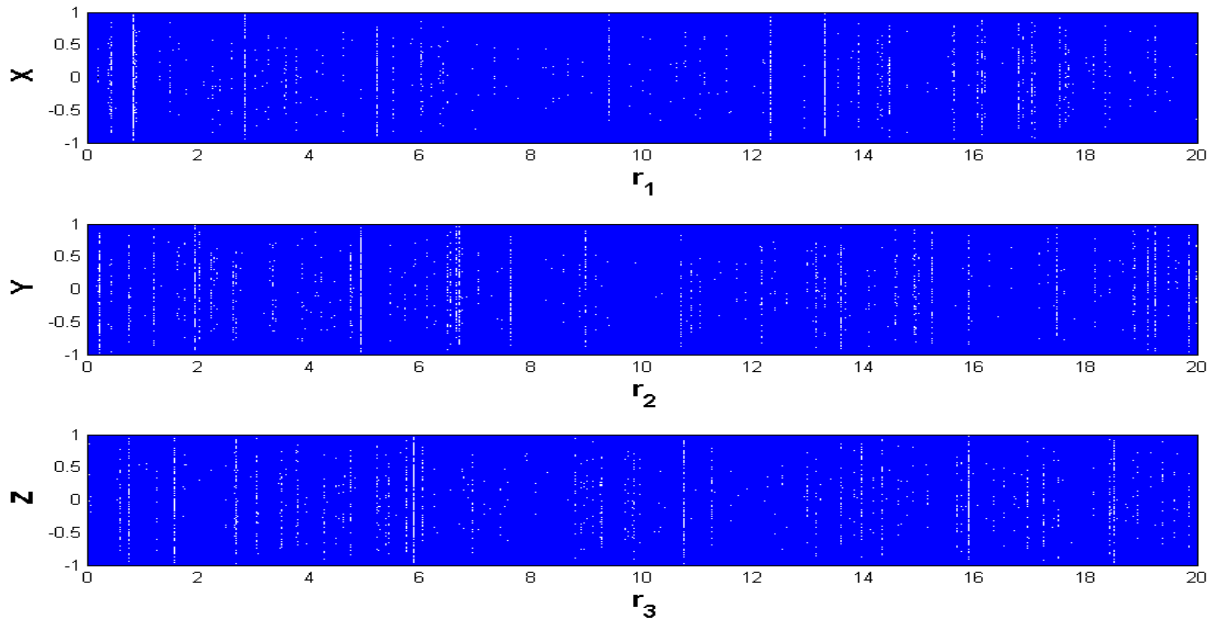
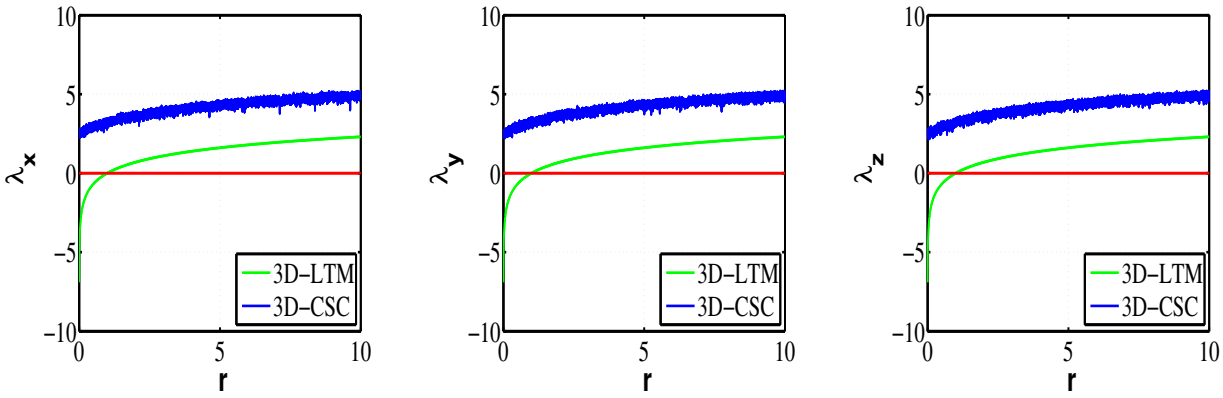


FIGURE 4.1 – Diagrammes de bifurcation pour chaque composante de 3D-CSC.

De la même manière, nous dessinons les graphiques de l'exposant de Lyapunov sur la figure 4.2a, 4.2b, et 4.2c. Nous pouvons remarquer clairement que les courbes de l'ensemble des trois composantes de 3D-CSC sont positives sur tous les intervalles du paramètre de contrôle r, i.e $r = r_1 = r_2 = r_3$; et ils sont approximativement égaux à 6 lorsque r=10, supérieurs aux courbes de l'exposant de Lyapunov de 3D-LTM [112], à noter aussi que λs de 3D-LTM possèdent des valeurs négatives pour r<1. Ces précieuses remarques prouvent le caractère aléatoire de notre carte chaotique pour toutes les valeurs de r_1, r_2, r_3, ainsi que la haute sensibilité envers elles.



(a) (b) (c)

FIGURE 4.2 – Exposant de Lyapunov pour chaque composante de 3D-CSC comparé au 3D-LTM. (a) Composante x. (b) Composante y. (c) composante z.

Pour concrétiser l'aspect chaotique du 3D-CSC, nous avons représenté ses échantillons dans un graphique 3D. Où figure 4.3 montre que la distribution de ces derniers est inégale et vraiment aléatoire.

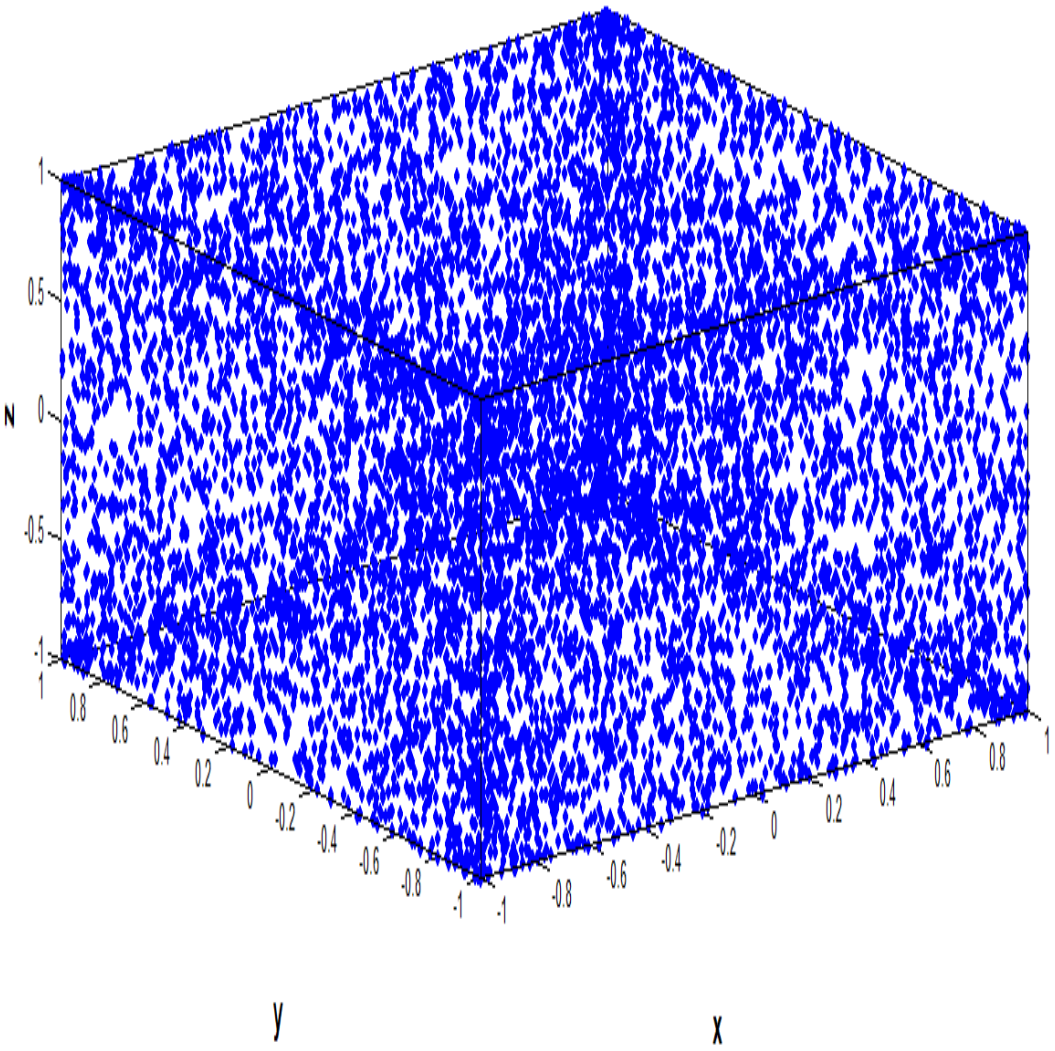


FIGURE 4.3 – Attracteur chaotique de 3D-CSC.

La sensibilité du système chaotique est l'un des plus importants critères. Sur ce fondement, à chaque fois nous opérons un très petit changement à l'une des conditions initiales (estimé par $10^{(-16)}$), ou à l'un des paramètres de contrôle (estimé par $10^{(-15)}$) tout en gardant les valeurs des autres paramètres inchangées. Les trajectoires des composantes du système avec et sans changement sont complètement différentes (Figure 4.5), sauf dans les toutes premières itérations. Par conséquent, notre système chaotique proposé est extrêmement sensible, ainsi, la sensibilité de ses conditions initiales atteint $10^{(-16)}$, tandis que celle de ses paramètres de contrôle est estimée par $10^{(-15)}$.

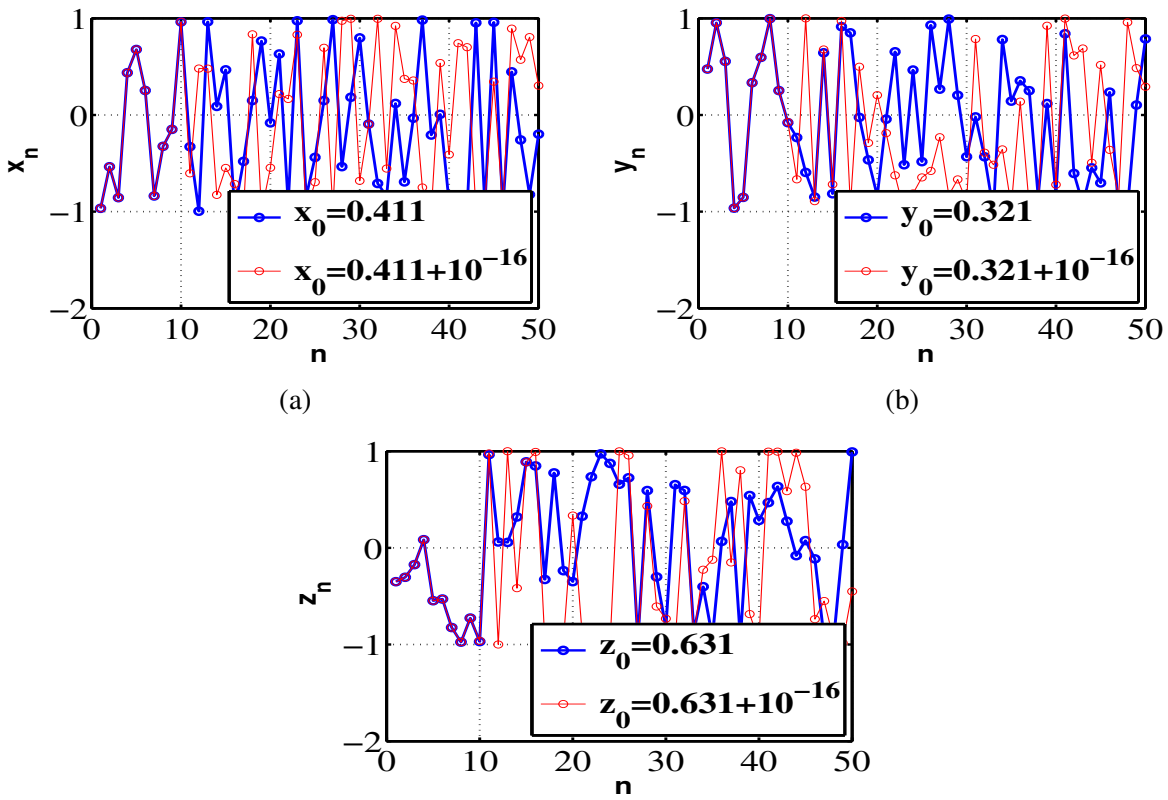


FIGURE 4.4 – Sensibilité de 3D-CSC à un changement minime de $10^{(-16)}$ aux valeurs initiales. (a) à x_0. (b) à y_0. (c) à z_0.

