Fractional chaotic pseudo-random number generator design and application to image cryptosystem

Chunxiao Yang

CENTRALE NANTES

THÈSE DE DOCTORAT DE

L'ÉCOLE CENTRALE DE NANTES

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Automatique, Productique et Robotique*

Par
Chunxiao YANG

Fractional chaotic pseudo-random number generator design and application to image cryptosystem

Thèse présentée et soutenue à Nantes, le 12 Décembre 2022
Unité de recherche : UMR 6004 Laboratoire des Sciences du Numérique de Nantes (LS2N)

Rapporteurs avant soutenance :

Christophe GUYEUX Professeur des universités, Université de Franche-Comté
Sergej ČELIKOVSKÝ Senior research fellow, Czech Academy of Sciences, République Tchèque

Composition du Jury :

Président :	Gilles MILLERIOUX	Professeur des universités, Université de Lorraine
Examinateurs :	Safwan EL ASSAD	Maître de conférences, HDR, Polytech Nantes, Nantes Université
	Jean-Pierre BARBOT	Professeur des universités, ENSEA Cergy-Pontoise
Dir. de thèse :	Jean-Jacques LOISEAU	Directeur de recherche CNRS, École Centrale de Nantes
Co-encadrante de thèse :	Ina TARALOVA	Maître de conférences, École Centrale de Nantes

ACKNOWLEDGEMENT

My time as a doctoral candidate has finally come to an end. I can still vividly remember when I found out I got admitted to this GEC-CSC (Groupe Ecole Centrale - Chinese Scholarship Council) program. After months of preparation and interviews with different professors and committee panels from GEC, I felt relieved, but at the same time very thrilled, when I finally got the admission letter from CSC. I was not wrong in being excited since a new journey was about to begin.

Looking back on my four years of Ph.D. study, I would say it was a rather thorny path full of obstacles and uncertainties. It was not as pleasant and fruitful as expected, especially initially. The difficulties lay in that I did not know much about Chaos and nonlinear dynamics from my previous study experience. Nor did I have any pre-acquired knowledge of control systems, let alone the fractional-order system. Fortunately, I received tremendous help from numerous people. My supervisor, Associate Professor Ina TARALOVA, and the thesis director, Professor Jean-Jacques LOISEAU, are the first people to be addressed.

Many thanks to Prof. TARALOVA for her patient guidance and devoted dedication. Her helpful directions and warm encouragement led me through the whole research process. The countless afternoons we spent in her office exchanging thoughts on research directions and paper writings are etched into my mind. Professor LOISEAU, on the other hand, as a prestigious expert on control systems and fractional delayed systems, took a significant amount of time explaining to me patiently the basics of control systems for my research. The one-on-one meeting we had together, aiming to equip me with sufficient background knowledge and the time contributed, proved to be inspiring and of great help.

Besides my supervisors, I would also like to express my gratitude to my CSI members, Prof. Safwan EL ASSAD and Prof. Christophe GUYEUX. They have proposed enlightening questions and valuable suggestions during my CSI meetings, which helped me dive further into my research topic. Especially for Prof. El ASSAD, who has been of great help to me with image encryption schemes design. His working style and timely responses are very well appreciated and influence me greatly.

I would also like to thank all my friends and colleagues for their support and help.

Especially those I met in Nantes, such as Dr. Hong ZHEN and Dr. Zongchao Qiao. There were many times when I was frustrated with the work or full of negative emotions; it is through consulting and talking with them that helped me relieve and regain peace in my mind.

Heartfelt gratitude goes to my parents, who provided me with an excellent environment in which I was raised. They were attentive to my need and decisions and spared every effort for my physical and mental well-being. Their dedication, love, and trust provide me with the soil where I can thrive.

Last but not least, I would express my appreciation to my husband, who has been caring and supportive since the second I decided to pursue my Ph.D. degree in France. I can not imagine how I would pass the last four years if he were not around.

List of Publications

i. Chunxiao Yang, Ina Taralova, Jean Jacques Loiseau, Safwan El Assad. A stream cipher based on fractional pseudo chaotic random number generator. 2020 15th International Conference for Internet Technology and Secured Transactions (ICITST). 1–6 (2020). doi:10.23919/ICITST51030.2020.9

ii. Chunxiao Yang, Ina Taralova, Jean Jacques Loiseau, Safwan El Assad. Design of a Fractional Pseudo-Chaotic Random Number Generator. International Journal of Chaotic Computing, Infonomics Society. 7(1), 166-178 (2021)

iii. Chunxiao Yang, Ina Taralova, Jean Jacques Loiseau. Fractional chaotic system solutions and their impact on chaotic behaviour. The 14th CHAOS 2021 International Conference, June 2021, Athens (turned into a virtual conference due to COVID-19).

iv. Chunxiao Yang, Ina Taralova, Jean Jacques Loiseau. Improving Chaotic Features of Fractional Chaotic Maps. 6th IFAC Conference on Analysis and Control of Chaotic Systems CHAOS 2021. IFAC-PapersOnLine. 54(17), 154–159 (2021)

v. Chunxiao Yang, Ina Taralova, Safwan El Assad, Jean Jacques Loiseau. Image encryption based on fractional chaotic pseudo-random number generator and DNA encryption method. Nonlinear Dynamics, Springer Verlag, 109, 2103-2127, (2022). 10.1007/s11071-022-07534-z

vi. Chunxiao Yang, Ina Taralova, Jean Jacques Loiseau. Impacts of the Numerical Calculation Methods on the Chaoticity of the Fractional Chaotic Systems. 1st IFAC Workshop on Control of Complex Systems, COSY 2022, Nov, Italy.

Resumé

La dynamique chaotique générée par les systèmes déterministes a intrigué de nombreux chercheurs depuis qu'elle a été introduite pour la première fois par Poincaré. Un système chaotique possède de nombreuses caractéristiques fascinantes, telles le comportement pseudo-aléatoire, la sensibilité élevée aux conditions initiales, l'ergodicité et la propriété de mélange topologique. Ces propriétés sont très cohérentes avec les exigences d'un cryptosystème et font du système chaotique un excellent candidat pour les applications cryptographiques.

Un crypto-système se compose de trois algorithmes cryptographiques : des algorithmes de chiffrement et de déchiffrement et un algorithme de génération de clé. Parmi les trois algorithmes, le terme chiffrement est utilisé pour le couple chiffrement et déchiffrement.

Au fil des années, les chercheurs ont étudié de manière assez exhaustive le cryptosystème basé sur une dynamique chaotique utilisant différentes structures et cartes ou systèmes chaotiques. De nombreux algorithmes de chiffrement basés sur le chaos ont été proposés. Cependant, nombre d'entre eux se sont avérés vulnérables à certaines attaques cryptographiques ; par conséquent, ils ne possèdent pas un niveau de sécurité suffisamment élevé.

Le chiffrement basé sur le chaos, d'une manière générale, peut être classé en deux types : le chiffrement par flux et le chiffrement par bloc. Le premier utilise un flux de clés pour masquer le texte en clair en continu en effectuant des opérations XOR (ou exclusif) avec le texte en clair. Le chiffrement par blocs, quant à lui, divise le texte en clair en blocs et les chiffre bloc par bloc. Les deux types de chiffrements impliquent l'utilisation d'un flux de clés qui est de préférence obtenu par un générateur de nombres pseudo-aléatoires chaotique (CPRNG), en raison de ses propriétés cryptographiques.

Une structure connue et efficace pour le chiffrement par blocs est le schéma de confusion-diffusion. Au cours du processus de confusion, l'algorithme de chiffrement effectue une opération complexe (qui consiste en une permutation et une substitution) entre le texte chiffré, la clé secrète et le texte en clair et masque les informations d'origine en un texte illisible. Pendant ce temps, le processus de diffusion vise à diffuser le changement à une position du texte en clair à l'ensemble du message, ce qui permet au changement de se

propager par cryptage. C'est-à-dire qu'un petit changement dans le texte en clair affectera considérablement le texte chiffré.

En ce qui concerne ce schéma de confusion-diffusion, les problèmes les plus courants sont l'inefficacité et l'insécurité. Certains chiffrements nécessitent plus d'un cycle de confusion et de diffusion pour obtenir des résultats de chiffrement satisfaisants. Avec l'augmentation des cycles, le temps de calcul augmente, ce qui accroît l'inefficacité. En même temps, la faible complexité de certains schémas de confusion-diffusion contribue également à l'insécurité de l'ensemble des chiffrements. Les attaquants peuvent profiter de ce manque de complexité pour annuler l'effet de diffusion ou de confusion, ce qui leur permet de craquer plus facilement le cryptosystème. De ce point de vue, le rôle important que joue une structure de confusion-diffusion bien conçue pour un chiffrement par blocs est indécible.

Pour de nombreux cryptosystèmes basés sur le chaos, on utilise des CPRNG pour générer un flux de clés. Cette remarque s'applique aux deux types de chiffrements basés sur le chaos : chiffrement par flux ou par bloc. Le CPRNG est souvent conçu à partir de plusieurs applications et systèmes chaotiques. Les conditions initiales et les paramètres du système chaotique adopté sont utilisés comme clé secrète du cryptosystème. Le CPRNG joue également un rôle crucial dans la sécurité du cryptosystème. Les propriétés pseudo-aléatoires et chaotiques obtenues par un excellent CPRNG cryptographique sont d'une grande importance pour que le cryptosystème résiste à toutes les sortes d'attaques cryptographiques, telles que les attaques à texte clair choisi, les attaques différentielles et les attaques statistiques. Pour neutraliser les attaquants qui tentent de casser le cryptosystème en utilisant une attaque par force brute, un CPRNG avec un espace clé suffisant peut également offrir une grande aide.

Pour conclure les points mentionnés ci-dessus, afin de concevoir un cryptosystème sécurisé basé sur le chaos, il convient de construire une structure de chiffrement efficace et sécurisée, et en même temps, d'avoir un fort flux de clés générées à partir d'un CPRNG correctement conçu avec un grand espace de clés.

Ces dernières années, avec la mise en œuvre technique du calcul fractionnaire et des systèmes fractionnaires dans de nombreuses disciplines, les chercheurs ont commencé à explorer les possibilités d'utiliser des systèmes chaotiques fractionnaires pour sécuriser la conception de cryptosystèmes. Le domaine est relativement ouvert, d'abord en raison des propriétés intrinsèques du calcul fractionnaire et des difficultés de mise en œuvre numérique, mais aussi en raison du fait que différentes méthodes de calcul de solutions

peuvent conduire à des comportements chaotiques distincts. Cependant, l'utilisation de systèmes chaotiques fractionnaires présente des avantages importants. D'abord, une plus grande complexité est introduite dans le cryptosystème. De plus, l'espace des clés est agrandi, car l'ordre de la dérivée fractionnaire est inclus dans la clé secrète. Mais ces avantages n'ont pas été pleinement exploités dans la littérature scientifique. Généralement, un système chaotique fractionnaire est utilisé pour confondre le texte en clair. La longueur du flux de la clé générée est relativement grande, ce qui réduit l'efficacité du cryptosystème. Le caractère aléatoire et les propriétés statistiques de la sortie du système ne sont pas testés, ce qui réduit sa sécurité.

En ce qui concerne la structure de chiffrement, l'utilisation du calcul basé ADN a retenu l'attention de nombreux chercheurs ces dernières années en raison de ses nombreux avantages, tels qu'un parallélisme massif, une énorme capacité de stockage et une faible consommation d'énergie. Convaincus que les difficultés techniques et les limites de la cryptographie ADN existant actuellement au niveau expérimental seront bientôt surmontées dans un avenir proche, un certain nombre de cryptosystèmes basés sur le chaos utilisant des méthodes de codage et de décodage de l'ADN ont été proposés. Cependant, il reste encore un long chemin à parcourir, non seulement dans pour la mise en œuvre dans la domaine de biochimie, mais également pour la conception d'algorithmes cryptographiques sécurisés pour l'ADN. Les travaux existants sur les crypto-systèmes d'images basés sur le codage et le décodage de l'ADN utilisent généralement un seul système chaotique pour effectuer la permutation des pixels d'image simples, en négligeant le caractère aléatoire et les caractéristiques statistiques des sorties du système chaotique. De plus, des opérations complexes sont apportées aux algorithmes de chiffrement pour surmonter le problème de diffusion insuffisante. Le nécessité et l'utilité de ces opérations fastidieuses ne sont pas concluantes et réduisent l'efficacité de l'ensemble du crypto-système.

Dans le présent travail, nous concentrons notre attention sur l'application de systèmes chaotiques fractionnaires au CPRNG et sur la conception d'un nouveau système de cryptographie d'image. Les idées et contributions essentielles sont résumées comme suit.

1. Pour implémenter des systèmes chaotiques fractionnaires, nous avons étudié les méthodes d'approximation numérique de différentes caractérisations de dérivées fractionnaires pour les systèmes multidimensionnels et les opérateurs fractionnaires. La méthode d'approximation constante par morceaux basée sur la caractérisation de Rienmann-Liouville (RL) est adoptée pour analyser l'état et le comportement chaotique du système logistique fractionnaire généralisé à double-humped.

Les méthodes de calcul de solutions numériques pour les caractérisations des systèmes fractionnaires de Gr*ü*nwald-Letnikov (GL) et Caputo sont discutées. En utilisant les deux méthodes, nous avons simulé et obtenu les solutions numériques pour deux systèmes chaotiques fractionnaires dérivés de systèmes classiques d'ordre entier, à savoir les systèmes fractionnaires de Chen et de Lu. Les impacts des deux méthodes sur le comportement chaotique des systèmes fractionnaires étudiés sont discutés.

2. La méthode correctrice-prédictive dite Adams-Bashforth-Moulton (ABM) est basée sur la définition de Caputo. Nous avons proposé sur cette base une nouvelle méthode de calcul non uniforme. Nous introduisons deux méthodes d'échantillonage non uniforme, qui sont définies en termes de transformation Tent Maps asymétriques chaotiques de façon à faire varier les pas de temps selon lesquels les solutions du système sont discrétisées. Nous choisissons cinq intervalles différents de 0,001 à 0,005 et divisons la plage des valeurs d'état possibles de la carte de tente asymétrique (0 à 1) en cinq intervalles sans chevauchement. Pour la première grille non uniforme proposée, un pas de temps est attribué à chaque état du système en fonction de l'intervalle dans lequel se trouve la sortie de la transformation Tent Map. Pour cette grille, nous avons trié les cinq intervalles dans un ordre séquentiel. Pour la deuxième grille non uniforme que nous définissons, nous avons utilisé deux transformations Tent Map asymétriques. La première d'entre elles est une bijection entre les cinq intervalles d'état et les cinq pas d'échantillonage, et celle de la seconde détermine le pas d'échantillonage en fonction de son intervalle correspondant. Nous avons ensuite utilisé les méthodes proposées pour la résolution numérique des systèmes chaotiques fractionnaires de Chen et de Lu. Les résultats de la simulation sont donnés, et les impacts sur le comportement chaotique des systèmes sont analysés.

3. Un CPRNG Fractionnaire (FCPRNG) qui réussit les tests aléatoires et statistiques est conçu pour la toute première fois. Nous avons construit une structure parallèle en adoptant trois systèmes chaotiques fractionnaires différents et deux cartes de tente asymétrique chaotique. Les sorties du système fractionnaire de Chen et du système de Lu calculées avec la première des deux grille proposées et les sorties de FGDHL acquises en utilisant la méthode d'approximation constante par morceaux ont été combinées à l'aide de trois opérations XOR. Avant d'être converties en valeurs 32 bits, les signaux de sorties des systèmes fractionnaires Chen et Lu sont

prétraitées par un mécanisme de repliement, et celle du FGDHL a été tronquée.

Le FCPRNG proposé passe avec succès la suite de tests statistiques du NIST, ce qui confirme la qualité du caractère pseudo-aléatoire des sorties du FCPRNG. Les tests effectués sur le chiffrement d'images prouvent le haut niveau sécurité et l'efficacité du cryptosystème d'image basé sur le générateur que nous avons proposé.

4. Un système de cryptage d'image sécurisé a été conçue en utilisant la méthode de codage et de décodage de l'ADN et un FCPRNG fonctionnant sur le mode d'enchaînement des blocs (Cipher Block Chaining (CBC)). Le flux de clés du cryptosystème est généré par le FCPRNG de la même manière que dans le schéma précédemment proposé, mais avec un espace de clé augmenté grâce à l'utilisation de notre seconde grille discrétisation non-uniforme, basée sur deux transformations Tent Maps asymétriques. Le processus de confusion du chiffrement a été réalisé par des méthodes dynamiques de codage et de décodage de l'ADN, avec une cat map modifiée fonctionnant au niveau symbolique de l'ADN. Les règles de codage et de décodage dynamiques sont déterminées par le flux de clés. Une application logistique avec une précision finie a été utilisée pour effectuer la diffusion. Le mode CBC adopté a encore amélioré les performances de la structure de confusion-diffusion et a augmenté la sécurité du cryptosystème.

   L'analyse de sécurité et les résultats expérimentaux ont démontré que le FCPRNG modifié possède d'excellentes propriétés cryptographiques, et que le chiffrement par blocs que nous proposons a des performances de confusion et de diffusion sécurisées et complexes et peut résister aux principales attaques connues.

En résumé, la thèse a analysé certaines des difficultés de la conception de cryptosystèmes basée sur le chaos. Les recherches ont abouti à la conception de générateurs de nombres pseudo-aléatoires et à la proposition d'un nouveau cryptosystème, basés sur des systèmes chaotiques fractionnaires. Les tests et analyses effectués ont prouvé la faisabilité et la fiabilité de l'algorithme cryptographique proposé.

Abstract

Chaotic dynamics yield by deterministic systems has been very intriguing ever since first sighted by Poincaré. The chaotic system possesses many fascinating characteristics, such as random-like behavior, high sensitivity to initial conditions, ergodicity, and topological mixing property. These properties are highly consistent with the requirement for an efficient cryptosystem and make the chaotic system an excellent candidate for cryptographic applications.

A cryptosystem consists of three cryptographic algorithms : encryption algorithm, and decryption algorithm, and a key generation algorithm. Among the three algorithms, the term "cipher" is used for the pair of encryption and decryption.

Over the years, researchers have been investigating quite exhaustively the cryptosystem based on chaotic dynamics employing different structures and chaotic maps or systems. Numerous chaos-based encryption algorithms have been proposed. However, many of them have been proved to be vulnerable to different cryptographic attacks; hence, they do not possess a sufficiently high level of security.

Chaos-based cipher, generally speaking, can be categorized into two types: stream cipher and block cipher. The former employs a keystream to mask the plaintext continuously by performing XOR (exclusive or) operations with the plaintext. The block cipher, on the other hand, divides the plaintext into blocks and encrypts them block by block. Both ciphers involve the use of a keystream which is preferably generated by a chaotic pseudo-random number generator (CPRNG) due to its good pseudo-randomness and cryptographic properties.

One popular and effective structure for the block cipher is the confusion-diffusion scheme. During the confusion process, the encryption algorithm performs a complex operation(which consists of permutation and substitution) between the ciphertext, the secret key, and the plaintext and masks the original information into some unreadable text. Meanwhile, the diffusion process aims at diffusing the bit change at a position of the plaintext to the entire message, making the change propagate through encryption. That is to say, a tiny change in the plaintext will significantly affect the ciphertext.

With respect to this confusion-diffusion scheme, the most commonly manifested prob-

lems are inefficiency and insecurity. Some ciphers need more than one round of confusion and diffusion operations to obtain satisfactory encryption results. With the increased rounds, the computational time augments, which builds up the inefficiency of the algorithm. At the same time, the low complexity of some confusion-diffusion schemes also contributes to the insecurity of the whole ciphers. The attackers can make use of this lack of complexity to remove the diffusion or confusion effect, making it easier for them to crack the cryptosystem. From this point of view, the significant role of a well-designed confusion-diffusion structure which a block cipher plays is primordial.

For many chaos-based cryptosystems, neglecting their category (stream cipher or block cipher), one can employ CPRNGs to generate the keystream. The CPRNG is often designed by properly handling several chaotic maps and systems. The initial conditions and parameters of the adopted chaotic system are used as the secret key to the cryptosystem. As a component of the cryptosystem other than the cipher, the CPRNG also plays a crucial role in the security of the cryptosystem. The pseudo-randomness and chaotic properties maintained by an excellent cryptographic CPRNG are of great importance for the cryptosystem to resist many cryptographic attacks, such as chosen-plaintext attacks, differential attacks, and statistical attacks. To exhaust attackers trying to break the cryptosystem using brute-force attacks, a CPRNG with sufficiently large key space is also in demand.

To conclude the above-mentioned points, in order to design a secure chaos-based cryptosystem, one should construct an effective and secure cipher structure, and, at the same time, have a strong keystream generated from a properly designed CPRNG with a large keyspace.

In recent years, with the engineering implementation of fractional calculus and fractional systems in many disciplines, researchers have started to explore the possibilities of employing fractional chaotic systems for secure cryptosystem design. The field is relatively deserted not only because of its intrinsic properties of fractional calculus and the difficulties for digital implementation but also due to the fact that different solution calculation methods can lead to distinct chaotic behavior as well. However, the merits are also prominent. By integrating fractional chaotic systems, greater complexity is introduced to the cryptosystem. In addition, larger keyspace comes along, with fractional derivative order included in the secret key. In the few papers discussing this matter, typically, one fractional chaotic system is employed to confuse the plaintext. The length of the generated keystream is relatively long, which decreases the efficiency of the cryptosystem. The

randomness and statistical properties of the system output are not tested, which could pose a security problem.

Regarding the cipher structure, the use of DNA (deoxyribonucleic acid) computing has caught many researchers' attention in recent years due to its numerous advantages, such as massive parallelism, huge storage capacity, and low energy consumption. Under the belief that the technical difficulties and limitations of DNA cryptography existing now at the experimental level will be soon conquered in the foreseeable future, quite a few chaos-based cryptosystems using DNA encoding and decoding methods have been proposed. However, a long way still lies ahead, not only in the sense of biochemistries implementation but the secure DNA cryptographical algorithms design as well.

The existing work on DNA encoding and decoding-based image cryptosystems usually employs one single chaotic system to perform permutation of the plain image pixels, neglecting the randomness and statistical features of the chaotic system outputs. In addition, complex operations are brought in for the cipher algorithms to overcome the problem of insufficient diffusion. The necessities and effects of these cumbersome operations are inconclusive and can increase the inefficiency of the whole cryptosystem.

For the research status and all the problems mentioned above, we draw our attention to apply fractional chaotic systems to the CPRNG design, and novel image cryptosystem construction. The main work carried out in the thesis is summarized as follows.

1. To implement fractional chaotic systems into digital application use, we investigated the numerical approximation methods under different fractional derivative characterizations for multi-dimension fractional systems and 1D fractional maps. The piecewise-constant approximation method based on Rienmann-Liouville (RL) characterization is adopted to analyze the states and chaotic behavior of the fractional generalized double-humped logistic system. The widely implemented numerical solution calculation methods based on Grünwald-Letnikov (GL) and Caputo characterizations for fractional systems are discussed. Employing both methods, we simulated and obtained the numerical results for two fractional chaotic systems, derived from classical integer order systems, namely, fractional Chen and Lu systems. The impacts of the two calculation methods on the chaotic behavior of the fractional chaotic systems under investigation are discussed.
2. Based on the numerical calculation method for fractional differential equations, the Adams-Bashforth-Moulton (ABM) corrector-predictor method, we proposed a type of non-uniform calculation method which can enlarge the range for the fractional

derivative order for which the systems is chaotic. Two non-uniform grids based on chaotic skew tent maps are introduced. The grid spaces on which the system solutions are discretized. We choose five different grid spaces from 0.001 to 0.005 and divide the range of the possible state values of the skew tent map (0 to 1) into five (non-overlapping) intervals. For the first proposed non-uniform grid (grid 1), a variable grid space is assigned to each system state depending on the interval in which the skew tent map output falls. Notice that for this grid, we sorted the five intervals in sequential order. For the second non-uniform grid, we employed two skew tent maps. The output of the first skew tent map constructs a map between the five intervals and the five grid spaces, and that of the second determines the grid space according to its corresponding interval. We then employed the proposed methods to calculate the fractional chaotic Chen and Lu systems. The simulation results are given, and the impacts on the chaotic behavior of the systems are analyzed.

3. A Fractional CPRNG (FCPRNG), which succeeded the randomness and statistical tests, is designed for the very first time. We designed a parallel structure adopting three different fractional chaotic systems and two chaotic skew tent maps. The outputs of the fractional Chen system and Lu system calculated on the proposed grid 1 and the outputs of FGDHL acquired employing the piecewise-constant approximation method have been combined using three XOR operations. Before converted to 32-bit values, the outputs of fractional Chen and Lu systems are pre-processed by a folding mechanism, and that of the FGDHL was truncated.

   The proposed FCPRNG passes successfully the NIST test suite and the statistical tests, which validates the randomness and statistical properties of the FCPRNG outputs. The tests conducted on the stream cipher for image encryption based on the proposed FCPRNG prove the security and liability of the image cryptosystem.

4. A secure image encryption cryptosystem was designed based on the DNA encoding and decoding method and FCPRNG working on the Cipher Block Chaining (CBC) mode. The keystream of the cryptosystem was acquired by the FCPRNG similar to the first FCPRNG but with larger key space brought in thanks to the numerical calculation method grid 2 with two extra skew tent maps. The confusion process of the cipher was achieved by dynamic DNA encoding and decoding methods with a modified cat map operating on the DNA symbol level. The dynamic encoding and decoding rules, together with the initial vectors for the modified cat map,

are generateed by the chaotic keystream. A logistic map with finite precision was employed to perform the diffusion. The adopted CBC mode further enhanced the performance of the confusion-diffusion structure and thus increased the security of the cryptosystem.

The security analysis and experimental results demonstrated that the modified FCPRNG possesses excellent cryptographic properties, and the proposed block cipher has secure and complex confusion and diffusion performances and can resist the main known attacks, such as statistical attacks, and brute-force attacks.

To sum up, the thesis has analyzed some of the existing problems for the chaos-based cryptosystem and oriented the research interests towards the novel and less discussed cryptosystem based on fractional chaotic systems and fractional pseudo-random number generator design. The tests and analyses conducted proved the feasibility, high security, and reliability of the proposed cryptographic application.

List of Tables

List of Figures

Table of Contents

INTRODUCTION

Research background and motivation

In mathematics and science, nonlinear systems are the systems whose change of outputs are not proportional to their change of inputs as for linear systems [Wikipedia contributors, 2022]. The dynamics of nonlinear systems are much more complex than linear systems, yet it is of great interest to researchers in engineering and science. Many real-life systems are nonlinear in nature, for instance, the turbulent flows studied by physicians, the population of species in biology, the process of weather evolution, etc. Due to the nonlinearity of the system equations, the behavior of a complex nonlinear system can be rather counterintuitive, unpredictable, and sometimes random-like. The chaos theory emerged to study the underlying deterministic laws and dynamics possessed by this random-like behavior of the nonlinear systems.

From the late 19th century till now, the scientific world has participated in and witnessed the rapid development and the flourishing of chaotic dynamics as time goes by. Some very first studies on Chaos are listed here: Poincaré [Poincaré, 1900], who first discovered a chaotic deterministic system and came up with the notion of chaos in his work discussing the three-body problem; in [Hadamard, 1898], Hadamard studied the chaotic motion of a free particle gliding frictionlessly on a surface of constant negative curvature and showed that all trajectories diverge exponentially from one another, with a positive Lyapunov exponent; Lorenz in [Lorenz, 1963] studied the evolution of the system starting from different initial conditions based on an idealized hydrodynamical system.

Since the 1990s, researchers in the domain of cryptography, attracted by the numerous characteristics exhibited by the chaotic systems (such as random-like, topological mixing, and high sensitivity to initial conditions and parameters), have started to explore the possibilities of employing the chaotic systems to the encryption algorithm design. The corresponding relationship between some properties of chaotic systems and the cryptographic algorithm design requirements [Qiao, 2021][Alvarez, 2006][Teh, 2020] are shown in Table.1. After years of study, researchers in the domain of cryptography now have reached the common understanding that chaotic systems can be considered a promising

Chaotic property	Cryptographic property	Description
Topological mixing	Confusion	The relation between input and output is intricate
High sensitivity to initial conditions and parameters	Diffusion	A tiny change in the input can cause a big difference at output
Random-like behavior	Random-like output	Any input can produce a random-like output.
Deterministic systems	Deterministic encryption /decryption algorithm	A deterministic scheme can generate pseudo-random outputs.
Complex nonlinear dynamics	Nonlinear transfromation and complex algorithm	A simple system has a high nonlinear dynamical complexity.

Table 1 – A partial list of the relationship between the properties of chaotic system and cryptosystem

tool when it comes to cryptosystem design, especially for image encryption applications [Abraham, 2013].

Many chaos based cipher algorithms have been proposed and discussed, both encrypting the input plaintext block by block (block cipher), or taking the input as a whole and performing the XOR operation to the plaintext and a keystream (stream cipher). One popular way to integrate chaotic systems into the encryption algorithms is to use a chaotic pseudo-random number generator (CPRNG) to generate pseudo-random outputs, which are then adopted as the key stream of the cipher. Being responsible for achieving the confusion and diffusion of the plaintext to obscure the original message, the crucial part that the CPRNG outputs play is not to be neglected. Hence, the performance and quality of the CPRNG adopted are of great importance for the design of a well-functioning and secure cryptosystem.

As for the CPRNG, its outputs should satisfy the statistical requirements for the outputs of a Pseudo-Random Number Generator (PRNG). Normally, this means that the outputs of the generator should be pseudo-random and uniformly distributed over a certain range of values. To satisfy the latter, it is common to employ several chaotic systems or maps, and the initial conditions and parameters of the chaotic systems adopted are considered the generator's seed. From the aspect of cryptography, the seed constitutes the secret key of the cryptosystem, and the CPRNG outputs will work as the keystream. In the meantime, it is known that large keyspace and high sensitivity to the secret key components ensure the security of the cryptosystem against brute-force attacks. Therefore, to have excellent quality and to achieve outstanding security performance, the designed

CPRNG for cryptographic use should not only satisfy the requirements for the PRNG algorithm but preserve satisfactory chaotic properties as well.

The history of fractional calculus as a branch of mathematical analysis is almost as long as integer-order differential calculus. It was also in Gottfried Wilhelm Leibniz's maneuver in 1695 where the fractional derivative made its first appearance [Katugampola, 2011]. The more well-established idea of fractional calculus was introduced independently by both Niels Henrik Abel, and Liouville in around the 1830s in [Niels, 1823] and [Liouville, 1832], respectively. Since then, many mathematicians have discussed the perception of fractional calculus and given characterizations (definitions or types) of their own, such as the fractional derivatives of Grünwald–Letnikov, Riemann–Liouville, and Caputo type [S, 1993][Podlubny, 1999].

Though having been discussed mathematically by many mathematicians, fractional calculus remained a purely mathematical problem for centuries. Over the past few decades, due to its natural memory effect and hereditary properties [Odibat, 2010], fractional calculus has finally made its grand entrance into engineering applications. Recent studies have proved that the use of fractional dynamic systems in defining and modeling real-life systems is applicable and suitable in many disciplines such as physics [Machado, 2011], biology [Arfan, 2021], economics [Tarasov, 2018], and composite behavior law [Krasnobrizha, 2016], and etc.

In the recent years, fractional-order system also caught the attention of researchers in the domain of secure communication and cryptography. The fact that some fractional-order systems also exhibit chaotic behavior provides a novel orientation for researchers in the field: the design of cryptosystems based on fractional chaotic systems. The advantages of employing fractional-chaotic systems are numerous:

1. The use of classical integer-order chaotic system to design encryption algorithm has been quite exhaustively investigated, while the fractional chaotic systems remain pretty deserted, and their use in cryptography are yet to be discovered.
2. The behavior of the fractional chaotic systems is different, adopting different fractional derivatives; thus, extra variable parameters will be introduced to the cryptosystem working as the secret key to enlarge the keyspace, which would probably lead to enhanced security.
3. Different fractional derivative characterizations and numerical calculation methods can lead to different chaotic behavior, which increases the complexity of the cryptosystem and renders the cryptosystem more challenging to break.

...

Organization of the thesis and the main contributions

In this section, the organization of the thesis and the main contributions will be presented. As the title of the thesis states, this thesis focus on nonlinear dynamics, especially chaotic system, and its application to the design of pseudo-random number generator. The fractional chaotic systems and their solutions are discussed, the pseudo-random number generator based on fractional systems is investigated, and a step forward is taken to propose secure chaos-based cryptosystems.

Concerning this, Chapter 1 firstly introduces the basis of chaotic systems and chaos-based cryptography. Then in Chapter 2, the fundamentals of fractional calculus and fractional chaotic systems are illustrated. We also analyzed the current work and progress on chaos-based cryptography and chaotic pseudo-random number generator design with existing problems stipulated. In addition, we oriented the research direction to the new perspective of employing fractional chaotic systems in the chaos-based image cryptosystem.

The main contributions are summarized as follows and are given in the following chapters.

Chapter 3 : Illustrate and compare different solution calculation approaches for fractional systems based on Caputo and Grunwald characterization of fractional derivatives, respectively.

To implement the fractional system into cryptography, the solutions of the systems must be calculated numerically. In chapter 3, we illustrated one numerical approach for a one-dimensional (1D) fractional chaotic map employed to solve the fractional generalized double-humped logistic map (FGDHL), which was later used for our FCPRNG design. The widely-used calculation methods were introduced for multi-dimensional fractional systems with fractional derivatives defined (characterized) by both Caputo and Grunwald characterizations, respectively. We then employ the discussed methods to solve two fractional chaotic systems, namely fractional Chen and Lu systems which are derived from their corresponding integer order three-dimensional (3D) chaotic systems. The simulation results were compared and revealed the fact that chosen numerical approximation approach will impact the chaotic behavior of the fractional systems under investigated.

Chapter 4 : Propose a non-uniform grid numerical calculation method for multi-dimension fractional chaotic systems.

To further exploit the advantage of a fractional chaotic system concerning the additional parameters introduced to the cryptosystem, in Chapter 4, based on the well-accepted and widely-implemented fractional Adam-Bashforth corrector predictor calculation method for fractional systems, we propose a non-uniform calculation method. The original fractional corrector predictor calculation method employs a uniform grid for the numerical calculation. Based on this trivial method, 1D skew tent maps are introduced to form a non-uniform grid on which the solutions of the fractional systems are calculated. The method is employed to find the solutions of several well-known multiple dimension fractional chaotic systems. The bifurcation diagram, Lyapunov exponents, as well as time comparison of different response results, show that enhanced chaotic performance have been successfully obtained and expressed in terms of the ranges of fractional orders and parameters.

Chapter 5 : Employ the proposed non-uniform grids to design the fractional chaotic pseudo-random number generator, which passes the statistical and randomness tests.

We introduce an original pseudo-random number generator scheme which consists of 3 different fractional chaotic systems and maps, namely, the fractional Chen system, fractional Lu system, and fractional generalized double-humped logistic system (FGDHL). The former two systems are fractional 3D systems which are solved using our proposed non-uniform grid method. The designed pseudo-random number generator based on the fractional chaotic system (FCPRNG) passes the NIST randomness test successfully and possesses great cryptography properties. After this, we design and test a new stream cipher based on the proposed FCPRNG. The performance analysis proves that the new stream cipher owns good statistical and security properties.

Chapter 6 : Design a new secure chaos-based image cryptosystem (block cipher) with good confusion and diffusion properties applying the proposed FCPRNG and DNA computing.

For a well-designed encryption scheme, a sufficient level of diffusion and confusion is of great importance. Large keyspace size is also in demand for the scheme to possess a high level of security. In Chapter 6, a secure cryptosystem for image encryption based on chaotic components and DNA encoding methods is discussed, which consists of an efficient FCPRNG and a block cipher. The dynamic DNA encoding and decoding algorithms, together with one cat map proceeding at the DNA level, constitute the confusion of the

encryption scheme. A discrete chaotic logistic map with finite precision of 32 bits is employed to achieve diffusion. The proposed block cipher works on cipher block chaining mode (CBC), which further reinforces the confusion and diffusion performance of the cipher.

The proposed FCPRNGs can produce pseudo-random numbers with good cryptographic properties and they can be used in any designs of stream ciphers or block ciphers for encryption purposes. In addition to this, it also can be applied in FCPRNG required applications.

At last, in Chapter 7, we conclude the work has been carried out successfully and discuss the prospective future research directions for the thesis.

Chapter 1

Chaotic dynamics and chaos based cryptography

1.1 Introduction

In this chapter, the introduction of chaotic dynamics and chaos-based cryptography are discussed. Firstly we briefly introduce the history and major discovery over the time of the chaotic dynamics in Section 1.2. Then some basics and features of chaotic system are presented in Section 1.3. Next, in 1.4, the introduction to the chaos-based cryptosystem including cryptography, cryptosystems and crypto-attack is given. After that, we give the state of art of the cryptosystem based on chaotic systems. The literature analysis focusing on the chaos-based encryption scheme and the design of CPRNG is discussed in Section 1.5. In the end, a conclusion is given in Section 1.6.

1.2 Historical overview of chaotic dynamics

When speaking of chaos, many may think of the famous 'butterfly effect' which gained its name because of the title of the paper published by Edward Lorenz in 1973. The paper is entitled 'Predictability: Does the Flap of a Butterfly's Wings in Brazil set off a Tornado in Texas?' As the metaphor and the system (the flapping wing of the butterfly in Brazil) may cause a huge difference in the trajectory and lead to the unpredictability of the overall system (a Tornado in faraway Texas). Edward Lorenz accidentally started his research in chaos through his work on weather prediction in 1961. Yet nearly a century before his pioneer work on meteorology, the spark for chaos theory had ignited when some other problems in physics were addressed.

As one of the third major physics discoveries in the 20th century (together with relativity theory and quantum mechanics [Overman, 1996]), chaotic dynamics was first sighted in the late 19th century by the prestigious french mathematician Henri Poincaré. Huge

complexity was observed by him with no accurate solutions when conducting the study on the three-body problem. Being also a theoretical physicist, engineer, and philosopher of science, Henri Poincare applied his knowledge of dynamics and topology to the three-body problem and discovered that within a certain range, the solution to the problem is random [Poincaré, 1890]. This finding has revolutionary significance since it is the first work that reveals that even for a deterministic system, the system's trajectory can be extremely unstable. Different system behavior could be manifested with any slight change in the initial condition. This also raised the consciousness of many researchers of the inherited randomness in some deterministic systems [Poincaré, 2017][Diacu, 1999].

Eighteen years after the discovery of Henri Poincaré, in 1898, french mathematician Jacques Hadamard published his work on the study of geodesics, where chaotic dynamics is also observed [Hadamard, 1898]. In this paper, he discussed the motion of a free particle gliding frictionlessly on a surface of constant negative curvature. The trajectories of the particles diverge exponentially with a positive Lyapunov exponent, showing that they are unstable.

Though it was not until 1960s that a rapid development in chaotic dynamic was witnessed, many initial insights of chaos theory have already emerged during the first half century when the mathematicians at that time studied the non-linear differential equations inspired from physics.

For example, the British mathematician Mary Lucy Cartwright who collaberated with John Edensor during the world war 2 studied the differential equations which were used to model radio and radar waves [Cartwright, 1947]. Their work and finding was noticed after 1960s by many mathematicians and developed into the foundations of chaos theory [Dataiku, 2021]. Soviet mathematician Andrey Kolmogorov researched into the mathematical model behind the turbulence of the incompressible fluid in [Kolmogorov, 1991], and discussed the persistence of quasiperiodic motions under small perturbations in [Kolmogorov, 1954]. The latter was later proven by his student Vladimir Arnold and Jürgen Moser and led to the famous KAM theorem in hamiltonian mechanics, which laid the foundation of Chaos theory [Salamon, 2004].

The time comes to the 1960s, with the development of electronic computers, the study of chaotic dynamic entered a time of boosting. Much more in-depth research into Chaos theory started to be carried out. In 1963, while researching meteorology, Lorenz accidentally found that the evolution of weather is closely related to initial conditions and proposed a 3D autonomous system to discuss his founding in paper [Lorenz, 1963].