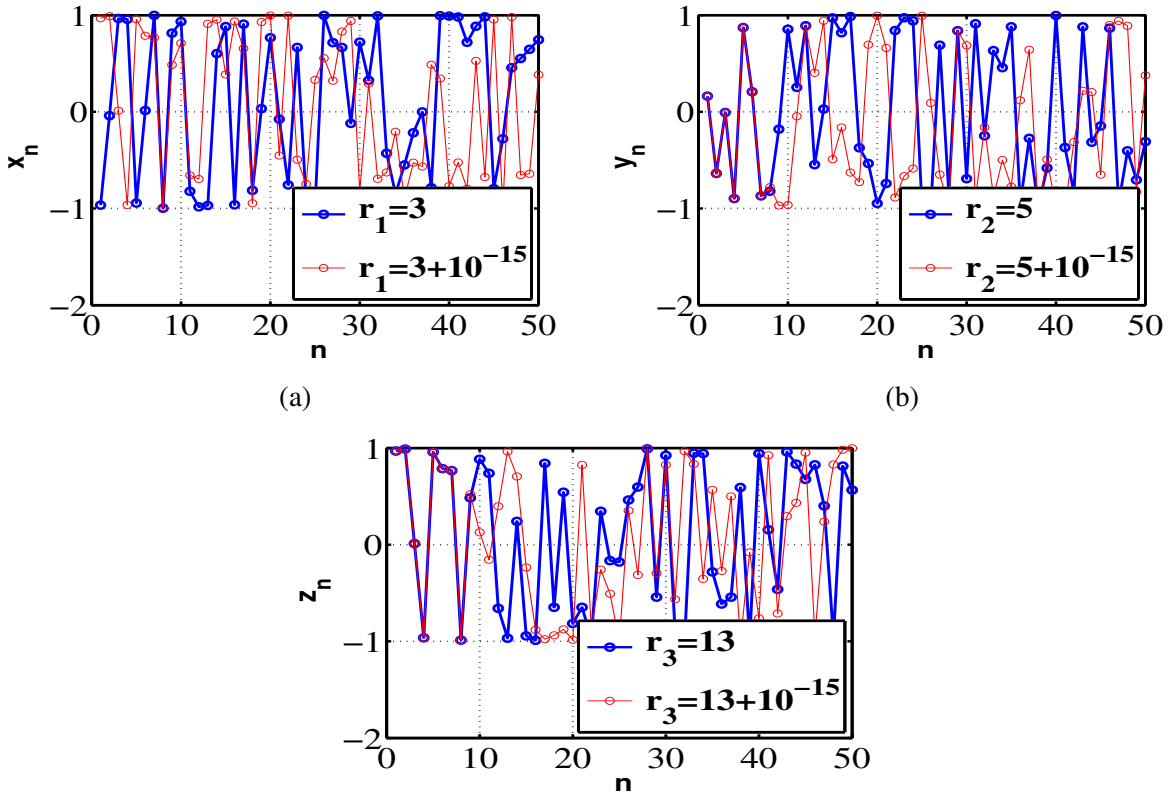


FIGURE 4.5 – Sensibilité de 3D-CSC à un changement minime de $10^{(-16)}$ aux valeurs des paramètres de contrôle. (a) au r_1. (b) au r_2. (c) au r_3.

4.3.2 Évaluation statistique

Afin de mener une étude numérique et statistique sur notre système, nous avons appliqué deux tests du chaos. Le test NIST, et le test DIEHARD. Les plages acceptables de la valeur p du test NIST et du test DIEHARD sont respectivement [0,1] et [0,1[. Les résultats obtenus sont satisfaisants, où nous pouvons remarquer dans les tableaux 4.1 et 4.2 que tous les tests sont réussis.

TABLEAU 4.1 – Résultats de test de NIST de notre système et d'un autre système.

Test	p-value de 3D-CSC			p-value de ([113])		
	x	y	z	x	y	z
Frequency	0.95	0.90	1.00	0.55	0.32	0.45
Block Frequency	0.98	0.93	0.66	0.83	0.57	0.43
Runs	0.89	0.58	0.97	0.54	0.19	0.23
Longest Runs Of Ones	0.90	0.63	0.99	0.80	0.79	0.78
Rank	0.72	0.99	0.52	0.97	0.26	0.33
Spectral	1.00	1.00	0.77	0.03	0.94	0.45
Non Overlapping Template Matching	0.71	0.95	0.61	0.23	0.47	0.58
Overlapping Template Matching	1.00	1.00	0.92	0.49	0.08	0.78
Universal	1.00	0.68	0.81	0.93	0.05	0.05
Linear Complexity	1.00	1.00	1.00	0.79	0.27	0.92
Serial	0.95	0.99	0.99	0.75	0.84	0.34
Approximate Entropy	0.09	0.12	0.38	0.61	0.08	0.36
Cumulative Sums	0.98	0.92	0.88	0.16	0.94	0.63
Random Excursions	0.99	0.39	0.78	0.65	0.08	0.65
Random Excursions Variant	0.99	0.39	0.78	0.56	0.05	0.56

TABLEAU 4.2 – Résultats de test de DIEHARD de notre système en comparant avec un autre système.

test	P valeurs de 3D-CSC			P valeurs de ([113])		
	x	y	z	x	y	z
Birthday spacing	0.774320	0.772605	0.249720	0.546556	0.623562	0.435636
Overlapping permutation	0.464080	0.123819	0.810897	0.572362	0.463465	0.567473
Binary rank	0.515743	0.451585	0..323937	0.637347	0.435625	0.563457
Bitstream	0.797546	0.514673	0.829590	0.456778	0.514673	0.447568
OPSO	0.903456	0.875265	0.551700	0.356257	0.346235	0.347673
OQSO	0.783671	0.679432	0.730728	0.365835	0.626525	0.567884
DNA	0.642364	0.544538	0.566541	0.475481	0.546847	0.467848
Count the ones	0.547261	0.692197	0.698977	0.367324	0.435734	0.574579
Parking Lot	0.655385	0.832579	0.876831	0.346756	0.367346	0.568547
Minimum distance	0.831462	0.174412	0.777604	0.546736	0.475489	0.579256
3DS spheres	0.258140	0.920332	0.103942	0.627934	0.580245	0.589207
Squeeze	0.876672	0.451267	0.867129	0.235637	0.375591	0.402486
Overlapping sum	0.967031	0.064126	0.425792	0.573468	0.495427	0.368025
Runs	0.562094	0.692791	0..711610	0.467803	0.358246	0.483682
Craps	0.858143	0.691649	0.78	–	–	–

4.4 Crypto-système proposé

Nous proposons un schéma de cryptage d'image efficace, se servant essentiellement de 3D-CSC. Étant donné que les images sont constituées d'un grand nombre de pixels, avec une forte corrélation, il sera nécessaire de prendre en compte le facteur temps. De même, l'importance de l'intégration de chaos manifeste dans le cassage de cette corrélation entre les pixels. L'idée principale est d'appliquer deux processus de confusion-diffusion avec la mise en œuvre de 3D-CSC. L'algorithme proposé est basé sur une architecture de type permutation-diffusion simple et efficace. Il peut être appliqué à différents types d'images. Les tests d'évaluation prouvent l'efficacité et la vélocité de notre crypto-système. Le processus de cryptage suit les trois étapes suivantes :

— En utilisant 3D-CSC, nous le faisons appel quatre fois, en changeant à chaque fois les conditions initiales, pour avoir au final 24 clés.

— Un processus de diffusion à base de principe de récursivité, l'opération XOR, et l'arith-

métique modulaire. L'élément clef ici est l'utilisation de deux vecteurs chaotiques, pour localiser les positions des pixels dans matrices construites par 3D-CSC.

— Un processus de confusion-diffusion simultané, dans lequel on effectue un décalage circulaire aléatoire de l'image en question et ou exclusif avec une matrice chaotique construite par 3D-CSC.

— Finalement, un processus de confusion, on rend la taille de l'image carrée (si ce n'est pas le cas) puis on la divise horizontalement et verticalement en quatre moitiés égales. Après ça, on prend deux vecteurs d'un morceau et deux vecteurs du morceau opposé d'une manière aléatoire, et on fait échanger les positions des vecteurs qui appartiennent au même morceau.

Pour aller loin, les détails de ces étapes seront expliqués dans la sous-section ci-dessous.

4.4.1 Schéma de cryptage

Dans cette partie, nous allons expliquer notre algorithme de cryptage de manière plus profonde ; le schéma fonctionnel du processus de cryptage/décryptage est présenté sur la figure 4.8, tandis que les détails sont les suivants :

A. Soit IMG l'image couleur de taille (s×M×N) reformée en trois matrices en cascade [R G B] de taille (M×n), avec n=3×N.

B. En utilisant 3D-CSC, nous générons trois vecteurs (x1,y1,z1) de taille (M×N), puis nous construisons les trois premières matrices indispensables (MATRIX1) dans le processus de diffusion selon les formules suivantes :

$$vector1 = abs\left(Arrondi\left(10^{14} \times x1 \ mod\, 256\right)\right). \tag{4.3}$$

$$vector2 = abs\left(Arrondi\left(10^{14} \times y1 \ mod\, 256\right)\right). \tag{4.4}$$

$$vector3 = abs\left(Arrondi\left(10^{14} \times z1 \ mod\, 256\right)\right). \tag{4.5}$$

$$matrix1 = reformer\left(vector1, M, N\right). \tag{4.6}$$

$$matrix2 = reformer\left(vector2, M, N\right). \tag{4.7}$$

$$matrix3 = reformer\left(vector3, M, N\right). \tag{4.8}$$

$$MATRIX1 = \left[matrix1 \ \ matrix2 \ \ matrix3\right]. \tag{4.9}$$

Les trois autres matrices (MATRIX2) sont construites de la même façon ; nous devons

donc seulement changer les paramètres initiaux de 3D-CSC, pour obtenir de nouvelles séquences. Nous aurons en effet des nouveaux vecteurs (x2, y2, z2). Soit X1, Y1, et X2, Y2 les vecteurs de localisation créés par l'un des derniers vecteurs (x1,y1,z1) ou (x2,y2,z2) comme suit :

$$\mathrm{X1} = \mathrm{abs}\left(\mathrm{Arrondi}\left((\mathrm{M}-1)\times \mathrm{x1}\left(1:\mathrm{M}\right)\right)\right)+1. \tag{4.10}$$

$$\mathrm{Y1} = \mathrm{abs}\left(\mathrm{Arrondi}\left((\mathrm{n}-1)\times \mathrm{y1}\left(1:\mathrm{n}\right)\right)\right)+1. \tag{4.11}$$

Il en va de même pour X2 et Y2 (mais en utilisant x2 et y2, ou des valeurs différentes de x1 et y1). X1 et X2 sont les vecteurs lignes de MATRIX1 et MATRIX2 respectivement ; aussi, Y1 et Y2 sont leurs vecteurs colonnes. Nous allons les utiliser pour modifier les valeurs des pixels de IMG en suivant la démarche suivante :

---

for $i = 1$ **to** M **do**
  for $j = 1$ **to** n **do**
    $IMG(i,j) = ((IMG(i,j) + MATRIX1(X1(i),Y1(j))) \ mod \ 256) \oplus MATRIX2(X2(i),Y2(j)))$
  end
end

L'image résultante subira à une opération XOR récursive, horizontalement et verticalement :

$$IMG\left(i,j+1\right) = IMG\left(i,j+1\right) \oplus IMG\left(i,j\right). \tag{4.12}$$

Où $i = 1....M$, $j = 1...n-1$.

$$IMG\left(i+1,j\right) = IMG\left(i+1,j\right) \oplus IMG\left(i,j\right). \tag{4.13}$$

Où $i = 1...M-1, j = 1...n$.

À noter que les deux opérations ci-dessus sont très importantes, vu que leur absence crée un large gap au niveau des valeurs de NPCR et UACI.

C. Le deuxième sous-processus est l'opération de Confusion-Diffusion simultanée, où nous allons décaler chaque vecteur de IMG (chaotiquement) verticalement et horizontalement tout en appliquant l'opération XOR entre le vecteur décalé et le vecteur correspondant dans la matrice (MATRIX3). Les valeurs de décalage circulaire horizontal et vertical sont créés

en utilisant 3D-CSC pour la troisième fois :

$$shiftverti = Arrondi\left(M * x3\left(1:n\right)\right). \tag{4.14}$$

$$shifthoriz = Arrondi\left(n * y3\left(1:M\right)\right). \tag{4.15}$$

$$MATRIX3 = reformer\left(Arrondi\left(abs\left(255 * z3\right)\right), M, n\right). \tag{4.16}$$

Le processus est comme suit :

---

for $i = 1$ **to** M **do**
$\quad IMG1(:, j) = circshift(IMG(:, j), shiftverti(j)') \oplus MATRIX3(:, j)$
end
for $j = 1$ **to** n **do**
$\quad IMG2(i, :) = circshift(IMG1(i, :)', shifthoriz(i)') \oplus MATRIX3(i, :)'$
end

D. Nous revenons ici à la forme basique de $IMG2$ (M×N×3). Ensuite, nous reformons chaque matrice en matrice carrée (si elle ne l'est pas). Dans le cas de manque des pixels, nous pouvons ajouter des pixels supplémentaires. L'idée principale de cette dernière étape est d'interchanger chaotiquement les lignes et les colonnes de chaque moitié de l'image (pour plus de détails voir la figure 4.6) ; cette dernière doit faire appel une fois de plus à 3D-CSC (x4,y4,z4) ; le processus est résumé dans les points suivants :

$$xx = abs\left(Arrondi\left(\left(\frac{mm}{2} - 1\right) \times x4\right)\right) + 1. \tag{4.17}$$

$$yy = abs\left(Arrondi\left(\left(\frac{mm}{2} - 1\right) \times y4\right)\right) + \frac{mm}{2}. \tag{4.18}$$

$$zz = abs\left(Arrondi\left(\left(\frac{mm}{2} - 1\right) \times z4\right)\right) + \frac{mm}{2}. \tag{4.19}$$

$$vec\_up = reformer\left(xx\left(1 : \frac{3 \times mm}{2}\right), \frac{mm}{2}, 3\right). \tag{4.20}$$

$$vec\_right = reformer\left(xx\left(1 + \frac{3 \times mm}{2} : 2 \times \frac{3 \times mm}{2}\right), \frac{mm}{2}, 3\right). \tag{4.21}$$

$$vec\_left = reformer\left(yy\left(1 : \frac{3 \times mm}{2}\right), \frac{mm}{2}, 3\right). \tag{4.22}$$

$$vec\_down = reformer\left(zz\left(1 : \frac{3 \times mm}{2}\right), \frac{mm}{2}, 3\right). \tag{4.23}$$

mm est la nouvelle taille de IMG2 (mm×mm×s).

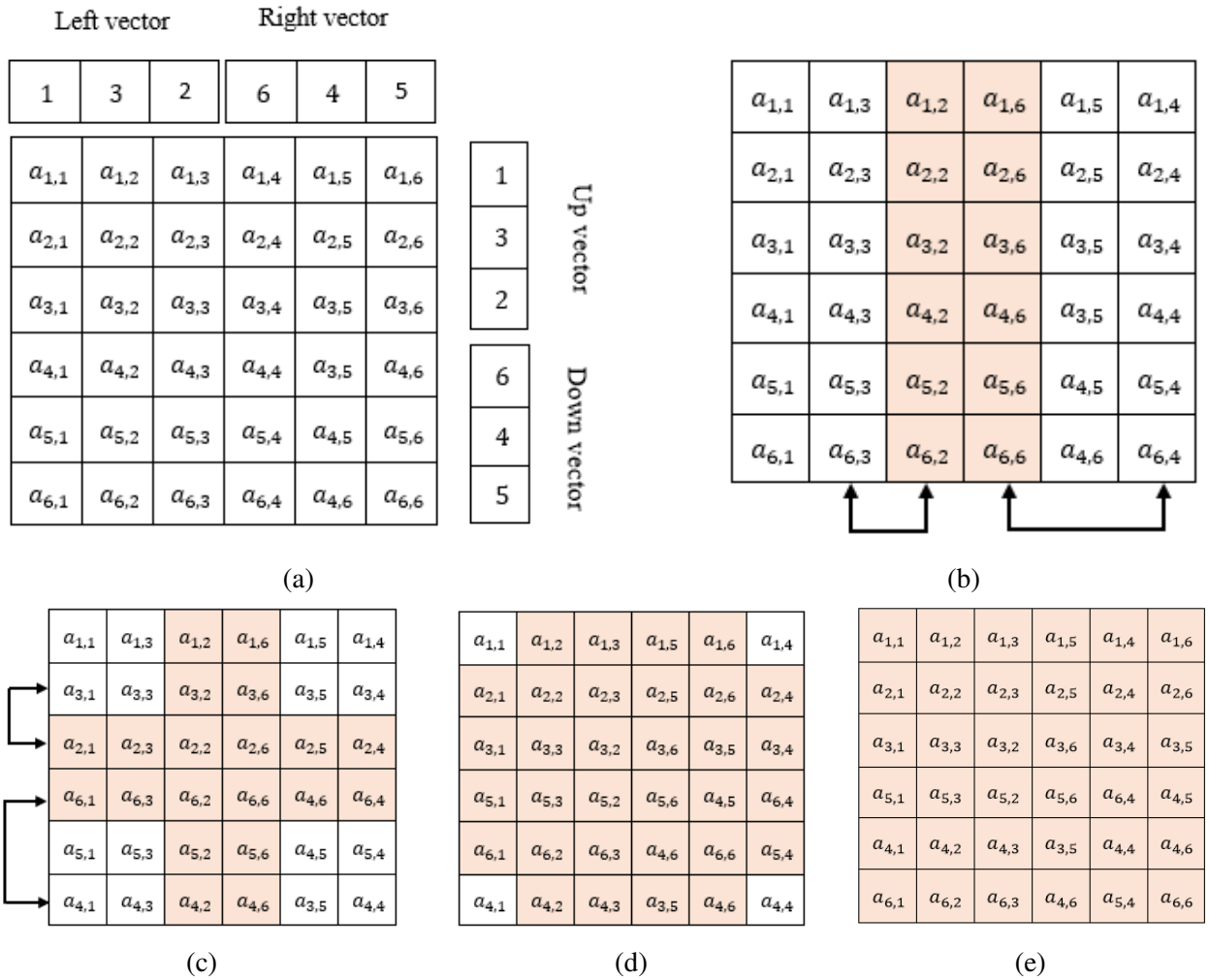


FIGURE 4.6 – Représentation du dernier processus appliqué sur une matrice 4×4 en fonction des valeurs de vecteurs right/left et up/downs. (a) La matrice avec les vecteurs de position avant le début du processus de confusion. (b) La première étape du processus de confusion (inter-changer les deux colonnes de milieu). (c) La deuxième étape : échanger les deux lignes du milieu. (d) Étape 1 et étape 2 pour le deuxième vecteur horizontal/vertical à partir du milieu. (e) Étape 1, 2 pour le dernier vecteur horizontal/vertical à partir du milieu.

for $ii = 1$ **to** s **do**

$k = \frac{mm}{2} + 1$

$j = 1$

for $i = \frac{mm}{2}$ **to** n **do**

$img2$(:,[i $vecleft$(ii,j) k $vecright$(ii,j)],ii)=$img2$(:,[$vecleft$(ii,j) i $vecright$(ii,j) k],ii)

$img2$([i $vecup$(ii,j) k $vecdown$(ii,j)], :,ii)=$img2$(:,[$vecup$(ii,j) i $vecdown$(ii,j) k],ii)

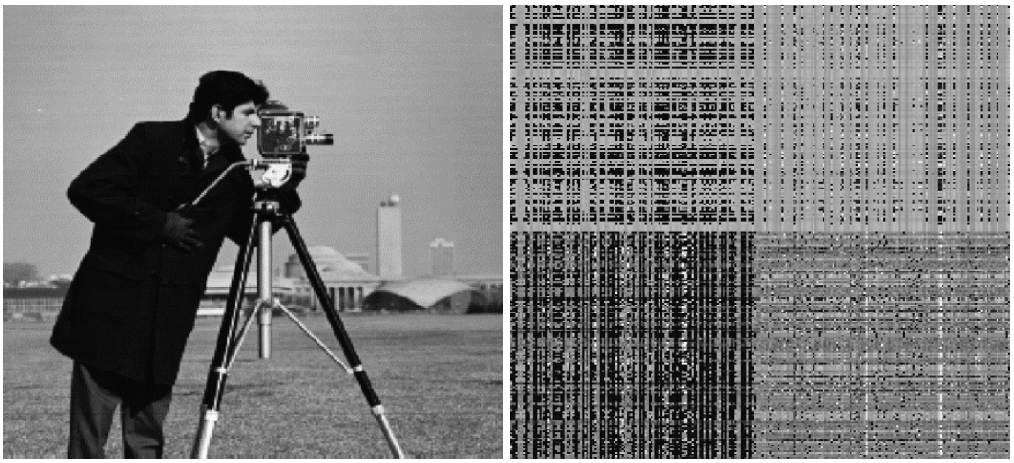
$k = k + 1$

$j = j + 1$

end

end

Cette méthode débute par le milieu jusqu'à balayer tous les quarts (haut-bas et droite-gauche). La figure ci-dessus représente l'application de notre dernier processus directement sur l'image originale de cameraman pour mieux voir à quoi il s'agit.



(a) (b)

FIGURE 4.7 – Résultat de l'application de la dernière étape directement sur l'image originale.

4.4.2 Schéma de décryptage

L'algorithme de décryptage est le même que l'algorithme de cryptage mais inversement, en partant de la dernière méthode vers la première méthode (les méthodes mentionnée ci-dessus). Pour cela, il faut remettre les vecteurs horizontaux et verticaux à leurs places, puis appliquer l'opération XOR avec MATRIX3 ainsi qu'une opération de décalage circulaire inverse est effectuée simultanément. En outre, en fonction des vecteurs d'emplacement (de localisation) des pixels, on effectue le cryptage inverse par l'arithmétique modulaire et XOR. Sans oublier que cette opération va être précédée et achevée par deux opérations xor récursives.

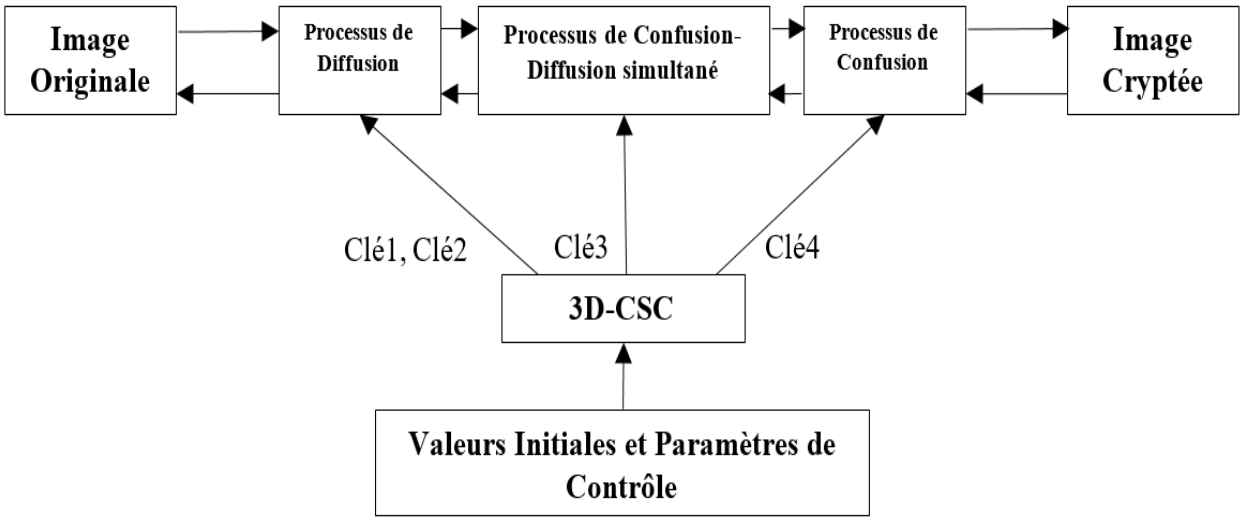


FIGURE 4.8 – Système proposé de cryptage/décryptage.