This deterministic system showed high sensitivity to initial conditions, which is in line with the typical feature of Chaos. The proposed Lorenz system turned out to be the very first mathematical model describing Chaos. Later in Lorenz's another paper, the famous metaphor 'Butterfly effect' was also given to this phenomenon where any slight change in initial conditions leads to a huge difference in system outcome and overall behavior [Lorenz, 1972].

In 1976, Hénon map was introduced by French mathematician and astronomer Michel Hénon in his paper "A two-dimensional mapping with a strange attractor". This famous and most studied two-dimensional discret-time chaotic map, which captures the stretching and folding dynamics of chaotic systems, is a simplified model inspired by Lorenz system [Wen, 2014]. In the same year, Australia scientist Robert May published a paper "Simple mathematical model with complex dynamics characteristics" in "Nature". The paper focuses on the animal population dynamics and proposed a 1D prey-predator model : logistic map, where chaotic behavior of period-doubling bifurcation and chaos existed.

Also, in the year of 1976, the American mathematical physicist Mitchell Feigenbaum observed the underlying mechanism within Chaos. He found that "When an ordered system begins to break down into Chaos, a consistent pattern of rate doubling occurs" [Mitchell, 1989]. Two constants indicating the convergence ratio that lead from period-doubling bifurcation to Chaos were discovered and published in 1978 in [Feigenbaum, 1978]. This founding of the universality of Chaos laid the foundation for the study of the chaotic behavior of 1D maps.

In 1989, American mathematician Robert Luke Devaney gave his definition of chaotic systems, formulated in his book "An Introduction to Chaotic Dynamical Systems" [DEVANEY, 1986]. This mathematical definition implies that for a system to be chaotic, it should have sensitive dependence on initial conditions; it should be topologically transitive (for any two open sets, some points from one set will eventually hit the other set); and its periodic orbits should form a dense set [Boccara, 2004]. Though this definition is later proved redundant, since the first property comes with the latter two, Devaney's definition of chaotic systems remains simple and widely used.

With the emergence of cheaper, faster, and more powerful computers, the research into Chaos theory and chaotic dynamic experience speedy advance after the 1990s. Nowadays, while many researchers are still working on the investigation of theoretical properties of chaotic systems [Billings, 2001][Feltekh, 2014], works and applications in other domains have also prospered, such as mathematics, physics [Schöll, 2001], biology, information

theory, economics [Basalto, 2005] and etc,.

1.3 Introduction to chaotic system

1.3.1 Fundamentals of nonlinear systems

A nonlinear dynamical system is used to describe a physical model that can be represented by a set of nonlinear equations. Different from linear dynamic system where the system equation can be described using a set of linear functions, the change of the output of nonlinear systems is not proportional to the change of the input [Hardesty, 2010]. Hereafter some fundamentals related to chaotic systems and pay special attention to some features which are particularly important cryptography applications are introduced.

Continuous-time nonlinear systems

Most continuous-time nonlinear dynamical systems can be described by a differential equation :

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t; \mathbf{p}), t \in [t_0, \infty). \tag{1.1}$$

In the equation, $\mathbf{x} = \mathbf{x}(t)$ is the *state* if of the system that usually belongs to a bounded region $\Omega_{\mathbf{x}} \subset \mathbf{R}_r$, where n denotes the dimension of state variable $\mathbf{x}$; the initial time is t_0 and the initial condition is $\mathbf{x} = \mathbf{x}(t_0) \in \Omega_{\mathbf{x}}$; $\mathbf{p} \subset \mathbf{R}_m$ is the system parameter vector that usually varies within a bounded range. Normally, $m \leq r$; $\mathbf{f}$ is a nonlinear or piece-wise linear function that depicts explicitly a specfied system.

Discrete-time system

The discrete time nonlinear system, different from continuous-time system, only has states at given instants. The system can be described by a difference equation or a map:

$$\mathbf{x_{n+1}} = \mathbf{f}(\mathbf{x}_n, n, \mathbf{p}) \tag{1.2}$$

In the equation, n is the time index; $\mathbf{x}_0$ is the initial condition for the state $\mathbf{x}$.

Notions of nonlinear systems

Phase space. Also called *state space*, represents the entire space that comprises all states of a dynamical system. For an n-dimensional system, the phase space expands by its evolving states $\mathbf{x}$. For a 1D discrete system, the phase space can be constructed by the iteration states in the space (x_n, x_{n+1}), and the graph shown in the phase space coincides with the graph of the 1D function.

Orbits. The evolution of a dynamical system is embodied in a trajectory of the states traveled from the initial state x_0 in the phase space. This trajectory is called an orbit of the system. The dynamics of the system can be observed by its orbits in phase space.

Deterministic system. A dynamical system is deterministic, if there is a unique consequence to every change of the system's parameters or initial conditions. Otherwise, the system is *stochastic* or *random*, if there exists more than one possible consequence for a change in its parameters or initial conditions according to some probability distribution [Chen, 1998].

It is to be emphasized here that chaotic system is also deterministic. This is to say the orbit starts from one initial condition does not intersect with the trajectory starting from another. In addition, the orbit itself can be reproduced again with the same initial condition. These properties make it possible to establish a link between chaotic system and cryptography. The uniqueness of the keystream (numbers generated by CPRNG) to the secret key and the possibility to reproduce the same keystream at both sender and receiver end can be guaranteed.

Fixed point. Also called *equilibrium point* and *invariant point*, is an equilibrium state $\mathbf{x}_{fp}$ of a dynamical system. According to [Holmgren, 2000], the definition of the fixed point is given here: If f is a function and $f(c) = c$, then c is a fixed point (x_{fp}) of f. For the continuous time system, it has

$$\mathbf{f}(x_{fp}; \mathbf{p}) = \mathbf{0}. \tag{1.3}$$

For the case of discrete-time system (as given in equation 1.2), if a 1-dimensional map is concerned, then the fixed point is the intersection of the system equation $\mathbf{f}(x; \mathbf{p})$ with the function $\mathbf{f(x) = x}$ which gives

$$x_{fp} = \mathbf{f}(x_{fp}; \mathbf{p}). \tag{1.4}$$

For the points that lie on the intersection, the orbits stay at the same fixed point regardless of the iterations, which indicates that the orbits remain locked.

When a three dimensional (3D) dynamic system is considered, the equilibirum can be catagorized into the following 4 different types according to the eigenvalues of Jacobian matrix of the system at the equilibrium: node, saddle, focus node, saddle focus. Some more details on this will be given in Chapter 2 Section 2.4 with respect to this point.

Generally speaking, if the nearby orbits of the fixed point drive toward it, the fixed point is considered to be stable or attractive; otherwise (move away from it), the fixed point is said to be unstable or repulsive. We did not investigated profondly the fixed point in our work. But one should know that for the cryptosystem design, it is preferable to avoid the fixed points of the chaotic systems as much as possible. Otherwise, the undesired result of locking into the fixed point may lead to a lack of security of the cryptosystem.

Attractor. This term is for the set of states where a system is tend to evolve [Milnor, 1815]. It can be defined as a compact subset Q in phase space M which saticfies the following conditions:

— Q is an invariant with respect to the dynamics of the system. $f^n(Q) = Q$ for any given n dynamics which remains on the attractor Q;
— there exists a neighbourhood U, which is compressed to compressed to Q. $Q \subset U \subset M$ if the condition is satisfied n such that $f^n(U) \subseteq A$;
— Q can not be decomposed into two or more disjoint invariant subsets.

There are different types of attractors, including the fixed point. Among these types, there is one called the strange attractor. The characteristic of the strange attractor is that a non-smooth structure can be observed. One can also found strong sensitivity to initial conditions for the strange attractor. Thus, the appearance of this type of attractor is considered as a strong evidence of the existence of chaotic trajectories.

Bifurcation. In dynamical system, a bifurcation diagram shows the values visited by the system as the function of parameter change. It determines the qualitative change in the behavior of the system under the variation of a certain system (control) parameter. We give here the bifurcation diagram of the discrete-time Logistic map as an example. The logistic map takes the form as following,

$$x_{n+1} = \rho x_n(1 - x_n) \tag{1.5}$$

In the above equation, ρ is the control parameter. For each value of ρ, the logistic map is

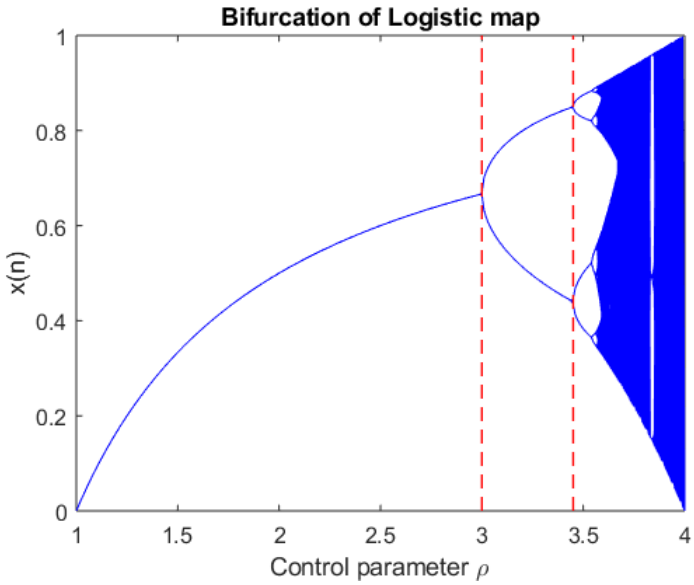


Figure 1.1 – Bifurcation diagram of logistic map

iterated 10 thousands times, and only the last 2000 states are plotted in Fig. 1.1.

The solid curve in the figure stands for the stable values obtained from corresponding control parameters ρ. It is clear that parameter changes can give rise to qualitative new behaviour such as fixed point and stable periodic behavior, or chaotic attractor.

1.3.2 Definition of Chaos

Chaos is a random-like behavior exhibited by many nonlinear dynamic systems where a small change of the initial conditions of the system will lead to huge difference in the corresponding system overall behavior. Though the chaotic systems may exhibit the apparent disorder, it is in fact controlled by certain deterministic differential equations. The system states may be predictable at the beginning, but in the long term, the orbit of the system is unpredictable.

Though the domain of Chaos theory has been established and discussed for decades, there is still no universally agreed definition for the term 'Chaos'. Scientists and researchers have come up with different definitions mathematically from different aspects over the years. For example, Li-Yorke definition [Li, 1975] describes the Chaos from the orbits aspect; Devaney definition [DEVANEY, 1986] defines the Chaos from the topological aspect; and Smale definition from the geometrical aspect. In the following, we introduce the most well-known chaos descriptive definition, the Devaney definition.

Devaney Definition [DEVANEY, 1986]. The definition states that a map, $M : S \to S$, where S is generally a compact and invariant set (under M) in $\boldsymbol{R^n}$, is said to be chaotic if:

1. M is transitive on S, namely, for any pair of nonempty open sets U and V in S, there is an integer, $k \in \boldsymbol{Z}^+$ and satisfy $f^{[k]}(U) \cap V \neq \varnothing$;
2. M displays a sensitive dependence on initial conditions, namely, $\exists \epsilon > 0$, $\forall$ and its neighborhood U, there exists $x, y \in U$, $n \in \boldsymbol{Z}^+$ that satisfies

$$|f_{[n]}(x) - f_{[n]}(y)| > \epsilon$$

3. the periodic points of are dense in S.

To explain it in a simpler way, the definition suggests that a dynamical system is classified as chaotic if 1) it is topologically transitive; 2) it is sensitive to initial conditions; 3) it has dense periodic orbits. The first and second requirement indicates that the chaotic dynamics has randomness features, and the third implies that regularity can be found within the chaotic dynamics.

1.3.3 Features of chaotic dynamics

Though there has not been a universally-agreed definition for Chaos, some common concepts and characteristics of chaotic dynamic have been well-acknowledged by the community studying Chaos theory. These concepts and properties are of greater importance especially for engineering applications than the precise mathematical definition. Hereafter, we introduce several features which worked as compass for our work.

High sensitivity to initial conditions. The most prominent feature of a chaotic system is its extreme sensitivity to initial conditions. This property allows two initial conditions with only a slight difference to result in two different trajectories. In another word, for a chaotic system, an arbitrarily small change or perturbation in the states may lead to significantly different future outcomes. The well-known 'Butterfly effect' is a metaphor for this phenomenon. The term as mentioned in Sectionsec:History comes from Lorenz's work on atmospheric studies. In his paper [Lorenz, 1972], the author drew an analogy between the small change in the initial condition of the studied dynamical system and the flapping wing of the butterfly. This flapping at the very beginning leads to a series

of events which altered tremendously the outcome weather. The huge difference in the evolving orbits is caused by the seemingly trivial change.

It should be emphasized again that as a deterministic system, the chaotic system is short-term predictable within a certain allowable tolerance. However, it is unpredictable in the long run due to its sensitivity to initial conditions. this property can be used to distinguish the chaotic system from other deterministic dynamical systems [Chen, 1998].

Positive Leading Lyapunov Exponents. The Lyapunov exponent (LE) or Lyapunov characteristic exponent can describe the characteristic of the behavior of a dynamical system both quantitatively and qualitatively. It measures the average exponential rate of the divergence of infinitesimally close trajectories. For a chaotic system, its sensitive dependence on initial conditions leads to an exponential growth of small perturbation. Thus, the existence of a positive leading Lyapunov exponent came with exponential growth is one of the characteristics possessed by the chaotic system. In fact, the LE is also the most convenient characteristic when verifying the existence of Chaos in engineering applications. From the aspect of chaoticity, we can say that the larger the positive Lyapunov exponent is, the more chaotic the system is.

Quantitatively speaking, the rate of which two trajectories with a separation of $\delta\mathbf{x_0}$ diverge can be given by the equation below,

$$|\delta\mathbf{x}(t)| \approx e^{\lambda t} |\delta\mathbf{x}_0|, \tag{1.6}$$

where λ is the Lyapunov exponent, $\delta\mathbf{x}(\mathbf{t})$ is the separation at time t. Therefore the LE can be expressed by

$$\lambda = \lim_{t\to\infty} \lim_{|\delta\mathbf{x_0}\to\mathbf{0}} \frac{1}{t} \ln \frac{\delta\mathbf{x}(t)}{\delta\mathbf{x}_0} \tag{1.7}$$

For a n-dimensional dynamic system $\dot{x_i} = f_i(x)$, $(i = 1, 2, ..., n)$there is a spectrum of LEs composed of $\{\lambda_1, \lambda_2, ..., \lambda_n\}$. To define this set of LEs, the Jacobian matrix of the form in equation 1.8 is used.

$$\mathbf{J}_{ij}(t) = \left.\frac{df_i(x)}{dx_j}\right|_{x(t)}, \tag{1.8}$$

The evolution of the tangent vectors (denoting matrix $\mathbf{Y}$) can then be expressed as equation 1.9.

$$\dot{\mathbf{Y}} = \mathbf{JY}, \tag{1.9}$$

where $\mathbf{Y}$ describes how a small change at the point $\mathbf{x}(0)$ propagates to $\mathbf{x}(t)$, with $\mathbf{Y_{ij}}(0) =$

δ_{ij} denoting its initial condition. With the above notations established, the LEs λ_i can be acquired by calculating the eigenvalues of matrix $\mathbf{\Lambda}$ which holds the following form,

$$\mathbf{\Lambda} = \lim_{t\to\infty} \frac{1}{2t} \log\left(\mathbf{Y}\left(t\right)\mathbf{Y}^{T}\left(t\right)\right), \tag{1.10}$$

The leading LE is generally the most important one among all the λ_i. As long as there is one such leading LE greater than 0, the chances for the multi-dimensional system to be chaotic are great. But it is to be noted that a positive leading may also be the sign of the production of an unbounded system trajectory. Therefore, LE greater than zero alone is not sufficient to signify chaos.

Except for the aforementioned features, there are also many other criteria for a chaotic system, such as strange attractor, Fractal and self-similarity, finite Kolmogorov-Sinai entropy, positive topological entropy, etc. We have only introduced the above ones because they are the most distinct ones and are most relevant to our work. A more detailed chaos features introduction can be found in book [Chen, 1998].

1.4 Introduction to chaos-based cryptography

1.4.1 Cryptography and Cryptosystem

Cryptography is the practice and study of techniques for secure communication in the presence of adversarial behavior [Rivest, 1991]. This scientific domain investigates the algorithms used to mask the information into an unreadable or non-understandable form to prevent private messages or information from being read by a third party. In the modern computer-centered world, cryptography is often associated with an encryption process, where the plaintext (original message) is scrambled and encrypted to cipher text, and its reversal process, the decryption.

A typical cryptosystem consists of three algorithms: encryption and decryption algorithms, which compose a cipher; and a key generation algorithm. With respect to different secret key compositions, two classes of cryptosystems are often addressed. One type is composed of the symmetric-key (private-key) algorithm and another with the asymmetric-key (public-key) algorithm [Goldreich, 2001]. For the former type, the secret keys for the cipher remain unchanged, which means the same set of keys is used for both encryption and decryption process. Whereas for asymmetric cryptosystem, different keys are employed. In most cases, there is one public key which could be known to anyone for

the encryption use, and a private key which is dedicated to the proper receiver for the information retrieval.

Hereafter we give the formal mathematical notations and of the relations between the cryptosystem components [Stinson, 2005].

Cryptosystem mathematical definition

A cryptosystem can be defined as a tupe $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \pm, \mathcal{D})$ consisting of five sets with following properties,

1. $\mathcal{P}$ is a finite set called 'plaintext space' which is composed by the possible plaintexts.
2. $\mathcal{C}$ is a finite set called 'ciphertext space' which is composed by the possible ciphertexts.
3. $\mathcal{K}$ is called 'key space' where the elements of the set are possible keys.
4. $\varepsilon = E_k : k \in \mathcal{K}$ represents a set of encryption functions which can be expressed by $E_k : \mathcal{P} \to \mathcal{C}$.
5. $\mathcal{D} = E_k : k \in \mathcal{K}$ is a set of decryption functions satisify $D_k : \mathcal{C} \to \mathcal{P}$.
6. For each encryption rule $e \in \mathcal{K}$, there is a corresponding decryption rule $d \in \mathcal{K}$ such that $D_d(E_e(p)) = p$ for all $p \in \mathcal{P}$.

Compared to well designed symmetric-key encryption, the asymmetric-key encryption are relatively slow. Thus, it is usually used for the application where small amount of data manipulations are demanded, such as secret key arrangement, authentication, digital signature and etc,. In this thesis, we focus on the symmetric cryptosystem which is believed to be fast and efficient, and more suitable for tackling problem when large amount of data is involved [Alvarez, 2006].

Two groups of ciphers are classified for symmetric-key cryposystem: **stream cipher** and **block cipher**.

Stream cipher encrypts plaintext digits by combining every single bit with a bit from keystream at a time. The keystream of stream cipher is typically pseudorandom and can be generated by pseudo-random number generator (PRNG) with the seed working as

the secret key. One typical stream cipher in practice is acquired by directly applying the XOR (exclusive OR) operation between the plaintext and the keystream bit by bit continuously. For the decryption process, the plaintext is recovered by using the XOR operation between the ciphertext and the identical keystream.

Block cipher operates on the blocks with fixed-length groups of bits and encrypts the plaintext block by block. Most block cipher algorithms are classified as iterated block ciphers where blocks of plaintext are transformed into ciphertext by repeating the application of an invertible function (round function). Each iteration is referred to as a round [Knudsen, 2011]. Among the algorithms of iterated block ciphers, one important and popular type is a substitution–permutation network (SPN) where the plaintext blocks and the key are fed. At the output, the ciphertext is obtained by applying several rounds consisting of a substitution stage followed by a permutation stage [Keliher, 1999].

1.4.2 Chaos-based cryptosystem

As mentioned in previous sections and the Introduction, chaotic system possesses many properties that are similar to the properties of the cryptosytem. This indicates that the chaotic system bears advantages which are in favor of cryptographic use. Thus, the domain of crytosystems adopting chaotic systems and maps emerges. Most chaos-based cryptosystems that have now investigated are symmetric-key cryptosystem. Therefore, hereafter we use the term chaos-based cryptosystem for the simplicity.

The mechanism of chaos-based crytosystem is to introduce chaotic elements to the cipher and keystream generation algorithm. These processes can be described by 1.2. The chaotic outputs generated and controlled by the chaotic key are used as the keystream to mask the plaintext into ciphertext. At the authorized receiver end (if a communication system is concerned), the plaintext can be recovered by decrypting the ciphertext using the keystream generated from the identical chaotic pseudo-random number generator.

There are ways to generate chaotic keystream for encryption use and design the cryptosystem. However, to achieve better confusion and diffusion, the PRNG based on chaotic systems (CPRNG) which can generate chaotic pseudo-random number is preferred. The outputs of CPRNG have excellent pseudo-randonmess and high sensitivity to initial conditions which lead to a keystream with large key space, high sensitivity to the secret key and pseudo-random properties. These properties are crucial for both stream cipher and block cipher design.

In 1.3, a block cipher based on confusion and diffusion with the keystream provided

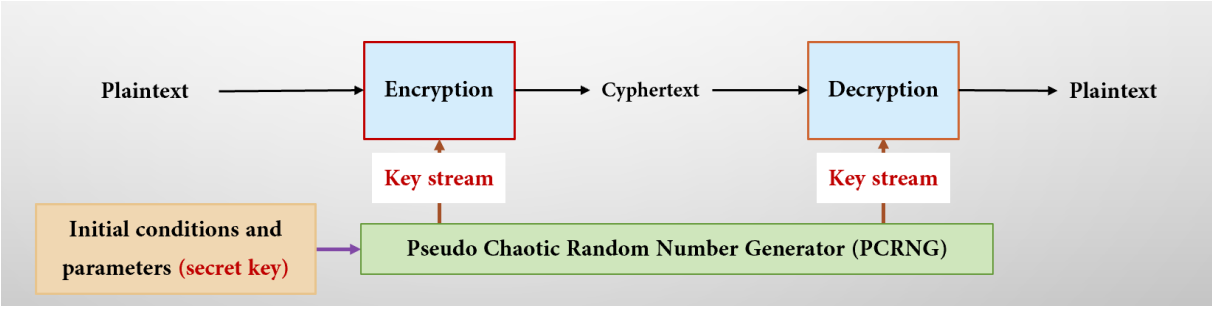


Figure 1.2 – Scheme of the chaos-based cryptosystem

by CPRNG. **Confusion** means using transformations to complicate the dependence of the statistics of the ciphertext on the statistics of the plaintext. **Diffusion** aims to spread the influence of a single element of the plaintext over as many elements of the ciphertext as possible [Kocarev, 2001].

For image encryption cryptosysystem, which will be discussed in our work, the confusion layer is employed to conceal and complicate the relationship among the keystream, the plain image and cipher image; whereas the diffusion layer focuses on diffusing the influence of each plain image bit to the rest of the image and change the statistical properties of the plain image. Typically, the confusion layer contains permutation operations which are used to relocate the pixel positions, and substitution operations which change the pixel values [Özkaynak, 2018a][Patidar, 2009a].

1.4.3 Necessary requirements for efficient CPRNG

Like any PRNG is required, the CPRNG should also generate pseudo-random outputs with uniform distribution. The uniformity can be tested by the histogram of the outputs and the Chi-square statistical test. The pseudo-randomness can be confirmed applying NIST test suite [Qiao, 2020] [Bassham, 2010].

Histogram

A histogram is an approximate representation of the distribution of numerical data. It is used to give the audience a rough idea of the distribution and the density of the data. The term is first introduced by English mathematician and biostatistician Karl Pearson [Pearson, 1894]. As mentioned above, for a well-designed CPRNG, its outputs should have a uniform distribution. This means that the output sequence must have an almost

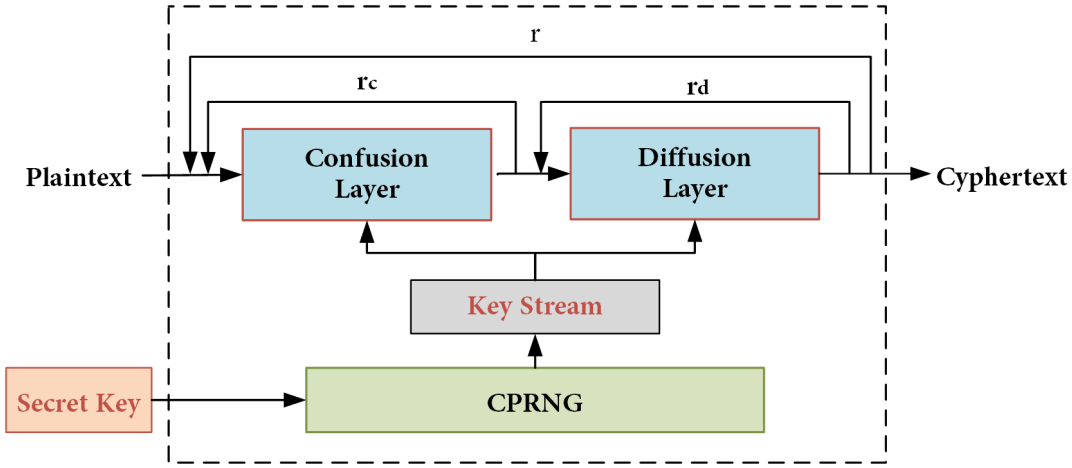
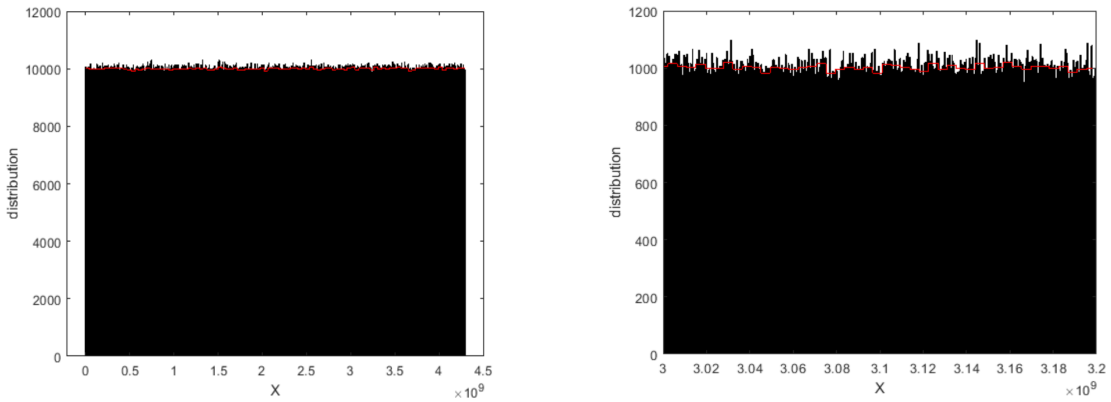


Figure 1.3 – Scheme of the confusion-diffusion in chaos-based cryptosystem



(a) 10^7 chaotic samples ploted in 1000 classes (b) Partial histogram

Figure 1.4 – Histogram of the outputs of CPRNG proposed in [Qiao, 2019]

equal probability for its elements to take one of the values among all the possible output values. As an example, two diagrams of histogram of the successfully proposed CPRNG in [Qiao, 2019] with a uniform distribution are give below. The 10000000 CPRNG outputs are divided into 1000 different bins and the histogram is plotted in Fig . 1.4a. A zoomed-in figure within the CPRNG output number range of 3×10^9 to 3.2×10^9 is given in Fig. 1.4b.

Chi-square test

A chi-square test (also Chi-square or χ^2 test) is a statistical hypothesis test which is often used to determine whether there is a significant difference between the expected and

observed occurrences of discrete data. To do the chi-square test, one should first set a null hypothesis H_0 and a level of significance α. If the test statistic computed from the observations follows a χ^2 distribution (the test statistic is smaller than the critical value (χ_c^2) under given α and degree of freedom), then the null hypothesis can not be rejected.

For the Chi-square evaluating the distribution of the CPRNG outputs, the H_0 is that the outputs distribute uniformly over the range of all the possible values. The level of significance is set to 0.05, and the test statistics can be calculated by the formula given below,

$$\chi_e^2 = \sum_{k=1}^{n} \frac{O_i - E_i}{E_i} \tag{1.11}$$

The χ_e^2 represents the test statistic; n is the number of bins in the histogram; O_i is the observed values in bin i; E_i is the expected value in bin i.

For the histogram example given above, 1000 bins are chosen for the FCPRNG outputs. The degree of freedom of the test is $1000 - 1 = 900$. The critical value χ_c^2 under given degree of freedom and level of significance$\alpha = 0.05$ can be acquired equal to 1073.6427. So when the test statistic χ_e^2 smaller than 1088.4871, the null hypothesis is accepted which concludes that the distribution is indeed uniform.

NIST test suite

NIST (National Institute of Standard and Technology) test suite is nowadays one of the the most powerful and widely used tool for randomness check. The test suite consists of 15 different randomness tests outputting 188 test results. The tests are given in the Apprendix A.

The methodology behind each test is based on statistical test and the sequence generated by the generator is analyzed as follows:

1. The null hypothesis H_0 which states that the sequence is random is assumed.
2. The statistic for the tests are calculated.
3. The probability value $P \in [0, 1]$ is calculated.
4. The probability value P is compared with the significance level α of the test, where $\alpha \in [0.001, 0.01]$. If $P \geq \alpha$ then the H_0 is accepted, otherwise it is rejected.

To do the NIST test, a total of 10^8 bits (100 blocks of 10^6 bits) is recommended for the sequence to be tested. The tests results for the 100 blocks are consolidated. With a total pass rate greater than 96/100, the test is considered passed.

1.4.4 Security analysis of the cryptosystem

Cryptanalysis refers to the process of analyzing information systems in order to understand hidden aspects of the systems. In other word, cryptanalysis studies the way to breach cryptosystem. This domain of science is of great importance, since the security is the foremost concern for a cryptosystem. When a cryptosystem is designed, the security analysis should always be carried out. Though it is not possible to run a exhaustive list of security tests, analysis against some common attacks must be covered to evaluate the security of the designed cryptosystem.

Several types of security attacks are discussed in the following.

Cryptographic attacks

According to [Stinson, 2005], there are four levels of common known attacks on cryptography. The attacks are ordered from the hardest to easiest to carry out.

1. Ciphertext-only attack : the attacker can only access to some cipheretexts.
2. Known-plaintext attack : the attacker has access to some pairs of plaintexts and the corresponding ciphertexts.
3. Chosen plaintext attack : the attacker has temporary access to the encryption process and he can encrypt some specific plaintexts to obtain their corresponding ciphertexts.
4. Chosen ciphertext attack : the attacker has temporary access to the decryption process and he can use some specific ciphertexts in decryption to obtain their corresponding plaintexts.

Apart from the four attacks listed above, there are some other more specialized attacks based on them. For example, the differential attack is also one of the important attacks that need to be checked carefully when designing a cryptosystem with block cipher.

Differential attack

The differential cryptanalysis is a chosen-plaintext attack and was introduced for the first time by [Biham, 1991]. To practice this attack, adversary will trace the differences in the ciphertexts acquired after making a small change in the plaintext. By observing whether the ciphertext exhibits non-random behavior following the small change in plaintext and exploiting such properties, the secret key might be recovered. In [Chen, 2004],

authors introduced a variation of differential cryptanalysis applied to image encryption. To defeat the differential attack, the ciphertext should have a high sensitivity to even a tiny change in the plaintext.

Brute-force attack

The brute-force attack, as its name indicates, involves exhaustively testing all the possible keys brutally and hoping to guess the correct one. The quicker one can acquire the key of the cryptosystem through this blind and brutal testing process, the weaker the cryptosystem is. Therefore, the keyspace size of the cryptosystem is vital in resisting brute-force attack. With the computer speed of today, It is generally agreed that to provide a sufficient security against brute-force attacks, the key space size should be greater than 2^{128} [Özkaynak, 2018b].

Statistical attack

Statistical attack aims to extract the relationships between the plain image and the ciphered image by exploiting the statistical weaknesses in a cryptosystem. According to Shannon's theory of information and communication, it is possible to break many types of cryptosystems by statistical attack [Shannon, 1949].

If a PRNG is used to generate the keystream, then to resist against the statitical attack, its output sequence must pass all statistical tests for randomness, and the period of the pseudo-random sequence should be as large as possible [Alvarez, 2006] where the NIST test suits discussed in Section is always applied. From the perspective of image encryption, statistical attacks can be resisted if the encrypted images have a uniform distribution and there is no statistical correlation in the pixels of encrypted images.

For all the above-discussed attacks, the aim of the attacker is to break the cryptosystem by gaining access to the key. That is to say, for a well-designed cryptosystem, even if the attacker can access the whole cryptosystem but the specific key used by the user, it would still be extremely difficult for him to break the cryptosystem and retrieve the messages. The importance of the secret key is indisputable. A sufficient size of the key space, as well as an efficient confusion and diffusion process, are essential for secure block ciphers.

1.5 State of art of existing work

There are many conventional and standard encryption algorithms, for example, AES (Advanced Encryption Standard) working on a symmetric-key algorithm, and RSA (Rivest-Shamir-Adleman) public key cryptosystem. Some of these conventional algorithms are still in use but become less competent due to their computation complexity. In addition, most of these algorithms do not have sufficient diffusion level. When dealing with the big bulk of data such as digital images and video, the strong correlation between pixels makes it difficult to gain high-security level results with these conventional algorithms that are incompetent in diffusion [Ahmed, 2007][Liu, 2010]. Therefore, new encryption methods with improved security performance for large-size data encryption should be designed.

As has been illustrated in Section 1.3.3 and Introduction, chaotic dynamics is a perfect candidate for cryptography, due to its advantageous nature such as its sensitivity to initial conditions and random-like behavior. The first try of chaos-based cryptosystem design is conducted by Robert Mattews in 1989 where a chaotic logistic map is employed [Matthews, 1989]. Since then, research applying chaotic systems to cryptographic uses has grown considerably.

For a chaos-based cryptosystem encrypting large size of data, two crucial components need to be designed. The encryption scheme consists of confusion and diffusion process, and the key-stream generation, which normally involves the design of CPRNG. In the following, we first discuss the existing chaos-based cryptosystems and focus on those with confusion-diffusion schemes in section 1.5.1. Then in section 1.5.2, we will discuss the current research status for PRNG and CPRNG.

1.5.1 Chaos-based cryptography

As mentioned before, the block cipher with confusion-diffusion structure has been proved efficient to design large-size data encryption, such as the encryption of image or video [Shannon, 1949] [Patidar, 2009b].

The first cipher with confusion-diffusion structure is proposed by Fridrich in his work [Fridrich, 1998] in 1998. He adopted a invertible two dimensional chaotic map to construct the cipher which proved to be able to eliminate the visual redundancy among pixels. In the meantime, the applicability of chaotic systems and the confusion-diffusion structure for image encryption applications is confirmed. Ever since then, the confusion-diffusion cipher structures with different encryption approaches have been proposed by many researchers.

As described in Section 1.4.1, generally speaking, the confusion process consists of permutation and substitution operations. The diffusion process, if an image encryption algorithm is considered, aims at spreading the influence of each bit in the plain image to the cipher image.

For the *permutation* process applying chaotic systems, three approaches have been widely discussed. The first method involves the permutation of image pixels based on a sorting process [Hua, 2018]. The chaotic sequence is first rearranged in a certain order (descending order for example). Then the permutation of plain image is conducted according to the sorted chaotic sequence. It is easy to notice that this approach requires the length of the chaotic sequence to be equal to the plain image size. Hence, when the size of plain image is large, the burden of generating chaotic sequences with the same length leads to heavy computational consumption and hinders the efficiency of the entire cryptosystem.

The second permutation method works on the bit-level. The pixel values of the original plain image are first converted to binary bits, then the permutation is accomplished by shifting bits. This type of permutation possesses the advantage of achieving confusion and diffusion at the same time, since the pixel values are changed during the process which also has the diffusion effect [El Assad, 2016][Li, 2017][Cao, 2018][Li, 2020b]. However, similar to the first pixel-level approach, a considerable high computation power is needed when a large size plain image is to be encrypted.

For *substitution* purposes, the substitution box (S-box) is usually employed. The S-box is a type of basic nonlinear element in the encryption algorithm which is also the most important part of the Advanced Encryption Standard (AES) and Data Encryption Standard (DES) encryption algorithm [Farah, 2020a]. Many researchers have proposed cryptosystem for image encryption with S-box based on different cipher scheme and diverse chaotic maps [Chen, 2007][Wang, 2012][Khan, 2013]. For example, in [Wang, 2014], the author divided the plain image into groups and apply the dynamic S-box strategy based on the logistic map and tent map for different groups.

For the diffusion algorithms, two important categories can be found in the existing work. The first type as discussed in [Farajallah, 2018a] and [Wang, 2019] diffuses a slight change in the plaintext directly to the ciphertext. Yet, one problem arises: the diffusion performance is related to the position of the changing pixel. Thus, several rounds are often needed to achieve good encryption results which demand greater computational power and time. The second type of approach for diffusion makes use of the change in plaintext to

alter the initial conditions of the chaotic system. Thus, total different chaotic sequences are acquired due to the sensitivity of the system to initial conditions. Subsequentially, distinguished ciphertexts are obtained [Diab, 2018][Zhang, 2011][Liu, 2012].

Apart from the confusion-diffusion structure, there are many other techniques and algorithms incorporating chaotic dynamics that have been investigated and developed for encryption applications, such as compressed sensing [Zhou, 2016][Chai, 2017], quantum coding [Luo, 2020], fractional Fourier transform [Farah, 2020b], DNA encoding [Chai, 2019a][Zhang, 2014] and so on. We would like to draw readers' attention to the use of DNA encoding and decoding in the deisgn of chaos-based cryptosystem. This orientation has emerged and received unceasing attention in recent years, which is also one of our research interest.

Since the first DNA computation experiment conducted by Adleman [Adleman, 1994] in 1994, the idea of implementing DNA computation in cryptography has been brought to light. DNA computation possesses many advantages such as massive parallelism, huge storage capacity, and low energy consumption [Xue, 2020]. Intuitively, these advantages can also result in an efficient secure cryptosystem if designed properly, although there are still technical limitations at the experimental level in biochemistry for DNA cryptography where the DNA molecule is used to carry information. Scientists believe that these difficulties will be overcome in the foreseeable future. Therefore, many researchers have been triggered to explore the possibility of designing an efficient, practical, and implementable encryption algorithm using DNA encoding and decoding methods for future biochemistry implementation.

Regarding the application of image encryption, a dynamic DNA-based cipher with two chaotic maps, namely fractional Chen's system and Lorenz system, has been investigated in [Zhang, 2016]. In [Wu J, 2018], the authors analyzed an encryption algorithm combining a novel proposed 2D Hénon-sine map DNA ciphering methods and achieved better performance in terms of security. A new one-time pad encryption scheme has been designed based on the coupled map lattices (CML system) and DNA diffusion sequence in [Wang, 2020] with good security analysis results. In [Wen, 2020], authors proposed a color light field image encryption algorithm using DNA sequence and well-known chaotic systems (logistic map and Chen system), which is proved to be applicable, reliable, and secure enough.

With respect to the chaotic systems adopted for the chaos-based cryptographical system, the classical chaotic maps and systems with integer derivative order are not the only

ones that can be used. In recent years, the possibility of employing the *fractional chaotic systems* in the cryptosystem design also caught up a couple of researchers' eyes. The larger secret key space size which can be acquired, together with the intrinsic complexity the fractional calculus bears, makes the fractional chaotic system a promising candidate when comes to the cipher design.

In [Radwan, 2012], the authors employed the fractional-order Lorenz system to design a simple encryption algorithm based only on pixel confusion. To realize the confusion process, a multiplexing mechanism is adopted to choose the system states starting from three different initial conditions. For every color layer on each pixel position, exclusive-or (XOR) operations are performed between the chosen states and the pixel value on the color layers.

In [Yang, 2020], authors explored and analyzed the characteristic of a new fractional-order hyperchaotic system based on the Adomian decomposition method (ADM). Then a novel image encryption algorithm is proposed applying the discussed system and Galois field (GF). The image pixels scrambling is achieved by column cyclic shift. The diffusion process is conducted through the GF algorithm.