4.5 Résultats expérimentaux et analyse de la sécurité

Pour évaluer la performance du système proposé, nous effectuons de multiples tests. À noter que nous pouvons appliquer le schéma proposé à différents types d'images, provenant de plusieurs bases de données, MATLAB, USC-SIPI et autres, ainsi que de la vie réelle.

4.5.1 Analyse d'histogrammes

Un histogramme d'image est une représentation graphique de la distribution des valeurs des pixels, où il révèle certaines informations statistiques sur l'image. Une image cryptée par un système de cryptage d'image sécurisé doit avoir un histogramme plat, afin de résister à toute attaque statistique.

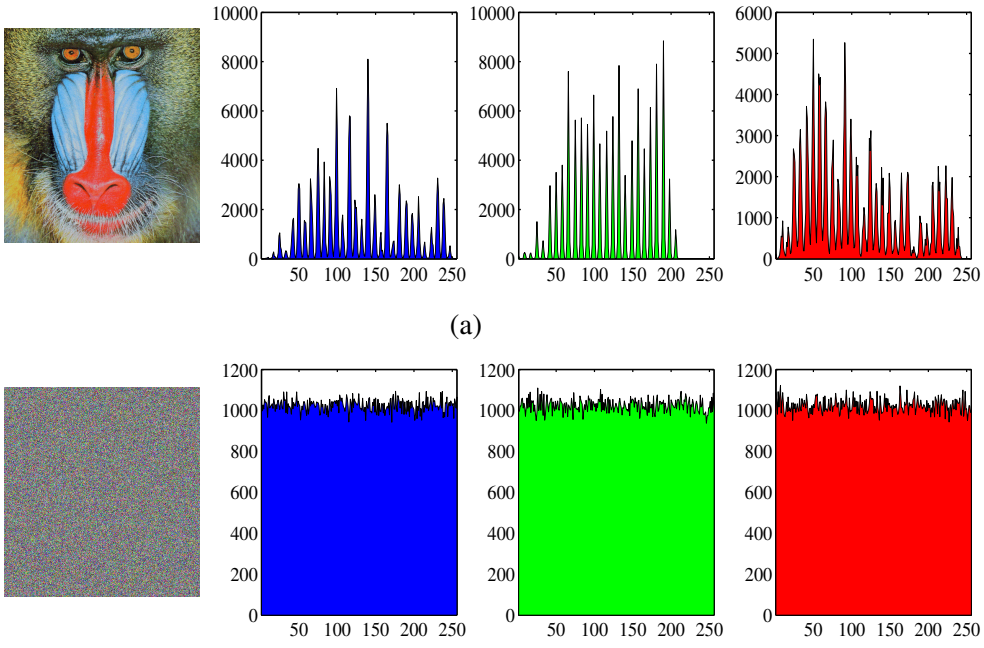


FIGURE 4.9 – Histogrammes d'une image couleur en clair et de son image chiffrée.

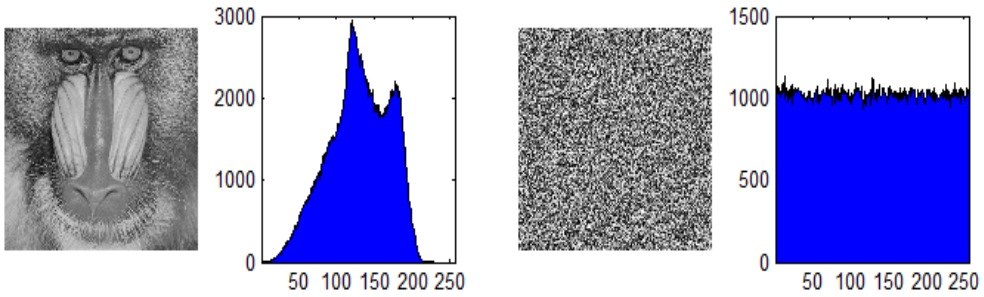


FIGURE 4.10 – Histogrammes d'une image en niveau de gris en clair et de son image chiffrée.

Figures 4.9 et 4.10 présentent les histogrammes des images claires/chiffrées en couleur et en niveaux de gris. Nous pouvons constater que les histogrammes des images chiffrées sont uniformes et significativement différents de ceux de leurs images claires respectives. Par conséquent, aucune information statistique ne peut être extraite de ces histogrammes.

4.5.2 Analyse des coefficients de corrélation

En analysant les valeurs des coefficients de corrélation du tableau 4.3, et les graphes de corrélation de la figure 4.11, nous pouvons clairement constater qu'il y a une grande différence entre les valeurs des pixels voisins dans les trois directions (horizontale, verticale et diagonale) de l'image cryptée, contrairement à celles de l'image originale, et que la comparaison des résultats de notre algorithme avec les algorithmes de l'état de l'art prouve non seulement qu'il n'y a pas de corrélation dans l'image cryptée, mais aussi que les coefficients de corrélation de notre méthode sont inférieurs à ceux des autres références citées.

TABLEAU 4.3 – Comparaison des résultats obtenus du coefficient de corrélation entre la méthode proposée et d'autres méthodes.

Images	verticale	Horizontale	Diagonale
Image claire "peppers"	0.9893	0.9933	0.9855
Notre méthode	-0.0013	-0.0008	-0.0006
Réf [107]	0.0012	-0.0021	-0.0015
Réf [108]	-0.0015	-0.0009	-0.0024
Réf [109]	0.0015	0.0016	-0.0015
Réf [75]	-0.0140	0.0024	-0.0009
Réf [114]	0.0008	-0.0006	0.0002

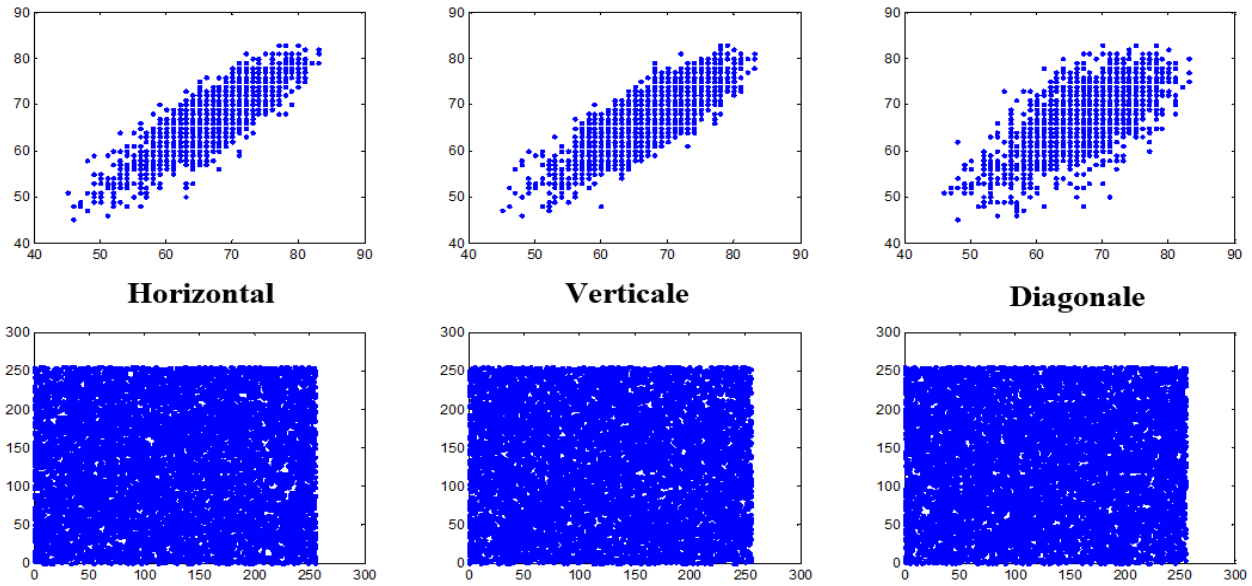


FIGURE 4.11 – Distribution des pixels voisins dans différentes directions. La première ligne représente l'image claire, la deuxième ligne représente l'image chiffrée.

4.5.3 Analyse de l'entropie de l'information

L'information d'entropie est l'un des importants paramètres ; elle est définie comme la mesure de l'incertitude du caractère aléatoire. Plus les résultats de l'entropie sont proches de 8, plus la distribution aléatoire dans l'image cryptée est importante. Comme le montrent les tableaux 4.4 et 4.5 ; les valeurs d'entropie du système proposé sont supérieures à la valeur moyenne mentionnée dans [115]. En outre, en comparant avec d'autres références, nous pouvons confirmer que notre système est à la hauteur.

TABLEAU 4.4 – Analyse d'entropie de l'information dans le cas d'images en couleur.

Images	Image claire	Notre méthode	Réf [107]	Réf [108]	Réf [109]	Réf [75]
Football	7.1796	7.9993	7.9992	7.9993	7.9992	7.9993
Airelane	7.7622	7.9998	7.9998	7.9997	7.9997	7.9998

TABLEAU 4.5 – Analyse d'entropie de l'information dans le cas d'images en niveaux de gris.

Images	Plain image	Notre méthode	Réf [116]	Réf [74]	Réf [75]
Fingerprint	1.2117	7.9993	7.9975	7.9972	7.9985
Image médicale	5.4658	7.9993	7.9993	7.9993	7.9993
Barbara (256×256)	7.4948	7.9974	7.9972	7.9975	7.9977
Image noire (256×256)	0	7.9972	7.9972	7.9974	7.9969

4.5.4 Attaque différentielle

Les attaques différentielles sont une sorte d'attaque en texte clair choisi ; elles peuvent être considérées comme l'un des défis de la sécurité les plus intéressants. Le taux de changement du nombre de pixels (NPCR) et l'intensité moyenne unifiée changée (UACI) sont deux mesures utilisées pour évaluer la sensibilité d'un schéma de cryptage d'image vis-à-vis de ce type de cryptanalyse. En fait, les valeurs théoriques de NPCR et UACI sont autour de 100% et 33,46% respectivement. D'après les tableaux 4.6, 4.7, et 4.8, 4.9, nous pouvons remarquer que les valeurs de NPCR et UACI du schéma proposé sont trop proches des deux valeurs optimales. En outre, la comparaison avec les résultats des autres travaux, montre que nos résultats sont dans la gamme des bonnes valeurs. Et donc, notre système proposé peut résister aux attaques différentielles.

TABLEAU 4.6 – Résultats de l'analyse de NPCR.

Images	Notre méthode	Réf [107]	Réf [108]	Réf [109]	Réf [75]
Lena (512×512)	R :99.59	R :99.60	R :99.60	R :99.63	R : 99.63
	G :99.64	G :99.61	G :99.56	G :99.60	G : 99.60
	B :99.56	B :99.60	B :99.61	B :99.61	B : 99.61
Pears	R :99.60	R :99.61	R :99.60	R :99.61	R :99.61
	G :99.23	G :99.59	G :99.60	G :99.58	99.58
	B :99.38	B :99.62	B :99.62	B :99.59	B : 99.59
Tree	R :99.52	R :99.59	R :99.57	R :99.58	R :99.58
	G :99.62	G :99.62	G :99.60	G :99.62	G :99.62
	B :99.55	B :99.58	B :99.61	B :99.61	B :99.61

TABLEAU 4.7 – Résultats de l'analyse d'UACI.

Images	Notre méthode	Réf [107]	Réf [108]	Réf [109]	Réf [75]
Lena (512×512)	R :33.47	R :33.41	R :33.54	R :33.35	R : 33.47
	G :32.83	G :33.47	G :33.44	G :33.51	G :30.35
	B :32.67	B :33.43	B :33.50	B :33.33	B : 33.46
Pears	R :33.40	R :33.51	R :33.47	R :33.40	R :33.46
	G :33.23	G :33.53	G :33.44	G :33.44	33.45
	B :33.39	B :33.45	B :33.42	B :33.47	B : 33.55
Tree	R :33.54	R :33.53	R :33.46	R :99.43	R :99.43
	G :33.05	G :33.47	G :33.48	G :33.51	G :33.55
	B :33.02	B :33.40	B :33.62	B :33.53	B :33.45

TABLEAU 4.8 – Résultats de NPCR pour les images en niveau de gris.

Images	Notre méthode	Réf [116]	Réf [74]	Réf [75]
Médicale image	99.64	99.62	99.60	99.60
Cameraman	99.60	99.61	99.63	99.63
Circbw	99.53	99.64	100	99.61

TABLEAU 4.9 – Résultats de UACI pour les images en niveau de gris.

Images	Notre méthode	Réf [116]	Réf [74]	Réf [75]
Médicale image	33.47	33.44	33.37	33.50
Cameraman	33.30	33.48	33.32	33.60
Circbw	32.47	33.45	25.48	33.42

4.5.5 Analyse de SSIM et PSNR

Nous avons employé ces deux métriques pour évaluer la qualité de notre schéma proposé. Les valeurs recommandées pour PSNR et SSIM doivent être respectivement inférieures à 10 et proches de 0. Elles ont été calculées entre l'image cryptée et l'image en clair. Les résultats de PSNR et SSIM présentés dans le tableau 4.10 et le tableau 4.11 prouvent la validité de notre système, surtout lorsqu'il est comparé à d'autres travaux antérieurs.

TABLEAU 4.10 – Analyse de SSIM et PSNR pour une image en couleur.

Algorithme	SSIM			PSNR		
	R	G	B	R	G	B
Notre schéma	0.0328	0.0250	0.0251	8.1440	7.3951	7.3154
Réf [107]	0.315	0.0303	0.0337	7.8396	8.5452	9.6134
Réf [108]	0.0301	0.0291	0.0368	8.4606	9.3217	9.0126
Réf [109]	0.0307	0.0286	0.0329	7.8616	8.5517	9.5953
Réf [75]	0.0353	0.0272	0.0339	8.4613	9.2855	9.9980

TABLEAU 4.11 – Analyse de SSIM et PSNR pour une image en niveau de gris.

Algorithme	SSIM	PSNR
Notre schéma	0.0117	8.4275
Réf [116]	0.0289	9.8857
Réf [74]	0.0289	9.2239
Réf [75]	0.0328	9.8830

4.5.6 Analyse de l'espace clé

Puisque la taille de l'espace clé doit être $> 2^{100}$, plus l'espace clé est grand, plus le système de cryptage d'images est sécurisé. Dans notre cas, nous avons douze clés (x_{0i}, y_{0i}, z_{0i}) avec une précision de 10^{16}, les autres douze clés (r_{1i},r_{2i},r_{3i}) sont avec une précision de $10^{(15)}$, sachant que i=1,2,3,4. Leur espace total sera $10^{(16\times12)} \times 10^{(15\times12)} = 10^{(192+180)} \approx 2^{3.322\times372} \approx 2^{1235}$ ce qui est très grand et plus grand que la taille de la clé de certains travaux (tableau 4.12).

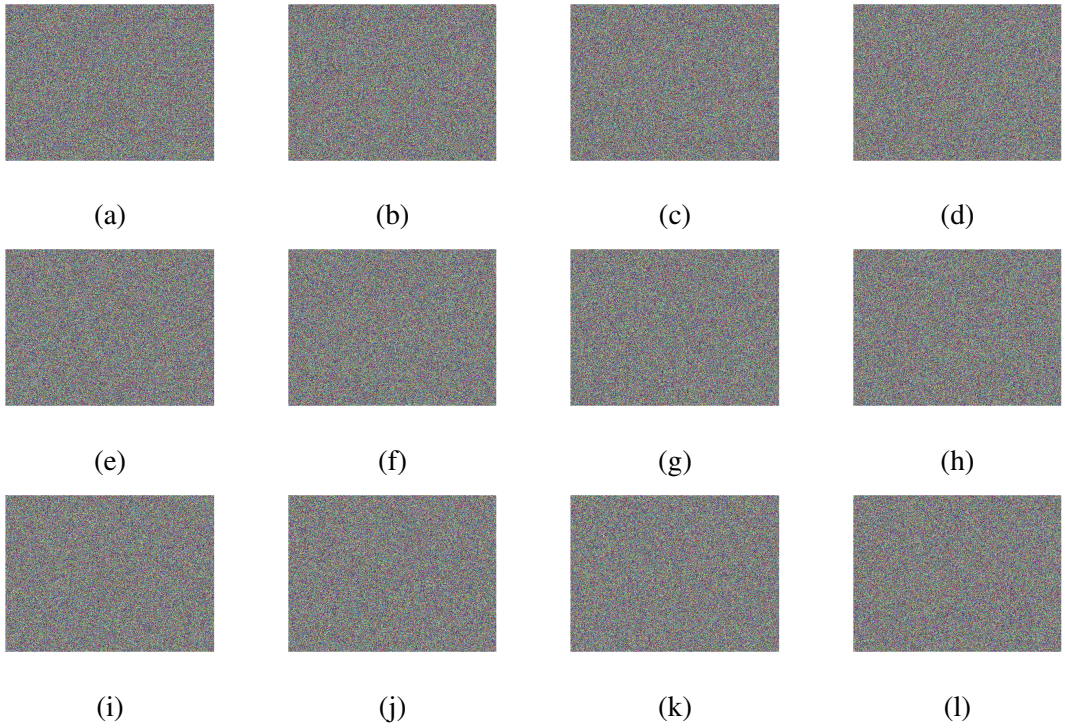
TABLEAU 4.12 – Analyse de l'espace clé.

Schéma	Taille de l'espace clé
Notre méthode	2^{1235}
Réf [109]	2^{138}
Réf [110]	2^{195}
Réf [113]	2^{216}
Réf [104]	2^{391}
Réf [116]	2^{420}
Réf [117]	2^{216}
Réf [114]	2^{279}

4.5.7 Sensibilité de la clé

Plus la sensibilité de la clé est élevée, plus le système de cryptage est performant. Ainsi, on considère la sensibilité de la clé est l'une des caractéristiques de cryptage d'image les plus significatives. Nous allons donc l'étudier en toute minutie. D'abord, à chaque fois, nous ferons un changement minuscule dans une seule valeur de la clé. Pour la clé 1 (x_{01},y_{01}, z_{01}, x_{02}, y_{02}, z_{02}, x_{03}, y_{03}, z_{03}, x_{04}, y_{04}, z_{04}) le changement est de l'ordre de $10^{(-16)}$. Pour la clé 2 (r_{11},r_{12},r_{13}, r_{21}, r_{22}, r_{23}, r_{31}, r_{32}, r_{33}, r_{41}, r_{42}, r_{43}), il vaut $10^{(-15)}$. Les résultats visuels sont présentés dans les figure 4.12 et 4.13, où l'on peut bien observer qu'un tout petit changement, que ce soit dans les valeurs initiales ou dans les paramètres de contrôle endommage complètement l'image décryptée. Quant aux résultats quantitatifs obtenus par le calcul de PSNR de ces images décryptées, ils sont présentés dans le tableau 4.13, où leurs valeurs démontrent que la méthode proposée offre une sécurité suffisante et une bonne imperceptibilité.

Ensuite, nous changeons un seul élément de la clé de chiffrement k_1 pour obtenir $k_1^{'}$. Nous chiffrons la même image par k_1 pour obtenir C1 (Figure 4.14b), puis par $k_1^{'}$ pour obtenir C2 4.14c. Après cela, nous comparons les deux images chiffrées C1 et C2 (Figure 4.14d). Les résultats obtenus en modifiant n'importe quelle partie de la clé sont tous dans la plage acceptable de NPCR/UACI (figure 4.15). Ce qui confirme encore plus la haute sensibilité de la clé, ainsi la capacité de système proposé à résister contre les attaques par force brute.

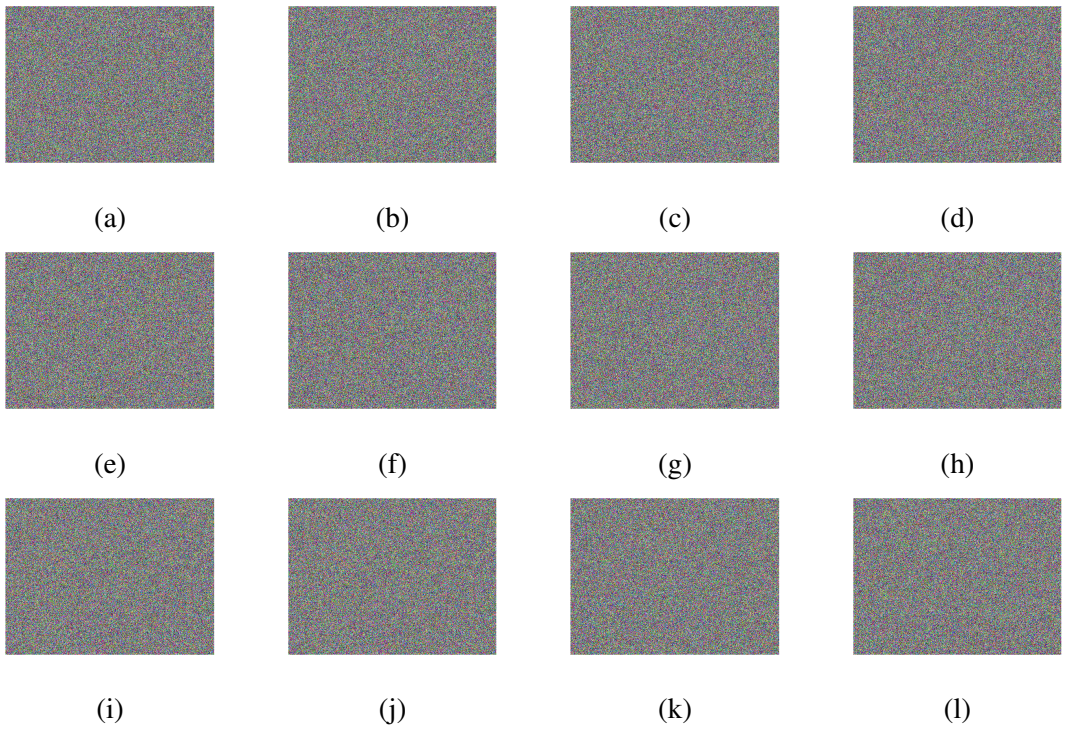


(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

FIGURE 4.12 – Image décryptée de "Peppers" avec un léger changement dans l'une des conditions initiales. (a)$x_{01}^{'} = x_{01} + 10^{-16}$. (b)$y_{01}^{'} = y_{01} + 10^{-16}$. (c)$z_{01}^{'} = z_{01} + 10^{-16}$. (d)$x_{02}^{'} = x_{02} + 10^{-16}$. (e)$y_{02}^{'} = y_{02} + 10^{-16}$. (f)$z_{02}^{'} = z_{02} + 10^{-16}$. (g)$x_{03}^{'} = x_{03} + 10^{-16}$. (h)$y_{03}^{'} = y_{03} + 10^{-16}$.(i)$z_{03}^{'} = z_{03} + 10^{-16}$.(j)$x_{04}^{'} = x_{04} + 10^{-16}(k)y_{04}^{'} = y_{04} + 10^{-16}$.(l)$z_{04}^{'} = z_{04} + 10^{-16}$



(a) (b) (c) (d)

(e) (f) (g) (h)

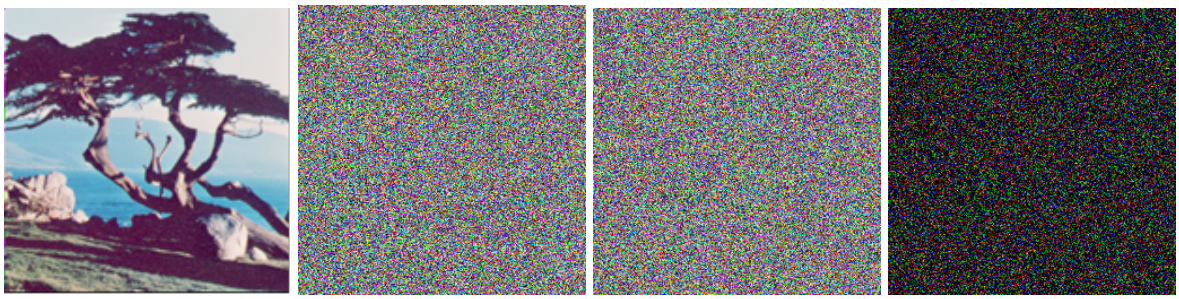
(i) (j) (k) (l)

FIGURE 4.13 – Image décryptée de "Peppers" avec un léger changement dans l'un des paramètres de contrôle. (a) $r_{11}^{'} = r_{11}+10^{-15}$. (b) $r_{12}^{'} = r_{12}+10^{-15}$. (c) $r_{13}^{'} = r_{13}+10^{-15}$. (d) $r_{21}^{'} = r_{21}+10^{-15}$. (e) $r_{22}^{'} = r_{22} + 10^{-15}$. (f) $r_{23}^{'} = r_{23} + 10^{-15}$. (g) $r_{31}^{'} = r_{31} + 10^{-15}$. (h) $r_{32}^{'} = r_{32} + 10^{-15}$.(i) $r_{33}^{'} = r_{33} + 10^{-15}$.(j) $r_{41}^{'} = r_{41} + 10^{-15}$. (k) $r_{42}^{'} = r_{42} + 10^{-15}$.(l) $r_{43}^{'} = r_{43} + 10^{-15}$

.