In a recently published paper [Gao, 2021], the characteristics of a fractional hyperchaotic system are investigated. A multi-image cryptosystem is then proposed based on the fractional system. The system output sequence is used to fuse multiple grayscale images into a color image. The scrambling and diffusion are then achieved using again the output chaotic sequences.

For the above-mentioned papers, some problems also arise which affect the secure performances and efficiency of the cryptosystem. Firstly, all of the proposed cryptosystems only employed one unique fractional chaotic system. In addition, most of them directly adopt the system states as the keystream used to permute the image pixels. Hence, the complexity of the generated keystream and the encryption scheme was relatively minor. Secondly, the chaotic and pseudo-random properties of the system outputs over the parameters have not been tested and confirmed. Therefore, there is no guarantee of the increase of keyspace size, which authors deemed in their papers (such as in [Radwan, 2012] and [Gao, 2021]). Last but not the least, most of the existing cryptosystems based on the fractional chaotic system require iterating the system many times (at least as many iterations as the size of the image). Hence, a great computational power is needed.

1.5.2 Chaotic pseudo-random number generator

The pseudo random number generator (PRNG), unlike True random number generator (TRNG) which is generated from real-life phenomena, is generated by a deterministic system with an initial seed (initial state for the system). As its name suggests, the numbers generated by PRNG are not truly random, they only appear to be random. Since the pseudo-random numbers are generated from a deterministic system, the PRNG is easier to control and reproduce compared to TRNG. The possibility of reproduction using the same initial seed also make it possible to use PRNG for cryptography applications. Because if a symmetric-key cryptosystem is considered, the same keystream needs to be reproduced for the decryption process.

The typical properties of pseudo-random numbers are uniform distribution, independence between two pseudo-random sequences produced by different seeds, and long periods[Hellekalek, 1998]. To evaluate the randomness of a produced pseudo-random number sequence, some commonly used tests like NIST test suite TestU01 [LECUYER, 2007], ENT and DIEHARD can be adopted. It is to be remarked that the NIST test suite is the most frequently used randomness test in the cryptographic applications, since it is easy to implement and can work with different data format.

The Chaotic Pseudo-Random Number Generator (CPRNG) is the PRNG designed by chaotic maps or systems. Compared to some commonly used PRNGs based on numerical methods (such as linear congruential generator, lagged Fibonacci generator, and linear feedback shift registers-based generator(LFSR)), the sequence generated by a properly designed CPRNG has a higher security level and greater efficiency. The properties, such as longer periods, lower computational requirements, and complex nonlinear behavior, make the PCRNG an excellent alternate for many applications demanding the use of PRNG, including the design of a secure cryptosystem.

For some chaos-based cryptosystems, the CPRNG is used to generate pseudo-random keystream. The initial conditions and parameters of systems composing the CPRNG work as the secret keys. According to Kerckhoffs' principle and Shannon's information security theory, the CPRNG is crucial for the security of a cryptosystem. To establish a solid foundation for the design of a cryptosystem with a high level of security, the adoption of a properly designed CPRNG is extremely important. The CPRNG should not only possess a large keyspace, but it must be able to produce keystreams with pseudo-randomness and high sensitivity properties as well.

The CPRNG can take one of the structures, either cascaded or parallel. The former

takes the output of one chaotic system as the input of another, which can enhance the chaotic property of the CPRNG. But the errors caused by dynamical degradation may accumulate through this cascaded algorithm over platforms of finite precision [Lan, 2018]. A parallel structure, however, arranges the chaotic systems adopted in parallel. For this type of structure, as the initial conditions and parameters of all the employed systems can work as the secret key, the keyspace of the resulting CPRNG and cryptosystem is effectively enlarged [Li, 2020a]. Nevertheless, the way to integrate the chaotic sequences to enhance the chaotic property and ensure the pseudo-random characteristic remains to be considered and investigated.

With respect to the problems mentioned above, researchers have been working hard to propose effective integrating approaches. O. Garasym et al. proposed a CPRNG by exploring a chaotic coupling method with a topology network in their work [Garasym, 2016]. In [Sahari, 2018], authors coupled two chaotic maps, a piecewise linear chaotic map (PWLCM) and a 2D logistic map, to construct a CPRNG for color image encryption cipher. C. Zhu, in his work [Zhu, 2019], introduced a CPRNG based on a coupled logistic-tent chaotic system; better chaotic behavior and greater parameter range have proved to be acquired successfully.

Apart from the coupling methods which has been proven effective by the mentioned work and many others, mixing is anther strategy to integrate effectively the multiplex chaotic sequence. A robust CPRNG connecting in parallel the skew tent map and PWLCM was proposed and implemented for a block cipher in [El Assad, 2016]; the large period for all generated sequence was ensured by a linear feedback shift resgister in the CPRNG structure. The CPRNG introduced in [Hamza, 2017] adopted Chen chaotic system and combined its three coordinates; the cascading structure and mixing of the orbit samples were proposed to resolve the degradation problem.

Other techniques for CPRNG design are also discussed in the existing literature. For instance, a CPRNG sheme based on coupled map lattice with time-varying delay was proposed in [Lv, 2018]; an image encyrption algorithm applying the proposed CPRNG was then dicussed and proven to be able resist the differential attack. In [Guyeux, 2010], authors proposed a fast CPRNG combing ISACC (indirection, shift, accumulate, add, and count) cipher and XOR shift generators using the chaotic iterations.

For most of the CPRNG investigated in current literature, low-dimensional chaotic maps are employed. The advantages of applying such maps lie in the fact that they are relatively simple and easier to implement numerically. Yet, there are some researchers

working on the higher dimensional chaotic systems despite of their relatively more complex dynamics and the numerical implementation difficulties. Taking the CPRNG proposed in [Lynnyk, 2015] as an example, the author employed a generalized Lorenz system to propose two CPRNGs with slight difference in the way which the three dimensional coordinates were combined; the output sequences passed the NIST test suite successfully.

As mentioned in the Introduction of this manuscript, fractional calculus in engineering applications has started to be studied in the last decade. The possibility of integrating fractional systems into the PRNG design has also been partially investigated. In [Ozkaynak, 2020], the author studied the fractional order chaotic Chua system and proposed a fractional CPRNG (FCPRNG) by converting the generated outputs to numbers between 0 and 15; some simple and naive tests were carried out like Monobit test and Chi-square test to evaluate the pseudo-randomness of the generated sequence. However, compared to the exhaustively investigated CPRNG based on integer order chaotic maps and systems, very little work on FPCRNG design has been conducted and awaits more in-depth research.

1.5.3 Problem statements

Regarding the image encryption purpose, one can find numerous works employing different algorithms and schemes. However, among these papers, some do not possess the high security as they claimed and are not sufficiently secure against certain attacks.

In paper [Zhang, 2013b], a chaos-based image cryptosystem using logistic map to perform the dependent diffusion has been proposed whose security is analyzed by [Farajallah, 2018b]. The latter pointed out that the diffusion effect achieved by the cryptosystem can in fact be removed, because its argument is exposed in the ciphered image. The resulting reduced keyspace and the recovery of the permuted version of the ciphered image make the cryptosystem vulnerable to brute-force attack and chosen plaintext attack. Though it has been proved that iterating several times (rounds) the encryption scheme may lead to a more secure cryptosystem, as it is for the case of [Zhang, 2013b] (after iterating the encryption scheme twice, the cryptosystem is secure). The low efficiency comes along with the iterations results in the inefficiency of the whole cipher. In addition, the trick of multi-rounds is not always a guarantee for secure cryptosystem. The [Chen, 2017] broke the cryptosystem proposed in [Zhou, 2015] by differential attack, even if the encryption algorithm has been performed several rounds.

Though researchers in the domain have tried to design one-round cryptosystem with

sufficient efficiency and security, the problems still persist. Two groups of secret key were used separately for the chaotic system and rectangular ransformation permutation in [Wu, 2017]. But in [Zhu, 2018], authors remonstrated that the rectangular transformation does not work for all pixels. What's more, [Muhammad, 2019] proved that the scheme proposed in [Wu, 2017] can be cracked using brute-force attach and chosen plaintext attack. The cryptosystem discussed in [Boriga, 2014] has been broken by [Wen, 2017] using differential attack; the latter pointed out that the scheme can be degraded to a diffusion-only algorithm and permutation-only algorithm.

The cracking of many insecure cryptosystems shows the following common problems in image encryption algorithm design. The structure consists of confusion and diffusion is either insufficient where multiple rounds are needed, thus, reducing the efficiency of the scheme [Zhang, 2013b][Farajallah, 2018b][Zhou, 2015], or lack of complexity and insecurity, which leads to the leak of keystream information or even secret keys [Wu, 2017][Zhu, 2018][Muhammad, 2019][Huang, 2020]. Another commonly incurred problem is the exposure of the secret key due to the lack of cryptographic features of the keystream [Liu, 2018][Ma, 2020][Wang, 2018]. To overcome the abovementioned problems, an encryption scheme with a sufficiently secure and complex confusion-diffusion structure, and a keystream facilitating the cryptographic use is in demand. Adopting a properly designed CPRNG could be a possible solution for the latter problem. The output sequence with good chaotic and random properties can work as the keystream of the cryptosystem.

From the aspect of the system integration into the image encryption algorithm, a new trend has been under investigation using fractional chaotic systems in recent years with the expectation that the introduced fractional derivatives can enlarge the key space size. As stipulated in Section 1.5.1, the existing work on this topic typically employs only one single fractional chaotic system. Thus, the complexity and security of the keystream are not secured. In addition, with the system states directly used as the keystream, many proposed image cryptosystems require a long sequence (the size of the image, for instance) for the encryption, which renders the encryption algorithm time-consuming and less efficient. What is more, similar problems of the encryption algorithm based on classical order chaotic system exist, for instance, the lack of complexity and security of confusion-diffusion structure, and the unguaranteed cryptographic performance of the keystream.

Compared to the use of classical chaotic maps and systems, the chaos-based cryptosystem adopting fractional chaotic system is still relatively undeveloped. The intrinsic

complexity of fractional calculus and its digital implementation add to the research difficulties. The fact that the chaotic behavior of the fractional chaotic system is dependent on the adopted numerical calculation methods and fractional calculus characterizations also slows the broader application of such systems. Yet, great cryptographic potential lies behind the intricacy of fractional chaotic system implementation and FPCRNG design.

1.6 Conclusion

.

In this chapter, some fundamentals on chaotic systems and chaos-based cryptography have been introduced to facilitate the understanding of our research topic and background. Focusing on the confusion-diffusion encryption scheme, we analyzed some literature and stipulated the existing problems. The work applying DNA computing is also introduced as one direction for image cryptosystem design. Regarding the CPRNG, we demonstrated its current development and emphasized its crucial rule in designing secure encryption algorithms. In addition, we oriented the readers' attention to the use of fractional chaotic systems in cryptosystem and FCPRNG design.

Chapter 2

Fractional calculus and fractional chaotic systems

2.1 Introduction

In the previous chapter, we introduced the basics and fundamentals of chaotic systems and the state of the art of chaos-based cryptosystems. In this chapter, we give introductions to another significant component of the thesis, the fractional calculus, and fractional dynamic system. First, the definition and different fractional integral and derivative characterizations are displayed. We then introduce the fractional chaotic systems, particularly those used for our Fractional chaotic pseudo-random number generator design.

2.2 Basics on fractional calculus

2.2.1 Definition

Fractional calculus is a branch of mathematical analysis that studies the several different possibilities of defining real number powers or complex number powers of the differentiation operator and integration operator. The continuous integro-differential operator is dedined as follows, where a and t are the bonds of the operation and $\alpha \in \mathrm{R}$ [Petráš, 2011]

$$ {}_aD_t^\alpha = \begin{cases} \dfrac{d^\alpha}{dt^\alpha}, & \alpha > 0 \\ 1, & \alpha = 0 \\ \displaystyle\int_a^t (d\tau)^\alpha, & \alpha < 0 \end{cases} \tag{2.1} $$

There are many definitions (characterizations) for fractional calculus, for example, Gr*ü*nwald-Letnikov (GL) definition, Riemann-Liouville (RL) definition, Caputo defini-

tion, etc,. These definitions were brought up and discussed by different mathematicians independently, who gave them different names and forms over time. However, from the aspect of researchers in the automatic domain, these definitions are equivalent under certain initial conditions. Therefore, instead of using the term 'definitions' as some researchers do, we adopt the term 'characterization' in the following.

2.2.2 Fractional derivatives under Grünwald-Letnikov (GL) fractional characterization

The fractional derivative of GL type can be derived from integer order derivatives. For a continuous function $f(t)$, its first derivative can be expressed as,

$$\frac{d}{dt}f(t) = f'(t) = \lim_{h\to 0}\frac{r(t) - f(t-h)}{h}. \tag{2.2}$$

Using the equation (2.2) twice, following expression can be obtained for the second derivative of $f(t)$,

$$\begin{aligned}\frac{d^2}{dt^2} = f''(t) &= \lim_{h\to 0}\frac{f'(t) - f'(t-h)}{h} \\ &= \lim_{h\to 0}\frac{f(t) - 2f(t-h) + f(t-2h)}{h^2}.\end{aligned} \tag{2.3}$$

Repeating the process, a general formula for n-derivative of the function $f(t)$ by t for $n \in \mathrm{N}, j > n$ can be derived as following for positive value of n,

$$\frac{d^n}{dt^n}f(t) = f^{(n)}(t) = \lim_{h\to 0}\frac{1}{h^n}\sum_{j=0}^{n}(-1)^j\binom{n}{j}f(t-jh), \tag{2.4}$$

where the binomial coefficients is expressed as,

$$\binom{n}{j} = \frac{n(n-1)(n-2)...(n-j+1)}{j!} = \frac{n!}{j!(n-j)!}. \tag{2.5}$$

The formula (2.4) can be rewritten as flowing equation with $-n$ substituting n,

$$\frac{d^{-n}}{dt^{-n}}f(t) = f^{-n}(t) = \lim_{h\to 0}\frac{1}{h^n}\sum_{j=0}^{n}f(t-jh), \tag{2.6}$$

where n is a positive integer number and $\begin{bmatrix} n \\ j \end{bmatrix}$ is defined as,

$$\begin{bmatrix} n \\ j \end{bmatrix} = \frac{n(n+1)...(n+j-1)}{j!}. \tag{2.7}$$

With the above derived classical integer-order derivatives, the fractional-order derivative definition of order $\alpha, (\alpha \in \mathbf{R})$ as given in the box below.

Grünwald-Letnikov (GL) fractional derivative

The Grünwald-Letnikov (GL) characterization of fractional derivatives can be writen as

$${}_aD_t^\alpha f(t) = \lim_{h\to 0} \frac{1}{h^\alpha} \sum_{j=0}^{[\frac{t-a}{h}]} (-1)^j \binom{\alpha}{j} f(t-jh) \tag{2.8}$$

where $[x]$ means the integer part of x, a and t are the bounds of operation for ${}_aD_t^\alpha f(t)$, and $\binom{\alpha}{j}$ is defined as

$$\binom{\alpha}{j} = \frac{\alpha!}{j!(\alpha-j)!} = \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \tag{2.9}$$

The $\Gamma(.)$ denotes the Euler Gamma function holding the form as following,

$$\Gamma(\alpha) = \int_0^\infty \frac{t^{\alpha-1}}{e^t} dt \tag{2.10}$$

2.2.3 Rienmann-Liouville fractional integrals and derivatives

For the Rienmann-Liouville characterisations of integrals and derivatives, the Riemann-Liouville n-fold integral folding the form as follows is first considered,

$$\int_a^t \int_a^{t_n} \int_a^{t_{n-1}} ... \int_a^{t_3} \int_a^{t_2} f(t_1) dt_1 dt_2 ... dt_{n-1} dt_n = \frac{1}{\Gamma(n)} \int_a^t \frac{f(\tau)}{(t-\tau)^{1-n}} d\tau \tag{2.11}$$

for $n \in \mathrm{N}, n > 0$. The equation (2.11) can be rewrite by substituting the n by the fractional order α as follows,

$$ {}_a\int_t^\alpha f(t) \equiv {}_a D_t^{-\alpha} f(t) = \frac{1}{\Gamma(-\alpha)} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha+1}} d\tau \tag{2.12} $$

for $\alpha, a \in \mathrm{R}, \alpha < 0$.

From the relation given in equation (2.12), the fractional derivative of Riemann-Liouville type can be rewritten as the equation given in the box below.

Rienmann-Liouville fractional derivative

The fractional derivative of order α defined by Rienmann-Liouville definition can be illusctrated as the form below,

$$ {}_aD_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \tag{2.13} $$

where $(n-1 < \alpha < n)$, a and t are the limits of operation ${}_aD_t^\alpha f(t)$.

In equation (2.13), $\Gamma(.)$ stands for the Euler gamma function defined in (2.10).

For the case where fractional order α is greater than 0 but less than 1 $(0 < \alpha < 1)$, and $f(t)$ being a causal function of t, that is $f(t) = 0$ for $t < 0$, the fractional integral is defined as:

$$ {}_0D_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}} d\tau \tag{2.14} $$

for $0 < \alpha < 1, t > 0$ and the expression for the fractional order derivative is:

$$ {}_0D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_0^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} = d\tau \tag{2.15} $$

2.2.4 Fractional derivatives of Caputo type

The fractional derivative under Caputo characterization can be written as given in the box [Caputo, 1967].

Caputo characterisation

The Caputo definition of fractional derivatives can be written as:

$$ {}_aD_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \int_a^t \frac{f^{(n)}(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau, \text{for n} - 1 < \alpha < \text{n} \quad (2.16) $$

It is to be remarked that under the homogenous initial conditions, the Riemann-Liouville and the Caputo derivatives are equivalent. If the notions ${}^{R}L_aD_t^{\alpha} f(t)$ and ${}_a^{C}D_t^{\alpha} f(t)$ to denote the fractional derivative under RL and Caputo characterizations, then the relation between them is:

$$ {}_a^{RL}D_t^{\alpha} f(t) = {}_a^{C} D_t^{\alpha} f(t) + \sum_{k=0}^{n-1} \frac{(t-a)^{k-\alpha}}{\Gamma(k-\alpha+1)} f^{(k)}(a) \qquad (2.17) $$

for $f^{(k)}(a) = 0, (k = 0, 1, ..., n-1)$.

It is also worth mentioning that the initial conditions for the fractional-order differential equations with the Caputo derivatives are in the same form as for the integer-order differential equations. This provides the fractional derivatives of this type clear interpretations for their initial conditions ($f(a), f'(a), f''(a)$ and etc.), which is required by the applied problems. Therefore, the fractional derivative under Caputo characterization is the most popular and widely used definition when it comes to engineering applications.

2.3 Fractional order dynamic systems

It has been found that in interdisciplinary fields, there are many systems which possess inherited properties, such as visco-elastic systems [Bagley, 1991], electromagnetic waves [Heaviside, 1971], diffusion wave [El-Sayed, 1996], quantum evolution of complex systems and quantitative finance [Laskin, 2000]. All the precedent states of these systems influence their current states , which fits the idea of fractional calculus.

In the fields of dynamical systems and control theory, a fractional-order system is a dynamical system that can be modeled by a fractional differential equation containing derivatives of non-integer order [Monje, 2010].

2.3.1 Fractional Linear time invariant system

A general fractional-order system can be described by a fractional differential equation of the form,

$$\begin{aligned} a_n D^{\alpha_n} y(t) + a_{n-1} D^{\alpha_{n-1}} y(t) + ... + a_0 D^{\alpha_0} y(t) \\ = b_m D^{\alpha_m} u(t) + b_{m-1} D^{\beta m-1} u(t) + ... + b_0 D^{\beta_0} u(t) \end{aligned} \tag{2.18}$$

where $D_\gamma \equiv_0 D_t^\gamma$ denotes the Gr*ddotu*wald-Letnikov, the Rienmman-Liouville or the Caputo's fractional derivative [Podlubny, 1999]. The corresponding transfer function of *incommuensurate* real orders has the following form,

$$G(s) = \frac{b_m s^{\beta_m} + ... + b_1 s^{\beta_1} + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + ... + a_1 s^{\alpha_1} + a_0 s^{\alpha_0}} = \frac{Q(s_{\beta_k})}{P(s_{\alpha_k})} \tag{2.19}$$

In the frequency domain, equation (2.18) can be expressed as following [Petras, 2000],

$$G(j\omega) = \frac{b_m (j\omega)^{\beta_m} + ... + b_1 (j\omega)^{\beta_1} + b_0 (j\omega)^{\beta_0}}{a_n (j\omega)^{\alpha_m} + ... + a_1 (j\omega)^{\alpha_1} + a_0 (j\omega)^{\alpha_0}} \tag{2.20}$$

where a_k $(k = 0, ..., n)$, b_k $(k = 0, ..., m)$ are constants, and α_k $(k = 0, ..., n)$, β_k $(k = 0, ..., n)$ are arbitrary real or rational numbers and without loss of generality they can be arranged as $\alpha_n > \alpha n - 1 > ... > \alpha_0$, and $\beta_m > \beta m - 1 > ... > \beta_0$.

The incommensurate order system 2.3.1 can also be expressed in commensurate form by the multivalued transfer function [Merrikh-Bayat, 2009],

$$H(s) = \frac{b_m s^{m/\nu} + ... + b_1 s^{1/\nu} + b_0 s^{1/\nu} + b_0}{a_n s^{n/\nu} + ... + a_1 s^{1/\nu} + a_0 s^{1/\nu} + a_0}, \quad \nu > 1. \tag{2.21}$$

The fractional-order linear time-invariant (LTI) system can also be represented by the following state-space model

$$\begin{aligned} {}_0D_t^{\mathbf{q}} &= \mathbf{A}x(t) + \mathbf{B}u(t) \\ y(t) &= \mathbf{C}x(t) \end{aligned} \tag{2.22}$$

where $x \in \mathrm{R}^n$, $u \in \mathbf{R}^r$ and $y \in \mathrm{R}^p$ are the state, input and output vectors of the system and $\mathbf{A} \in \mathrm{R}^{n \times n}$, $\mathbf{B} \in \mathrm{R}^{n \times r}$, $\mathbf{C} \in \mathrm{R}^{r \times p}$, and $\mathbf{q} = [q_1, q_2, ..., q_n]^T]$ are the fractional orders.

It is to be kept in mind is that, for the LTI expressed as the way given in (2.22), if $q_1 = q_2 = ... = q_n \equiv \alpha$, system (2.22) is called a commensurate-order system, otherwise it is an incommensurate-order system.

2.3.2 Fractional nonlinear system

The fractional nonlinear system of Caputo type can be expressed as the expression (2.23) in the box below.

Fractional nonlinear system

$$\begin{aligned} &D_t^{q_i} x_i(t) = f_i\left(x_1(t), x_2(t), ..., x_n(t), t\right) \\ &x_i(0) = c_i, i = 1, 2, ..., n. \end{aligned} \tag{2.23}$$

where $c_i, i = 1, 2, ..., n$ denote the initial conditions of different system components x_i; q_i is the fractional derivatives order for each differential equations composing the system; and f_i is the non-linear function.
The vector representation of (2.23) is :

$$D^{\mathbf{q}}\mathbf{x} = \mathbf{f}(\mathbf{x}) \tag{2.24}$$

where $\mathbf{q} = [q_1, q_2, ..., q_n]^T, i = (1, 2, ..., n)$ and $\mathbf{x} \in \mathrm{R}^n$.

The equilibra of the system (2.24) can be obtained the same way for classical integer-order systems following the equation below,

$$\mathbf{f}(\mathbf{x}) = \mathbf{0} \tag{2.25}$$

The denotation $E^* = (x_1^*, x_2^*, ..., x_n^*)$ is employed to denote the equilibrium point of the system.

For this thesis, we mainly consider the commensurate fractional-order nonlinear systems with identical fractional derivative order α in the range of (0,1). That is to say, q_i, equal to an identical value α $(i = 1, 2, ..., n)$, and $\alpha \in (0, 1)$.

2.3.3 The stability of the fractional nonlinear systems

The stability of the fractional-order nonlinear system is very complex and many definitions of stability exist (asymptotic, global, orbital, etc.). For fractional nonlinear systems, the exponential stability cannot be used to characterize asymptotic stability [Matignon, 1996]. Oustaloup et al. have introduced a new definition [Oustaloup, 2008].

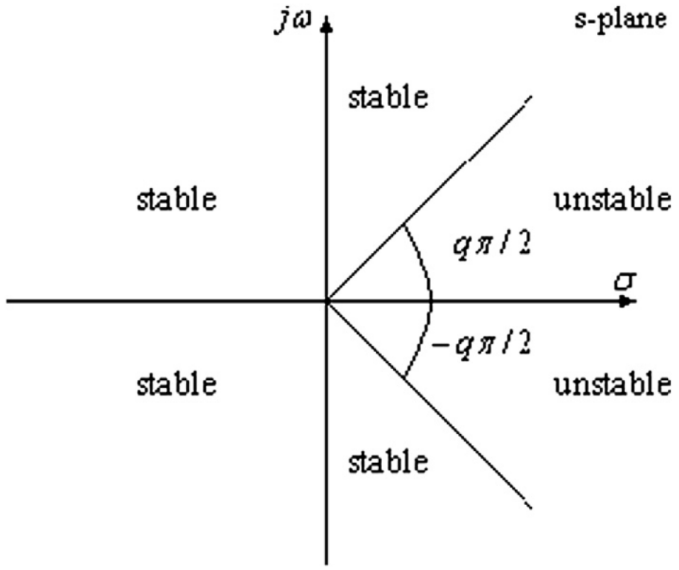


Figure 2.1 – Stability regions of the fractional-order system ([Petráš, 2011])

Definition 2.3.1. Trajactory $x(t) = 0$ of the system (2.23) is t^{-q} asymptotically stable if there is a positive real q so that

$$\forall \|x(t)| \text{ with } t \leq t_0, \exists N(x(t)), \text{such that } \forall t \geq t_0, \|x(t)| \leq Nt^{-q}. \tag{2.26}$$

The fact that the component of $x(t)$ slowly decay towards 0 following t^{-q} leads to fractional systems sometimes called long memory systems. Power law stability t^{-q} is a special case of Mittag-Leffler stability [Li, 2009].

Theorem 2.3.1. *According to stbility theorem defined in [Tavazoei, 2008a], the equilibirum points are asymptotically stable for $q_1 = q_2 = ... = q_n \equiv q$ if all the eigenvalues $\lambda_i, (i = 1, 2, ..., n)$ of the Jacobian matrix $\boldsymbol{J} = \boldsymbol{\partial f / \partial x}$, where $\boldsymbol{f} = [f_1, f_2, ..., f_n]^T$, evaluated at the equilibrium E^*, satisfy the condition [Tavazoei, 2007a]:*

$$|\arg(eig(\boldsymbol{J})| = |\arg(\lambda_i)| > q\frac{\pi}{2}, i = 1, 2, ..., n. \tag{2.27}$$

The stable and unstable region in this case is shown in Fig. 2.1.

2.4 Fractional chaotic system

As mentioned at the beginning of this section, there are systems and processes that can be modeled by fractional differential equations with non-integer order derivatives. Among them, fractional order differential systems derived from integer-order chaotic system with chaotic behavior exist. It is found that the chaotic properties of many classical and well-known integer-order chaotic systems are preserved with order smaller than three acquired with fractional derivatives [Lu, 2006].

In the following section, two three-dimensional fractional chaotic systems extended from classical integer-order chaotic systems will be introduced. But before that, we first illustrate one necessary condition that a fractional order system must meet to remain chaotic, especially for the existence of double-scroll attractor for a 3D fractional system.

2.4.1 Necessary condition to be chaotic and having double-scroll attractor

To introduce the requirement of fractional derivative order that must be satisfied for the system to be chaotic, the definition and types of saddle points are to be mentioned.

According to [Tavazoei, 2007a], for a 3D nonlinear dynamic system, a *saddle point* is an equilibrium point on which the equivalent linearized model has at least one eigenvalue in the stable region and one eigenvalue in the unstable region. In addition, the saddle point is called saddle point of index 1, if the equilibrium possesses one unstable eigenvalue, and the other eigenvalues are stable. In contrary, if only one eigenvalue is stable, the others are unstable, then the saddle point is of index 2. It has been proved that the chaotic scrolls can only be generated around the saddle point of index 2, whereas saddle point of index 1 are responsible for connecting scrolls [Chua, 1986][Silva, 1993][Cafagna, 2003][Lu, 2004].

Suppose a 3D chaotic system has three fixed point, then in order to have double scrolls, the system should have one saddle point of index 1 and two of index 2. Then the following definition is established as the necessary condition for the existence of double-scroll attractor.

Definition 2.4.1. Suppose that the unstable eigenvalues of the two scroll focus points are of the 3D fractional system : $a \pm jb$. The necessary condition to exhibit double-scroll attractor of system 2.23 is the eigenvalues λ_1, λ_2 remaining in the unstable region

[Tavazoei, 2008b]. The condition for commensurate derivatives order is,

$$q > \frac{2}{\pi} \arctan \frac{|b_i|}{a_i}, i = 1, 2. \tag{2.28}$$

According to [Tavazoei, 2007a], equation (2.28) can be used to determine analytically the minimum order q for a fractional nonlinear system to display chaotic behavior.

Hereafter, we introduce two 3-dimensional fractional-order chaotic systems, namely fractional Chen and Lu systems which are derived from classical integer-order Chen and Lu system. The systems are used for our further study of the application of fractional chaotic pseudo-random number generator design. Their system equations are given and the stability of the equilibria are analyzed analytically.

2.4.2 Fractional chaotic Chen systems investigated in our work

In 1999, Chen and Lu studied a simple 3D system that has a chaotic attractor, which can be described by the following equations [Lu, 2002],

$$\begin{array}{rcl} \frac{dx_1(t)}{dt} & = & a_c(x_2(t) - x_1(t)) \\ \frac{dx_2(t)}{dt} & = & (c_c - a_c)x_1(t) - x_1(t)x_3(t) + c_c x_2(t). \\ \frac{dx_3(t)}{dt} & = & x_1(t)x_2(t) - b_c x_3(t) \end{array} \tag{2.29}$$

In the equation, $(a_c, b_c, c_c) \in \mathbf{R}^3$ and are the parameters of the system.

The equilibrium points of this Chen system can be acquired by setting the right hand side system equation equal to zeros as follows,

$$\begin{cases} a_c\left(x_2\left(t\right) - x_1\left(t\right)\right) = 0 \\ \left(c_c - a_c\right) x_1\left(t\right) - x_1\left(t\right) x_3\left(t\right) + c_c x_2\left(t\right) = 0 \\ x_1\left(t\right) x_2\left(t\right) - b_c x_3\left(t\right) = 0 \end{cases} \tag{2.30}$$

Three equilibria of the system (2.29) are obtained by solving equation (2.30), $E_1 = (0, 0, 0)$, $E_2 = (\sqrt{b_c(2c_c - a_c)}, \sqrt{b_c(2c_c - a_c)}, 2c_c - a_c)$,
$E_3 = (-\sqrt{b_c(2c_c - a_c)}, -\sqrt{b_c(2c_c - a_c)}, 2c_c - a_c)$.

The singularity of the equilibrium points can also be acquired by evaluating the eigenvalue of the Jacobian matrix of the system at equilibrium points which takes the form as follows,

$$\mathbf{J}_c = \begin{bmatrix} -a & a & 0 \\ c - a - x_3^* & c & -x_1^* \\ x_2^* & x_1^* & -b \end{bmatrix}, \tag{2.31}$$

where $\mathbf{J}$ denotes the Jacobian matrix and the equilibrium point $E^* = (x_1^*, x_2^*, x_3^*)$

It is known that when $(a, b, c) = (35, 3, 28)$, the chaotic attractor exists. With this set of parameters, three equilibrium points are acquired $E_1 = (0, 0, 0)$,$E_2 = (-7.9373, -7.9373, 21)$, and $E_3 = (7.9373, 7.9373, 21)$ by solving equation. (2.30).

The eigenvalues of the Jacobian matrix at the equilbria are then calculated by following equation,

$$\mathbf{det}\left(\boldsymbol{\lambda}\mathbf{I} \ - \mathbf{J}_c\right) = \begin{bmatrix} \boldsymbol{\lambda} + a_c & -a_c & 0 \\ -c_l + a_l + x_{i,3}^* & \boldsymbol{\lambda} - c_c & x_{i,1}^* \\ -x_{i,2}^* & -x_{i,1}^* & \boldsymbol{\lambda} + b_c \end{bmatrix} = 0. \tag{2.32}$$

$\mathbf{J}_c$ in equation (2.32) represents the Jacobian matrix of the system equation, $\mathbf{I}$ is the identity matrix, $\boldsymbol{\lambda}$ denotes the eigenvalues, and $(x_{i,1}^*, x_{i,2}^*, x_{i,3}^*)$ stands for the equilibrium point E_i $(i = 1, 2, 3)$.

For E_1, the eigenvalues are $\lambda_1 = -3$,$\lambda_2 = 23.8359$, and $\lambda_3 = -30.8359$; for E_2, we have $\lambda_1 = -18.4380$, and an identical pair of conjugate eigenvalue $-4.2140 \pm 14.8846i$ for λ_2 and λ_3; for E_3, the three eigenvalues are $\lambda_1 = -18.4380$, $\lambda_{2,3} = -4.2140 \pm 14.8846i$. According to he above illustrated categories for equilibrium, E_1 is a saddle which is of index 1, whereas E_2 and E_3 are saddle-focus points (saddle of index 2). All of them satisfy the stability condition to keep chaotic behavior.

As mentioned before, the fractional chaotic system can be obtained by extending the classical integer order chaotic system to fractional derivative order. The fractional Chen system is therefore described as (we only considered the systems with same fractional derivative order for its system equation) [Lu, 2006],

Fractional Chen system's equation

$$\begin{cases} D^{q_1} x_1(t) = a_c\left(x_2(t) - x_1(t)\right) \\ D^{q_2} x_2(t) = \left(c_c - a_c\right) x_1(t) - x_1(t)\, x_3(t) + c_c x_2(t) \\ D^{q_3} x_3(t) = x_1(t)\, x_2(t) - b_c x_3(t) \end{cases} \tag{2.33}$$

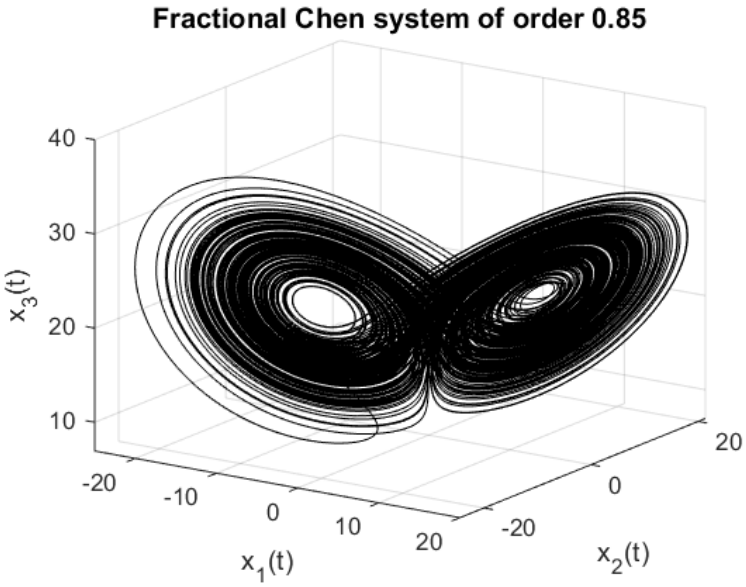


Figure 2.2 – Fractional Chen system with order $\alpha_C = 0.85$ and parameters $(a_c, b_c, c_c) = (35, 3, 28)$

In the equation (2.33), q_1, q_2 and q_3 denote the fractional derivative orders; (a_c, b_c, c_c) are the parameters of the system.

It is to be emphasized that in our work, we have only considered the commensurate fractional derivative order $q_1 = q_2 = q_3$. This commensurate fractional order for Chen system is represented by α_c in the following, where $0 < \alpha_c < 1$.

The equilibria of the system and the singularity of them are preserved for the fractional chaotic system extended from corresponding integer-order chaotic system. Therefore, when the same set of parameter $(a_c, b_c, c_c) = (35, 3, 28)$ is considered, the equilibrium points, and the singularity of the equilibrium points are the same as for the integer-order Chen system illustrated above, and are given in Table 3.1.

From the analytical point of view, according to the stability criteria and the condition introduced by equation (2.27) and (2.28), the fractional-order Chen system with given parameter values should have a fractional order $\alpha_c > 0.8244$ in order to exhibit chaotic behavior. In addition, the double scroll attractor appears around the equilibria E_2 and E_3 which are the saddle focuses.

We display here a figure of fractional Chen system attractor of order 0.85 in Fig. 2.2. The numerical calculation is done by the method given in [Petráš, 2011] which will be discussed in next chapter.

Table 2.1 – Fractional Chen system equilibria and their stability

System		**Fractional Chen system** $(a_c, b_c, c_c) = (35, 3, 28)$		
Equilibrium		(0,0,0)	(-7.9373,-7.9373,21)	(7.9373,7.9373,21)
Eigenvalue	λ_1	-30.8359	-18.4280	-18.4280
	λ_2	23.8359	4.2140+14.8846i	4.2140+14.8846i
	λ_3	-3	4.2140-14.8846i	4.2140-14.8846i
Singularity		Saddle of index 1 (Section 2.4.1)	Saddle of index 2 (Section 2.4.1)	Saddle of index 2 (Section 2.4.1)
Necessary condition to be chaotic (eq.(2.28))		$\alpha_c > 0.8244$		

2.4.3 Fractional chaotic Lu system investigated in our work

The system equation for fractional chaotic Lu system extended from integer-order Lu system can be described as follows [Deng, 2005]:

Fractional Lu system's equation:

$$f_l(x) = \begin{cases} D^{\alpha_l} x_1(t) = a_l(x_2(t) - x_1(t)) \\ D^{\alpha_l} x_2(t) = -x_1(t) x_3(t) + c_l x_2(t). \\ D^{\alpha_l} x_3(t) = x_1(t) x_2(t) - b_l x_3(t) \end{cases} \tag{2.34}$$

For system (4.17), D^{α_l} denotes the fractional derivative with commensurate order α_l; a_l, b_l, and c_l are the parameters of the system. The equilibrium of the system can be obtained easily by solving the following equations,

$$\begin{cases} a_l(x_2(t) - x_1(t)) = 0 \\ -x_1(t) x_3(t) + c_l x_2(t) = 0 \\ x_1(t) x_2(t) - b_l x_3(t) = 0 \end{cases} \tag{2.35}$$

The equilibrium points are calculated as $E_1 = (0, 0, 0)$, $E_2 = (\sqrt{b_l c_l}, \sqrt{b_l c_l}, 20)$, $E_3 = (-\sqrt{b_l c_l}, -\sqrt{b_l c_l}, 20)$. The sigularity of the equilibria can be acquired by calculating the eigenvalues of the Jacobian matrix of the system at the given equilibria.

Take the parameters of the fractional Lu system equal to $(a_l, b_l, c_l) = (36, 3, 20)$ where the integer order Lu system exhibit chaotic behavior as an example, the equilibrium points of the system is calculated as $E_1 = (0, 0, 0)$, $E_2 = (7.460, 7.460, 20)$ and $E_3 = (-7.460, -7.460, 20)$ through the above given symbolic expression.