TABLEAU 4.13 – Valeurs de PSNR après une petite modification dans un élément de la clé.

Valeurs initiales	PSNR			Paramètres de contrôle	PSNR		
	R	G	B		R	G	B
x_{01}	8.1624	7.3992	7.3116	r_{11}	8.1625	7.3796	7.3259
y_{01}	8.1570	7.4127	7.3109	r_{12}	8.1824	7.4184	7.3315
z_{01}	8.1665	7.4280	7.3248	r_{13}	8.1934	7.4305	7.3437
x_{02}	8.1501	7.3861	7.2998	r_{21}	8.1729	7.4358	7.3322
y_{02}	8.1455	7.4074	7.3140	r_{22}	8.1993	7.4226	7.3328
z_{02}	8.1854	7.4113	7.3163	r_{23}	8.1594	7.4139	7.3220
x_{03}	8.1474	7.4033	7.3296	r_{31}	7.3153	7.4051	8.1736
y_{03}	8.1509	7.4163	7.3308	r_{32}	8.1576	7.4104	7.3141
z_{03}	8.1535	7.3947	7.3140	r_{33}	8.1708	7.3951	7.3294
x_{04}	8.1776	7.4372	7.3337	r_{41}	8.1545	7.4086	7.3173
y_{04}	8.1923	7.4226	7.3586	r_{42}	8.1420	7.4167	7.3038
z_{04}	8.1756	7.4207	7.3521	r_{43}	8.1791	7.3925	7.3264



(a) (b) (c) (d)

FIGURE 4.14 – Test par attaque en texte clair (a) Image claire. (b) Image chiffrée par la clé k_1. (c) Image chiffrée par la clé k_2. (d) Image de différence entre C1 et C2.

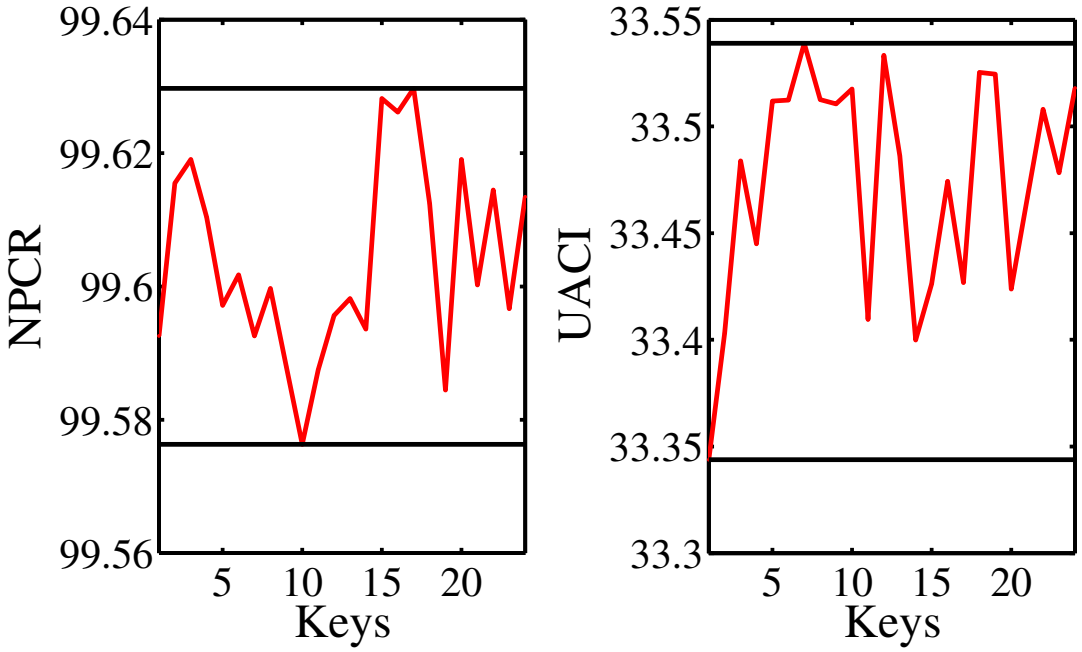


FIGURE 4.15 – Résultat d'analyse de la sensibilité de la clé secrète en employant NPCR et UACI dans la phase de décryptage.

4.5.8 Analyse de vitesse

La rapidité est un facteur primordial dans le domaine de télécommunications en général, encore moins quand il s'agit de la sécurité. Dans notre cas, l'algorithme que nous proposons assure un niveau élevé de rapidité par rapport aux autres travaux disponibles. Le tableau 4.14 expose le temps d'exécution de notre système, comparé à d'autre précédentes recherches pour une image de taille (256×256).

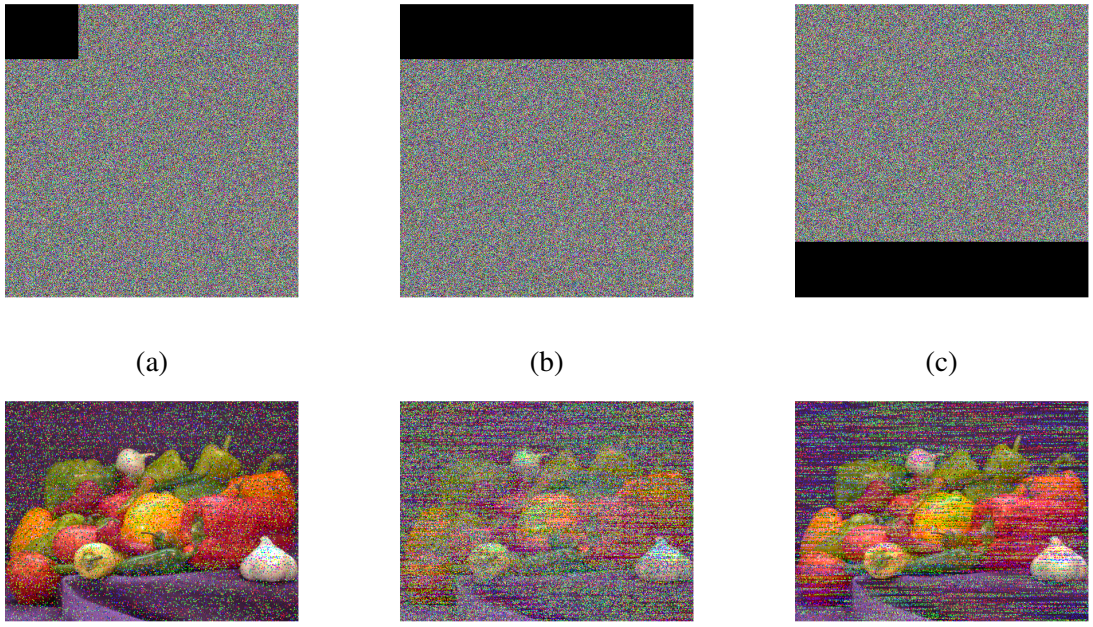
TABLEAU 4.14 – Temps d'exécution.

Notre méthode	Réf [118]	Réf [119]	Réf [120]
0.386794	0.40585	2.2234	0.42601

4.5.9 Perte de données

La perte de données est un très courant phénomène, dans lequel les données sont endommagées, effacées ou rendues méconnaissables ; elle pourrait être due à un acte intentionnel ou accidentel. Dans cette partie, nous allons provoquer des pertes de données dans l'image chiffrée comme illustré sur les figures 4.16a, 4.16b et 4.16c. Où nous pouvons explicitement voir que les images décryptées résultantes sont encore reconnaissables, malgré le grand espace de perte. Ces constatations nous amènent à décider que l'algorithme proposé peut résister à n'importe quelle suppression éventuelle

de données.



(a) (b) (c)

(d) (e) (f)

FIGURE 4.16 – Test de perte de données avec : (a) 96×128 de données perdues, (b) 96×512 de données perdues, de haut de l'image cryptée, (c) 96×512 de données perdues, de bas de l'image cryptée. images décryptées correspondantes(d), (e), (f) respectivement.

4.6 Conclusion

Ce chapitre est dédié à la présentation de notre système tridimensionnel (3D-CSC), construit à partir de la combinaison des fonctions simple (sinus et cosinus), et facile à implémenter. Il exhibe un niveau élevé de comportement chaotique sur une plage infinie de ses trois paramètres de contrôle. L'efficacité de ce système a été prouvée par le biais de divers tests de performance, tels que les diagrammes de bifurcations, l'exposant de Lyapunov, l'analyse de la sensibilité aux conditions initiales et aux paramètres de contrôle, et les tests statistiques. La particularité de notre système ressort dans la possibilité d'étendre sa dimension jusqu'à 3 fois N (N est presque illimité), en gardant la même haute performance et avec plus de paramètres et de valeurs initiales, c.à.d plus de clés. Nous avons proposé aussi, un nouveau schéma de cryptage d'images couplé avec 3D-CSC. Les résultats expérimentaux et l'analyse de sécurité démontrent que cette méthode qui base essentiellement sur la nouvelle carte chaotique 3D pour la génération des clés est appropriée pour protéger les informations des images contre tout type de cryptanalyse. De plus, elle possède un grand espace de clé (2^{1100}), une haute sensibilité et une bonne imprévisibilité. Surtout lorsqu'on la compare à d'autres schémas de cryptage d'images. Il est à noter également que cette méthode

pourrait être appliquée sur plusieurs types d'images, ce qui signifie l'utilisabilité dans de multiples domaines.

Conclusion générale

Le cryptage d'image à base des systèmes chaotiques est un sujet largement discuté dans le monde de la sécurité informatique, de nombreuses approches ont été communiquées dans ce cadre, car ils représentent une solution efficace aux failles de la sécurité. Cependant, Les systèmes chaotiques utilisés ont connu beaucoup d'inconvénients tels que la petite plage chaotique. De plus, les schémas proposés souffrent de plusieurs insuffisances comme la difficulté de leur mise en pratique, la complexité et le temps d'exécution, la compatibilité avec tous les types d'images, ainsi que la taille de l'espace clé. C'est dans ce contexte et dans le but de répondre à ces exigences et satisfaire ces besoins de la sécurité, que nous avons intervenu dans cette thèse de doctorat avec deux contributions différentes. En premier lieu, nous avons introduit une nouvelle carte chaotique, nommée 1-DCE, construite à partir de la carte cubique classique, du modulaire arithmétique et de la fonction exponentielle. Ensuite, nous avons étudié son usage dans un système de cryptage d'images basé sur la confusion et une nouvelle approche de diffusion. Comme seconde contribution, nous avons présenté une nouvelle carte chaotique 3D, à base des fonctions trigonométriques (sinus et cosinus), on l'a donné le nom, 3D-CSC. Et puis, nous l'avons implémenté dans un nouveau schéma de cryptage d'images reposant sur deux processus de confusion-diffusion.