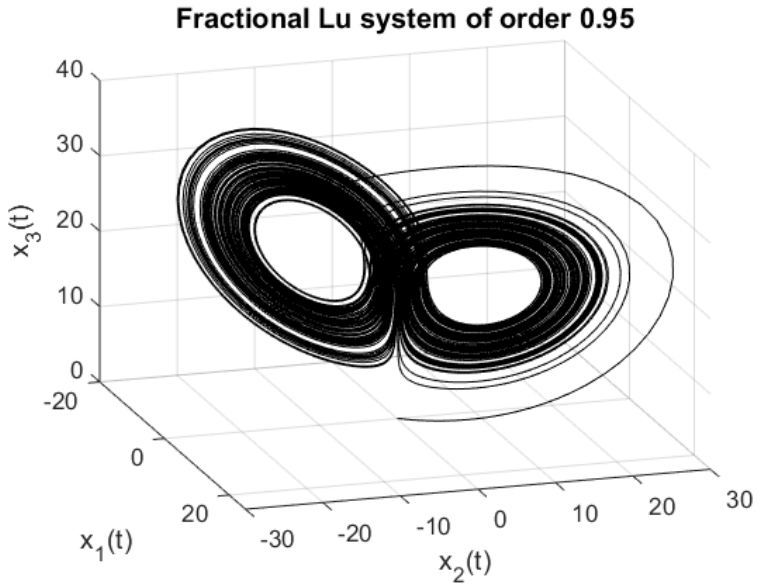


Figure 2.3 – Fractional Lu system with order $\alpha_l = 0.95$ and parameters $(a_l, b_l, c_l) = (36, 3, 20)$

The Jacobian matrix $\mathbf{J}_l$ of the fractional Lu system with given parameters at the equilibrium $E_i, (i = 1, 2, 3)$ can be expressed as,

$$\mathbf{J}_l = \begin{bmatrix} -36 & 36 & 0 \\ -x^*_{i,3} & 20 & -x^*_{i,1} \\ x^*_{i,2} & x^*_{i,1} & -3 \end{bmatrix} \tag{2.36}$$

For the equilibrium $E_1 = (0, 0, 0)$, the eigenvalues are (-36,20,-30). For the other two equilibria E_2 and E_3, their characteristic equation for eigenvalues calculation can be expressed as,

$$\lambda^3 + 19\lambda^2 + 108\lambda + 4320 = 0 \tag{2.37}$$

By solving equation (2.37), we get three roots $\lambda_1 = -22.6515$, $\lambda_{2,3} = 1.8258 \pm 13.6887i$, and $|\arg| = 7.4974$. Since there are one real eigenvalue with positive sign and two conjugate eigenvalues with opposite sign to the real one, both equilibria E_2 and E_3 are saddle focuses (saddle point of index 2) and are surrounded by chaotic double scroll attractor (shown in Fig. 2.3).

According to the relation given by equation (2.28), the necessary condition for the system to be chaotic is to have a fractional order $\alpha_l > 0.9156$ (for which we observe a double scroll). The singularities of the equilibrium points are given in Table. 2.2.

Table 2.2 – Fractional Lu system equilibria and their stability

System		**Fractional Lu system** $(a_l, b_l, c_l) = (36, 3, 20)$		
Equilibrium		(0,0,0)	(-7.746,-7.746,20)	(7.746,7.746,20)
Eigenvalue	λ_1	-36	-22.6516	-22.6516
	λ_2	20	1.8258+13.6887i	1.8258+13.6887i
	λ_3	-3	1.8258-13.6887i	1.8258-13.6887i
Singularity		Saddle of index 1 (Section 2.4.1)	Saddle of index 2 (Section 2.4.1)	Saddle of index 2 (Section 2.4.1)
Necessary condition to be chaotic (eq.(2.28))		$\alpha_l > 0.9156$		

2.5 Conclusion

In this Chapter, we have introduced some terminology and preliminaries of the fractional calculus and fractional dynamic systems. Since our aim is to investigate into the application of fractional nonlinear dynamic systems with chaotic behavior, we also discuss the stability of fractional nonlinear system and the necessary condition for the fractional system to be chaotic. Two fractional 3D chaotic systems which are employed for our further study of the pseudo-random number generator design application are given. Their stability with given parameters are discussed from the analytical analysis point of view.

Chapter 3

Numerical calculation methods for fractional systems

3.1 Introduction

To solve the fractional systems and implement them into engineering problems, the approximation or the numerical solutions of the systems must be acquired. Globally speaking, one can either to solve the systems applying frequency domain method or time domain method. However, for the implementation of fractional chaotic systems, it is believed that the frequency domain approximation methods are not always reliable in detecting chaos in nonlinear dynamics [Tavazoei, 2008b][Tavazoei, 2007b]. The time domain approximation methods on the other hand, though are sometimes more complicated and time consuming due to the long memory characteristics of fractional derivatives, can provide more reliable results.

In the following sections of this chapter, different numerical time domain calculation approaches for both one dimensional fractional system and multi-dimensional fractional systems are introduced. One fractional generalized doubled-humped logistic system (FGDHL) which we employed for our FCPRNG design is discussed using the calculation method illustrated. The numerical solutions of fractional Chen and Lu systems is also calculated applying two different numerical approaches. The results and comparison among the two methods are analyzed.

3.2 Numerical calculation method of one dimensional fractional chaotic system

In this section, we explain one time domain approximation approach for one dimensional fractional system. The approach is based on piecewise-constant approximation

method. One fractional chaotic system, namely generalized double-humped logistic system (FGDHL) is employed to illustrate the discretization process of the method. The approximated system are analysed analytically and the simulation results are discussed.

3.2.1 Piecewise-constant approximation for one-dimensional fractional system

The one dimensional fractional chaotic system we employed to explain the dicretization process is fractional generalized double-humped logistic system (FGDHL) which is inspired by the integer-order generalized double-humped logistic map.

For the classical integer-order, the double humped logistic map holds the form as below,

$$x_{n+1} = \rho(x_n - 1)^2(1 - (x_n - 1)^2) \tag{3.1}$$

where ρ is the growth rate and also the sole control parameter of the map. The generalized double-humped logistic map is a generalized version of (3.1) which has been discussed in [Ismail, 2018] and has the following form,

$$x_{n+1} = \rho(x_n - c)^2(c - (x_n - c)^2) \tag{3.2}$$

where ρ and c are the control parameters of the chaotic map.

Inspired by map (3.2) , the fractional differential equation for the FGDHL can be expressed as follows,

$$D^{\alpha_g}x(t) = \rho(x(t) - c)^2(c^2 - (x(t) - c)^2), \quad t > 0 \tag{3.3}$$

In equation (3.3), α_g stands for the fractional derivative order between $(0, 1)$, ρ and c denote the parameters, $x(0)$ is the initial condition.

With the introduction of piecewise constant arguments, the corresponding FGDHL system equation can be rewritten as,

$$D^{\alpha_g}x(t) = \rho\Big(x\left(\left[\frac{t}{r}\right]r\right) - c\Big)^2\left(c^2 - \left(x\left(\left[\frac{t}{r}\right]r\right) - c\right)^2\right) \tag{3.4}$$

with $x(0) = x_0$ its initial condition.

The discretization process has been discussed in [El Raheem, 2014], the author used the method to discretize the fractional logistic differential equation. One property of

fractional differentiation and integration is recalled,

$$D^{\alpha} f(t) = I^{n-\alpha} D^{n} f(t), D = \frac{D}{dt}, \tag{3.5}$$

where n is the greatest integer smaller than α, i.e. $\alpha - 1 < n < \alpha$.

Discretization process

a. Let $t \in [0, r)$, the $\frac{t}{r} \in [0, 1)$, the equation is transformed into

$$D^{\alpha_g} x(t) = \rho(x_0 - c)^2 \left(c^2 - (x_0 - c)^2\right), t \in [0, r) \tag{3.6}$$

Employing the property given in equation (3.5) and fractional integration given in Section 2.2.3, the solution can be derived as,

$$\begin{aligned} x_1(t) &= x_0 + I_r^{\alpha_g} \rho(x_0 - c)^2 \left(c^2 - (x_0 - c)^2\right) \\ &= x_0 + \rho(x_0 - c)^2 \left(c^2 - (x_0 - c)^2\right) \int_0^t \frac{(t-s)^{\alpha_g - 1}}{\Gamma(\alpha_g)} ds \\ &= x_0 + \rho(x_0 - c)^2 \left(c^2 - (x_0 - c)^2\right) \frac{t^{\alpha_g}}{\Gamma(\alpha_g + 1)} \end{aligned} \tag{3.7}$$

b. Let $t \in [r, 2r)$, then $\frac{t}{r} \in [1, 2)$, we get,

$$D^{\alpha_g} x(t) = \rho(x_1 - c)^2 \left(c^2 - (x_1 - c)^2\right), t \in [r, 2r) \tag{3.8}$$

The corresponding solution is,

$$\begin{aligned} x_2(t) &= x_1(r) + I_r^{\alpha_g} \rho(x_1 - c)^2 (c^2 - (x_1 - c)^2) \\ &= x_1(r) + \rho(x_1 - c)^2 (c^2 - (x_1 - c)^2) \int_r^t \frac{(t-s)^{\alpha_g - 1}}{\Gamma(\alpha_g)} ds \\ &= x_1(r) + \rho(x_1(r) - c)^2 (c^2 - (x_1(r) - c)^2) \frac{(t-r)^{\alpha_g}}{\Gamma(\alpha_g + 1)} \end{aligned} \tag{3.9}$$

c. Repeating the above process, one can deduce that the solution for $t \in [nr, (n+1)r)$,

$$\begin{aligned} x_{n+1}(t) &= x_n(nr) + \rho(x_n(rn) - c)^2(c^2 - (x_n(nr) - c)^2) \int_{nr}^{t} \frac{(t-s)^{\alpha_g - 1}}{\Gamma(\alpha_g)} ds \\ &= x_n(nr) + f(x_n(nr)) \frac{(t-nr)^{\alpha_g}}{\Gamma(\alpha_g + 1)} \end{aligned} \tag{3.10}$$

d. Let $t \to (n+1)r$, we have,

$$x_{n+1}((n+1)r) = x_n(nr) + \frac{r_{\alpha_g}}{\Gamma(1+\alpha_g)} \rho(x_n(nr) - c)^2(c^2 - (x_n(nr) - c^2))^2 \tag{3.11}$$

Then, the following formula below for the discretization is finally derived,

$$x_{n+1} = x_n + \frac{r^{\alpha_g}}{\Gamma(1+\alpha_g)} \rho(x_n - c)^2(c^2 - (x_n - c)^2) \tag{3.12}$$

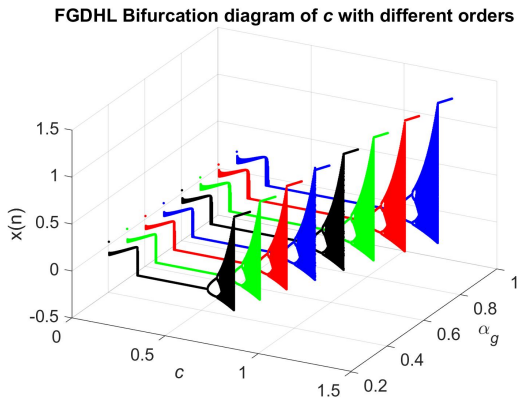
3.2.2 Fractional generalized double-humped logistic system numerical results

We notice that, with different r values, the solutions of the system can be very different. So, for the sake of simplification and consistency, in the following discussion, we set $r = 0.2$.

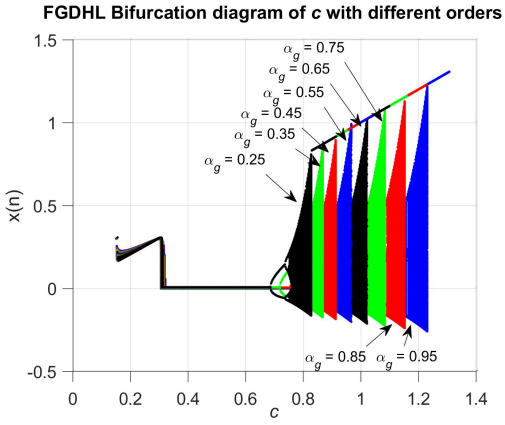
To briefly discuss the chaotic property of the proposed FGDHL system, the bifurcation diagrams, Lyapunov exponent results, and histogram of the states of the system are given and analyzed from the experimental simulation point of view.

The effect of the control parameter c through bifurcation diagram for different fractional orders α_g from 0.25 to 0.95, while ρ = -4.3 is shown in Fig. 3.1a and 3.1b . It can be seen from the figure that with higher fractional order ($0 < \alpha_g < 1$), a greater c value is needed for the system to exhibit chaotic behavior. Besides, the vertical scale of $x(n)$ is proportional to the value of parameter c. That is to say, with the increase of c, the system states fall into a wider range of values.

The bifurcation diagrams for the parameter ρ over different fractional orders are given in Fig.3.1a and 3.1b. The parameter c is set to 0.9. It is observable that the range for the system state remains approximately the same. In terms of the chaotic behavior, for fractional orders from 0.25 to 0.95, the bifurcation point for the parameter ρ shifted leftwards with the increase of the fractional order value. That is to say that for the

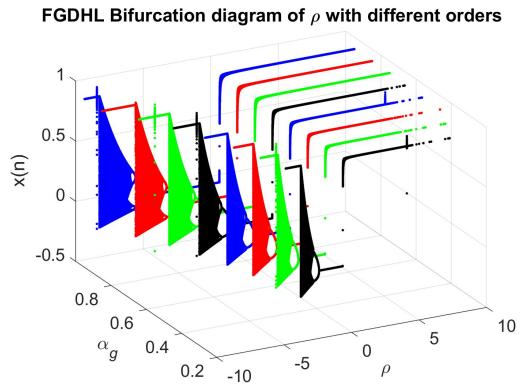


(a) Bifurcation diagram $\rho = -4.3$ (3D)

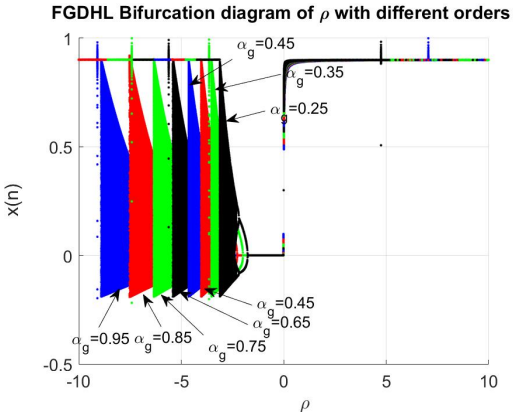


(b) Bifurcation diagram $\rho = -4.3$

Figure 3.1 – FGDHL bifurcation diagram parameter c v.s fractional derivative α_g while $\rho = -4.3$

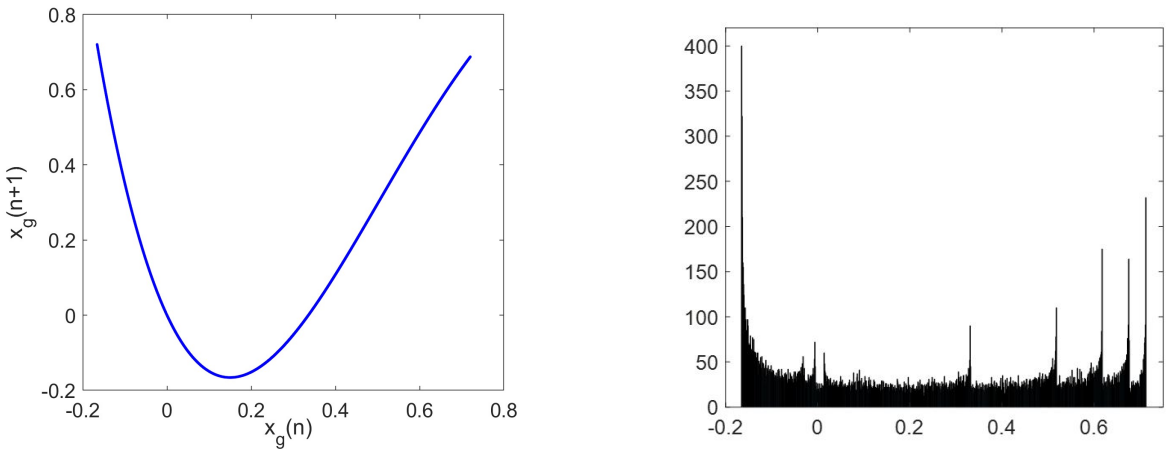


(a) Bifurcation diagram $c = 0.9$ (3D)



(b) Bifurcation diagram $c = 0.9$

Figure 3.2 – FGDHL bifurcation diagram parameter ρ v.s fractional derivative α_g while $c = 0.9$



(a) Phase space FGDHL $\alpha_g = 0.85$ (b) Histogram of FGDHL states

Figure 3.3 – Histogram of FGDHL $\rho = -10.3$, $\alpha_g = 0.85$, $c = 0.85$, $x(0) = 0.7$

system to exhibit chaotic behavior, a smaller ρ value is required with the increase of the non-integer order.

By setting the initial condition $x(0)$ to 0.7, the fractional order α_g, control parameter c and ρ as 0.85, 0.85 and -10.3. The phase delay and histogram of 31250 states are obtained and shown in Fig.3.2a and 3.2b, respectively.

3.3 Numerical calculation methods for fractional-order multi-dimensional systems

In the following, two different time domain calculation approaches are to be discussed. The first one is based on Grünwald-Letnikov characterization, and second is the fractional Adam-Bashforth-Moulton corrector-predictor method based on Caputo type fractional derivative.

3.3.1 Numerical calculation method based on Grünwald-Letnikov characterisation

The explicit numerical approximation of α-th derivative under GL characterisation at the points $kh, (h = 1, 2, ...)$ is expressed as follows [Podlubny, 1999]

$$_{(k-L_m)/h}D^{\alpha}_{t_k} f(t) \approx h^{-\alpha} \sum_{j=0}^{k} (-1)^j \begin{pmatrix} \alpha \\ j \end{pmatrix} f(t_{k-j}). \tag{3.13}$$

In expression (3.13), L_m is the memory length; $t_k = kh$, where h is the calculation time step; the binomial coefficient $(-1)^j \binom{\alpha}{j}$ can be denoted as $c_j^{(\alpha)} (j = 0, 1, ...)$ which is expressed use the following expression [Dorcak, 1994],

$$c_0^{(\alpha)} = 1, c_j^{(\alpha)} = \left(1 - \frac{1+\alpha}{j}\right) c_{j-1}^{(\alpha)}. \tag{3.14}$$

Thus, the general numerical solution of fractional differential equation described by equation(3.15) can be expressed as given in (3.16).

$$_aD_t^{\alpha} y(t) = f(y(t), t) \tag{3.15}$$

$$y(t_k) = f(y(t_k), t_k)h^{\alpha} - \sum_{j=\nu}^{k} c_j^{(\alpha)} y(t_{k-j})) \tag{3.16}$$

The sum in (3.16) stands for the memory term. If a 'long memory effect' is considered, then the lower index $\nu = 1$ for all k, otherwise $\nu = 1$ for $k < (L_m/h)$ and $\nu = k - L_m$ for $k > (L_m/h)$.

3.3.2 Fractional Adams-Bashforth-Moulton corrector predictor method

The fractional Adams-Bashforth-Moulton (ABM) corrector predictor method is derived based on integer-order corrector-predictor calculation method for differential equations. The method is first discussed in [Diethelm, 2002].

The fractional differential equation considered is expressed as follows,

$$D_*^{\alpha} y(x) = f(x, y(x)) \tag{3.17}$$

and the initial condition of the equation is given by

$$y^k(0) = y_0^k, k = 0, 1, 2...m - 1. \tag{3.18}$$

where $m := \lceil \alpha \rceil$.

Note that the Caputo charaterization is used to derive the algorithm for this numerical methods.

The reason for the chosen charaterization lies in the fact that the additional conditions that need to be specified for the equation have a well understood physical meaning and can be measured, thus it is more flexible and easier to implement.

From the analytical property point of the view, the initial fractional differential problem is equivalent to Volterra integral equation which holds the following form,

$$y(x) = \sum_{k=0}^{\lceil \alpha \rceil - 1} y_0^{(k)} \frac{x^k}{k!} + \frac{1}{\Gamma}(\alpha) \int_0^x (x-t)^{\alpha-1} f(t, y(t)) dt \tag{3.19}$$

Recall on the classical Adams-Bashforth-Moulton algorithm

The agorithm is a generatlization of the classical Adams-Bashforth-Moulton integrator that is well known for the numerical solution of first-order problems.

$$Dy(x) = f(x, y(x)), y(0) = y(0) \tag{3.20}$$

The function f is assumed to be such that a unique solution exists on some interval $[o, T]$. To simplify the problem, a uniform grid $\{t_n = nh : n = 0, 1, ...N\}$ is chosen to develope the algorithm. The basic idea of the algorithm is to obtain the approximation of the latter point on the grid from the previous point, the equation is shown below:

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(z, y(z)) dz \tag{3.21}$$

Assuming the approximation of $y(t_n)$ is already calculated and expressed by $y_h(t_n)$, then with the integral replaced by the two-point trapezodial quadrature formular

$$\int_a^b g(z) dz \approx \frac{b-a}{2}((g(a) + g(b)) \tag{3.22}$$

This gives the following expression,

$$y_h(t_{n+1}) = y_h(t_n) + \frac{h}{2}[f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))] \tag{3.23}$$

While substituting $y(t_n)$ and $y(t_{n+1})$ with their approximations, the implicit one-step Adams-Moulton method is yield.

$$y_h(t_{n+1}) = y_h(t_n) + \frac{h}{2}[f(t_n, y_h(t_n)) + f(t_{n+1}, y_h(t_{n+1}))] \tag{3.24}$$

Since the unknown approximation $y_h(t_{n+1})$ appears in both sides of the equation. The so-called predictor, a preliminary approximation $y_h^p(t_{n+1})$ is added to the equation. The predictor is obtained with the help of rectangle rule,

$$\int_a^b g(z) \approx (b-a)g(a) \tag{3.25}$$

which leads to the formula below,

$$y_h^P(t_{n+1}) = y_h(t_n) + hf(t_n, y_h(t_n)) \tag{3.26}$$

This formula is also known as forward Euler or one-step Adams Bashforth method. Then, the one-step Adams-Bashforth-Moulton technique is deduced and holds the following form,

$$y_h(t_{n+1}) = y_h(t_n) + \frac{h}{2}[f(t_n, y_h(t_n)) + f(t_{n+1}, y_h^P(t_{n+1})] \tag{3.27}$$

Fractional order predictor-corrector approach

With the concept introduced, the modification will be made to the problem in order to develop the suitable formula for the fractional order problem. To obtain the modified equation, the product trapezoidal quadrature formula as given below is used in replacement of the integral in equation (3.19).

$$\int_0^{t_{n+1}} (t_{n+1} - z)^{\alpha-1} g(z)dz \approx \int_0^{t_{n+1}} (t_{n+1} - z)^{\alpha-1} \widetilde{g}_{n+1}(z)dz, \tag{3.28}$$

where $\widetilde{g}_{n+1}$ is the piecewise linear interpolant for g with nodes and knots chosen at the $t_j, j = 1, 2, ...n+1$. The standard techniques from quadrature theory is used to rewrite

the right-hand side of equation (3.28).

$$\int_0^{t_{n+1}} (t_{n+1} - z)^{\alpha-1} \widetilde{g}_{n+1}(z)dz = \frac{h^\alpha}{\alpha(\alpha+1)} \sum_{j=0}^{n+1} a_{j,n+1} g(t_j) \tag{3.29}$$

where

$$a_{j,n+1} = \begin{cases} n^{\alpha+1} - (n-\alpha)(n+1)^\alpha, & \text{if } j = 0, \\ (n-j+2)^{\alpha+1} + (n-j)^{\alpha+1} - 2(n-j+1)^{\alpha+1}, & \text{if } 1 \le j \le n, \\ 1, & \text{if } j = n+1. \end{cases} \tag{3.30}$$

With the equation above, the fractional variant of the one-step Adams-Moulton method is given as follows,

$$\begin{aligned} y_h(t_{n+1}) = &\sum_{k=0}^{\lceil\alpha\rceil-1} \frac{t_{n+1}^k}{k!} y_0^{(k)} + \frac{h^\alpha}{\Gamma(\alpha+2)} f(t_{n+1}, y_h^P(t_{n+1})) \\ &+ \frac{h^\alpha}{\Gamma(\alpha+2)} \sum_{j=0}^{n} a_{j,n+1} f(t_j, y_h(t_j)), \end{aligned} \tag{3.31}$$

Applying the same method, replacing the integral by the product rectangle rule with the following expression,

$$\int_0^{t_{n+1}} (t_{n+1} - z)^{\alpha-1} g(z)dz \approx \sum_{j=0}^{n} b_{j,n+1} g(t_j) \tag{3.32}$$

where

$$b_{j,l+1} = \frac{h^\alpha}{\alpha}((n+1-j)^\alpha - (n-j)^\alpha) \tag{3.33}$$

gives the predicted value $y_h^P(t_{n+1})$ determined by the fractional Adams-Bashforth method,

$$y_h^P(t_{n+1}) = \sum_{k=0}^{\lceil\alpha\rceil-1} \frac{t_{n+1}^k}{k!} y_0^{(k)} + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{n} b_{j,n+1} f(t_j, y_h(t_j)). \tag{3.34}$$

Then the fractional Adams-Bashforth-Moulton basic algorithm is fully defined with the weights $a_{j,n+1}$ and $b_{j,n+1}$.

Note that using above approach, the Fractional Differential Equations (FDEs) can be

reduced to Volterra type integral equations. Therefore, the numerical scheme for Volterra type integral equations can be applied to find the numerical solution of FDEs which is a great advantage [Wang, 2013].

It is obvious that for the numerical calculation of fractional differential equations, unlike integer differential equations, the expression involves all the previous calculated values. And this leads to great computational cost. To reduce the arithmetic complexity, the fixed memory principle of Podlubny [Podlubny, 1999] can be used. And in [Deng, 2007], the author combined the short memory principle with the predictor-corrector approach, and obtain an efficient reduction of the computation complexity. In [Ford, 2001], a nested memory concept is also brought up which can be applied to the predictor-corrector.

3.4 3D fractional chaotic systems numerical simulation results

3.4.1 Fractional chaotic systems applying calculation method based on Grünwald-Letnikov characterisation

With the numerical solution of fractional differential equation calculated under Grünwald-Letnikov (GL) method derived given by equation (3.16), the calculation for the states of fractional Chen system and fractional Lu system (systems (2.33) and (4.17)) can be expressed by the following identities (3.35) and (3.36), respectively.

$$\begin{cases} x_1(n) = (a_c(x_2(n) - x_1(n-1)))h^{\alpha_c} - \sum\limits_{j=\nu}^{n} c_j^{(\alpha_c)} x_1(n-j) \\ x_2(n) = ((c_c - a_c)x_1(n) - x_1(n)x_3(n-1) + c_c x_3(n-1))h^{\alpha_c} - \sum\limits_{j=\nu}^{n} c_j^{(\alpha_c)} x_2(n-j) \\ x_3(n) = (x_1(n)x_2(n) - b_c x_3(n-1))h^{\alpha_c} - \sum\limits_{j=\nu}^{n} c_j^{(\alpha_c)} x_3(n-j) \end{cases} \tag{3.35}$$

$$\begin{cases} x_1(n) = (a_l(x_2(n-1) - x_1(n-1)))h^{\alpha_l} - \sum\limits_{j=\nu}^{n} c_j^{(\alpha_l)} x_1(n-j) \\ x_2(n) = (-x_1(n)x_3(n-1) + c_l x_2(n-1))h^{\alpha_l} - \sum\limits_{j=\nu}^{n} c_j^{(\alpha_l)} x_2(n-j) \\ x_3(n) = (x_1(n)x_2(n) - b_l x_3(n-1))h^{\alpha_l} - \sum\limits_{j=\nu}^{n} c_j^{(\alpha_l)} x_3(n-j) \end{cases} \tag{3.36}$$

To be mentioned is that in our work, the 'long memory effect' is adopted applying GL method which means that the number ν in equaitons (3.35) and (3.36) is equal to 1. The

System	Equilibrium	Eigenvalue		singularity
		λ_1	λ_2, λ_3	
Fractional	(0, 0, 0)	-30.8359	23.8359,-3	Saddle
Chen	(-7.9379,-7.9379,21)	-18.4280	4.2140±14.8846i	Saddle Focus
system	(7.9379,7.9379,21)	-18.4280	4.2140±14.8846i	Saddle Focus
Fractional	$(0, 0, 0)$	-36	$20, -3$	Saddle
L$\ddot{u}$	$(-7.460, -7.460, 20)$	-22.6516	$1.8258 \pm 13.6887i$	Saddle Focus
system	$(7.460, 7.460, 20)$	-18.4280	$1.8258 \pm 13.6887i$	Saddle Focus

Table 3.1 – Fractional Chen and Lu systems' equilibria and their singularity

time step h in above equations is set to a fixed value 0.001.
We plotted the phase portraits of the two systems with fractional orders $\alpha_c = 0.9$ and $\alpha_l = 0.95$ in Fig.3.4a and 3.4b, respectively. The parameters and initial conditions for Chen system are $(35, 3, 28)$ and $(-9, -5, 14)$. Those of Lu system are chosen to be $(36, 3, 20)$ and $(0.2, 0.5, 0.3)$.

3.4.2 Fractional chaotic systems applying ABM corrector-predictor approach

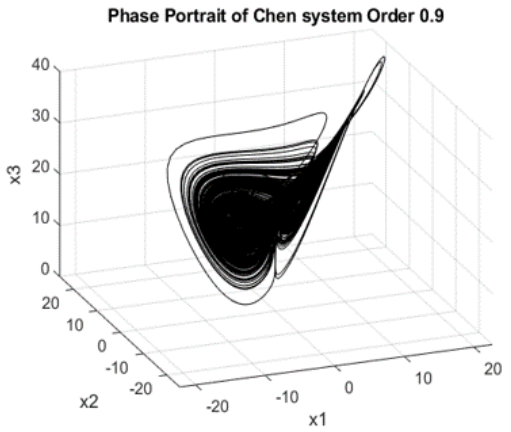
Based on the fractional ABM corrector-predictor numerical calculation approach for the solution of fractional differential equations given in equations (3.30)-(3.34), the states of fractional Chen system applying ABM predictor corrector approach can be expressed as follows,

$$\begin{aligned} \mathrm{X}_c(n+1) = \mathrm{X}_c(0) &+ \frac{h^{\alpha_c}}{\Gamma(\alpha_c+2)} f_c(\mathrm{X}_c^P(n+1)) \\ &+ \frac{h^{\alpha_c}}{\Gamma(\alpha_c+2)} \sum_{j=0}^{n} a_{j,n+1} f_c(\mathrm{X}_c(j)) \end{aligned} \tag{3.37}$$

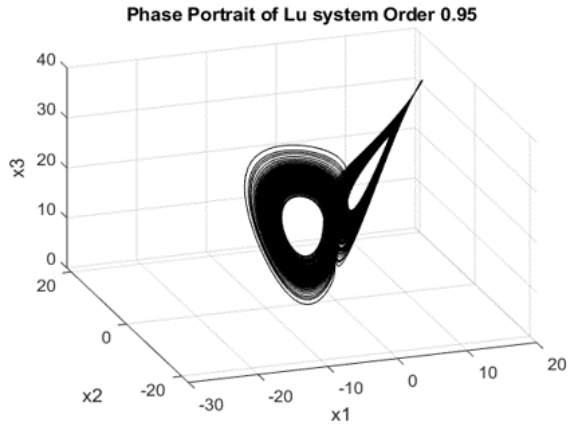
$$\mathrm{X}_c^P(n+1) = \mathrm{X}_c(0) + \frac{1}{\Gamma(\alpha_c)} \sum_{j=0}^{n} b_{j,n+1}^{1} f_c(\mathrm{X}_c(j))$$

$$a_{j,n+1} = \begin{cases} n^{\alpha_c+1} - (n-\alpha_c)(n+1)^{\alpha_c}, & \text{if} j = 0, \\ (n-j+2)^{\alpha_c+1} + (n-j)^{\alpha_c+1} - 2(n-j+1)^{\alpha_c+1}, & \text{if} 1 \le j \le n, \\ 1, & \text{if} j = n+1. \end{cases} \tag{3.38}$$

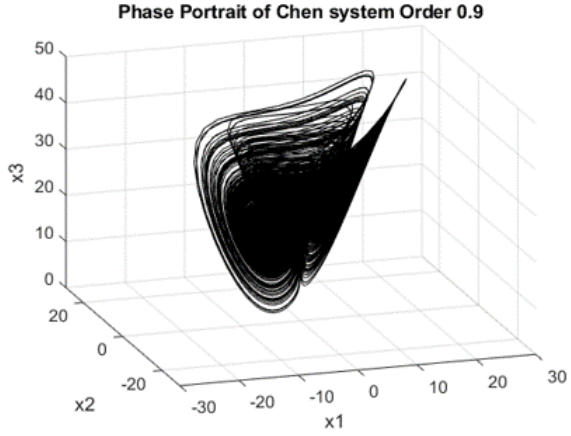
$$b_{j,l+1} = \frac{h^{\alpha_c}}{\alpha_c}((n+1-j)^{\alpha_c} - (n-j)^{\alpha_c})$$



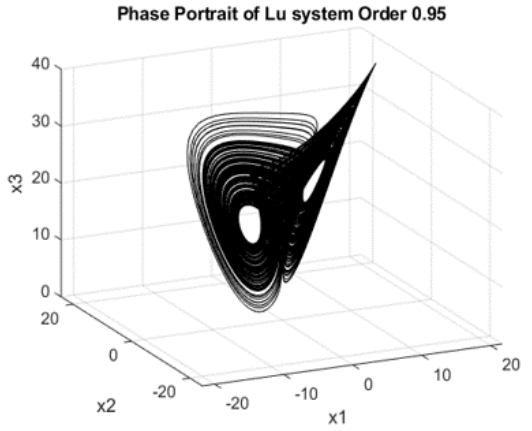
(a) GL method Chen system ($\alpha_c = 0.9$)



(b) GL method Lu system ($\alpha_l = 0.95$)



(c) ABM method Chen system ($\alpha_c = 0.9$)



(d) ABM method Lu system ($\alpha_l = 0.95$)

Figure 3.4 – Phase Portrait of fractional Chen and Lu systems characterized by GL and ABM method

In the above expressions, $\mathrm{X}_c(n+1)$, $\mathrm{X}_c(n)$ and $\mathrm{X}_c^P(n+1)$ are state vectors composed of all the state components x_1, x_2, and x_3; α_c is the fractional order between $(0, 1)$; f_c stands for the Chen system equations.

The formula for the calculation of the states of fractional Lu system can be obtained by substituting the state vectors, fractional order and system equations in equations (3.37)-(3.38) with X_l, α_l and f_l where $0 < \alpha_l < 1$. The phase portraits of the two systems acquired employing ABM corrector-predictor approach are given in Fig.3.4c and 3.4d, respectively. The fractional orders, parameters and initial conditions are the same as those for the GL method.

3.4.3 Impact on system chaoticity

For the work in this section, we used the same parameters and initial conditions for the two systems as adopted in the previous section, which are $(a_c, b_c, c_c) = (35, 3, 28)$, $\mathrm{X}_c(0) = (-9, -5, 14)$ for Chen system; $(a_l, b_l, c_l) = (36, 3, 20)$, $\mathrm{X}_l(0) = (0.2, 0.5, 0.3)$ for Lu system, respectively. The time step h is set to 0.005. The MATLAB code [Garrappa Roberto, 2021] for ABM corrector-predictor method and [Danca, 2018] is employed for the following simulation and the calculation of LE.

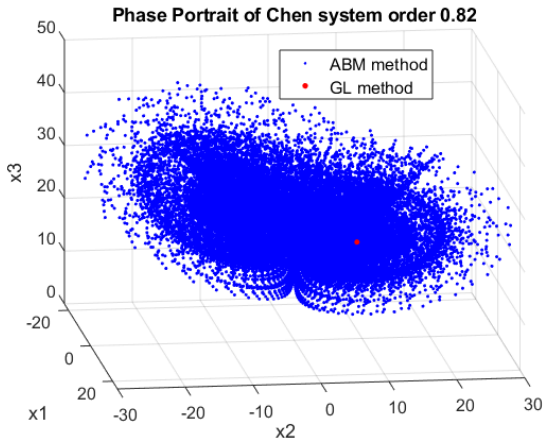
In the previous Chapter (Chapter 2), we have discussed the necessary conditions for the fractional system to be chaotic. The minimum fractional derivative order for both fractional Chen and Lu system to exhibit chaotic behavior, especially to have double-scroll attractor has been computed. We recall here the minimum derivative order with given parameters. For fractional Chen system,the minimum order is approximately 0.8244. For fractional Lu system, this order is 0.9156.

In Fig. 3.5, we plot the phase portrait of fractional Chen system at boundary fractional values 0.82 and 0.83 applying both GL and ABM corrector-predictor methods. The time response of the last 2000 iterated states obtained through both methods are also given. The states calculated by GL method are in red and ABM corrector-predictor in blue.

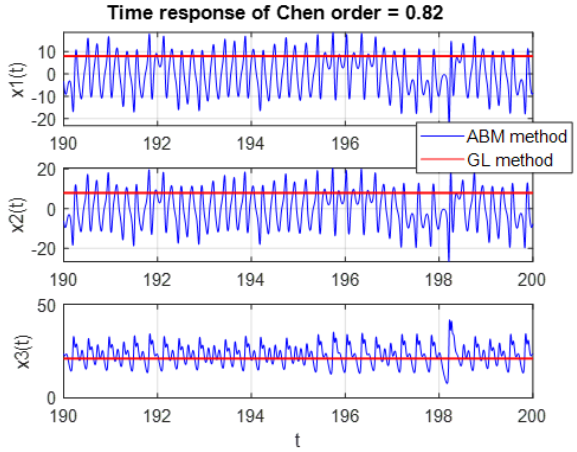
It is not difficult to observe from Fig. 3.5a and 3.5c that with order 0.82 there are only one red point in the figure, which suggests that the states of the systems remain at the same fixed point applying GL method. Whereas for the applied ABM method(blue dots), they appear to have a shape of the attractors. When the system order is equal to 0.83, both methods display the attractors shape. This indicates that when applying GL calculation method with long memory effect, the system's dynamic behavior is in accordance with the stability criteria given by equation (2.28). While the ABM calculation method applied allows the system to have a smaller derivative order for the system to be chaotic.

The time responses given by Fig. 3.5b and 3.5d confirms the finding. The blue curve stands for the states obtained through ABM method and red for GL. It is clear that for a derivative order 0.82, the red lines stays at the same value for the three state vector components x_1, x_2 and x_3, while the blue curves appear to be oscillating.

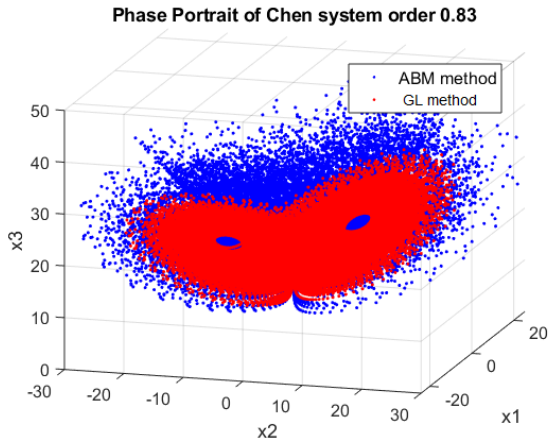
We also give the Lyapunov exponent and bifurcation diagrams over different fractional orders of fractional Chen and Lu systems in Fig.3.6. For each fractional derivative orders, 10^4 states were generated and the LEs were calculated throughout the iterations. The LE spectrum curves in 3.6a and 3.6b are obtained by combining the LE values of the last iterations for each evaluated derivative order.



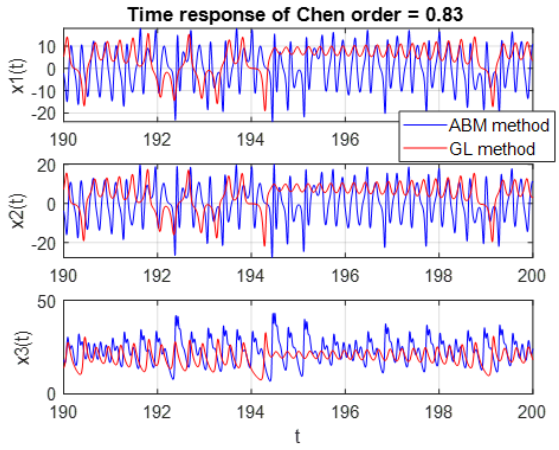
(a) Chen system with order 0.82



(b) Time response with order 0.9



(c) Chen system with order 0.83



(d) Time response with order 0.83

Figure 3.5 – Phase Portrait and time response of Chen system at boundary fractional order values

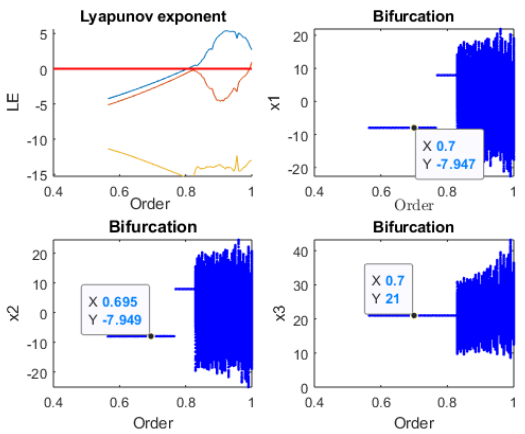
The plots show that only x_1 component possesses LE value greater than 0, when applying both methods. It can be observed that when applying ABM corrector-predictor approach, for the fractional Chen system, the LE of x_1 greater than 0 appears before the fractional order $\alpha_c = 0.53$, whereas for GL method, the LE exceeds 0 after fractional order of 0.8.

The LEs for fractional Lu system calculated by both methods show similar results, with ABM method having a smaller chaotic fractional derivative value. This is in accordance with our previous findings concerning the phase portrait and time responses which draws to the conclusion that GL method give a more accurate approximation of the original fractional system. Apart from this, from the y-coordinates of the bifurcation diagram where the system is non-chaotic, it can be observed that the solution obtained using ABM method stays at the equilibrium point, as obtained through the analytical study.

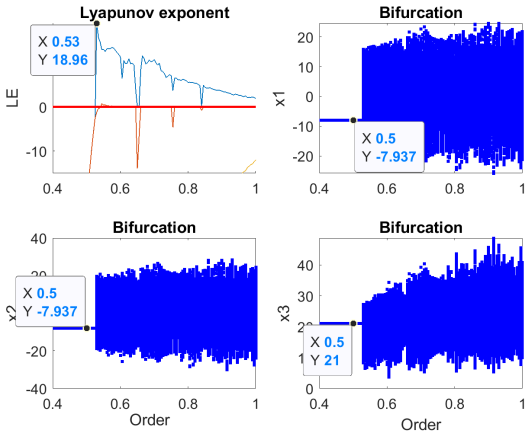
The LEs results and bifurcation diagram over different parameters of fractional Lu system are also given in Fig.3.7 to illustrate the dynamics of the system. We set the system fractional order to 0.9. It can be observed that applying different numerical calculation methods, the system dynamics is quite different. It is worth mentioning that the results for different parameters are conducted by changing one parameter at a time and fixing the other two unchanged.

3.5 Conclusion

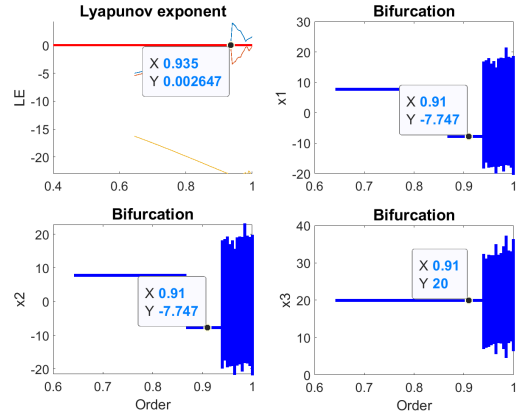
In this section, we first explained the piecewise argument calculation method for one-dimensional fractional map by employing the fractional generalized double humped logistic map. The solutions of the map has been calculated through the illustrated method. The chaotic behavior of FGDHL is discussed both from analytical point of view and from the numerical calculation. Then, two time domain numerical calculation methods based on both Grünwald-Letnikov and Caputo fractional derivative characterizations were discussed. We also employed two 3D fractional chaotic systems and applyed both calculation methods to the systems. The systems characteristics and the impacts on their chaotic behavior when applying different approaches have been discussed.



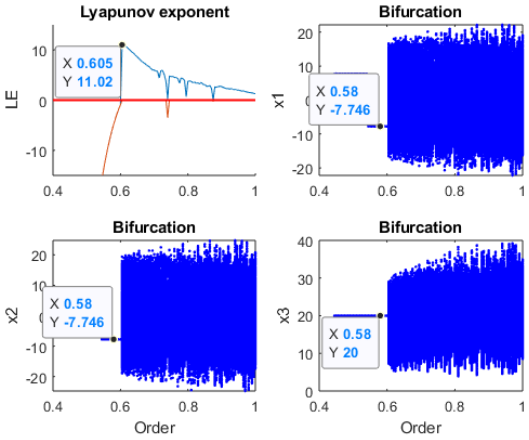
(a) Chen system GL method



(b) Chen system ABM method

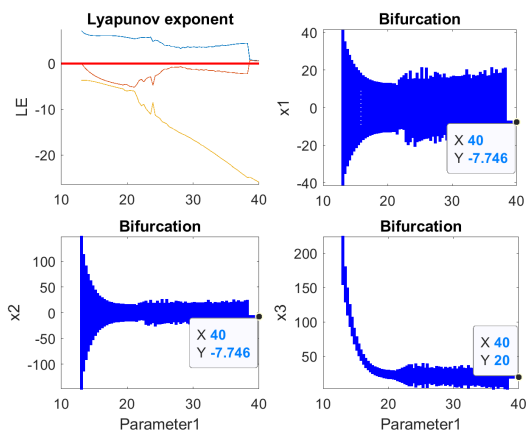


(c) Lu system GL method

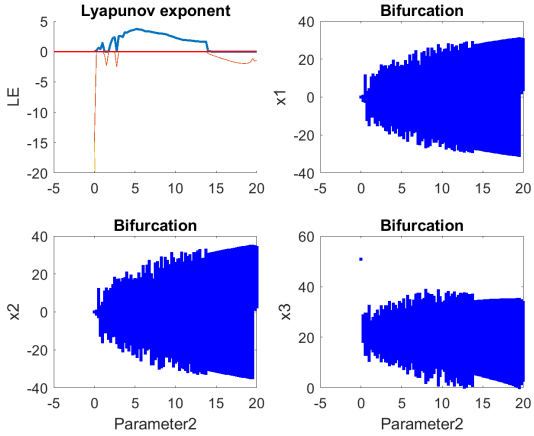


(d) Lu system ABM method

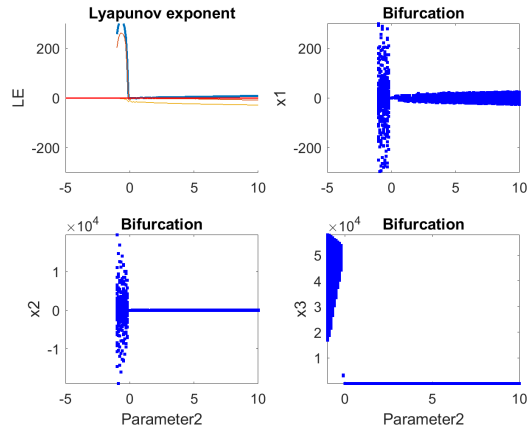
Figure 3.6 – LE and bifurcation results for Chen and Lu systems over different fractional derivatives employing different calculation methods $(a_c, b_c, c_c) = (35, 3, 28)$, $(a_l, b_l, c_l) = (36, 3, 20)$



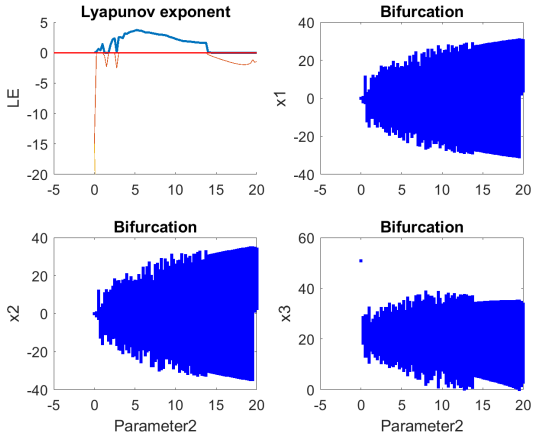
(a) Lu system GL method a_l LE results



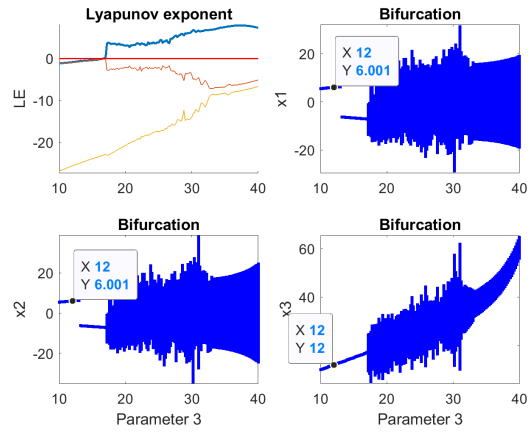
(b) Lu system ABM method a_l LE results



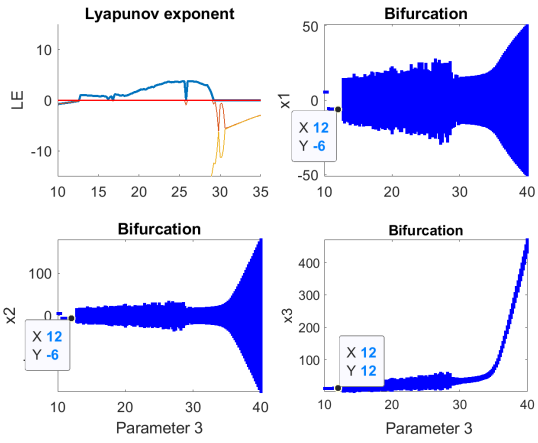
(c) Lu system GL method b_l LE results



(d) Lu system ABM method b_l LE results



(e) Lu system GL method c_l LE results



(f) Lu system ABM method C_l LE results

Figure 3.7 – LE and bifurcation results for Chen and Lu systems over different fractional parameters employing different calculation methods

Chapter 4

Non-uniform grid calculation methods for fractional chaotic systems

4.1 Introduction

In this chapter, a non-uniform grid numerical calculation method based on the classical fractional ABM Corrector-Predictor method for the calculation of fractional chaotic system solutions will be proposed. To form the non-uniform grid on which the solutions are calculated, skew-tent maps have been adopted to vary the step size (grid space). The numerical simulations and their analysis show that the proposed non-uniform grid method brings different dynamics and chaoticity to the fractional system with respect to the classical ABM Corrector-Predictor method. Positive impacts in favor of the use of fractional chaotic system for cryptography propose are observed.

In the following section, Section 4.2 provide some basic knowledge on skew-tent map. Then, in Section 4.3, the non-uniform grid calculation method is thoroughly illustrated. Two different non-uniform grid are constructed using skew-tent maps and are illustrated in Section 4.3.1 and 4.3.2. After that, the simulation for 3D fractional systems are carried out in Section 4.4, and the analysis and comparison of the proposed and classical ABM methods are demonstrated in 4.5. Finally, we draw conclusion in Section 5.4.

4.2 Chaotic map used for non-uniform grid

In this section, we give some preliminaries on real domain skew tent map. The map is used to vary the step size and form the non-uniform grid on which the fractional system is calculated. The phase portrait, bifurcation diagram and Lyapunov exponent results of

he skew tent map are also illustrated.

4.2.1 Skew tent map in real domain

The skew tent map is derived from classical tent map, which is in the range of $(0, 1)$ holding the form as below,

$$x(n+1) = \begin{cases} \mu x(n), & \text{for } x(n) < \dfrac{1}{2} \\ 1 - \mu x(n), & \text{for } \dfrac{1}{2} \leq x(n) < 1 \end{cases} \tag{4.1}$$

where μ is the control parameter. The skew tent map defined in he real domain can be expressed as follows,

$$\mathrm{Xst}(n+1) = \begin{cases} \dfrac{\mathrm{Xst}(n)}{p}, & \text{for } 0 \leq \mathrm{Xst}(n) < p \\ \dfrac{1 - \mathrm{Xst}(n)}{p}, & \text{for } p \leq \mathrm{Xst}(n) \leq 1 \end{cases} \tag{4.2}$$

where $\mathrm{Xst}(n), n = 1, 2, ...$ stand for the states, and p is the control parameter in the range of $(0, 1)$.

4.2.2 Simulation results for the adopted chaotic map

The fixed point of a dynamic map is the intersection of $\mathbf{f} = (\mathbf{x}, p)$ and $\mathbf{f} = \mathbf{x}$, which indicates that the orbits remain locked with respect to the change of iterations. For the given one-dimensional skew-tent map, the fixed point can be obtained by solving the equation $\mathrm{Xst}(n+1) = \mathrm{Xst}(n)$.

Two unstable fixed points $\mathrm{Xst}_1^* = 0$ and $\mathrm{Xst}_2^* = \frac{1}{2-p}$ exist for the given value of $p \in (0, 1)$. In Fig.4.1, we give the shape of the skew tent map and their first-hint fixed points with different control parameter (p) value of $1/4, 1/2, 9/10$, respectively.

It is to be remembered that the fixed points should be excluded when one aims to design maps with chaotic behavior. One should also avoid the initial values that are the (backward iterates) pre-images of the fixed points because they also lead to the fixed points after iterating forward.

As discussed in Section 1.3.1, the bifurcation diagram is a graph that gives a visual illustration of the system states values versus the evolution of the parameters. It shows the

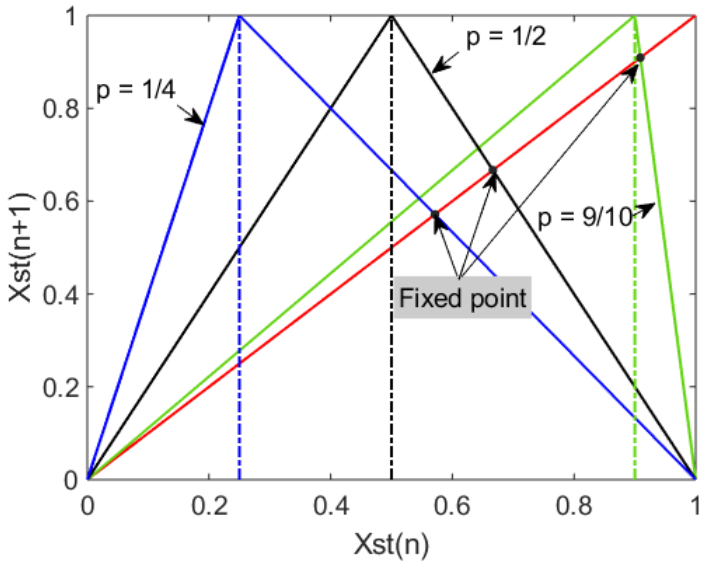


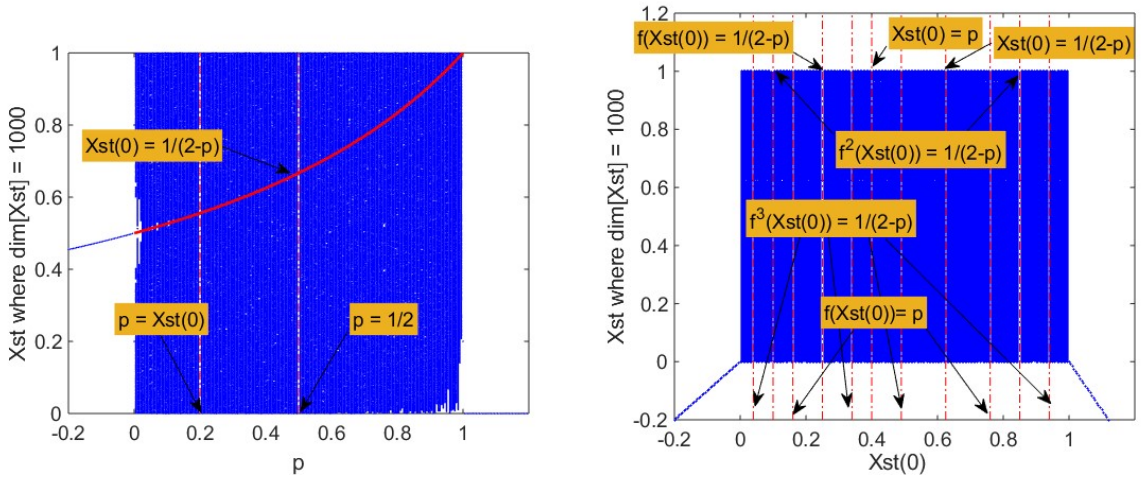
Figure 4.1 – Skew tent map with different control parameter p values

changes in the dynamic behavior of the chaotic map with the variation of the parameter values.

We give the bifurcation diagrams of the skew-tent map with initial condition $\mathrm{Xst}(0) = 0.2$ in Fig.4.2a. It can be observed that when p is in the range of -0.2- to 0 and 1 to 1.2, a continuous blue line appears in the figure. This means the values of the skew-tent map states, $\mathrm{Xst}(n)$, remain unchanged after many iterations for every p in these ranges. However, after the transient period, with p varying from 0 to 1, the outputs of the map scattered between $(0, 1)$. This verifies that the map is chaotic with its control parameter p chosen in the interval of $]0, 1[$.

It can be observed that there are two white vertical "gap" in Fig. 4.2a at $p = \frac{1}{2}$ and the initial value $p = 0.2 = \mathrm{Xst}(0)$. When $p = \mathrm{Xst}(0)$, the map maps to the fixed point 0 after 2 iterations, so, no chaotic behavior is displayed. When $p = 1/2$, it is easy to calculate that after several iterations, the map exhibits a periodic behavior with period 2 where the states take one of two values. (Take $\mathrm{Xst} = 0.2$ as an example, $\mathrm{Xst}(1) = \frac{0.2}{0.5} = 0.4$; $\mathrm{Xst}(2) = \frac{0.4}{0.5} = 0.8$;$\mathrm{Xst}(3) = \frac{1-0.8}{0.5} = 0.4$;$\mathrm{Xst}(4) = \frac{0.4}{0.5} = 0.8$. This means only after one iteration, the map starts to oscillate between two values 0.4 and 0.8.)

We also give in Fig.4.2b the impact of the different initial values on the chaotic behavior of the skew-tent map. The control parameter p is set to 0.4. One can observe that when $\mathrm{Xst}(0)$ is in the range of $[0, 1]$, the image of the map through iterations also lies in the



(a) Bifurcation diagram, Xst(0) = 0.2 (b) Impact of the initial value, $p = 0.4$

Figure 4.2 – Bifurcation of the skew tent map and impact of different initial condition values

same range and exhibits chaotic behavior throughout the interval except for a finite set of points. These specific values (where the white lines appear) are the fixed points of the map and their pre-images as mentioned previously. For example, within 3 iterations, the skew-tent map with control parameter $p = 0.4$, initial value $\mathrm{Xst}(0) = 0.04$ produces the state values $\mathrm{Xst}(1) = \frac{0.04}{0.4} = 0.1$, $\mathrm{Xst}(2) = \frac{0.1}{0.4} = 0.25$ and $\mathrm{Xst}(3) = \frac{0.25}{0.4} = 0.625$, which gives the fixed point $\frac{1}{2-p}$. Theoretically, as mentioned before, these initial values should be avoided when trying to acquire chaotic behavior since they reach the fixed point after finite iterations.

4.3 Non-uniform grid calculation method proposed for fractional chaotic system

In this section, we propose a numerical calculation method for fractional systems based on ABM fractional Corrector and Predictor method discussed in section 3.3.2.

The original method calculates the approximated solutions $x_h(t_j), (j = 1, 2, \ldots n)$ assuming a uniform grid $t_n = nh : n = 0, 1, \ldots, N$ with some integer N and $h := T/N$ is employed (interval $[0, T]$ is the interval where unique solution exists for the system). However, unlike the classical approach which is calculated with a fixed step size h, we propose two original non-uniform grids to vary the step size in each iteration. Generallly

speaking, the proposed method employs a varying step sizes obtained by a grid space switching mechanism involving the use of the skew-tent map discussed in Section 4.2. Simulations are runned by employing the widely applied MATLAB fractional differential equation solver FDE given by [Garrappa Roberto, 2021] with modifications made for our non-uniform grid.

4.3.1 Proposed numerical calculation method with first non-uniform grid

As mentioned before, the non-uniform grid fractional system solution calculation method we proposed is based on the classical fractional ABM Corrector and predictor method. We first proposed a variable discretization step $h(n)$ employing one single skew-tent map. The outputs of the map is used to alter the grid space and gives rise to the non-uniform grid 1 for the calculation of fractional system solution.

To be specific, the grid space $h(n)$ is obtained by multiplying a fixed value h by $i+1$ as shown in the following mathematical formula,

$$h(n) = h \times (i+1), \quad \text{if } \mathrm{Xst}(n) \in [0.2 \times i,\ 0.2 \times (i+1)[,\ i = 0, 1, 2, 3, 4. \tag{4.3}$$

The $\mathrm{Xst}(n)$ in the equation stands for the state of skew-tent map given by equation (4.2).

In the sequel, we adopted $h = 0.001$. Therefore, the non-uniform grid $h(n)$ takes one of the five values in the set of $\mathbf{S} = \{0.001, 0.002, 0.003, 0.004, 0.005\}$, and can be expressed as follows,

$$h(n) = \begin{cases} 0.001, & 0 < \mathrm{Xst}\,(n) \leq 0.2 \\ 0.002, & 0.2 < \mathrm{Xst}\,(n) \leq 0.4 \\ 0.003, & 0.4 < \mathrm{Xst}\,(n) \leq 0.6 \\ 0.004, & 0.6 < \mathrm{Xst}\,(n) \leq 0.8 \\ 0.005, & 0.8 < \mathrm{Xst}\,(n) \leq 1 \end{cases} \tag{4.4}$$

It is to be noticed that, here we choose 5 sequentially sorted intervals in which falls the $\mathrm{Xst}(n)$ to determine the corresponding value of $h(n)$. We can also adopt different sorting orders for the intervals, which will be discussed in Section 4.3.2.

With the non-uniform grid constructed, now we give in the following the expression for the proposed calculation method. We do not go into details of the formula deducing, since it is the same as the classical ABM method which can be found in section 3.3.2 and only give the deduced simplified following states calculation equations in (4.5)-(4.9).

Bearing in mind that the fractional derivative α in the equations are smaller than 1.

Non-uniform grid ABM Corrector-Predictor method with Grid 1

The states of the fractional system applying our proposed method with non-uniform grid 1 is calculated as follows,

$$\mathrm{X}\left(n+1\right)=\mathrm{X}\left(0\right)+\frac{h(n)^{\alpha}}{\Gamma\left(\alpha+2\right)}f\left(\mathrm{X}^{P}\left(n+1\right)\right)+\frac{h(n)^{\alpha}}{\Gamma\left(\alpha+2\right)}\sum_{j=0}^{n}a_{j,n+1}f\left(\mathrm{X}\left(j\right)\right), \tag{4.5}$$

where f stands for the system equations, $\Gamma(\cdot)$ is the gamma function given in (2.10). $h(n)$ is the non-uniform grid space (step size) acquired through following equation,

$$h(n)=0.001\times(i+1),\quad \text{if } \mathrm{Xst}(n)\in[0.2\times i,\ 0.2\times(i+1)[,\ i=0,1,2,3,4. \tag{4.6}$$

The parameter a in equation (4.5) takes the form as follows,

$$a_{j,n+1}=\begin{cases} n^{\alpha+1}-(n-\alpha)(n+1)^{\alpha}, & \text{if} j=0,\\ (n-j+2)^{\alpha+1}+(n-j)^{\alpha+1}-2(n-j+1)^{\alpha+1}, & \text{if} 1\leq j\leq n,\\ 1, & \text{if} j=n+1. \end{cases} \tag{4.7}$$

$\mathrm{X}^{p}(n+1)$ in equation (4.5) denotes the predicted value of $\mathrm{X}(n)$ and is formulated as,

$$\mathrm{X}^{P}(n+1)=\mathrm{X}(0)+\frac{1}{\Gamma(\alpha)}\sum_{j=0}^{n}b_{j,n+1}f(\mathrm{X}(j)),\quad 0<\alpha<1 \tag{4.8}$$

where parameter b is expressed as,

$$b_{j,n+1}=\frac{h(n)^{\alpha}}{\alpha}\left((n+1-j)^{\alpha}-(n-j)^{\alpha}\right) \tag{4.9}$$

It is to be remembered that in all the above equations, $\mathrm{X}(n+1)$ and $\mathrm{X}^{P}(n+1)$ stand for the state vectors with state components. The number of components is equal to the dimension of the system adopted.

4.3.2 Proposed method with another Non-uniform Grid 2

In this section, a more elaborated non-uniform grid, Grid 2, is discussed. The purpose is to study the impacts of the complication of non-uniform grid to the chaoticity of the fractional system, and to introduce more parameter to the Fractional chaotic pseudo-random number generator. The basic concept is the same as non-uniform grid 1, to vary the step size by the output of skew-tent map. But for grid 2, two skew-tent maps have been employed to construct the switching mechanism. One of them is used to assign different values to $h(n)$, and the other is adopted to increase the system's complexity by taking into account different step size allocation possibilities.

To introduce our proposed grid 2, we first introduce and recall some notations that are used to formulate the variable step size.

The two employed skew-tent maps with control parameter p_1 and p_2 respectively are given below,

$$\mathrm{Xst}_1(n) = \begin{cases} \dfrac{\mathrm{Xst}_1(n-1)}{p_1}, & 0 < \mathrm{Xst}_1(n-1) \leq p_1, \\ \dfrac{1-\mathrm{Xst}_1(n-1)}{1-p_1}, & p_1 < \mathrm{Xst}_1(n-1) < 1, \\ \mathrm{Xst}_1(n-1) - 0.05, \text{otherwise.} \end{cases} \tag{4.10}$$

$$\mathrm{Xst}_2(n) = \begin{cases} \dfrac{\mathrm{Xst}_2(n-1)}{p_2}, & 0 < \mathrm{Xst}_2(n-1) \leq p_2. \\ \dfrac{1-\mathrm{Xst}_2(n-1)}{1-p_2}, & p_2 < \mathrm{Xst}_2(n-1) < 1. \\ \mathrm{Xst}_2(n-1) - 0.05, \text{otherwise.} \end{cases} \tag{4.11}$$

An indicator function $\mathbf{1}_{\mathrm{A_i}}(.)$ is employed and it can be expressed as,

$$\mathbf{1}_{\mathrm{A}_i}(\mathrm{Xst}_1(n)) = \begin{cases} 1, & \mathrm{Xst}_1(n) \in \mathrm{A}_i \\ 0, & \mathrm{Xst}_1(n) \notin \mathrm{A}_i \end{cases}, i = 1,2,3,4,5. \tag{4.12}$$

where A_i denotes the following interval

$$\mathrm{A}_i = \left(y | \frac{1}{5}(i-1) < y \leq \frac{1}{5} i \right), i = 1,2,3,4,5. \tag{4.13}$$

The varying grid spaces still take one of the five values from the set **S** consisted of five values from 0.001 to 0.005 with a discrepancy of 0.001 as in Grid 1 ($\mathbf{S} = \{0.001, 0.002, 0.003, 0.004, 0.005\}$). A matrix of size 120×5 including all the possible combinations of the elements in set **S** is then constructed and is given below,

$$\mathbf{H} = \begin{pmatrix} 0.001 & 0.002 & 0.003 & 0.004 & 0.005 \\ 0.001 & 0.002 & 0.003 & 0.005 & 0.004 \\ 0.001 & 0.002 & 0.004 & 0.003 & 0.005 \\ 0.001 & 0.002 & 0.004 & 0.005 & 0.003 \\ & & \vdots & \vdots & \\ 0.005 & 0.004 & 0.003 & 0.002 & 0.001 \end{pmatrix} \tag{4.14}$$

With all the above introduced notations, the mathematical formulation of the variable step size $h(h(n)$ is expressed as follows,

$$\begin{aligned} h(n) &= f_h(\mathrm{Xst}_1(n), \mathrm{Xst}_2(n)) \\ &= \sum_{i=1}^{5} \mathbf{1}_{\mathrm{A}_i}(\mathrm{Xst}_1(n))\mathbf{H}(k,i) \\ k &= ((2^{32}-1) \times \mathrm{Xst}_2(n)) \,(\mathrm{mod}\ 120) + 1 \end{aligned} \tag{4.15}$$

$\mathbf{H}\,(k,i)$ in equation (4.15) denotes the element on k-th row and i-th column of the matrix **H** given in 4.14 and k is determined by second skew-tent map output $\mathrm{Xst}_2(n)$ according to equation (4.11).

A brief interpretation of the formula is given here. With an output of skew tent map in the range of $(0, 1)$, we introduce 5 intervals $\mathrm{A}_i, (i = 1, 2, \ldots 5)$ of the same size obtained by (4.13) for the assignment of $h(n)$. To match the intervals to the five possible step sizes in **S**, we construct the matrix **H** (equation (4.14)). By performing the modulo operation, the states $\mathrm{Xst}_2(n)$ are processed to acquire a row indice k for the matrix. Then, the interval A_i is matched with the corresponding steps sizes value on the i-th column and k-th row of **H**. Finally, the step size is assigned to $h(n)$ depending on which interval A_i (among the five) $\mathrm{Xst}_1(n)$ lies in.

In the following section, our proposed variable step size method is applied to numerically calculate two 3D fractional systems, fractional Chen and Lu systems respectively. The chaoticity of the maps is discussed.

4.4 Simulation results adopting the proposed non-uniform grid method

In this section, we employ the proposed non-uniform grid method to numerically solve fractional Chen and Lu systems. The impact of the different grid choices on the chaotic properties of the systems are also discussed.

4.4.1 Fractional Chen system applying non-uniform Grid 1

The fractional Chen system we employed here is the one given in Section (2.33) with one identical fractional order α_c for all its differential equations. Thus, it has the form as follows,

$$f_c(x) = \begin{cases} D^{\alpha_c} x_1(t) = a_c (x_2(t) - x_1(t)) \\ D^{\alpha_c} x_2(t) = (c_c - a_c) x_1(t) - x_1(t) x_3(t) + c_c x_2(t) \\ D^{\alpha_c} x_3(t) = x_1(t) x_2(t) - b_c x_3(t) \end{cases} \tag{4.16}$$

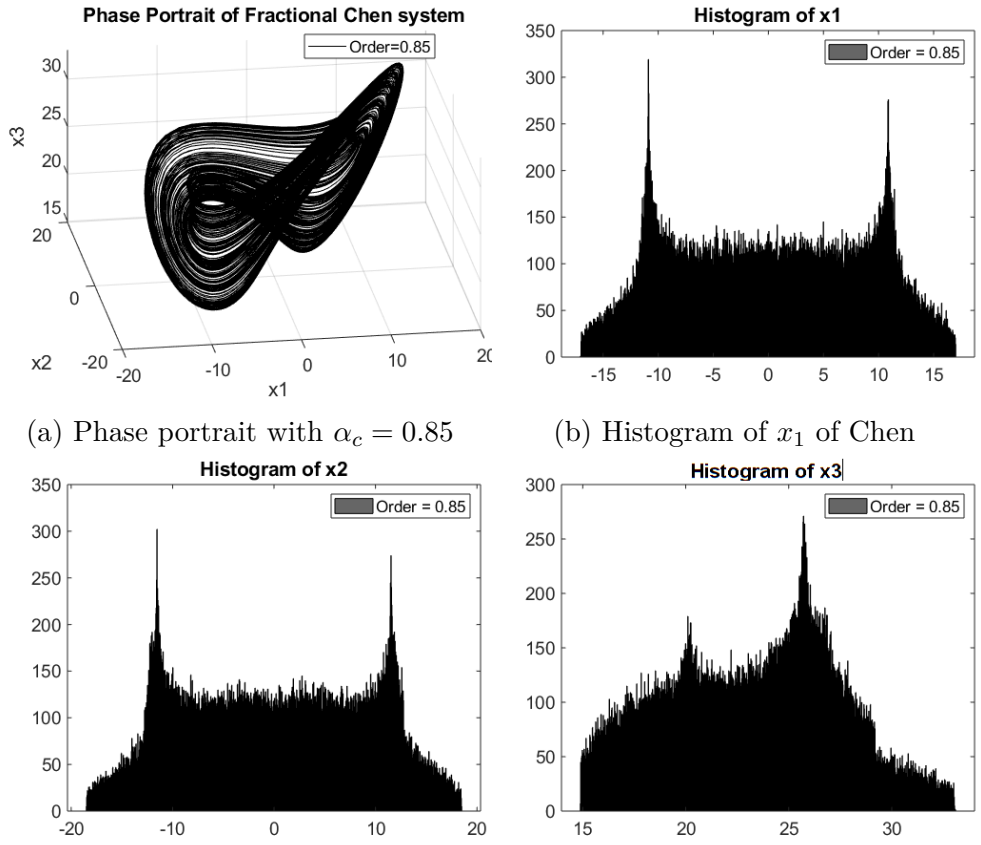
We employed our proposed calculation method illustrated in Section 4.3.1 to numerically solve the system given in equation (4.16).

10^6 states have been calculated and the last 100000 iterated states (after the transient) have been plotted in the Phase portrait figure (Fig.4.3a). The system parameters are $(a_c, b_c, c_c) = (35, 3, 28)$; the initial conditions are $(-9, -5, 14)$. We also gave the histograms of fractional Chen system with fractional derivative order $\alpha_c = 0.85$ to show the distribution of the output. The histograms of the state components with the last 100000 iterations have been classified into 1000 classes and have been plotted in Fig.4.3b, 4.3c and 4.3d.

It can be observed that the system with order 0.85 possesses a double-scroll attractor. For both state components x_1 and x_2, the values are quasi symmetric and scattered within a range approximate to $[-18, 18]$; whereas for component x_3, the values are always positive and are in ranges of $[15, 33]$.

As stipulated in Section 1.3.3, a LE value greater than 0 often indicates the existence of chaotic behavior. The Lyapunov Exponents are also computed for the fractional Chen system by modifying the MATLAB code given by [Danca, 2018].

The LE spectrum over different fractional derivative orders is given in Fig.4.4a. To acquire the LE values over different fractional derivative orders, 10^6 states have been iterated for each derivative order in the range of 0.45 to 1 with a gap of 0.01. The LEs



(a) Phase portrait with $\alpha_c = 0.85$

(b) Histogram of x_1 of Chen

(c) Histogram of x_2 of Chen

(d) Histogram of x_3 of Chen

Figure 4.3 – Phase Portrait and Histogram of fractional Chen systems with different derivative order $\alpha_c = 0.85$

have been calculated throughout the iterations, the last LE of different orders evaluated were then combined. The parameters are $(a_c, b_c, c_c) = (35, 3, 28)$. It can be observed that for the three state component x_1, x_2 and x_3, the x_1 possesses positive LE values. In addition, the first LE greater than 0 appears at $\alpha_c = 0.52$.

We then focused on the state component x_1 and plotted the LE values for state component x_1 over different parameters and fractional orders. It is to be remarked that when altering the parameters, we set the other two parameters unchanged, the same as for the fractional derivative order evaluation. For parameter a_c and c_c, the LEs are evaluated every 0.5 in the range of $[30, 50]$ and $[10, 35]$ respectively; for parameter b_c, the LE is evaluated every 0.25 in the range of $[-5, 15]$. For all the three parameters, eight fractional derivative orders from 0.65 to 0.99 have been employed to calculate 100000 system states. The LE results are given in Fig. 4.4. From the figure, one can observe that generally speaking, the smaller the fractional derivative order is, the greater LE the x_1 possesses. From the aspect of the values of parameters for which the LEs are positive, different orders have different ranges. But there are common ranges for all the fractional derivatives.

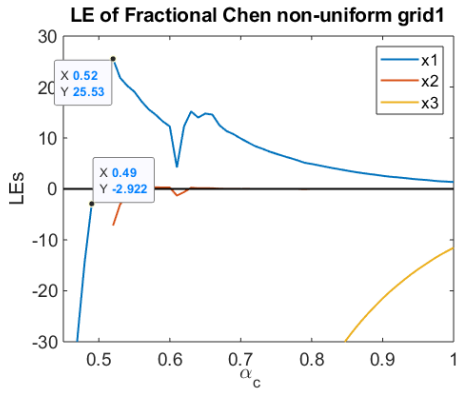
The bifurcation diagrams are also given to evaluate the chaotic behavior of the fractional Chen system calculated by the proposed non-uniform grid method with Grid 1. In Fig.4.5, 4.6 and 4.7, we displayed the bifurcation diagrams over a_c, b_c and c_c respectively with different fractional orders. It can be observed that with different fractional derivative orders, the system performs differently but is scattered all over the state values within certain parameter ranges.

According to both LE results and bifurcation diagrams, in our FCPRNG design, we have chosen $a_c \in [35, 40]$, $b_c \in [1.5, 3.5]$, $v_c \in [23, 28]$, in order to make sure the system possesses positive LE values and act chaotically.

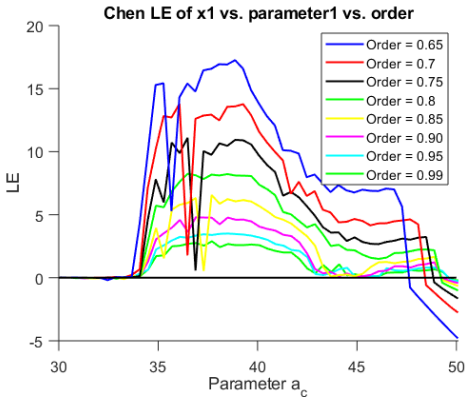
4.4.2 Fractional Lu system applying non-uniform Grid 1

The system equations of fractional Lu system is given as below with α_l denotes the commensurate fractional derivative order, and (a_l, b_l, c_l) the parameters.