L'évaluation de 1-DCE et 3D-CSC à l'aide de nombreux tests de performance a montré leur haute sensibilité aux conditions initiales et aux paramètres de contrôle, un très large intervalle des paramètres de contrôle, et un comportement extrêmement chaotiques. Quant aux deux schémas suggérés. Les résultats de la simulation et l'analyse de la sécurité ont prouvé que ces deux approches peuvent résister efficacement à de nombreuses attaques statistiques, différentielles et en texte clair, comme elles disposent un grand espace de clés et une forte robustesse contre la perte des données. Donc, un excellent niveau de sécurité. A noter que tous ces deux algorithmes présentés peuvent être appliqués à plusieurs types d'images, ce qui signifie qu'ils peuvent être utilisés dans de multiples domaines.

En perspective, nous envisageons à implémenter nos deux approches sur un circuit FPGA,

et à les tester avec d'autres types d'attaques pour jauger encore plus leur résistance. Vu que les images satellitaires ne sont pas sécurisées, nous comptons leur appliquer un système de cryptage, à base de techniques plus sophistiquées, comme les courbes elliptiques et les réseaux de neurones. Nous envisageons aussi à concevoir un système de cryptage des vidéos, vu le nombre décent de contributions dans ce domaine, malgré son importance accrue. Dernièrement, le développement des métriques d'évaluation de la sécurité est un sujet qui nous tient à cœur car, il est constaté qu'elles sont loin d'être efficaces pour l'analyse de la sécurité.

Bibliographie

[1] A. Yahi, T. Bekkouche, N. Diffellah, and M. E. H. Daachi, "Biometric image encryption scheme based on modified double random phase encoding system," in *Conference Proceedings ICCSA'2021*, p. 15.

[2] B. Norman, *Secret warfare ; : The battle of codes and ciphers*. Acropolis Books, 1973.

[3] R. Matthews, "On the derivation of a "chaotic" encryption algorithm," *Cryptologia*, vol. 13, no. 1, pp. 29–42, 1989.

[4] X.-Y. Wang, Y.-Q. Zhang, and L.-T. Liu, "An enhanced sub-image encryption method," *Optics and Lasers in Engineering*, vol. 86, pp. 248–254, 2016.

[5] J. Gleick, *La théorie du chaos :vers une nouvelle science*. Science d'aujourd'hui, 1989.

[6] H. Poincaré, *Calcul des probabilités 2ème édition*. JACQUES GABAY, 1987.

[7] J. Marsden and M. McCracken, "The hopf bifurcation and its applications springer," *New York etc*, 1976.

[8] D. Ruelle, "Small random perturbations of dynamical systems and the definition of attractors," *Communications in Mathematical Physics*, vol. 82, no. 1, pp. 137–151, 1981.

[9] D. Ruelle and F. Takens, "On the nature of turbulence," *Communications in Mathematical Physics*, vol. 20, pp. 167–192, 1971.

[10] J.-L. Pac, *Systèmes dynamiques-Cours et exercices corrigés*. DUNOD, 2016.

[11] X. T. Thun, *Le chaos et L'harmonie :la fabrication du réel*. Fayard, 1998.

[12] H. Dang-Vu and C. Delcarte, *Bifurcations et chaos : une introduction à la dynamique contemporaine avec des programmes en Pascal, Fortran et Mathematica*. Ellipses Ed. Marketing, 2000.

[13] J. Guckenheimer and P. Holmes, *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*, vol. 42. Springer Science & Business Media, 2013.

[14] C.-M. Marle, *Systèmes dynamiques-Une introduction*. ellipses, 2003.

[15] P. Manneville, "Systèmes dynamiques et chaos," *LadHyX, école Polytechnique*, 1998.

[16] J. E. Gentle, *Random number generation and Monte Carlo methods*, vol. 381. Springer, 2003.

[17] J. Walker, "Ent : a pseudorandom number sequence test program," *Software and documentation available at/www. fourmilab. ch/random/S*, 2008.

[18] E. Almaraz Luengo, B. Alaña Olivares, L. J. García Villalba, and J. Hernández-Castro, "Weaknesses in ent battery design," *Applied Sciences*, vol. 12, no. 9, p. 4230, 2022.

[19] G. Marsaglia, "Random number cdrom including the diehard battery of tests of randomness," *http ://stat. fsu. edu/pub/diehard/cdrom*, 1995.

[20] G. Marsalgia, "Diehard : a battery of tests of randomness (1997)."

[21] G. Marsaglia, "Keynote address : a current view of random number generators, proceedings, computer science and statistics," in *16th Symp. Interface*, Elsevier, 1985.

[22] G. Marsaglia and A. Zaman, "Monkey tests for random number generators," *Computers & mathematics with applications*, vol. 26, no. 9, pp. 1–10, 1993.

[23] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," tech. rep., Booz-allen and hamilton inc mclean va, 2001.

[24] C. Doty-Humphrey, "Practrand (practically random), a c++ library of pseudorandom number generators(prngs) and statistical tests for prngs," 2022.

[25] R. Álvarez, F. Martínez, and A. Zamora, "Improving the statistical qualities of pseudo random number generators," *Symmetry*, vol. 14, no. 2, p. 269, 2022.

[26] V. Lynnyk, *Chaos-based communication systems*. PhD thesis, Faculty of Electrical Engineering Department of Control Engineering, Prague, 2010.

[27] D. Bernardini and G. Litak, "An overview of 0–1 test for chaos," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 38, no. 5, pp. 1433–1450, 2016.

[28] G. A. Gottwald and I. Melbourne, "On the implementation of the 0–1 test for chaos," *SIAM Journal on Applied Dynamical Systems*, vol. 8, no. 1, pp. 129–145, 2009.

[29] F. V. Waugh, "Cobweb models," *American Journal of Agricultural Economics*, vol. 46, no. 4, pp. 732–750, 1964.

[30] M. Alawida, A. Samsudin, J. S. Teh, *et al.*, "Digital cosine chaotic map for cryptographic applications," *IEEE Access*, vol. 7, pp. 150609–150622, 2019.

[31] R. M. May, "Simple mathematical models with very complicated dynamics," in *The Theory of Chaotic Attractors*, pp. 85–93, Springer, 2004.

[32] N. K. Pareek, V. Patidar, and K. K. Sud, "Image encryption using chaotic logistic map," *Image and vision computing*, vol. 24, no. 9, pp. 926–934, 2006.

[33] M. Murillo-Escobar, C. Cruz-Hernández, L. Cardoza-Avendaño, and R. Méndez-Ramírez, "A novel pseudorandom number generator based on pseudorandomly enhanced logistic map," *Nonlinear Dynamics*, vol. 87, no. 1, pp. 407–425, 2017.

[34] T. Yoshida, H. Mori, and H. Shigematsu, "Analytic study of chaos of the tent map : band structures, power spectra, and critical behaviors," *Journal of statistical physics*, vol. 31, no. 2, pp. 279–308, 1983.

[35] I. Campos-Cantón, E. Campos-Cantón, J. Murguía, and H. Rosu, "A simple electronic circuit realization of the tent map," *Chaos, Solitons & Fractals*, vol. 42, no. 1, pp. 12–16, 2009.

[36] C. Zhu and K. Sun, "Cryptanalyzing and improving a novel color image encryption algorithm using rt-enhanced chaotic tent maps," *Ieee Access*, vol. 6, pp. 18759–18770, 2018.

[37] H. Alhumyani, "Efficient image cipher based on baker map in the discrete cosine transform," *Cybernetics and Information Technologies*, vol. 20, no. 1, pp. 68–81, 2020.

[38] V. I. Arnold and A. Avez, *Ergodic problems of classical mechanics*, vol. 9. Benjamin, 1968.

[39] B. Furht, E. Muharemagic, and D. Socek, *Multimedia encryption and watermarking*, vol. 28. Springer Science & Business Media, 2006.

[40] T. Ueta and G. Chen, "Bifurcation analysis of chen's equation," *International Journal of Bifurcation and Chaos*, vol. 10, no. 08, pp. 1917–1931, 2000.

[41] A. Akgul, S. Kacar, I. Pehlivan, and B. Aricioglu, "Chaos-based encryption of multimedia data and design of security analysis interface as an educational tool," *Computer Applications in Engineering Education*, vol. 26, no. 5, pp. 1336–1349, 2018.

[42] S.-T. Wu, "A secure real-time iot data stream based on improved compound coupled map lattices," *Applied Sciences*, vol. 12, no. 17, p. 8489, 2022.

[43] M. Ahmad, M. N. Doja, and M. Beg, "A new chaotic map based secure and efficient pseudo-random bit sequence generation," in *International symposium on security in computing and communication*, pp. 543–553, Springer, 2018.

[44] M. Z. Talhaoui, X. Wang, and M. A. Midoun, "A new one-dimensional cosine polynomial chaotic map and its use in image encryption," *The Visual Computer*, vol. 37, no. 3, pp. 541–551, 2021.

[45] H. Najm, H. Hoomod, and R. Hassan, “A new wot cryptography algorithm based on gost and novel 5d chaotic system,” 2021.

[46] B. A. Mezatio, M. T. Motchongom, B. R. W. Tekam, R. Kengne, R. Tchitnga, and A. Fomethe, “A novel memristive 6d hyperchaotic autonomous system with hidden extreme multistability,” *Chaos, Solitons & Fractals*, vol. 120, pp. 100–115, 2019.

[47] Q. Yang and M. Bai, “A new 5d hyperchaotic system based on modified generalized lorenz system,” *Nonlinear Dynamics*, vol. 88, no. 1, pp. 189–221, 2017.

[48] S. Singh, *Histoire des codes secrets : de l’Égypte des pharaons à l’ordinateur quantique*. JC Lattès, 2014.

[49] J. Dumas, J. Roch, E. Tannier, and S. Varrette, “Théorie des codes : Compression, cryptage et correction. collection sciences sup. dunod,” *Mars*, vol. 352, 2007.

[50] F. E. Abd El-Samie, H. E. H. Ahmed, I. F. Elashry, M. H. Shahieen, O. S. Faragallah, E.-S. M. El-Rabaie, and S. A. Alshebeili, *Image encryption : a communication perspective*. CRC Press, 2013.

[51] B. Schneier, “Cryptographie appliquée : Algorithmes, protocoles et codes source en c. 2eme édition.”

[52] P. Guillot, “Auguste kerckhoffs et la cryptographie militaire,” *Bibnum. Textes fondateurs de la science*, 2013.

[53] C. E. Shannon, “Communication theory of secrecy systems,” *The Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.

[54] S. Cimato and C.-N. Yang, *Visual cryptography and secret image sharing*. CRC press, 2017.

[55] G. S. Vernam, “Cipher printing telegraph systems : For secret wire and radio telegraphic communications,” *Journal of the AIEE*, vol. 45, no. 2, pp. 109–115, 1926.

[56] T. Young, “The bakerian lecture. on the theory of light and colours,” in *Abstracts of the Papers Printed in the Philosophical Transactions of the Royal Society of London*, no. 1, pp. 63–67, The Royal Society London, 1832.

[57] C. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin, “" experimental quantum cryptography" journal of cryptology vol. 5, no. 1,” 1992.

[58] P. W. Shor and J. Preskill, “Simple proof of security of the bb84 quantum key distribution protocol,” *Physical review letters*, vol. 85, no. 2, p. 441, 2000.

[59] C. H. Bennett, G. Brassard, and J.-M. Robert, “Privacy amplification by public discussion,” *SIAM journal on Computing*, vol. 17, no. 2, pp. 210–229, 1988.

[60] M. Salleh, S. Ibrahim, and I. F. Isnin, "Enhanced chaotic image encryption algorithm based on baker's map," in *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS'03.*, vol. 2, pp. II–II, IEEE, 2003.

[61] A. Palacios and H. Juarez, "Cryptography with cycling chaos," *Physics Letters A*, vol. 303, no. 5-6, pp. 345–351, 2002.

[62] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International journal of bifurcation and chaos*, vol. 16, no. 08, pp. 2129–2151, 2006.

[63] C. Cao, K. Sun, and W. Liu, "A novel bit-level image encryption algorithm based on 2d-licm hyperchaotic map," *Signal Processing*, vol. 143, pp. 122–133, 2018.