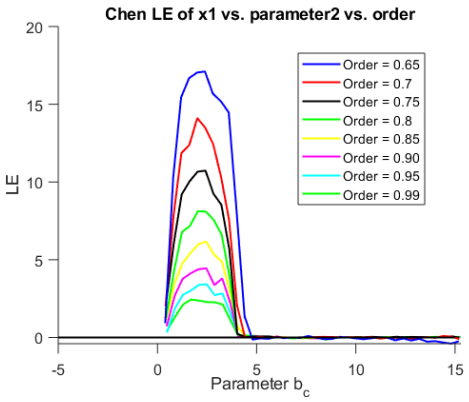
$$f_l(x) = \begin{cases} D^{\alpha_l} x_1(t) = a_l (x_2(t) - x_1(t)) \\ D^{\alpha_l} x_2(t) = -x_1(t) x_3(t) + c_l x_2(t) \\ D^{\alpha_l} x_3(t) = x_1(t) x_2(t) - b_l x_3(t) \end{cases}. \tag{4.17}$$



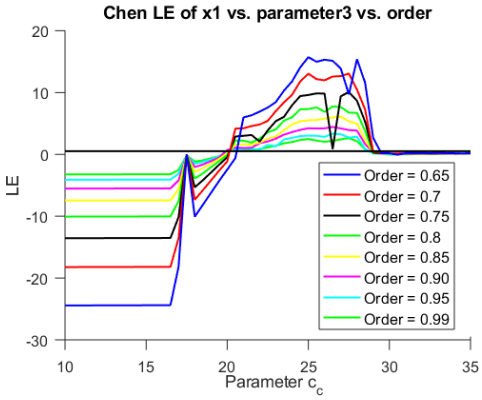
(a) LEs over different orders



(b) LE of x_1 v.s Parameter a_c

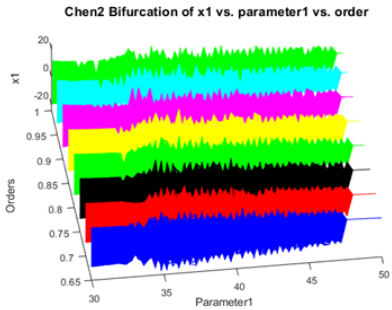


(c) LE of x_1 v.s Parameter b_c

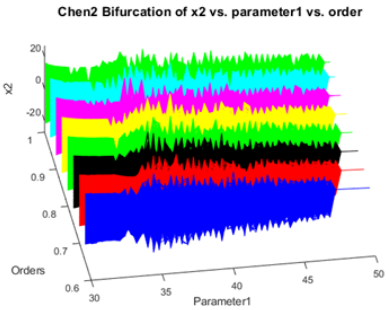


(d) LE of x_1 v.s Parameter c_c

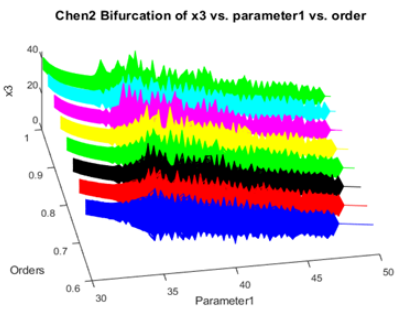
Figure 4.4 – Lyapunov Exponent results of fractional Chen systems with different parameters and orders



(a) x_1 vs a_c

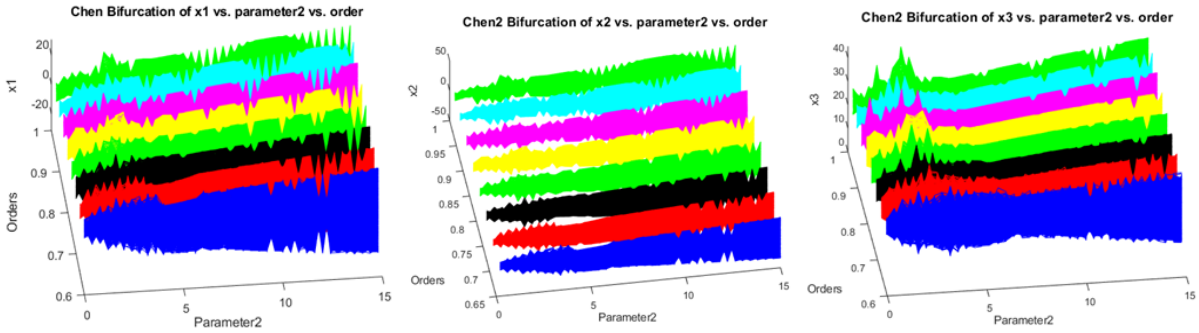


(b) x_2 vs a_c



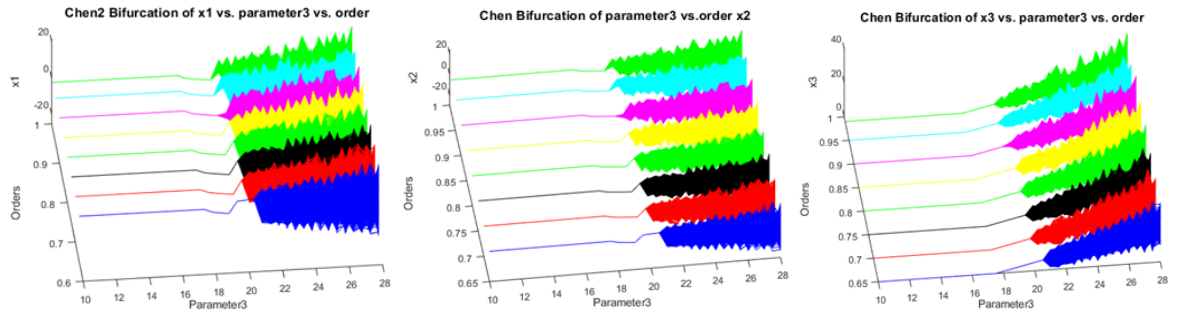
(c) x_3 vs a_c

Figure 4.5 – Bifurcation diagrams of parameter a_c for fractional Chen systems state components with different derivative orders



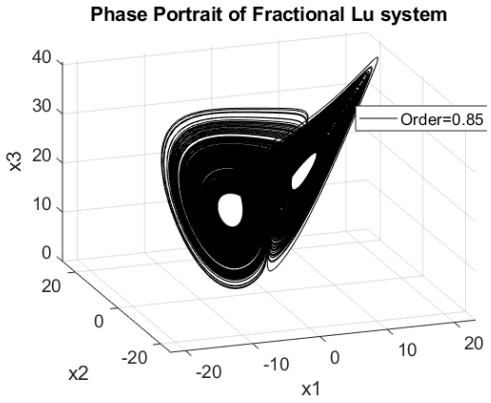
(a) x_1 vs b_c (b) x_2 vs b_c (c) x_3 vs b_c

Figure 4.6 – Bifurcation diagrams of parameter b_c for fractional Chen systems state components with different derivative orders

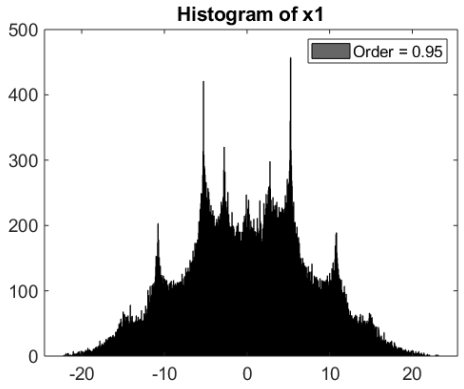


(a) x_1 vs b_c (b) x_2 vs b_c (c) x_3 vs b_c

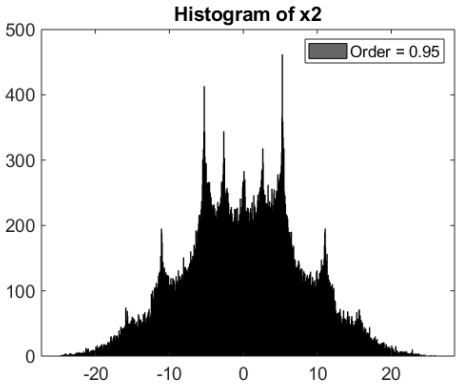
Figure 4.7 – Bifurcation diagrams of parameter c_c for fractional Chen systems state components with different derivative orders



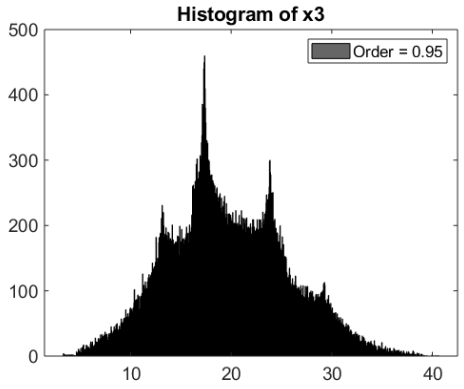
(a) Phase portrait with $\alpha_l = 0.95$



(b) Histogram of x_1 of Lu



(c) LHistogram of x_2 of Lu



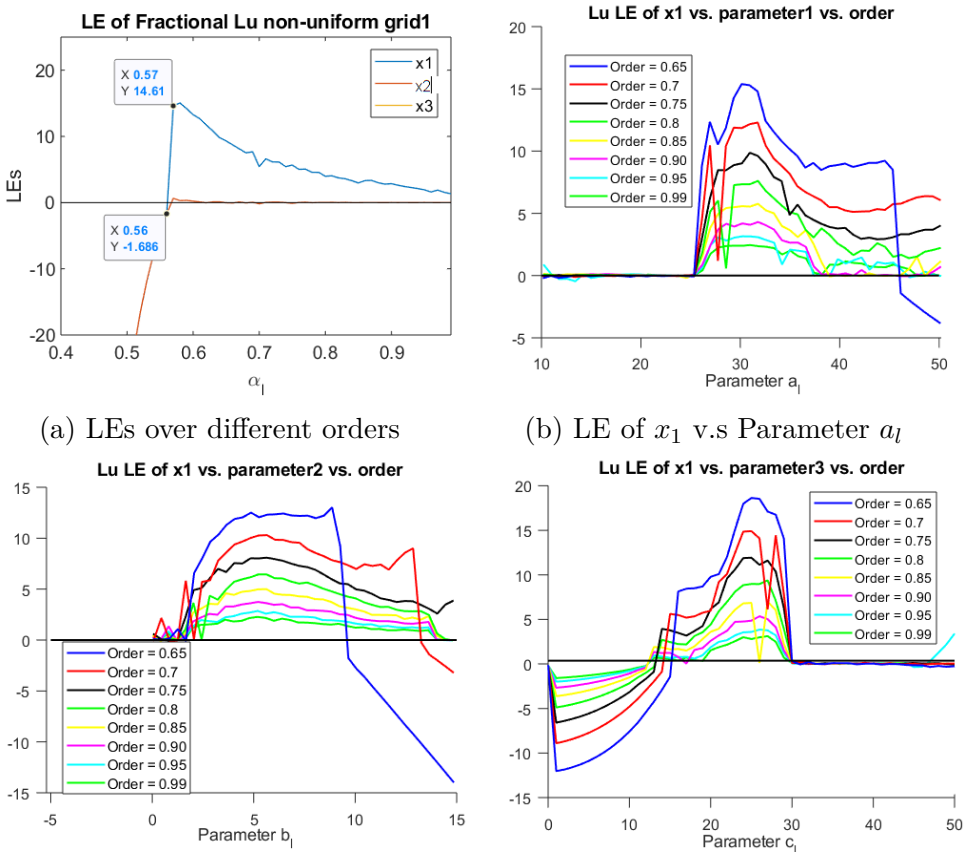
(d) Histogram of x_3 of Lu

Figure 4.8 – Phase Portrait and Histogram of fractional Lu systems with fractional derivative order $\alpha_l = 0.95$

Applying proposed Grid 1 method, 10^6 states have been calculated for fractional Lu system with parameter (a_l, b_l, c_l) equal to $(36, 3, 20)$ and initial condition $(x_1(0), x_2(0), x_3(0)) = (0.2, 0.5, 0.3)$. We plotted the phase portrait of the system with derivative order 0.95 in Fig.4.8a. The histograms of the state components x_1, x_2, x_3 are also given in Fig.4.8b, 4.8c and 4.8d, respectively. 1000 classes were chosen, and only the last 100000 states were employed to obtain the histogram.

Similar to the states of the fractional Chen system, a double-scroll appears for the phase portrait of the fractional Lu system with given parameters and order. The histogram for x_1 and x_2 are quasi-symmetric where x_1 and x_2 always lie in the range of $[-25, 25]$, and x_3 in a range of $[0, 40]$.

The LE results over different fractional derivatives with parameters $(a_l, b_l, c_l) = (36, 3, 20)$ are given in Fig.4.9a. The orders are evaluated every 0.01, and the smallest order with



(a) LEs over different orders

(b) LE of x_1 v.s Parameter a_l

(c) LE of x_1 v.s Parameter b_l

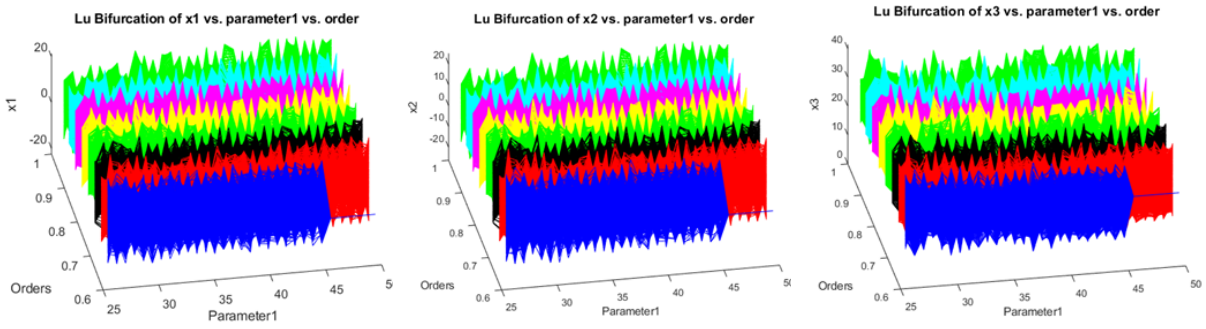
(d) LE of x_1 v.s Parameter c_l

Figure 4.9 – Lyapunov Exponent results of fractional Lu systems with different parameters and orders

LE greater than 0 is $\alpha_l = 0.57$.

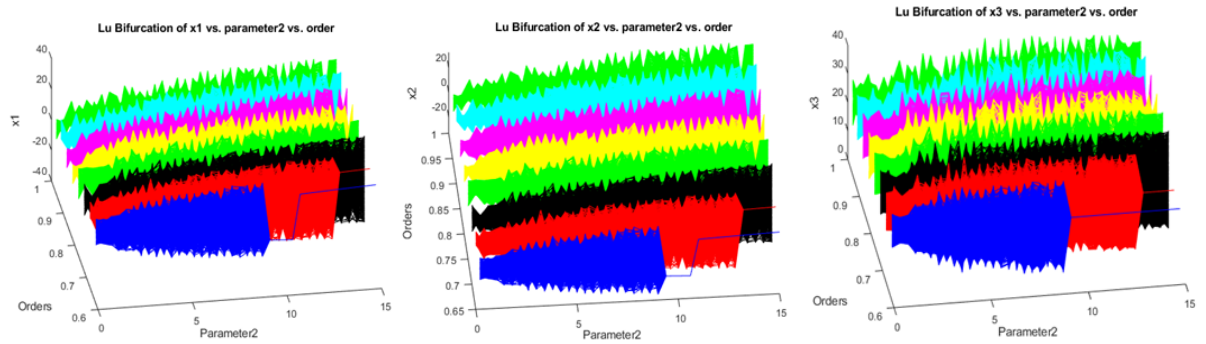
The LE results over different parameters have also been calculated and given in Fig.4.9. The a_l was calculated every 0.05 within the range of 10 to 50; b_c in the range of -5 to 15 was evaluated every 0.05; and c_c was varied with a gap of 0.05 within the range of $[0, 50]$. A similar conclusion can be drawn from the results of the fractional Chen system. With smaller fractional derivative orders, the LEs are smaller; the parameter values for the LEs greater than 0 are not the same for different fractional orders, however, common ranges exist.

We also give the bifurcation diagrams of fractional Lu system states calculated applying our proposed non-uniform grid method with different fractional derivative orders and parameters. The diagrams are given in Fig. 4.10, 4.11 and 4.12 for parameters a_l, b_l and c_l, respectively. The parameters are evaluated by varying one of the parameters and fixed



(a) x_1 vs a_l (b) x_2 vs a_l (c) x_3 vs a_l

Figure 4.10 – Bifurcation diagrams of parameter a_l for fractional Lu systems state components with different derivative orders



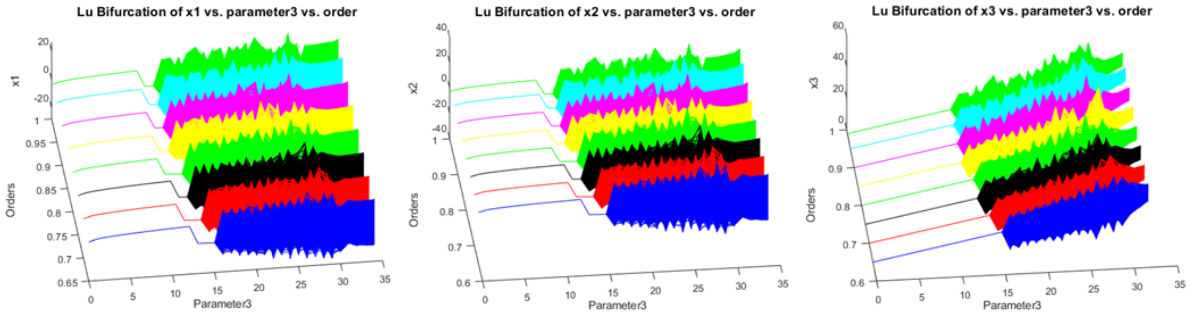
(a) x_1 vs b_l (b) x_2 vs b_l (c) x_3 vs b_l

Figure 4.11 – Bifurcation diagrams of parameter b_l for fractional Lu systems state components with different derivative orders

the others according to $(a_l, b_l, c_l) = (36, 3, 20)$

According to both LE results and bifurcation diagrams, in our FCPRNG design, we have chosen $a_l \in [35, 40]$, $b_l \in [3, 8]$, $c_l \in [20, 25]$, in order to make sure the system possesses positive LE values and acts chaotically.We also give the bifurcation diagrams of fractional Lu system states calculated by applying our proposed non-uniform grid method with different fractional derivative orders and parameters. The diagrams are given in Fig. 4.10, 4.11 and 4.12 for parameters a_l, b_l and c_l, respectively. The parameters are evaluated by varying one of the parameters and fixing the others according to $(a_l, b_l, c_l) = (36, 3, 20)$

According to both LE results and bifurcation diagrams, in our FCPRNG design, we have chosen $a_l \in [35, 40]$, $b_l \in [3, 8]$, $c_l \in [20, 25]$, in order to make sure the system possesses positive LE values and acts chaotically.



(a) x_1 vs c_l (b) x_2 vs c_l (c) x_3 vs c_l

Figure 4.12 – Bifurcation diagrams of parameter c_l for fractional Lu systems state components with different derivative orders

4.5 Different grid choices comparison

In this section, we compared the performance of the systems solved by adopting the proposed non-uniform grid methods and classical ABM corrector-predictor method.

4.5.1 LEs of proposed non-uniform grid and classical uniform grid

We analyzed first the LEs of fractional Chen system for different fractional derivative orders applying classical uniform grid ABM corrector predictor, and compared them with our proposed algorithm. The map has been evaluated on 55 values ranging from 0.45 to 1 with a discrepancy of 0.01. For each fractional derivative value, 10^6 states have been generated. The LEs have been first calculated over all the iterations, and the final values have been evaluated for each fractional order. They have been processed and combined together to form the LE spectrum curve given in Fig. 4.13.

It can be seen from the figure that for both methods, using uniform and non-uniform grids, among the three LEs, the LE for x_1 direction is greater than 0, indicating that the trajectories are expanding along this direction. One can also notice that the LE obtained through our proposed method exceeds zero at a smaller fractional derivative order ($\alpha_c = 0.52$) compared to the classical uniform grid predictor-corrector method ($\alpha = 0.54$). This indicates that with our proposed method, the fractional derivative range for the system to be chaotic has been enlarged. The same conclusion can also be detected from the bifurcation diagrams. In the diagram, the fixed focus can be observed with fractional order smaller than 0.52 for our proposed method and 0.54 for the uniform grid

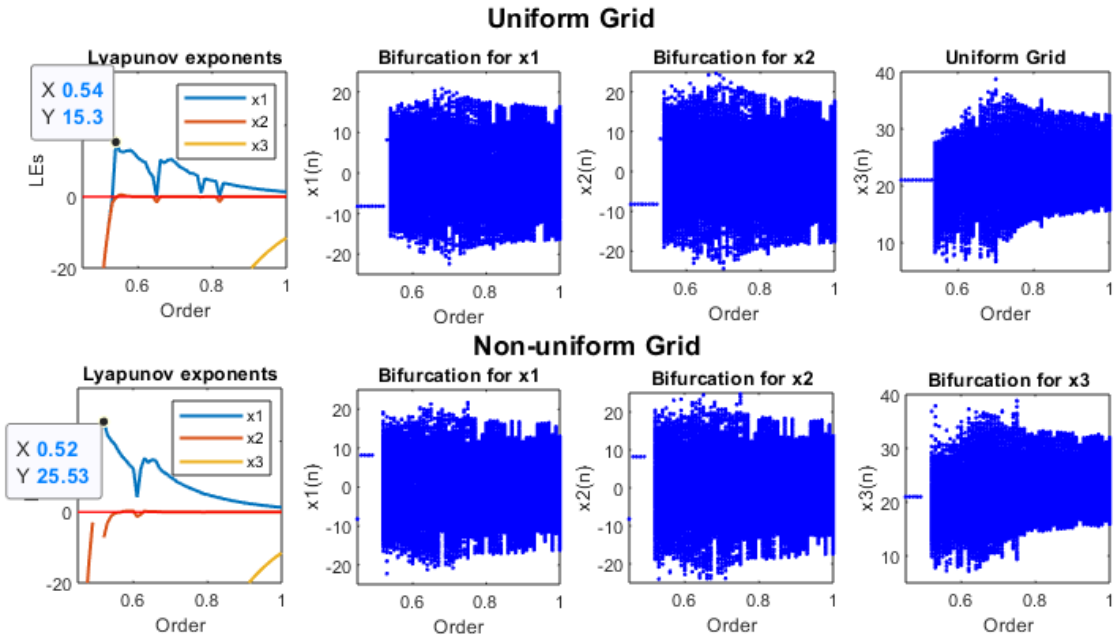


Figure 4.13 – Lyapunov exponent and Bifurcations for Chen system with fractional order α_c varying from 0.45 to 1 applying classical method and proposed non-uniform grid method

calculation method. After order 0.52 (0.54 for uniform grid), the bifurcation reveals the chaotic behavior.

We also give here the simulation results of fractional Lu systems for a clearer comparison between the two calculation methods. The LEs for fractional order from 0.55 to 1 are calculated every 0.01 and plotted for both methods in Fig. 4.14. A clear gap can be observed between the red line (uniform grid) and blue line (non-uniform grid), where the latter crosses the horizontal line (LE equals 0) between fractional derivatives orders of 0.56 and 0.57, while the former crosses the line between 0.6 and 0.61. This is also in accordance with our previous findings, which indicates that the implementation of non-uniform would enable the enlargement of the fractional order parameter range for which the system preserves its chaotic properties.

In Fig. 4.15 the comparison of the LEs of x_1 over different derivative orders obtained through our proposed non-uniform Grid 1, Grid 2 and classical uniform grid are given.

For the fractional Chen system, the orders have been evaluated every 0.01 from 0.45 to 1, whereas for the fractional Lu system, the evaluated derivative orders have been chosen as 0.55 to 1. It can be observed that for both fractional Chen and Lu systems, the curve of LE of Non-uniform Grid 1 (blue curve) across the horizontal line (LE equal to 0) at

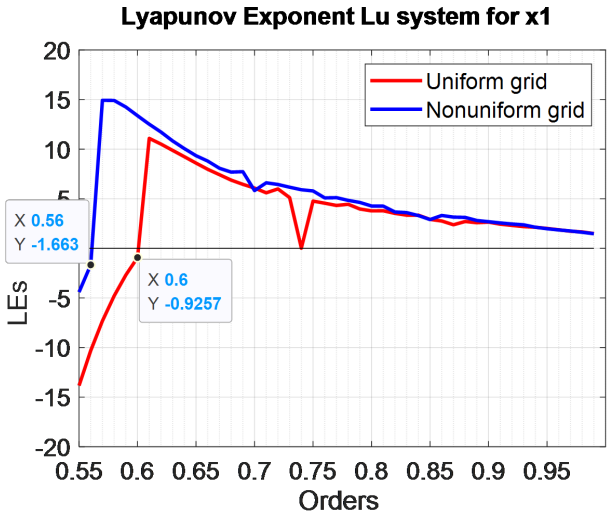
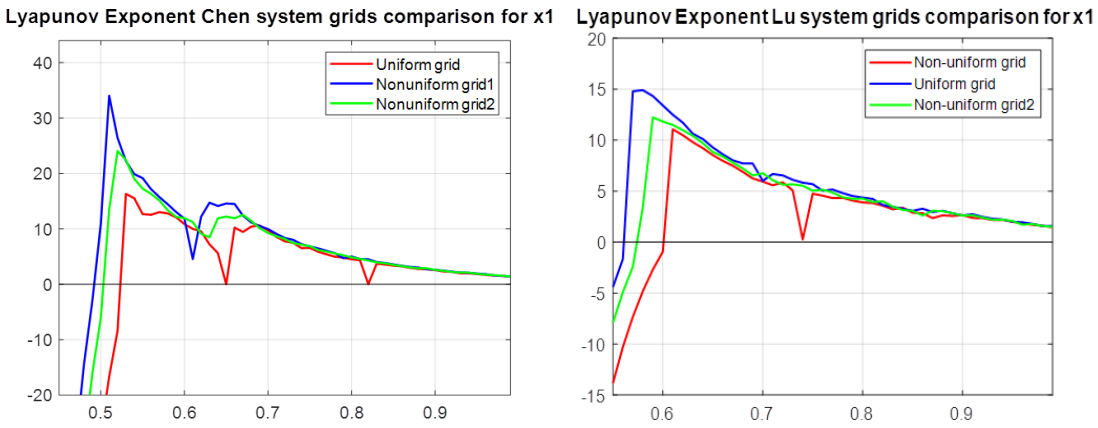


Figure 4.14 – LE for x_1 state component for fractional Lu systems applying both classical uniform grid method and proposed non-uniform Grid 1

the smallest fractional derivative orders with the largest LE values. The LE curve for the classical uniform grid (red curve) exceeds LEs equal to zero at the largest orders. The results for non-uniform Grid 2 lie between those of Grid 1 and the uniform grid. This indicates that our proposed non-uniform grid methods can enlarge the chaotic range of derivative order for both fracitonal Chen and Lu systems. But the Grid 1 possesses btter performance in terms of LE values and chaotic range.

Additional to the fractional derivatives, we have also analyzed the impact of the other control parameters. For the fractional Chen system, we have evaluated 50 successive values in the range of [20, 45], [1, 11], [20, 45] for its control parameters a_c, b_c, and c_c, respectively (change only one at a time, holding the others unchanged). The number of values whose LEs are greater than 0 among the 50 evaluated values has been acquired. For fractional Lu system, evaluation of 50 successive values in the range of [20, 45], [0, 10], [20, 45] for its control parameters a_l, b_l, and c_l has also been conducted. It can be observed from Table. 4.1 that our proposed method possesses more LEs greater than 0 for all the parameters as well as the fractional orders. This indicates that our proposed calculation method also introduces extra control parameters range for the system to be chaotic.

Knowing that the applied LE calculation method is only a qualitative measurement of chaotic properties for fractional order systems, we also calculate the percentage of



(a) LEs for Chen systems (b) LEs for Lu systems

Figure 4.15 – LE comparison of x_1 component of fractional Chen and Lu systems with different derivative orders applying classical uniform grid, proposed non-uniform Grid 1 and Grid 2

the number of LEs obtained applying our method, which exceeds that of the classical approach for a rough idea of the enhancement of chaoticity. Comparative results on the LEs obtained by our proposed non-uniform grid 1 calculation method and by the classical uniform grid approach are presented in Table. 4.1 for Chen and Lu systems. The results are also given in Table. 4.1. It can be concluded that for the evaluated parameter values that have LE greater than 0, the proposed method gives a LE which exceeds that of the classical method no less than 70% of the cases.

4.5.2 Time response and other results

To further evaluate the calculation methods and justify LE results from another perspective, we 'synchronized' the states of the Lu system calculated through the two methods by identifying and matching the states with identical time stamps (199961 'synchronized' states out of 10^6 iterated states). The last 150 'synchronized' states are plotted out in Fig. 4.16, which show in the time domain the evolution of the different system dynamics according to the grid choices. These figures confirm the analysis based on the LE (Fig. 4.14), for which with uniform grid, the system is not chaotic when the fractional derivative order α_l is smaller than 0.61, whereas with our proposed non-uniform gird, the system states exhibit chaotic behavior at fractional derivative order α_l starting at 0.57.

One can also observe from the y-coordinates of red and blue lines in Fig. 4.16a and

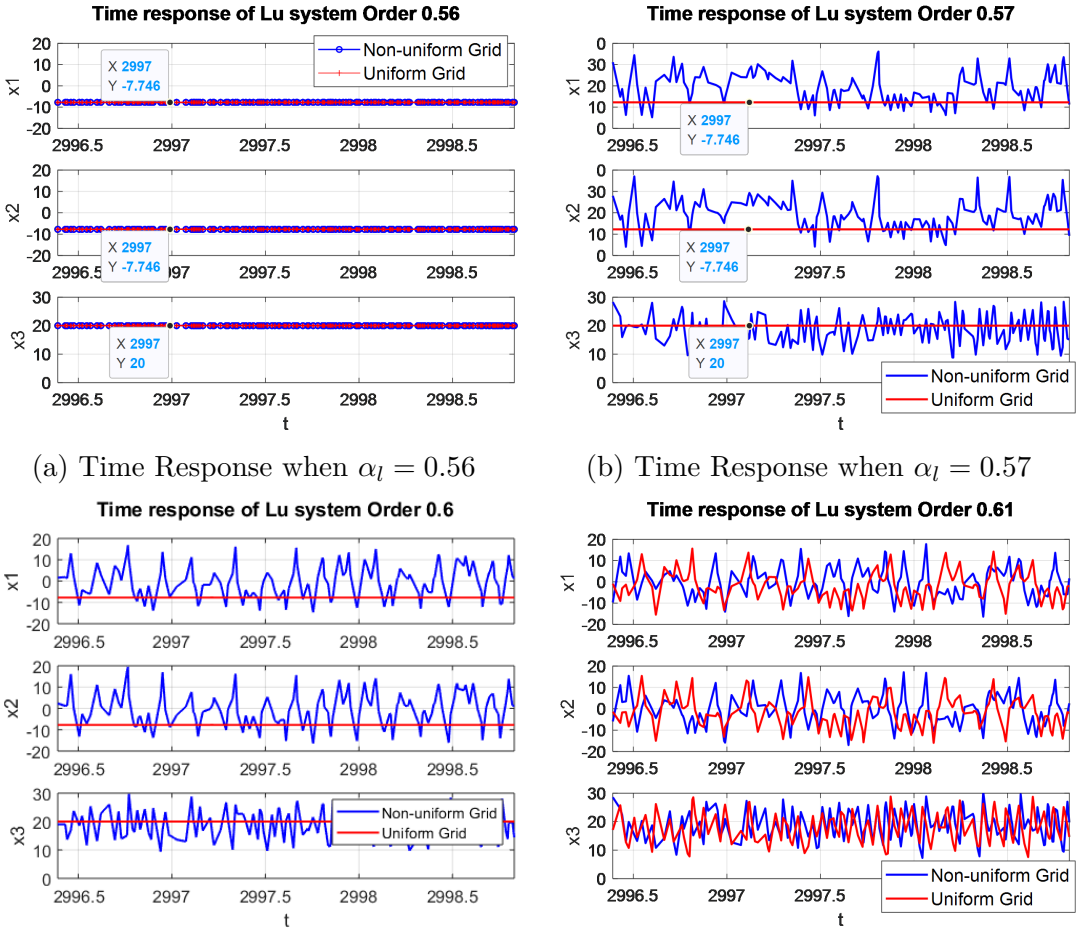
Table 4.1 – LE results for Chen and Lu system with different parameters and fractional orders

Systems	Parameters and Range		Number of values in the range	Percentage of LEs>0 (% in estimated values)		Proposed>classical (Greater in LE %)
				Uniform	Non-uniform	
Fractional Chen	α_c	0.5,1	50	45(90%)	49(98%)	89%
	a_c	20,45	50	48(96%)	49(98%)	80%
	b_c	1,11	50	39(78%)	43(86%)	78%
	c_c	20,45	50	49(98%)	50(100%)	70%
Fractional Lu	α_l	0.55,1	45	38(84%)	42(93%)	89%
	a_l	20,45	50	41(82%)	45(90%)	98%
	b_l	0,10	50	48(96%)	45(98%)	90%
	c_l	20,45	50	37(74%)	39(78%)	97%

4.16b that for order 0.56, both methods reach at the same point $(-7.746, -7.746, 20)$, as well as the states obtained through uniform grid calculation method for order 0.57 (red line in Fig. 4.16a, 4.16b and 4.16c). This indicates that under the evaluated initial conditions and parameters $(a_l, b_l, c_l) = (36, 3, 20)$, at fractional orders where there is no appearance of chaos. After the transient (sufficiently great number of iterations), the trajectories converge towards the fixed points as obtained from analytical study. The phase portraits of the attractors in Fig. 4.17 also confirm the coherence. There is only one fixed point for order 0.57 for the uniform grid method whereas for order 0.61 both methods possess LEs greater than 0, the phase portraits are chaotic and exhibit similar shape.

4.5.3 Computational time comparison

The computation time applying both methods for fractional Chen and Lu system are recorded and shown in Table. 4.2. We calculated the average computational time of different derivative orders evaluated. The time for non-uniform grid is acquired by averaging that of both Grid 1 and Grid 2. It can be observed that our proposed method takes less computational time with respect to the classical one (10^6 iterations). It should be mentioned here that this is also one merit of our proposed non-uniform calculation method in terms of design FCPRNG. Since the implementation of it can increase the efficiency of pseudo-random number generation, which is of great importance when it comes to PRNG design.



(a) Time Response when $\alpha_l = 0.56$

(b) Time Response when $\alpha_l = 0.57$

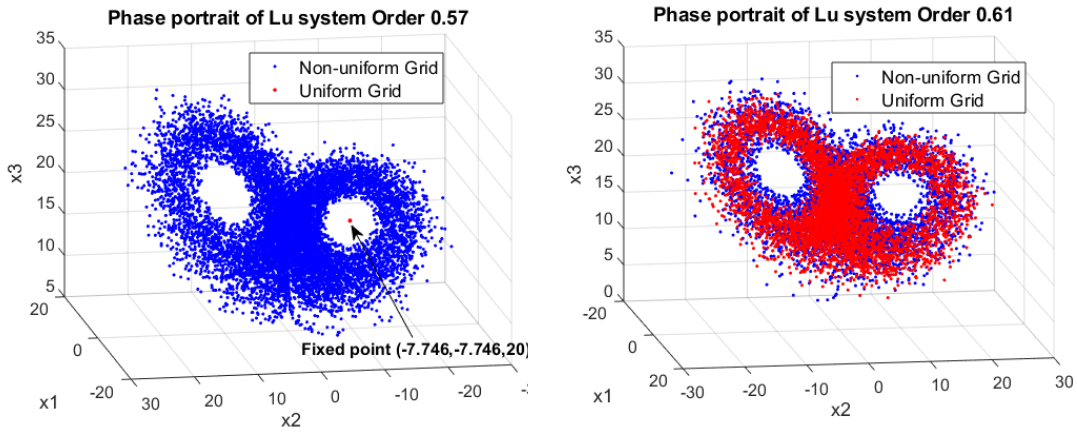
(c) Time Response when $\alpha_l = 0.6$

(d) Time Response when $\alpha_l = 0.61$

Figure 4.16 – Time response for Lu system with fractional order α_l equal to 0.56, 0.57 and 0.61

Table 4.2 – Computation time employing different grid

System	Computation time (s) for 10^6 iterations	
	Uniform grid	Non-uniform Grid
Chen	1171.511	1084.5942
Lu	1142.0571	1029.2975



(a) Time Response when $\alpha_l = 0.57$ (b) Time Response when $\alpha_l = 0.61$

Figure 4.17 – Phase Portrait (attractor) of fractional Lu system with fractional order α_l equal to 0.56, 0.57, 0.6, and 0.61

4.6 Conclusion

In this chapter, a type of innovative numerical calculation method for fractional system solutions is discussed. The basic idea is to construct a non-uniform grid as the step size of the calculation for the system states within every iteration. Two different non-uniform grids are constructed and discussed based on chaotic skew-tent map. The states of fractional Chen and Lu systems are calculated by the methods, and their simulations have been carried out. The comparison between the classical uniform calculation method and the proposed non-uniform grid methods proved positive impacts on the chaoticity of the behavior of the fractional system. The enlarged chaotic range for fractional derivative orders has been obtained for the systems applying the proposed non-uniform grid methods. Positive impacts on chaoticity of fractional systems in terms of parameters have been acquired using the non-uniform grid methods. Less time was consumed compared to the classical ABM method.

Chapter 5

Stream cypher based on fractional chaotic pseudo-random number generator

5.1 Introduction

In this chapter, we innovatively propose an efficient fractional chaotic pseudo-random number generator (FCPRNG) and apply it to design a secure stream cipher. The FCPRNG consists of two 3D fractional chaotic systems, namely fractional Chen and Lu system, and one fractional generalized double-humped logistic map (FGDHL). The multi-dimensional fractional systems are calculated on a non-uniform grid obtained through the introduction of a chaotic skew-tent map as discussed in the previous chapiter. Three exclusive-or (XOR) operations are performed to mix the outputs of fractional chaotic systems and map. The final output of the FCPRNG after the XOR operations works as the key stream of the stream cipher. The efficacy and security of it is tested and analyzed through its image encryption performance.

In the following, the structure of the designed FCPRNG and the stream cipher are discussed in Section 5.2. We then give the performance analysis of the stream cipher in Section 5.3. To be more specific, the statistical and randomness analysis for the proposed FCPRNG are discussed in Section 5.3.1; the cryptanalytic analysis of the cipher is presented in Section 5.3.2. After this, we draw the conclusion in the Section 5.4.

5.2 Proposed stream cipher

We give the structure of the proposed FPCRNG in Fig.5.1. In the figure, Fst[Xst(n)] denotes the classical skew-tent map as introduced in Section 4.2, holding the form as given

in equation 5.1. Fg[Xg(n)] stands for the FGDHL, and has the form as given in equation 5.2.

$$\mathrm{Xst}\,(n+1) = \mathrm{Fst}\,[\mathrm{Xst}\,(n)] = \begin{cases} \dfrac{\mathrm{Xst}\,(n)}{p}, 0 < \mathrm{Xst}\,(n) \leq p, \\ \dfrac{1-\mathrm{Xst}\,(n)}{1-p}, \mathrm{Xst}\,(n) > p. \end{cases} \tag{5.1}$$

$$\begin{aligned} &\mathrm{Xg}\,(n+1) = \mathrm{Fg}[\mathrm{Xg}\,(\mathrm{n})] \\ &= \mathrm{Xg}\,(n) + \frac{r^{\beta_g}}{\Gamma\,(1+\alpha)}\rho(\mathrm{Xg}\,(n)-c)^2\left(c^2 - (\mathrm{Xg}\,(n-1)-c)^2\right). \end{aligned} \tag{5.2}$$

$\mathrm{F_l}[\mathrm{Xst}(n), \mathrm{X_l}(n)]$ and $\mathrm{F_c}[\mathrm{Xst(n)}, \mathrm{X_c(n)}]$ in the figure represent the fractional Lu's and Chen's system calculated on the non-uniform grid whose grid space is determined by the outputs of the skew-tent map. The explicit illustration is given in Section 4.3.1. We recall here again the formula (equation (5.3)-(6.6)) for the calculation of the states of fractional Chen system. The same expressions are adopted for the fractional Lu system, only with corresponding Lu system's denotations.

$$\begin{aligned} \mathrm{X_c}(n+1) &= \mathrm{F_c}[\mathrm{Xst}(n+1),\ \mathrm{X_c}(n)] \\ &= \mathrm{X_c}(0) + \frac{h(n)^{\alpha_c}}{\Gamma(\alpha_c)+2} f_c(\mathrm{X_c}^{Pr}(n+1)) + \frac{h(n)^{\alpha_c}}{\Gamma(\alpha_c+2)}\sum_{j=0}^{n} a_{j,n+1} f_c(\mathrm{X_c}(j)), \end{aligned} \tag{5.3}$$

$$h(n) = h \times (i+1) \quad \text{if} \quad \mathrm{Xst}(n) \in [0.2 \times i, 0.2 \times (i+1)[, \quad i = 0,1,2,3,4. \tag{5.4}$$

$$\mathrm{X_c}^{Pr}\,(n+1) = \mathrm{X_c}\,(0) + \frac{1}{\Gamma\,(\alpha_c)}\sum_{j=0}^{n} b_{j,n+1} f_c\,(\mathrm{X_c}\,(j))\,, 0 < \beta_c < 1. \tag{5.5}$$

$$a_{j,n} = \begin{cases} n^{\alpha_c+1} - (n-\alpha_c)(n+1)^{\alpha_c}, & \text{if} j = 0, \\ (n-j+2)^{\alpha_c+1} + (n-j)^{\alpha_c+1} - 2(n-j+1)^{\alpha_c+1}, & \text{if} 1 \leq j \leq n, \\ 1, & \text{if} j = n+1. \end{cases} \tag{5.6}$$

$$b_{j,n+1} = \frac{h(n)^{\alpha_c}}{\alpha_c}\left((n+1-j)^{\alpha_c} - (n-j)^{\alpha_c}\right).$$

As shown in the previous sections, the states of the fractional systems discussed in the paper are not uniformly distributed. Therefore, to acquire the final output that satisfies the distribution requirement for the pseudo-random generator (uniformly distributed), we applied some adjustments to the outputs of the fractional systems. The states of the Chen's and Lu's 3D systems, $\mathrm{X_c}(n)$ and $\mathrm{X_l}(n)$ with decimal values are injected into the

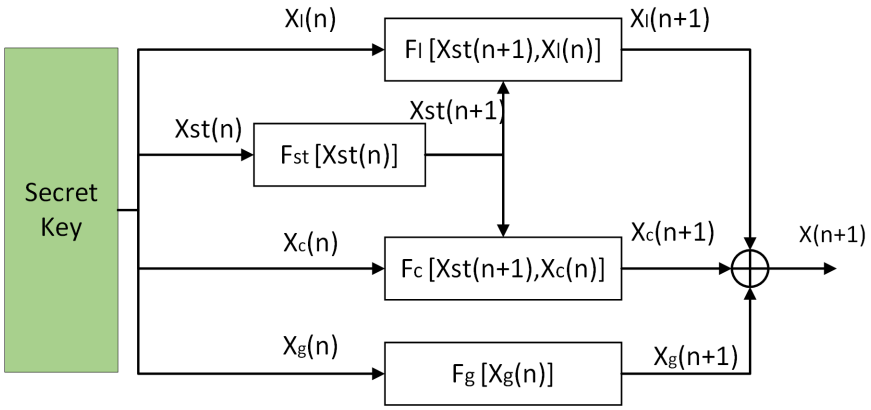


Figure 5.1 – FCPRNG structure

interval of $[-10, 10]$ by a folding mechanism as given below,

$$\mathrm{X_c}\left(n\right) = \begin{cases} 10 - \left(\mathrm{X_c}\left(n\right) - 10\right), & \text{if } \mathrm{Xc}\left(n\right) \geq 10, \\ -10 - \left(\mathrm{X_c}\left(n\right) - (-10)\right), & \text{if } \mathrm{X_c}\left(n\right) \leq -10, \\ \mathrm{X_c}\left(n\right), & \text{else.} \end{cases} \tag{5.7}$$

$$\mathrm{X_l}\left(n\right) = \begin{cases} 10 - \left(\mathrm{X_l}\left(n\right) - 10\right), & \text{if } \mathrm{X_l}\left(n\right) \geq 10, \\ -10 - \left(\mathrm{X_l}\left(n\right) - (-10)\right), & \text{if } \mathrm{X_l}\left(n\right) \leq -10, \\ \mathrm{X_l}\left(n\right), & \text{else.} \end{cases} \tag{5.8}$$

The states of FGDHL $\mathrm{X_g}(n)$ are truncated with a window of $[-0.15, 0.7]$. To evaluate the performance of the proposed generator and to use it in the following stream cipher, each decimal value of the systems states is injected into 32-bits values using following equations (5.9)-(5.11) and then converted into 32 bits binary values using MATLAB dec2bin function.

$$\mathrm{X_c}(n) = \frac{\mathrm{X_c}(n)}{10 - (-10)} \times (2^N - 1), N = 32. \tag{5.9}$$

$$\mathrm{X_l}(n) = \frac{\mathrm{X_l}(n)}{10 - (-10)} \times (2^N - 1), N = 32. \tag{5.10}$$

$$\mathrm{X_g}(n) = \frac{\mathrm{Xg}(n)}{0.7 - (-0.15)} \times (2^N - 1), N = 32. \tag{5.11}$$

The generator's final output $\mathrm{X}(n)$ is obtained by performing XOR operations among the first component outputs of fractional Chen's, fractional Lu's systems($x_{c,1}$,$x_{l,1}$ respec-

tively) , and the outputs of the FGDHL system following the equation below.

$$\mathrm{X}(n) = \mathrm{X_l}(n) \oplus \mathrm{X_c}(n) \oplus \mathrm{X_g}(n) \tag{5.12}$$

The secret key of the proposed FCPRNG is composed of the initial conditions of the skew tent map, fractional Chen and Lu systems (Xst(0), $\mathrm{X_c}(0)$, and $\mathrm{X_l}(0)$, respectively), and the parameters of the adopted systems((a_c, b_c, c_c) for fractional Chen system, (a_l, b_l, c_l) for fractional Lu system, ρ for FGDHL, p for skew tent map), and three fractional derivative orders (α_c, α_l, α_g).

A stream cipher based on the proposed FCPRNG for image encryption use is proposed applying the designed FCPRNG. The stream cipher is achieved by performing XOR operations between the plain image and the key stream generated by the FCPRNG bit by bit.

Several colored and grey images were encrypted by the stream cipher. We analyze in the following the performance of this stream cipher, applying tests that are currently used for quality evaluation of image encryption.

5.3 Performance analysis

In the following, the performance of the proposed FCPRNG and the stream cipher based on it is given and analyzed.

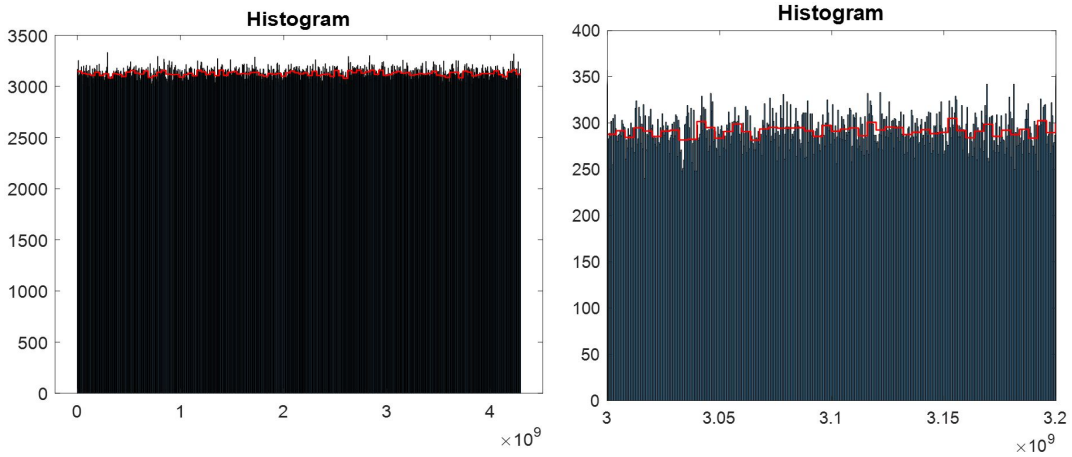
5.3.1 FCPRNG performance analysis

As introduced in Section 1.4.3, a well-designed CPRNG should give outputs with good statistical property and pseudo-randomness. Therefore, the histogram, χ^2 test, and NIST tests are employed to explore and verify the cryptographic and randomness performances of the proposed PCNG.

To do the NIST test, 100 chaotic sequences with 10^6 bits are needed. So we adopt 100 pairs of different secret keys to generate 100 different sequences with 31250 samples (in total $31250 \times 100 \times 32 = 100000000$ bits) for the performance evaluation. The parameters and fractional derivative orders of the systems are chosen randomly using the MATLAB random generation function **rand**. The ranges for all the parameters, derivative orders, and initial conditions of systems (secret key) employed for the proposed FCPRNG is given in Table. 5.1.

Table 5.1 – Keys and their ranges

Keys	**Ranges**	**Keys**	**Ranges**	**Keys**	**Ranges**	**Keys**	**Ranges**
$x_{c,1}(0)$	[-15,15]	$x_{c,2}(0)$	[-15,15]	$x_{c,3}(0)$	[0,30]	$x_{l,1}(0)$	[-15,15]
$x_{l,2}(0)$	[-15,15]	$x_{l,3}(0)$	[0,30]	Xst(0)	(0,1)	p	(0,1)
$\mathrm{X_g}(0)$	(0,0.3)	α_c	[0.75,1]	α_l	[0.75,1]	a_c	[35,40]
b_c	[1.5,3.5]	c_c	[23,28]	a_l	[35,40]	b_l	[3,8]
c_l	[20,25]	tr	[1000,1500]				



(a) Histogram of 3125000 samples (b) Partial histogram of the samples

Figure 5.2 – Histogram of FCPRNG outputs

1. **Histogram**

   The histogram of these 3125000 samples whose values are in the interval of $[0, 2n-1]$ ($n = 32$) is given in Fig.5.2. In total, 1000 statistical classes are chosen. The Fig.5.2a shows that the outputs of the proposed FCPRNG are uniformly distributed. To better observe the distribution, the histogram for the outputs ranging from $[3\times10^9, 3.2\times10^9]$ is also given in Fig.5.2b. It can be seen that this zoomed-in partial histogram holds a form that is qualitatively similar to its preceding histogram depicting the distribution of all the samples.

2. χ^2 **test**

   The χ^2 test which has been introduced in Section 1.4.3 is also applied to further validate the hypotheses of the uniformity of the FPCRNG outputs. To do the test, a null hypothesis is to be established, and a significance level α signifying the probability of rejecting the null hypothesis ($\mathrm{H_0}$) while it is true, is chosen. If one obtains a test statistic (experimental value) smaller than the Critical χ^2 Value

(CV) under the degree of freedom of the samples for the χ^2 test with a significance level of α, then the H_0 is considered to be true and validated.

We assume that hypothesis H_0 is that the outputs of the generator are uniformly distributed (numbers of samples in each of the 1000 classes are identical). The test statistic (experimental value) of Chi-Square is calculated by the following equation,

$$\chi_e^2 = \sum_{i=0}^{N_c-1} \frac{(O_i - E_i)^2}{E_i} \tag{5.13}$$

where N_c is the number of classes chosen, O_i is the number of samples in the i-th class that are observed and E_i represents the number of samples that are expected for a uniform distribution. Knowing that for a significant level of $\alpha = 0.05$, the critical Chi-square value for 1000 classes (degree of freedom = 1000-1 = 999) equals 1073.6427. Then, with an test statistic χ_e^2 equal to 1021.0521, the H_0 can not be rejected, thus, the uniformity of the generated sequence is validated.

3. **NIST test suite**

   The NIST (National Institute of Standard and Technology) test, as been introduced in Section 1.4.3, is a suite of 15 different bitwise tests used to investigate and measure the randomness of a sequence [Bassham, 2010]. A P-value greater than 0.01 indicates that the sequence tested is random with a confidence level of 0.99(99%).

   The NIST test suite result for 10^8 bits ($100 \times 31250 \times 32$) is shown in Table. 5.2. It shows that the sequence generated by FCPRNG passes all 15 tests successfully with P-values greater than 0.01.

5.3.2 Security analysis of the stream cipher

1. **Key space analysis**

   As mentioned in Section 1.4.4, to be able to resist brute-force attacks, the key space for a cryptosystem must be large enough. A secure cryptosystem should have a key space equal to or greater than 2^{128} as stipulated in [Ahmad, 2021].

   For the proposed stream cipher , the secret key consists of all the inputs of the FCPRNG: the initial conditions and parameters for all the systems employed, and the fractional orders of the fractional chaotic systems. Thus, the key space is composed by the parameters $(a_c, b_c, c_c, a_l, b_l, c_l, \rho, \beta_c, \beta_l, \beta_g, p)$, and the initial conditions

Table 5.2 – NIST test results

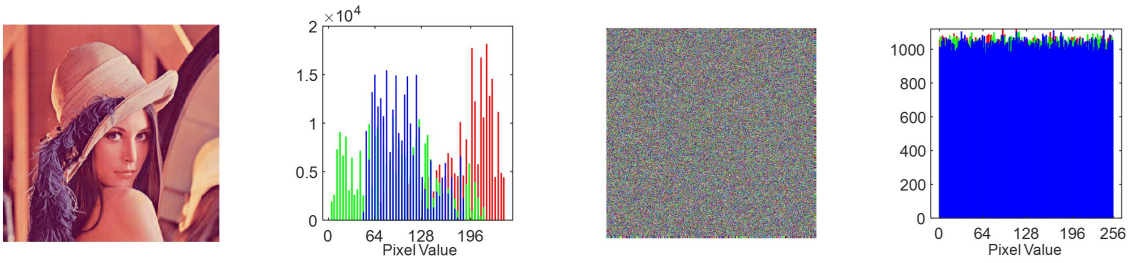
Test	P-value	Proportion
Frequency test	0.122	99.000
Cumulative-sum test	0.117	99.000
Longest-run test	0.019	99.000
FFT test	0.172	97.000
Overlapping-templates	0.760	99.000
Approximty entropy	0.679	98.000
Random-excursions-variant	0.334	99.171
Serial test	0.403	99.500
Runs test:	0.868	100.000
Rank test	0.419	99.000
Nonperiodic-templates	0.518	99.041
Universal	0.145	100.000
Random-excursions	0.464	99.440
Linear-complexity	0.740	98.000
Block-frequency test	0.679	99.000

$\mathbf{X_c}(0) = (x_{c,1}(0), x_{c,2}(0), x_{c,3}(0))$, $\mathbf{X_l}(0)(x_{l,1}(0), x_{l,2}(0), x_{l,3}(0))$, Xg(0), Xst(0), and the number of states deserted for the transient (tr) as given in Table. 5.1. With a computation precision of 10^{-14}, the key space is greater than 2^{128}. Hence, the stream cipher based on the proposed FCPRNG can resist the brute-force attack.

2. **Histogram and χ^2 test**

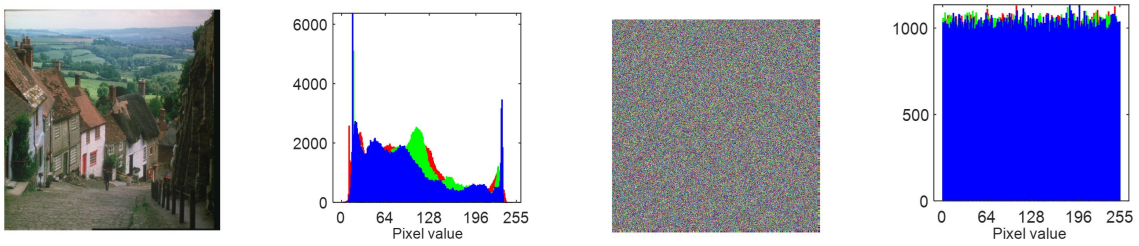
For image encryption, the pixel values of the ciphered image should follow a uniform distribution to resist the statistical attack. Thus, to evaluate the performance of the stream cipher in terms of the pixel value distribution after the encryption, the histogram and χ^2 test are employed.

In Fig. 5.3 and 5.4, the histograms of two different benchmark color images 'Lena' and 'Goldhill' are given. It can be seen from Fig. 5.3d and 5.4d that the ciphered images Fig. 5.3c and 5.4c have a uniform distribution in every color layer. Fig. 5.5d and 5.6d illustrate the encryption results for the grey images, 'boat' and a all-white image with black broader. The results also confirm the uniform distribution of pixel values after encryption.



(a) Lena (b) Lena Histogram (c) Cipher image (d) Cipher histogram

Figure 5.3 – Lena Results



(a) Goldhill (b) Histogram (c) Cipher image (d) Cipher Histogram

Figure 5.4 – Goldhill Results

We recall again the equation for test statistic calculation of the χ^2 test, ,

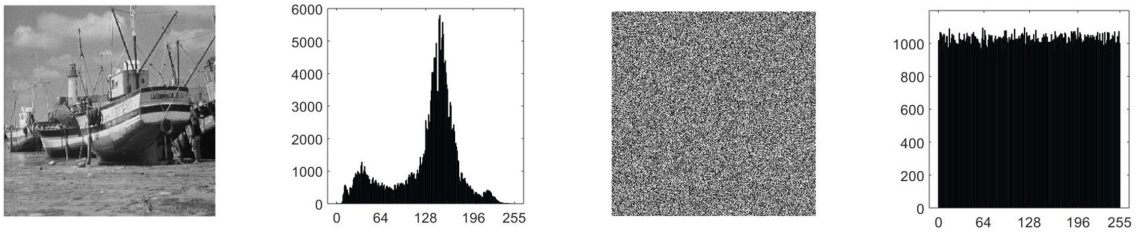
$$\chi_e^2 = \sum_{i=0}^{N_c-1} \frac{(O_i - E_i)^2}{E_i}. \tag{5.14}$$

Unlike the χ^2 test done for the FCPRNG output, to prove the uniform distribution of pixel values in cipher image, the number of classes and E_i are $N_c = 256$ (pixel value levels), $E_i = ImageSize/N_c$. The critical χ^2 value is then acquired and equal to 293.2478 (degree of freedom $= 256 - 1 = 255$) with a significance level of $\alpha = 0.05$.

The test statistic χ_e^2 for several benchmark image calculated by equation 5.14 are given in Table. 5.3. With all the experimental value smaller than the critical value, the tests confirm that the pixel values of the ciphered images are uniformly distributed.

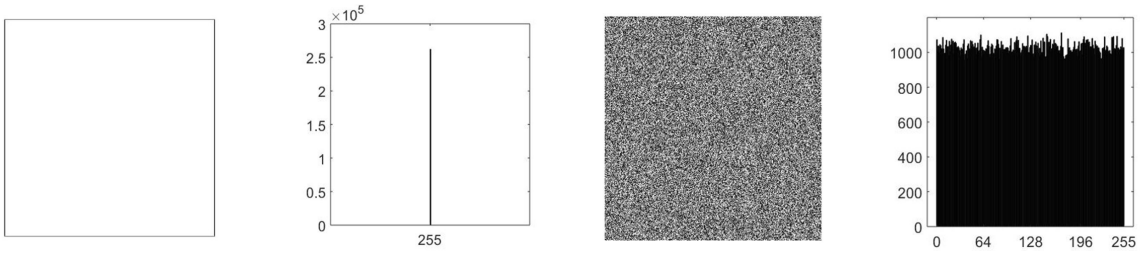
3. **Entropy test**

   In information theory, the entropy of a variable represents the average level of uncertainty inherent in the variable's possible outcome. From the aspect of image encryption, entropy can be used to evaluate the randomness of the image pixel



(a) Boat (b) Histogram (c) Cipher image (d) Cipher Histogram

Figure 5.5 – Boat Results



(a) White (b) Histogram (c) Cipher image (d) Cipher Histogram

Figure 5.6 – White Results

value and works as an indicator to estimate whether the cipher algorithm is robust or not.

If taking the pixel value as the variable, for the cipher algorithm to be robust, the occurrence probability, hence, the entropy, of different pixel values, should be equal or at least almost the same. The information entropy of the ciphered image is calculated by the following equation,

$$H(C) = \sum_{i=0}^{Q-1} Pro(c_i) \times \log_2 \frac{1}{Pro(c_i)} \tag{5.15}$$

where $H(C)$ stands for the entropy of the cipher image; Q represents the number of levels for pixel value ($Q = 256 = 28$); and $Pro(c_i)$ is the occurrences of c_i in each level ($i = 1, 2, \ldots, 256$). In the ideal case, for a well-ciphered image, each pixel value level of the image possesses equal occurrence probability $Pro(c_i)$, which is equal to $1/Q = 2 - 8$. Thus, the information entropy is given as follows,

$$H(C) = \sum_{i=0}^{Q-1} 2^{-8} \times \log_2 256 = 8 \tag{5.16}$$

Image	Chi-square	Entropy (H(P))	Entropy (H(C))	Mean HD
Lena Gray	248.5039	7.4116	7.9973	50.0118
Lena rgb	257.1031	5.6822	7.9998	49.9993
Baboon	254.2773	7.7073	7.9991	49.9861
Black	255.0596	0	7.9993	50.0010
White	262.4511	0	7.993	50.0014
Goldhill	258.0690	7.6220	7.9998	50.0023
Boat	257.1068	7.1914	7.9993	49.9944

Table 5.3 – Results of Chi-square and entropy test

The entropy test is performed on 7 different images. The entropy of each plain image ($H(P)$) and its cipher image ($H(C)$) are obtained by evaluating the average entropy over 50 cipher images encrypted by different secret keys. The results are given in Table 5.3. It can be seen that the average information entropy of the cipher image for all 7 tested images is close to the ideal value 8.

4. **Key sensitivity test**

   For a cipher stream to be robust, it must hold high sensitivity to the secret key. This can be evaluated through the calculation of Hamming distance (HD) between two ciphered images C_1 and C_2, which are obtained from one same plain image by changing the secret key of the stream cipher. The Hamming distance between these two cipher images is calculated as follows,

$$HD\left(C_1, C_2\right) = \frac{1}{lb} \sum_{k=1}^{lb} C_1\left[k\right] \oplus C_2\left[k\right] \tag{5.17}$$

   where lb is the bit length of the image.

   50 different secret keys are used for this experiment, and the average HDs given in Table. 5.3 show that for each pair of cipher images, the probability of bit changes is close to the optimal value of 50%. This proves that the stream cipher is sensitive to the secret key.

5. **Correlation analysis**

   The correlation between pixels is another feature tested to evaluate the security of the cryptosystem. A secure cryptosystem should break the high correlation between the pixels of the plain image. For the plain image and its corresponding cipher image, 8000 different pairs of adjacent pixels are selected in horizontal,

Image	Plain image			Ciphered image		
	Hor-D	**Ver-D**	**Dia-D**	**Hor-D**	**Ver-D**	**Dia-D**
Lenna Grey	0.9458	0.9727	0.9217	-0.0035	-0.0030	-0.0056
Lenna rgb	0.9750	0.9852	0.9652	-0.0011	-0.0012	-0.0029
Baboon	0.9538	0.9384	0.9175	-0.0005	-0.0025	0.0004
Goldhill	0.9775	0.9762	0.9601	0.0014	0.0039	0.0016
Boat	0.9385	0.9718	0.9227	0.0014	-0.00004	0.0001

Table 5.4 – Correlation results for different images

vertical, and diagonal directions respectively, to evaluate the correlation properties of the images. The correlation coefficient is calculated by the equation below.

$$\rho_{xy} = \frac{\sum_{i=1}^{N_p} \left[(x_i - \overline{x}) (y_i - \overline{y}) \right]}{\sqrt{\sum_{i=1}^{N_p} (x_i - \overline{x})^2} \sqrt{\sum_{i=1}^{N_p} (y_i - \overline{y})^2}} \tag{5.18}$$

For each image, the plain image is encrypted using 50 different secret keys. The correlation property of the ciphered image is obtained by averaging the correlation coefficients over these 50 different ciphered images. Table. 5.4 shows the correlation coefficients for 5 different images in horizontal, vertical, and diagonal directions. From the table, it can be observed that the correlation coefficients of the cipher images in all directions are around 0. This means that there is almost no correlation between pixels in the cipher images. The correlation results of the benchmark image 'Baboon' and the grey image 'airfield' given in Fig. 5.7 and Fig. 5.8 also visually confirm that the correlation between pixels in plain images is broken after encryption.

5.4 Conclusion

In this chapter, we designed, implemented and evaluated a stream cipher based on an innovative FCPRNG adopting two 3D fractional chaotic systems and one FGDHL map. To solve numerically the employed 3D fractional systems, we applied the non-uniform grid calculation method with Grid 1 discussed in Chapter 3 based on the fractional ABM Corrector-Predictor calculation method. With the non-uniform grid calculation method, greater range for the fractional derivative orders has been acquired and extra parameters have been introduced as the secret key. Thus greatly enlarged the key space size. The complexity of the FCPRNG structure also led to greater security level of the stream

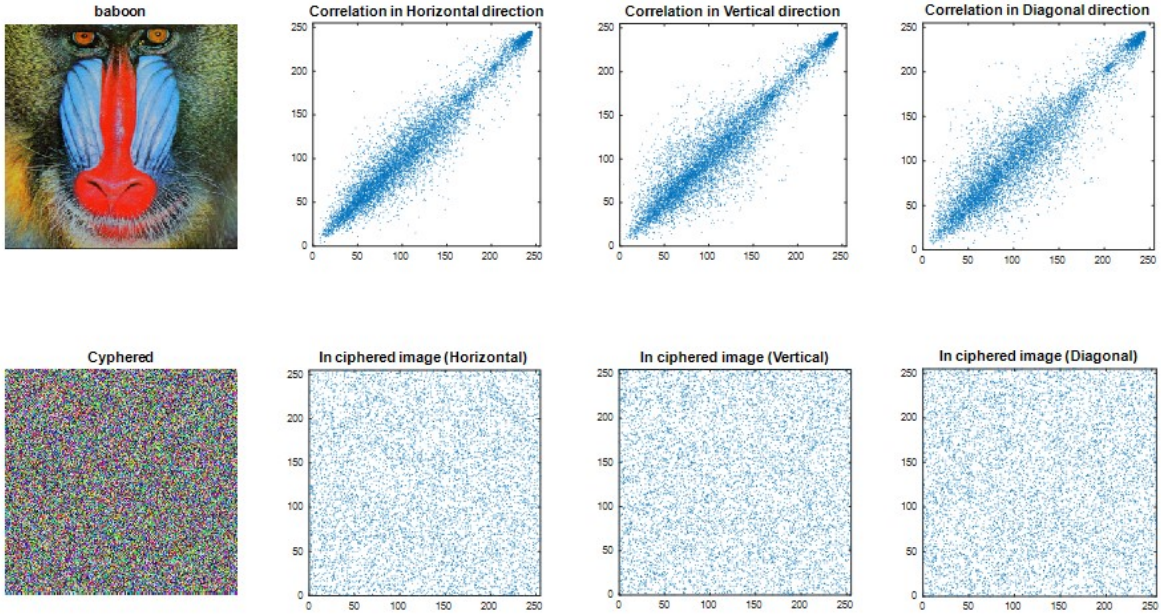


Figure 5.7 – Correlation in horizontal, vertical and diagonal directions of Baboon image

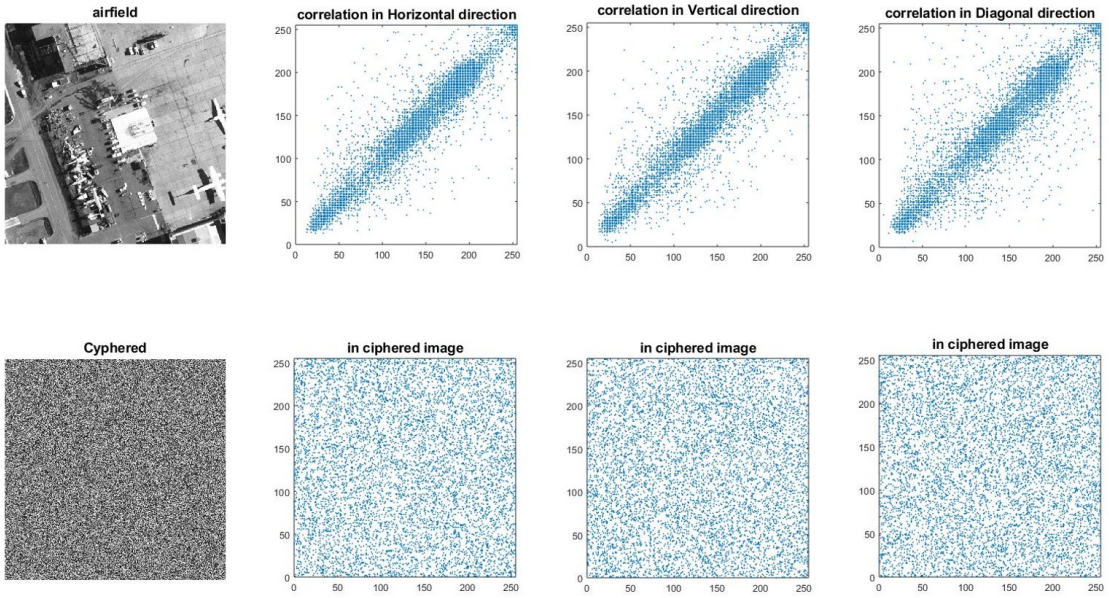


Figure 5.8 – Correlation in horizontal, vertical and diagonal directions of Airfield image

cipher.

The statistical analysis and the NIST test results of the proposed FCPRNG show that it possesses excellent characteristics in generating pseudo-randomness outputs. In addition, the experiment and tests conducted on the stream cipher show that the security of it is achieved, thus, confirms the excellent cryptographic performances of the proposed FCPRNG possesses.

Chapter 6

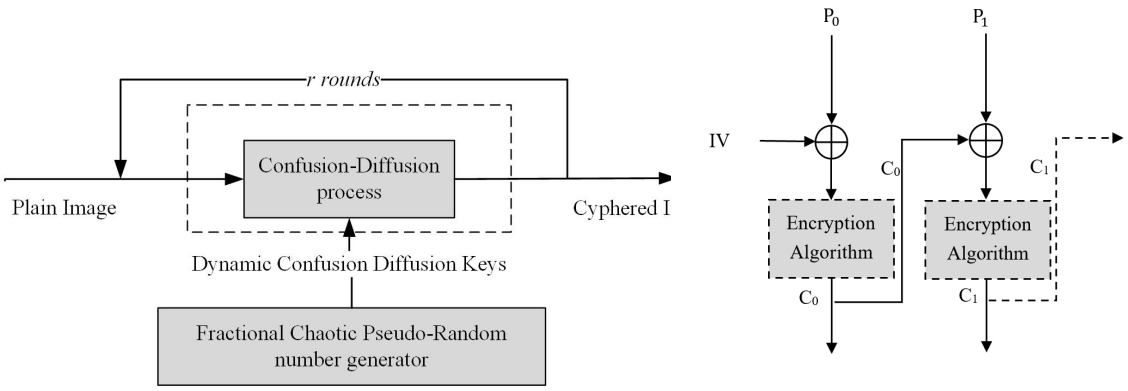
Image encryption based on fractional Chaotic pseudo-random number generator and DNA encoding and decoding method

6.1 Introduction

In this chapter, we propose, implement and evaluate a secure color image encryption cryptosystem based on DNA encoding and decoding method. The cryptosystem employs an FCPRNG to generate its key stream for its encryption and decryption algorithms and is realized through a block cipher in Cipher Block Chaining (CBC) mode.

For the designed FCPRNG, the same fractional chaotic systems and map as in chapter 5 are adopted. However, unlike the stream cipher, we employed our proposed non-uniform grid 2 as discussed in 4.3.2 to solve the 3D fractional chaotic systems. For the cipher algorithm, we adopt the DNA encoding and decoding, together with a 2D modified cat map for the confusion of the image pixels. The diffusion is achieved through the employment of an integer-order finite precision discrete logistic map.

In the following sections, Section 6.2 discuss the design of the proposed color-image cryptosystem. In particular, 6.2.2 introduces the principle of DNA encoding and decoding method; 6.2.3 illustrates the FCPRNG designed for the cryptosystem; the cipher is then explained in 6.2.4 and 6.2.5. The performance of the FCPRNG and the security of the cryptosystem are analyzed and displayed in Section 6.3. Finally, we draw our conclusion in Section 6.4.



(a) General block diagram (b) CBC mode encryption structure

Figure 6.1 – General Encryption concept

6.2 Proposed color-image cryptosystem

6.2.1 General concept of the encryption scheme

For the proposed encryption scheme, we employ a CBC mode block cipher of size $b_s = 1024$ (32*32 pixels) based on the encryption scheme discussed in [Farajallah, 2016] and a dynamic DNA encoding and decoding method. The general block diagram of the proposed encryption scheme is given in Fig. 6.1a. r rounds of this confusion and diffusion process are performed on the whole image in order to obtain the secured ciphered image. During each round, the images are encrypted block by block through the CBC mode by the encryption algorithm. The structure of the encryption scheme is illustrated in Fig. 6.1b. In the figure, P_0 stands for the first block of size 1024 bits from the plain image. IV is the Initial Vector pre-generated, C_0 is the first encrypted block. The cyphered block is then working as the initial vector to encrypt the next block etc.

Within each block, the permutation is performed by dynamic DNA encoding and decoding method with a 2D cat map, which will be explained in the following sections. A logistic map is then used to complete the encryption by further diffusing the resulting ciphered block.

6.2.2 DNA encoding/decoding methods

A DNA sequence in the biological sense is composed of four different nucleic acid bases, 'A' (Adenine),' T' (Thymine),' C' (Cytosine),' G' (Guanine). The composition

Table 6.1 – DNA encoding and decoding rules

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

of these nucleic acid bases follows the Watson–Crick principle, where 'A' and 'T' are complementary, and 'C' and 'G' likewise [Watson, 1953].

In DNA computing, other than the binary computation for traditional computers, the information is carried and expressed by these four acid bases 'A' (Adenine), 'T' (Thymine), 'C' (Cytosine), 'G' (Guanine). The transformation between the binary values and the DNA sequence involves the DNA encoding and DNA decoding process. Typically, to map a binary sequence by a DNA sequence, every two bits of binary sequence are grouped and encoded into one of the DNA nucleic acid-bases 'A', 'T', 'C', and 'G' through a specific DNA encoding rule. Its reverse procedure applies the DNA decoding rules and turns a DNA sequence back into a binary sequence. There are 24 (4!) combinations for the mapping of 2-bit binary symbols to DNA bases. Among them, if we consider '00' and '11' as a pair of complement, '01' and '10' another, then only 8 combinations satisfy the above-mentioned Watson-Crick complementary principle, which are shown in the Table.6.1 working as 8 different encoding and decoding rules [Chai, 2019b].

Intuitively, if the same rule has been chosen for both encoding and decoding processes, then the resulting binary value remains unchanged after the processing. Otherwise, the binary value is changed; hence, the original information is masked.

To further explain the application of this in image encryption, we take one 8-bit decimal pixel value '234' as an example to illustrate the DNA encoding process as shown in Fig. 6.2. The decimal value of '234' is first converted to binary bits' 11101010'. Then adopting DNA encoding rule 4, the corresponding DNA sequence 'CTTT' is obtained. Obviously, with different encoding rules, the same value can be transformed into distinguished DNA sequences (With rule 5, '11101010' turns to 'GAAA'). The same sets of rules are adopted for the DNA decoding to turn the DNA bases back to binary values. Take the previously obtained DNA sequence 'CTTT' as an example. If the DNA decoding rule 8 is taken, we obtain an 8 bits binary value of '10000000', whose corresponding decimal value is '128'. This decoding process is also illustrated in Fig.6.2.

For some early works of DNA encoding and decoding methods in cryptography, for

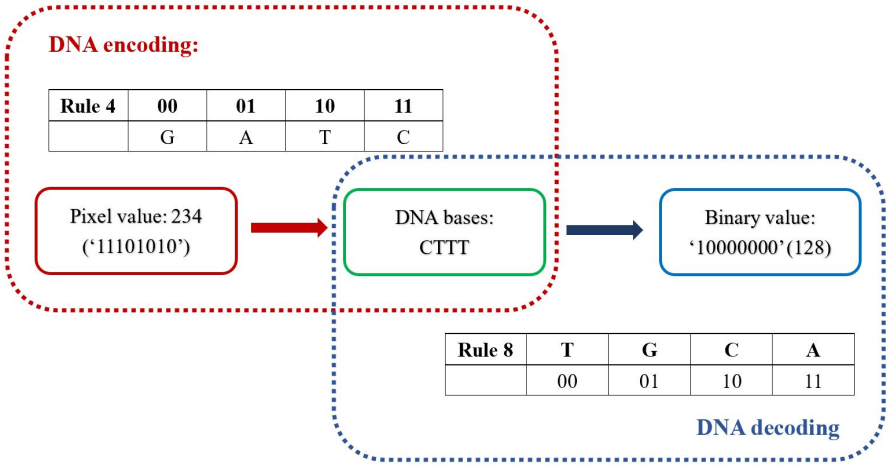


Figure 6.2 – Example of DNA encoding and decoding

instance, [Zhang, 2013a][Wei, 2012], the encoding and decoding rules have been fixed for the whole encryption process. However, it has been argued that an encryption scheme with a fixed rule can be easily detected and broken, thus it is not suitable for the design of cryptosystem [Wang, 2015][Akhavan, 2017]. Therefore, a dynamic DNA encoding and decoding method has been proposed. The principle of this method is to select different DNA encoding rules for the encryption of the plain text, which change dynamically during the whole encoding and decoding process. In this paper, we adopt the FCPRNG's outputs as the dynamic DNA encoding and decoding rules for the encryption of the image.

6.2.3 Proposed FCPRNG structure

The structure of the FCPRNG proposed is given in Fig. 6.3. The systems (control) parameters, together with the initial conditions constitute the secret key. This FCPRNG basically employs the same structure as discussed in the previous chapter. However, two sets of skew-tent maps with different initial conditions and parameters are adopted for the construction of the non-uniform Grids 2 (illustrated in 4.3.2) on which the fractional Chen and Lu systems are calculated on. The use or these Grid 2 further complicated the structure and enlarged the key space size.

In Fig. 6.3, $\mathrm{St}_i[\mathrm{Xst}_i(n-1)]$, $i = 1, 2, 3, 4$ are the skew tent maps used for the non-

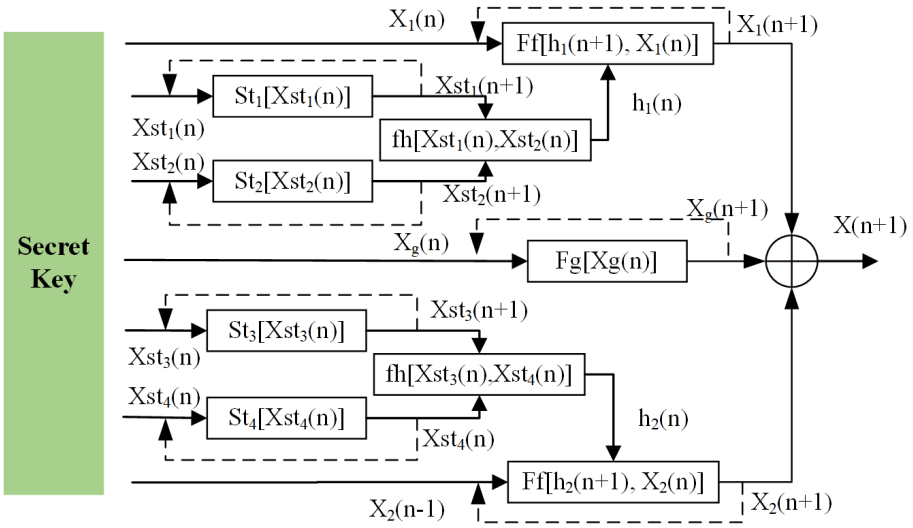


Figure 6.3 – Structure of the designed FCPRNG

uniform grid design holding the form as follows,

$$\begin{aligned} \mathrm{Xst}_i(n+1) &= \mathrm{St}_i[\mathrm{Xst}_i(n)] \\ &= \begin{cases} \dfrac{\mathrm{Xst}_i(n-1)}{p_i}, & 0 < \mathrm{Xst}_i(n-1) \leq p_i \\ \dfrac{1-\mathrm{Xst}_i(n-1)}{1-p_i}, & p_i < \mathrm{Xst}_i(n-1) < 1 \\ \mathrm{Xst}_i(n-1) - 0.05, & \text{otherwise} \end{cases}, \quad i = 1,2,3,4 \end{aligned} \tag{6.1}$$

With the notations (4.12)-(4.14) formulated in Section 4.3.2, the grid spaces $h_1(n)$ and $h_2(n)$ for fractional Chen and Lu systems respectively are obtained through fh[$\mathrm{Xst}_1(n)$, $\mathrm{Xst}_2(n)$] and fh[$\mathrm{Xst}_3(n)$, $\mathrm{Xst}_4(n)$] given by following formula,

$$\begin{aligned} h_1(n) &= \mathrm{fh}[\mathrm{Xst}_1(n), \mathrm{Xst}_2(n)] \\ &= \sum_{i=1}^{5} 1_{\mathrm{A}_i}(\mathrm{Xst}_1(n))\, \mathbf{H}(((2^{32}-1) \times \mathrm{Xst}_2) \,(\mathrm{mod}120) + 1, i) \end{aligned} \tag{6.2}$$

$$\begin{aligned} h_2(n) &= \mathrm{fh}[\mathrm{Xst}_3(n), \mathrm{Xst}_4(n)] \\ &= \sum_{i=1}^{5} 1_{\mathrm{A}_i}(\mathrm{Xst}_3(n))\, \mathbf{H}(((2^{32}-1) \times \mathrm{Xst}_4) \,(\mathrm{mod}120) + 1, i) \end{aligned} \tag{6.3}$$

We then calculate the states of fractional Chen's and Lu's systems, $\mathrm{X}_1(\mathrm{n})$ and $\mathrm{X}_2(\mathrm{n})$, on

the non-uniform grid $h_1(n)$ and $h_2(n)$ applying $\mathrm{Ff}[\mathrm{h_1(n)}, \mathrm{X_2(n-1)}]$ and $\mathrm{Ff}[\mathrm{h_2(n)}, \mathrm{X_2(n-1)}]$. The fomulations have been discussed in previous section, but we recall it here with the corresponding notations for the states of fractional Chen system $\mathrm{X_1}(n+1)$.

$$\begin{aligned}\mathrm{X_1}(n+1) &= \mathrm{F_1}[\mathrm{h}(n+1),\ \mathrm{X_1}(n)] \\ &= \mathrm{X_1}(0) + \frac{h_1(n)^{\alpha_1}}{\Gamma(\alpha_1)+2} f_c({\mathrm{X_1}}^{Pr}(n+1)) + \frac{h_1(n)^{\alpha_1}}{\Gamma(\alpha_c+2)} \sum_{j=0}^{n} a_{j,n+1} f_c(\mathrm{X_1}(j)),\end{aligned} \tag{6.4}$$

$${\mathrm{X_1}}^{Pr}(n+1) = \mathrm{X_1}(0) + \frac{1}{\Gamma(\alpha_1)} \sum_{j=0}^{n} b_{j,n+1} f_1(\mathrm{X_1}(j)), 0 < \alpha_1 < 1. \tag{6.5}$$

$$a_{j,n} = \begin{cases} n^{\alpha_1+1} - (n-\alpha_1)(n+1)^{\alpha_1}, & \text{if} j = 0, \\ (n-j+2)^{\alpha_1+1} + (n-j)^{\alpha_1+1} - 2(n-j+1)^{\alpha_1+1}, & \text{if} 1 \le j \le n, \\ 1, & \text{if} j = n+1. \end{cases} \tag{6.6}$$

$$b_{j,n+1} = \frac{h_1(n)^{\alpha_1}}{\alpha_1} \left((n+1-j)^{\alpha_1} - (n-j)^{\alpha_1}\right).$$

$\mathrm{Fg}[\mathrm{X_g(n+1)}]$ in the figure is the function employed to calculate the states of FGDHL as the same given by equation (5.2).