[64] J. Chen, F. Han, W. Qian, Y.-D. Yao, and Z.-l. Zhu, "Cryptanalysis and improvement in an image encryption scheme using combination of the 1d chaotic map," *Nonlinear Dynamics*, vol. 93, no. 4, pp. 2399–2413, 2018.

[65] W. Wen, Y. Zhang, M. Su, R. Zhang, J.-x. Chen, and M. Li, "Differential attack on a hyperchaos-based image cryptosystem with a classic bi-modular architecture," *Nonlinear Dynamics*, vol. 87, no. 1, pp. 383–390, 2017.

[66] Y. Wu, J. P. Noonan, S. Agaian, *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)*, vol. 1, no. 2, pp. 31–38, 2011.

[67] A. Hore and D. Ziou, "Image quality metrics : Psnr vs. ssim," in *2010 20th international conference on pattern recognition*, pp. 2366–2369, IEEE, 2010.

[68] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment : from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[69] C. Liu and Q. Ding, "A color image encryption scheme based on a novel 3d chaotic mapping," *Complexity*, vol. 2020, 2020.

[70] F. Masood, J. Ahmad, S. A. Shah, S. S. Jamal, and I. Hussain, "A novel hybrid secure image encryption based on julia set of fractals and 3d lorenz chaotic map," *Entropy*, vol. 22, no. 3, p. 274, 2020.

[71] A. H. Zahid, E. Al-Solami, and M. Ahmad, "A novel modular approach based substitution-box design for image encryption," *IEEE Access*, vol. 8, pp. 150326–150340, 2020.

[72] J. Wu, X. Liao, and B. Yang, "Color image encryption based on chaotic systems and elliptic curve elgamal scheme," *Signal Processing*, vol. 141, pp. 109–124, 2017.

[73] X.-Y. Wang, S.-X. Gu, and Y.-Q. Zhang, "Novel image encryption algorithm based on cycle shift and chaotic system," *Optics and Lasers in Engineering*, vol. 68, pp. 126–134, 2015.

[74] D. Herbadji, N. Derouiche, A. Belmeguenai, A. Herbadji, and S. Boumerdassi, "A tweakable image encryption algorithm using an improved logistic chaotic map," *Traitement du Signal*, vol. 36, no. 5, pp. 407–417, 2019.

[75] A. Mansouri and X. Wang, "A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme," *Information Sciences*, vol. 520, pp. 46–62, 2020.

[76] M. Z. Talhaoui, X. Wang, and A. Talhaoui, "A new one-dimensional chaotic map and its application in a novel permutation-less image encryption scheme," *The Visual Computer*, vol. 37, no. 7, pp. 1757–1768, 2021.

[77] Z. Ni, X. Kang, and L. Wang, "A novel image encryption algorithm based on bit-level improved arnold transform and hyper chaotic map," in *2016 IEEE International Conference on Signal and Image Processing (ICSIP)*, pp. 156–160, IEEE, 2016.

[78] Y. Liu, Z. Jiang, X. Xu, F. Zhang, and J. Xu, "Optical image encryption algorithm based on hyper-chaos and public-key cryptography," *Optics & Laser Technology*, vol. 127, p. 106171, 2020.

[79] L.-H. Gong, H.-X. Luo, R.-Q. Wu, and N.-R. Zhou, "New 4d chaotic system with hidden attractors and self-excited attractors and its application in image encryption based on rng," *Physica A : Statistical Mechanics and its Applications*, vol. 591, p. 126793, 2022.

[80] A. M. Rucklidge, "Chaos in models of double convection," *Journal of Fluid Mechanics*, vol. 237, pp. 209–229, 1992.

[81] M. Kaur, D. Singh, and V. Kumar, "Color image encryption using minimax differential evolution-based 7d hyper-chaotic map," *Applied Physics B*, vol. 126, no. 9, pp. 1–19, 2020.

[82] Q. Yang, D. Zhu, and L. Yang, "A new 7d hyperchaotic system with five positive lyapunov exponents coined," *International Journal of Bifurcation and Chaos*, vol. 28, no. 05, p. 1850057, 2018.

[83] M. Kaur, D. Singh, and V. Kumar, "Color image encryption using minimax differential evolution-based 7d hyper-chaotic map," *Applied Physics B*, vol. 126, no. 9, pp. 1–19, 2020.

[84] S. Zhang and T. Gao, "An image encryption scheme based on dna coding and permutation of hyper-image," *Multimedia Tools and Applications*, vol. 75, no. 24, pp. 17157–17170, 2016.

[85] R. Enayatifar, A. H. Abdullah, and I. F. Isnin, "Chaos-based image encryption using a hybrid genetic algorithm and a dna sequence," *Optics and Lasers in Engineering*, vol. 56, pp. 83–93, 2014.

[86] J. D. Watson and F. H. Crick, "Molecular structure of nucleic acids : a structure for deoxyribose nucleic acid," *Nature*, vol. 171, no. 4356, pp. 737–738, 1953.

[87] C. Lin and A. Kumar, "Matching contactless and contact-based conventional fingerprint images for biometrics identification," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 2008–2021, 2018.

[88] S. Zhang and T. Gao, "An image encryption scheme based on dna coding and permutation of hyper-image," *Multimedia Tools and Applications*, vol. 75, no. 24, pp. 17157–17170, 2016.

[89] A. N. Kengnou Telem, H. B. Fotsin, and J. Kengne, "Image encryption algorithm based on dynamic dna coding operations and 3d chaotic systems," *Multimedia Tools and Applications*, vol. 80, no. 12, pp. 19011–19041, 2021.

[90] Y. Zhang, "The image encryption algorithm based on chaos and dna computing," *Multimedia Tools and Applications*, vol. 77, no. 16, pp. 21589–21615, 2018.

[91] T. Bekkouche and S. Bouguezel, "A recursive non-linear pre-encryption for opto-digital double random phase encoding," *Optik*, vol. 158, pp. 940–950, 2018.

[92] T. Bekkouche and S. Bouguezel, "Digital double random amplitude image encryption method based on the symmetry property of the parametric discrete fourier transform," *Journal of Electronic Imaging*, vol. 27, no. 2, p. 023033, 2018.

[93] A. B. Joshi, D. Kumar, A. Gaffar, and D. Mishra, "Triple color image encryption based on 2d multiple parameter fractional discrete fourier transform and 3d arnold transform," *Optics and Lasers in Engineering*, vol. 133, p. 106139, 2020.

[94] J. Lang, "Color image encryption based on color blend and chaos permutation in the reality-preserving multiple-parameter fractional fourier transform domain," *Optics Communications*, vol. 338, pp. 181–192, 2015.

[95] K. Ratnavelu, M. Kalpana, P. Balasubramaniam, K. Wong, and P. Raveendran, "Image encryption method based on chaotic fuzzy cellular neural networks," *Signal Processing*, vol. 140, pp. 87–96, 2017.

[96] U. Erkan, A. Toktas, S. Enginoğlu, E. Akbacak, and D. N. Thanh, "An image encryption scheme based on chaotic logarithmic map and key generation using deep cnn," *Multimedia Tools and Applications*, vol. 81, no. 5, pp. 7365–7391, 2022.

[97] A. M. Abbas, A. A. Alharbi, and S. Ibrahim, "A novel parallelizable chaotic image encryption scheme based on elliptic curves," *IEEE Access*, vol. 9, pp. 54978–54991, 2021.

[98] P. Parida, C. Pradhan, X.-Z. Gao, D. S. Roy, and R. K. Barik, "Image encryption and authentication with elliptic curve cryptography and multidimensional chaotic maps," *IEEE Access*, vol. 9, pp. 76191–76204, 2021.

[99] M. Ramzan, T. Shah, M. M. Hazzazi, A. Aljaedi, and A. R. Alharbi, "Construction of s-boxes using different maps over elliptic curves for image encryption," *IEEE Access*, vol. 9, pp. 157106–157123, 2021.

[100] W. Steeb, F. Solms, T. K. Shi, and R. Stoop, "Cubic map, complexity and ljapunov exponent," *Physica Scripta*, vol. 55, no. 5, p. 520, 1997.

[101] R. M. May and G. F. Oster, "Bifurcations and dynamic complexity in simple ecological models," *The American Naturalist*, vol. 110, no. 974, pp. 573–599, 1976.

[102] J. C. Sprott and J. C. Sprott, *Chaos and time-series analysis*, vol. 69. Oxford university press Oxford, 2003.

[103] A. Mansouri and X. Wang, "A novel one-dimensional chaotic map generator and its application in a new index representation-based image encryption scheme," *Information Sciences*, vol. 563, pp. 91–110, 2021.

[104] M. A. Midoun, X. Wang, and M. Z. Talhaoui, "A sensitive dynamic mutual encryption system based on a new 1d chaotic map," *Optics and Lasers in Engineering*, vol. 139, p. 106485, 2021.

[105] G. Marsaglia, "Random number generators," *Journal of Modern Applied Statistical Methods*, vol. 2, no. 1, p. 2, 2003.

[106] R. Li, Q. Liu, and L. Liu, "Novel image encryption algorithm based on improved logistic map," *IET Image Processing*, vol. 13, no. 1, pp. 125–134, 2019.

[107] D. Herbadji, A. Belmeguenai, N. Derouiche, and H. Liu, "Colour image encryption scheme based on enhanced quadratic chaotic map," *IET Image Processing*, vol. 14, no. 1, pp. 40–52, 2020.

[108] R. Parvaz and M. Zarebnia, "A combination chaotic system and application in color image encryption," *Optics & Laser Technology*, vol. 101, pp. 30–41, 2018.

[109] C. Pak and L. Huang, "A new color image encryption using combination of the 1d chaotic map," *Signal Processing*, vol. 138, pp. 129–137, 2017.

[110] T. Bekkouche, N. Diffellah, and L. Ziet, "Hybrid image encryption based on digital pre-encryption and optical single random phase encoding," *Optica Applicata*, vol. 49, no. 4, pp. 559–569, 2019.

[111] T. Bekkouche, N. Diffellah, and S. Mokhnache, "Chaotic image encryption scheme based on permutation inside pixels and xor recursiveness in diffusion," in *Proceedings of the Int, Conf on pattern Analysis and recognition, ICPAR*, vol. 19, pp. 1–5, 2019.

[112] M. L. Sahari and I. Boukemara, "A pseudo-random numbers generator based on a novel 3d chaotic map with an application to color image encryption," *Nonlinear Dynamics*, vol. 94, no. 1, pp. 723–744, 2018.

[113] C. Liu and Q. Ding, "A color image encryption scheme based on a novel 3d chaotic mapping," *Complexity*, vol. 2020, 2020.

[114] A. Yahi, T. Bekkouche, M. E. H. Daachi, and N. Diffellah, "A color image encryption scheme based on 1d cubic map," *Optik*, vol. 249, p. 168290, 2022.

[115] V. M. Silva-García, R. Flores-Carapia, C. Rentería-Márquez, B. Luna-Benoso, C. A. Jiménez-Vázquez, and M. D. González-Ramírez, "Cipher image damage and decisions in real time," *Journal of Electronic Imaging*, vol. 24, no. 1, p. 013012, 2015.

[116] M. Shariatzadeh, M. J. Rostami, and M. Eftekhari, "Proposing a novel dynamic aes for image encryption using a chaotic map key management approach," *Optik*, vol. 246, p. 167779, 2021.

[117] K. Muhammad, J. Ahmad, N. U. Rehman, Z. Jan, and M. Sajjad, "Cisska-lsb : color image steganography using stego key-directed adaptive lsb substitution method," *Multimedia Tools and Applications*, vol. 76, no. 6, pp. 8597–8626, 2017.

[118] A. Girdhar, H. Kapur, and V. Kumar, "A novel grayscale image encryption approach based on chaotic maps and image blocks," *Applied Physics B*, vol. 127, no. 3, pp. 1–12, 2021.

[119] K. Xuejing and G. Zihui, "A new color image encryption scheme based on dna encoding and spatiotemporal chaotic system," *Signal Processing : Image Communication*, vol. 80, p. 115670, 2020.

[120] K. Shahna and A. Mohamed, "A novel image encryption scheme using both pixel level and bit level permutation with chaotic map," *Applied Soft Computing*, vol. 90, p. 106162, 2020.