The parameters and fractional derivative orders of Chen and Lu systems for the FCPRNG have been chosen as follows: $\alpha_1 \in [0.75, 1)$, $\alpha_2 \in [0.75, 1)$, $a_c \in [35, 40]$, $b_c \in [1.5, 3.5]$, $c_c \in [23, 28]$, $a_l \in [30, 35]$, $b_l \in [3, 8]$, $c_l \in [20, 25]$. The control parameters p_1, p_2, p_3, p_4 of the skew tent maps are all in the range of $(0, 1)$; and the initial conditions $\mathrm{Xst_1}(0)$, $\mathrm{Xst_2}(0)$(for the calculation of $h_1(n)$), and $\mathrm{Xst_3}(0)$, $\mathrm{Xst_4}(0)$ (for the calculation of $h_2(n)$) are also in the same range. In addition, the initial condition $\mathrm{X_g}(0)$ for FGDHL is equally adopted as a component of secret key, and it is in the range of $[0, 0.3]$.

It has been demonstrated that only the first component of the state vectors possess positive Lyapunov exponents (LEs) for both Lu and Chen 3D systems, which means that the first component among the three implies the chaotic dynamics of the whole system. (eg. $x_{11}(n)$ for Chen system states $\mathrm{X_1}(n) = [x_{11}(n), x_{12}(n), x_{13}(n)]$). Therefore, we only employ the first state component for further use as an output of the FCPRNG.

After converting $\mathrm{X_1(n)}$, $\mathrm{X_2(n)}$ and $\mathrm{X_g(n)}$ into 32 bits binary values using MATLAB **dec2bin** function, the final outputs of the FCPRNG X(n) are obtained by performing or-exclusive operations (XOR) between the outputs of these three fractional systems. It is worth mentioning that in order to improve uniformity for the output sequence distribution,

we inject the state values of the two fractional 3D systems into the interval of $[-10, 10]$ by a folding mechanism, and the states of FGDHL have been truncated with a window of $[-0.15, 0.7]$ as explained in Chapter 5 and by equation (5.7)-(5.11).

6.2.4 Encryption scheme of the proposed cryptosystem

The proposed encryption scheme is illustrated in Fig. 6.4.

For the first block of 1024 pixels, each pixel $p_0(k), (k = 1, ..., b_s)$ are first XOR-ed with the byte $(iv(k))$ of the initial vector (IV) of the same size given randomly by the function **randi** in MATLAB. Then, a value in the range of 1 to 8 is obtained by converting 3 bits of the generated FCPRNG outputs *KDNAe* to a decimal value. Adding one to this value gives us the dynamic DNA encoding rules $\mathrm{R_e(k)}$ to encode the 8 bits of the XOR-ed pixels. After that, the DNA complementary rules are employed to switch the encoded DNA bases to their complementary ones ('A'-'T', 'C'-'G'). After encoding all the pixels in the block, we obtain a block of DNA bases that has 64×64 bases (One DNA base consists of 2 binary bits, the total number of 1024×8 bits divided by 2, equal to $4096 = 64 \times 64$ DNA bases). To relocate the acquired DNA bases to a new position, the modified 2D cat map discussed in [Farajallah, 2013] is performed on the DNA base level. Up to this point, the confusion of the first block has been accomplished. For the diffusion layer, the permuted DNA bases are decoded to binary bits by the dynamic DNA decoding method, where the decoding rules $\mathrm{R_d(k)}$ are again acquired through FCPRNG outputs *KDNAd* the same way as introduced above for the dynamic DNA encoding. Then, a discrete logistic map of 32 bits is employed to construct the final cyphered output. During this process, every 4 pixels (32 bits) of decoded DNA bases are XOR-ed with the output of the discrete logistic map, and the input of the map is the 32 bits decimal value converted by the 4 pixels of the previously decoded DNA bases.

The encryption process of the second block to the last block, block number B_N, is almost the same(B_N denotes the total number of blocks in the image). However, rather than using the initial vector IV, each pixel $p_l(k), (l = 1, ..., B_N - 1)$ is XOR-ed with the pixel at the same position of the previous cyphered block($c_{l-1}(k)$) to achieve the CBC mode. In addition, the first input of the diffusion based discrete logistic map is acquired by processing the last 4 pixels of the previous cyphered block, namely $c_{l-1}(b_s-3)$, $c_{l-1}(b_s-2)$ and $c_{l-1}(b_s-1)$, $c_{l-1}(b_s)$ (in total 1024 pixels in each block). The four pixels are converted to 8 bits binary values $c^b_{l-1}(b_s - i)$, $(i = 3, 2, 1, 0)$ and form a 32 bits string which is then converted to the decimal value $x_l(0)$ as the input of the discrete logistic map.

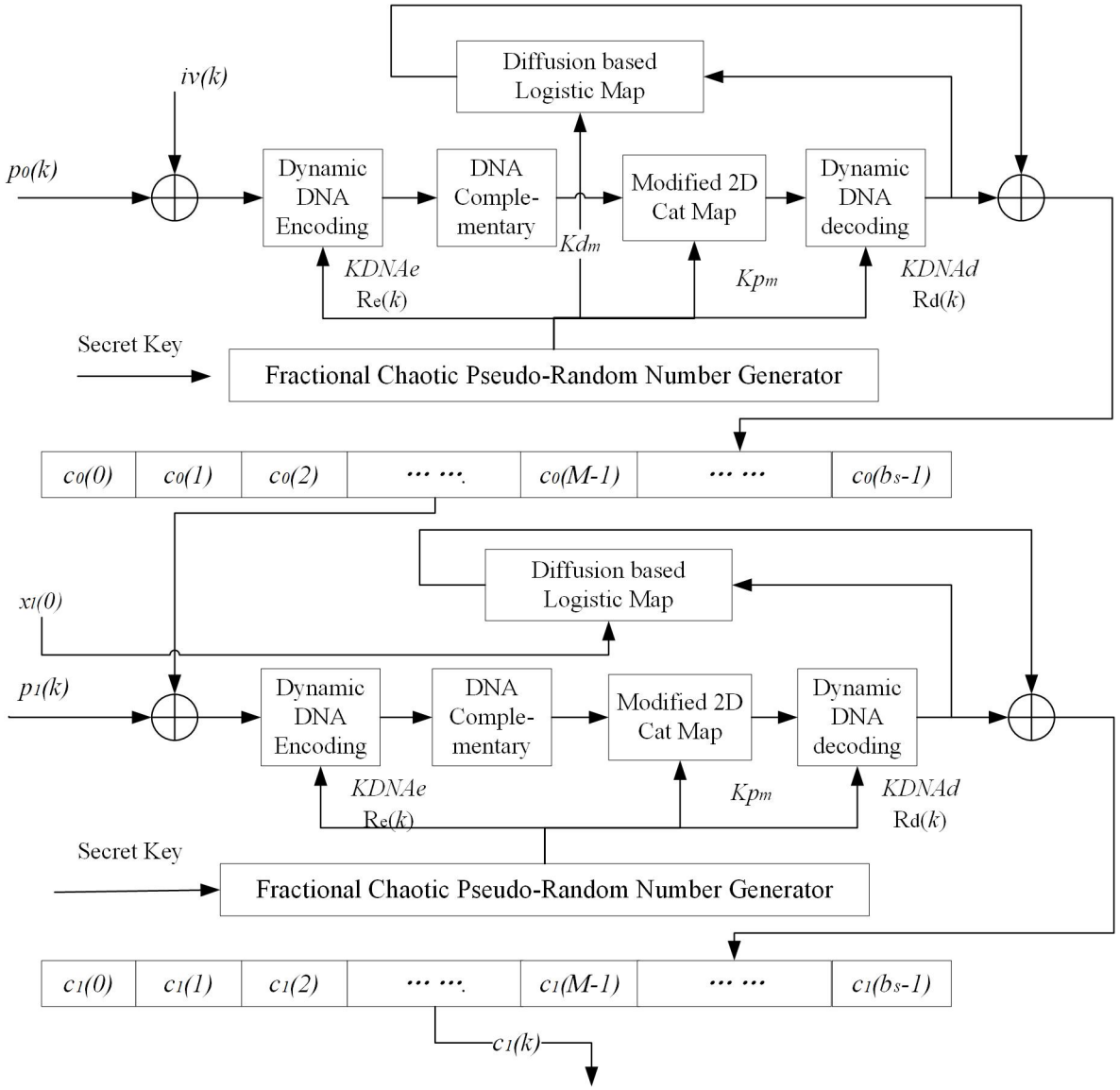


Figure 6.4 – Encryption structure of the cryptosystem

Algorithm 1 Encryption steps

Generate the IV values to encrypt the first block B_1 for $r = 1$
for $r = 1 : rg$ **do**
  for $j = 1 : Bn$ **do**
    if $r = 1$ and $j = 1$ **then**
      $iv(k) = IV(k)$
    else if $r \neq 1, j = 1$ **then**
      $iv(k) = C_{B_N}(k)$
    else
      $iv(k) = C_{j-1}(k)$
    end if
    Calculate $y_j(k) = P_j(k)$ XOR $iv(k)$
    Get $R_e(k)$ by converting 3 bits in $KDNAe$ to decimal value
    Encode $y_j(k)$ by the rule $R_e(k)$ given in Table.6.1 to $y_{j,\text{DNA}}$
    Apply the DNA complementary rules to change $y_{j,\text{DNA}}$ to $y'_{j,\text{DNA}}$
    Reshape $y'_{j,\text{DNA}}$ to a matrix of DNA bases My_{DNA} of size $\sqrt{4b_s} * \sqrt{4b_s}$
    for $i = 1 : \sqrt{4b_s}$ **do**
      for $l = 1 : \sqrt{4b_s}$ **do**
        Calculate (i_{new}, l_{new}) using Equation 6.7
        $My'_{\text{DNA}}(i_{new}, l_{new}) = My_{\text{DNA}}(i, l)$
      end for
    end for
    Reshape My'_{DNA} to a DNA bases string $y'_{j,\text{DNAnew}}$
    Get $R_d(k)$ by converting 3 bits in $KDNAd$ to decimal value
    Decode $y'_{j,\text{DNAnew}}$ to $y_{j,new}$ applying $R_d(k)$
    if $j = 1$ and $r = 1$ **then**
      Get $x_l(0)$ from KMp
    else if $r \neq$ and j=1 **then**
      $x_l(0)$ equals the 32 bits decimal value converted from the binary string consisted of $[c_{B_n}(b_s - 3), c_{B_n}(b_s - 2), c_{B_n}(b_s - 1), c_{B_n}(b_s)]$
    else
      $x_l(0))$ equals the 32 bits decimal value converted from the binary string consisted of $[c_j(b_s - 3), c_j(b_s - 2), c_j(b_s - 1), c_j(b_s)]$
    end if
    Calculate $s(0) = f_l(x_l(0)$ using Equation 6.9
    Convert $y_{j,new}$ to string $y'_{j,new}$ consisted of 32-bits decimal values
    for $t = 1 : b_s/4$ **do**
      if $t = 1$ **then**
        $s(1) = f_l(s(0))$ using Equation 6.9
      else
        Calculate $s(t) = f_l(x(t - 1))$ using Equation 6.9
      end if
      $x(t) = y'_{j,new}(t)$ XOR $s(t)$
    end for
    Get $C_j(k)$ from converting the string of x to 8 bits values
  end for
end for

The 2D map adopted here for the permutation is derived from the Arnold's cat map and has been discussed in [El Assad, 2016]. The equations for the map is defined as follows,

$$\begin{bmatrix} i_{new} \\ j_{new} \end{bmatrix} = mod\left(A_0 \times \begin{bmatrix} i \\ j \end{bmatrix} + \begin{bmatrix} rl + rc \\ rc \end{bmatrix}, \begin{bmatrix} M \\ M \end{bmatrix} \right) \tag{6.7}$$

The matrix A_0 in equation (6.7) is defined as

$$A_0 = \begin{bmatrix} 1 & u \\ v & 1 + u \times v \end{bmatrix} \tag{6.8}$$

The determinant of matrix A_0 is equal to 1, which indicates that each point (i, j) of the square matrix is transferred to a unique point at position (i_{new}, j_{new}). The parameter u, v, rl, rc are the outputs of FCPRNG and form the dynamic key (Kp_m). The use of rc, rl allows to overcome the fixed point problem, encountered in the Arnald Cat map [Qiao, 2020].

It is to be noticed that these dynamic keys u, v, rl, rc are fed by the proposed FCPRNG, and values change for each block.

For the digital implementation, we need a 32 bits discrete logistic map employed for the diffusion, which is expressed as follows [Farajallah, 2016],

$$\mathrm{X}_{k+1} = f_l(\mathrm{X}_k) = \begin{cases} \left\lfloor \dfrac{\mathrm{X}_k \times (2^N - \mathrm{X}_k)}{2^{N-2}} \right\rfloor, & \text{if } \mathrm{X}_k \neq [3 \times 2^{N-2}, 2^N] \\ 2^N - 1, & \text{if } \mathrm{X}_k = [3 \times 2^{N-2}, 2^N] \end{cases} \tag{6.9}$$

where X_{k+1} stands for the new output calculated from its previous one X_k; N is the number of bits representing the integer output of the discrete logistic map. In our proposed cryptosystem, since a 32-bit discrete logistic map is applied, we have $N = 32$.

To determine the optimal value of the rounds r (rg in pseudo-code of **Algorithm 1** and **2**) needed for the encryption scheme to pass successfully the security tests, we evaluate first the confusion and diffusion performance by calculating the Hamming distance (HD) between two ciphered images corresponding to original images with only one bit difference. The calculation is given in the form below,

$$\mathrm{HD}\left(C_1, C_2\right) = \frac{1}{lb} \sum_{k=1}^{lb} C_1\left[k\right] \oplus C_2\left[k\right] \tag{6.10}$$

In the above equation, C_1, C_2 represent two cipher images; lb is the bit length of the image which is calculated by $lb = N_{pix} \times l \times L$. The N_{pix} is the number of pixels in the image; l is equal to 1 if a grey image is encrypted and equals 3 for a colored image; L denotes the number of bits for each pixel.

Three images of different types (grey or colored) with different sizes and features have been tested. Twenty pairs of C_1 and C_2 have been obtained for each plain image by randomly changing one bit of a pixel in the original image. Then, equation (6.10) has been employed to calculate the HDs of the images. The average HD over these 20 pairs of cyphered images shows that when $r = 2$, the HD is already close to 50%. The result indicates that the probability of a bit change is 0.5, which is the optimal value indicating the diffusion is effective.

6.2.5 Decryption proposed of the proposed cryptosystem

The diagram for decryption scheme is given in Fig.6.5. It is the reversed process of the encryption scheme.

For each round, the decryption starts from the last block c_{B_n-1} to the first block c_0. For the block $c_l(l = 1, 2, ...B_n - 1)$, XOR operation is first operated between the cyphered pixels and the output of the logistic map. The input $x_l(0)$ for the first 4 pixels $c_l(k), (k = 1, ..., 4)$, is the decimal value of the 32-bit string converted from the last 4 pixels of the previous cyphered block c_{l-1}; and the inputs for the rest of the cyphered pixels are obtained from performing XOR operation between the 4 current cyphered pixels and 4 previously XORed pixels. A dynamic DNA encoding method applying DNA encoding rules $R_d(k)$ obtained from $KDNAd$ is employed to turn the XORed pixels' values into DNA bases. In the same way as for the encryption process, a DNA base matrix of size 64×64 is constructed. After the matrix has been acquired, the modified 2D cat map with Kp_m generated by FCPRNG and DNA complementary rules permutates the bases. The dynamic DNA decoding process with decoding rules $R_e(k)$ is then employed, followed by XOR operations between the obtained sequence and the cyphered pixels of previous blocks c_{l-1}.

For the first block, c_0 for round $r(\ r \neq 1)$, the first input of the logistic map is acquired by the last four pixels of the whole image. The final XOR operation over the block is performed between the obtained block and the decyphered c_{B_n-1} of last round $r - 1$. Whereas for $r = 1$, the first input $x_l(0)$ is given by Kd_m from FCPRNG, and the XOR is carried out with initial vector IV.

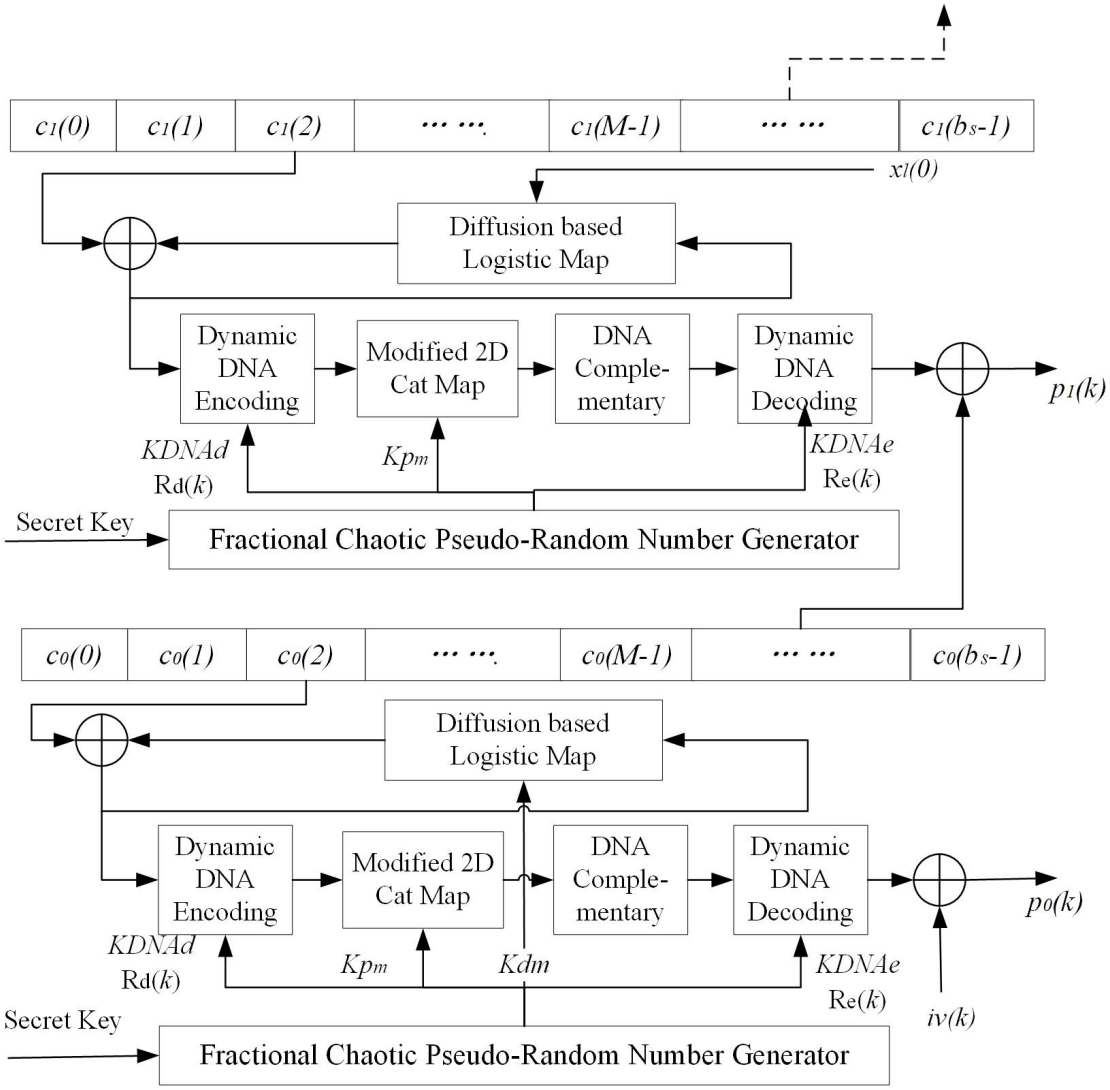


Figure 6.5 – Decryption structure of the cryptosystem

To prove that the proposed FCPRNG and cryptosystem are reliable, a series of well-recognized tests and indicators have been adopted, and the performances have been analyzed in the next section.

6.3 Cryptosystem performance analysis

The following performance and security analyses are divided into two parts. We will 1) discuss the performance of the proposed FCPRNG structure through statistical analysis and NIST (National Institute of Standard and Technology)test [Bassham, 2010]; 2) the

Algorithm 2 Decryption steps

Generate the IV values to decrypt the first block B_1 for $r = 1$
for $r = rg : 1$ **do**
  for $j = Bn : 1$ **do**
    if $j = 1$ and $r = 1$ **then**
      Get $x_l(0)$ from KMp
    else if $r \neq$ and j=1 **then**
      $x_l(0)$ equals the 32 bits decimal value converted from the binary string consisted of $[D_{B_n}(b_s - 3), D_{B_n}(b_s - 2), D_{B_n}(b_s - 1), D_{B_n}(b_s)]$
    else
      $x_l(0))$ equals the 32 bits decimal value converted from the binary string consisted of $[D_j(b_s - 3), D_j(b_s - 2), D_j(b_s - 1), D_j(b_s)]$
    end if
    Calculate $s(0) = f_l(x_l(0))$ using Equation.6.9
    Convert D_j to string Dy_j consisted of 32-bits decimal values
    for $t = 1 : b_s/4$ **do**
      if $t = 1$ **then**'
        Calculate $s(1) = f_l(s(0))$ using Equation.6.9
      else
        Calculate $s(t) = f_l(Dy_j(t - 1))$
      end if
      $x(t) = Dy_j(t)$ XOR $s(t)$
    end for
    Get$Dy_j^{'}$ from converting the string of x to 8 bits value
    Get $R_d(k)$ by converting 3 bits in $KDNAd$ to decimal value
    Encode $Dy_j^{'}$ to $Dy_{j,DNA}^{'}$ applying $R_d(k)$
    Reshape $Dy_{j,DNA}^{'}$ to a matrix of DNA bases MDy_{DNA} of size $\sqrt{4b_s} * \sqrt{4b_s}$
    for $i = 1 : \sqrt{4b_s}$ **do**
      for $l = 1 : \sqrt{4b_s}$ **do**
        Calculate (i_{new}, l_{new}) using Equation.6.7
        $DMy_{\mathrm{DNA}}^{'}(i_{new}, l_{new}) = MDy_{\mathrm{DNA}}(i, l)$
      end for
    end for
    Reshape $DMy_{\mathrm{DNA}}^{'}$ to a DNA bases string $Dy_{j,\mathrm{DNAnew}}^{'}$
    Apply the DNA complementary rules to change $Dy_{j,\mathrm{DNA}}^{'}$ to $Dy_{j,\mathrm{DNA}}$
    Get $R_e(k)$ by converting 3 bits in $KDNAe$ to decimal value
    Decode $Dy_{j,\mathrm{DNA}}$ by the rule $R_e(k)$ given in Table.6.1 to Dy_j
    if $r = 1$ and $j = 1$ **then**
      $iv(k) = IV(k)$
    else if $r \neq 1, j = 1$ **then**
      $iv(k) = P_{B_n}(k)$
    else
      $iv(k) = P_{j-1}(k)$
    end if
    Calculate $P_j(k) = Dy_j(k)$ XOR $iv(k)$
  end for
end for

whole cryptosystem's security has been evaluated through various tests on the encrypted images. It is to be reminded that we use the platform MATLAB with floating-point data for the numerical implementation, which inevitably leads to the degradation of the system's chaoticity. However, the following analysis illustrates that the use of floating-point data does not compromise the security of our proposed cryptosystem.

6.3.1 FCPRNG statistical analysis

1. Distribution tests

a) Histogram of the FCPRNG outputs

Same as for the FCPRNG tested in the previous Chapter, we firstly employ the histogram to evaluate the distribution of the outputs of the proposed FCPRNG. One hundred sequences with one million (10^6) bits are generated with 100 pairs of different secret keys(3125000 samples in total). The distribution of these outputs is shown in Fig.6.6. It can be observed that these 3125000 values are almost uniformly distributed, which meets the uniformity requirement of the pseudo-random number generator.

b) Chi-square test

Apart from the histogram, which can visually illustrate the uniform distribution of the outputs, the Chi-square test is also employed. Same as in the previous Chapter, the significance level α signifying the probability of rejecting the null hypothesis (H_0 : outputs are uniformly distributed) while it is true is chosen as 0.05. The calculation of χ_e^2 experimental value is recalled as given below.

$$\chi_e^2 = \sum_{i=0}^{N_c-1} \frac{(O_i - E_i)^2}{E_i}. \tag{6.11}$$

In equation (6.11), N_c denotes the number of classes, O_i is the number of samples in the i-th class, and E_i represents the number of expected samples for a uniform distribution.

The critical χ^2 value is equal to 1073.6427 with a degree of freedom equal to 999 and a significance level of 0.05. The calculated experimental value χ_e^2 with $N_c = 1000$, $E_i = 3125000/N_c = 3125$ equals to 999. The fact that $\chi^2 < \chi_c^2$ leads to the acceptance of H_0, which in turn confirms that the FCPRNG outputs exhibit a uniform distribution.

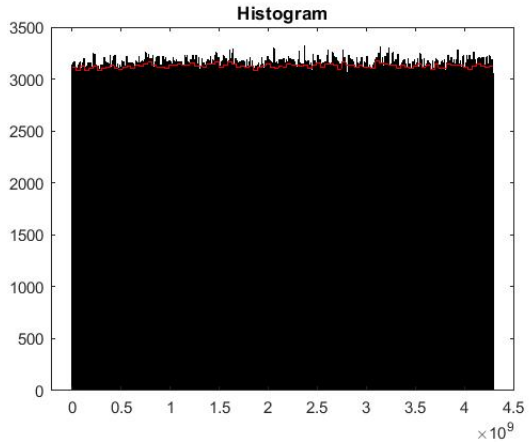


Figure 6.6 – Histogram of 31250000 output samples

The same chi-square test is also employed to test the uniformity of the ciphered images' pixel values later in the cryptosystem's security analysis. However, the number of classes N_c and degree of freedom is different since there are only 256 different possibilities for the pixel values (0-255).

2. Randomness test

1) NIST test suite

As in the previous Chapter, to evaluate the pseudo-randomness of the FCPRNG outputs, we employ the NIST test suite. The tests results are given in Table.6.2. With all the p-values greater than 0.01, and proportions greater than 96.000, the pseudo-randomness of the generated outputs is certified.

After having demonstrated the performances of the proposed FCPRNG, we investigate the performances of the whole cryptosystem.

6.3.2 Cryptosystem security analysis

To evaluate the security and the performance of the proposed encryption algorithm, we encrypted eight benchmark images in black and RGB colored as given in Fig. 6.7 by our proposed encryption algorithm. Some well-recognized tests have been adopted to evaluate the confusion and diffusion performance of the proposed cryptosystem, and the results will be reported in the following.

Table 6.2 – NIST test suite results

Test	P-value	Proportion	Result
Frequency test	0.834	99.000	Pass
Block-frequency test	0.115	100.000	Pass
Cumulative-sums test	0.698	99.000	Pass
Runs test	0.192	98.000	Pass
Longest-run test	0.290	100.000	Pass
Rank test	0.276	98.000	Pass
FFT test	0.016	100.000	Pass
Nonperiodic-templates	0.532	98.885	Pass
Overlapping-templages	0.740	99.000	Pass
Universal	0.290	97.000	Pass
Approximity entropy	0.419	100.000	Pass
Random-excursions	0.437	99.000	Pass
Random-excursions-variant	0.240	99.138	Pass
Serial test	0.382	99.000	Pass
Linear-complexity	0.437	99.000	Pass

(a) Baboon (b) Lena (c) Goldhill (d) Pepper

(e) Airfield (f) Boat (g) Black (h) White with rim

Figure 6.7 – Eight test images with different sizes and features

Statistical property

As mentioned in Chapter 1, there is a kind of attack that aims to crack the cryptosystem by extracting the relationships between the plain and ciphered image and exploiting its statistical weakness of it. For an image cryptosystem to resist statistical attack, it must produce a ciphered image with uniform distribution and no statistical correlation between the images. The following tests have been carried out to evaluate our proposed cryptosystem.

(1) Hisotogarm and χ^2 test

As carried out in Section 5.3.2, we employed the histogram to check the distribution of the plain and ciphered images. In Fig. 6.8, histograms of colored images 'Lena' and 'Goldhill' are illustrated. One can easily observe that the histograms of the original plain images in the three color layers follow certain patterns over the pixel values (0 to 255) but do not satisfy the conclusion for uniform distribution. In comparison, those of the ciphered images possess much uniformly distributed pixel values in all color layers.

The histograms of the grey images 'Airfield' and 'Black' are also given in Fig. 6.9. The same observations can be made, which signifies that the pixel values of the ciphered image satisfy the essential requirement of having a uniform distribution.

The χ^2 test is also adopted here to assess the uniformity of the pixel values. The test statistic χ_e^2 for this test is calculated the same way as given in equation 6.11, but within the number of classes N_c equal to 256 since there are in total 0 to 255, 256 possible pixel values; and $E_i = ImSize/N_c$. The critical value χ_c^2 for the test with $\alpha = 0.05$ and degree of freedom 255 ($N_c - 1$) can be found equal to 293.2478.

The averaged χ_e^2 over the 50 ciphered images obtained from different secret keys for the tested images are given in Table. 6.4. With all the experimental values smaller than 293.2478, the pixel values of cyphered images obtained after encryption are proved to be uniformly distributed.

(2) Correlation analysis

The correlation efficiency between adjacent pixels has also been calculated to evaluate the ability of the encryption scheme to resist statistical attacks. 8000 different pairs of pixels have been chosen in horizontal, vertical, and diagonal directions for the tested plain images.

For each image, the correlation coefficients for different color layers are evaluated over

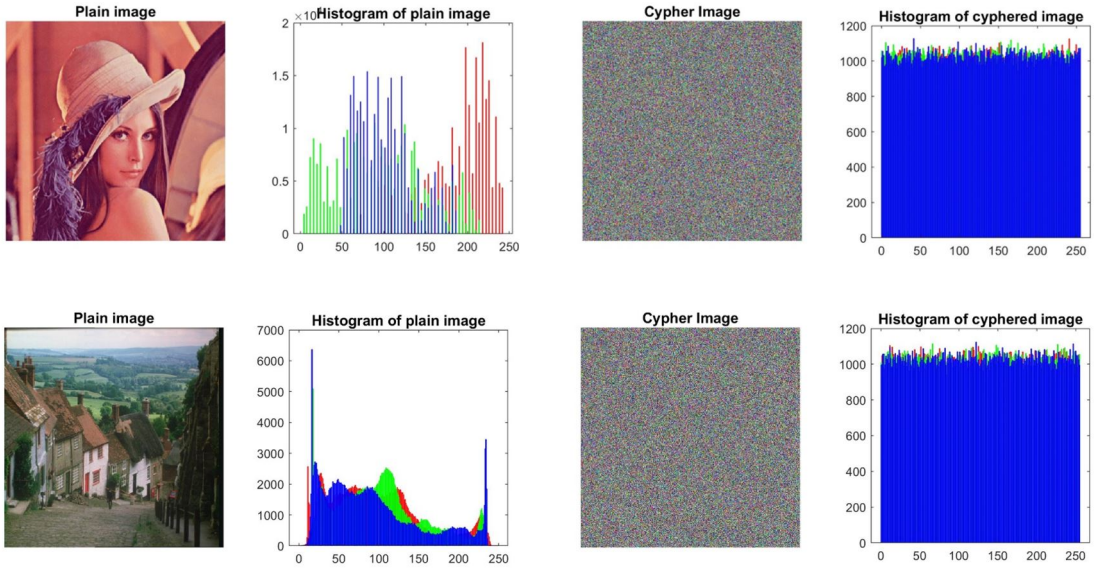


Figure 6.8 – Histogram of plain and cypher colored images

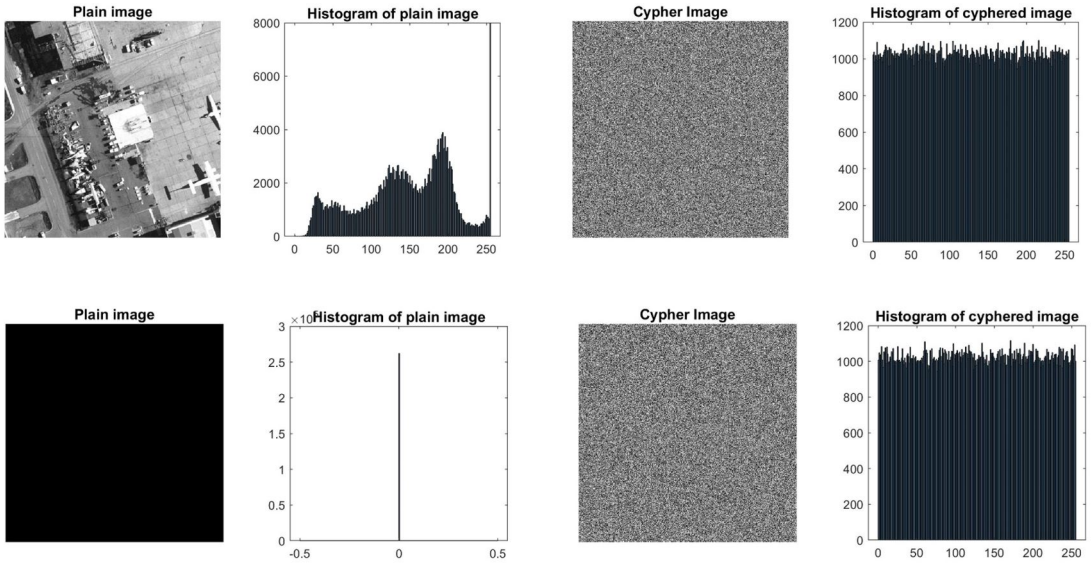


Figure 6.9 – Histogram of plain and cypher colored images

Table 6.3 – Correlation coefficient of several images in different directions

Image	**Size**	**Direction**	**Plain Image**			**Cyphered Image**		
			Correlation coefficient			**Correlation coefficient**		
			Red	Green	Blue	Red	Green	Blue
Baboon	256*256*3	Horizontal	0.9543	0.8845	0.9288	-0.0015	0.0006	-0.0007
		Vertical	0.9343	0.8561	0.9281	-0.0010	0.0011	-0.0031
		Diagonal	0.9175	0.8125	0.8924	-0.0004	0.0001	-0.0018
Lena	512*512*3	Horizontal	0.9753	0.9666	0.9337	-0.0004	0.0007	0.0005
		Vertical	0.9852	0.9803	0.9558	0.0026	-0.0002	0.0005
		Diagonal	0.9653	0.9528	0.9177	0.0030	0.0023	0.0007
Goldhill	512*512*3	Horizontal	0.9778	0.9819	0.9845	0.0022	0.0005	0.0006
		Vertical	0.9763	0.9850	0.9864	0.0008	0.0009	0.0007
		Diagonal	0.9604	0.9700	0.9733	-0.0007	0.0004	-0.0048
Airfield	512*512	Horizontal		0.9399			-0.0005	
		Vertical		0.9418			0.0011	
		Diagonal		0.9053			0.0009	

100 different cipher images. These cipher images have been obtained by encrypting the images differing by one bit from the original image in their pixel value for a random position.

The average correlation coefficients are tabulated in Table. 6.3. In all cases, the correlation coefficients in every color layer and each direction of the ciphered images have a value close to zero. This implies that the encryption scheme is highly resistant to statistical-based attacks. We also illustrated the correlation in the three directions for both plain and ciphered images of the colored image 'Pepper' and the grey image 'Boat' in Fig. 6.9 and Fig. 6.10, respectively. One can easily observe that in the plain images, the correlation between pixels is evident, while in the ciphered images, same as given by the coefficients, the high correlation is broken.

(3)Information entropy

We adopted the calculation of information entropy as given in equation (5.15) to evaluate uncertainty and randomness properties in the ciphered images. The average information entropy results of the plain and ciphered images for the tested images are given in Table.6.4. From the obtained results, we remark that the information entropy values of the ciphered images are close to the ideal value, i.e. 8. This indicates that the proposed encryption scheme gives rise to randomly distributed image pixel values.

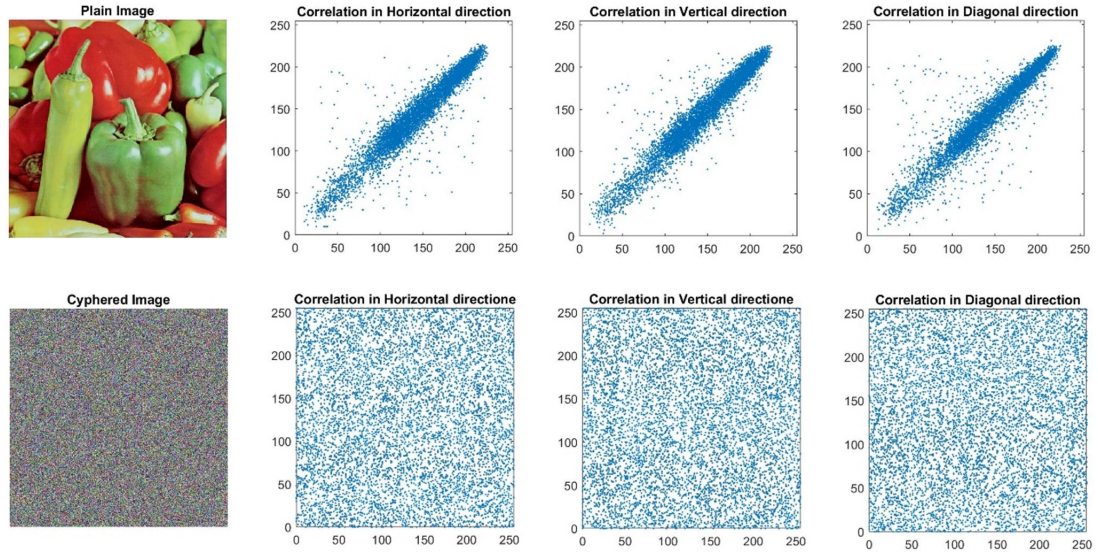


Figure 6.10 – Correlation results in 3 directions of colored image 'Pepper'

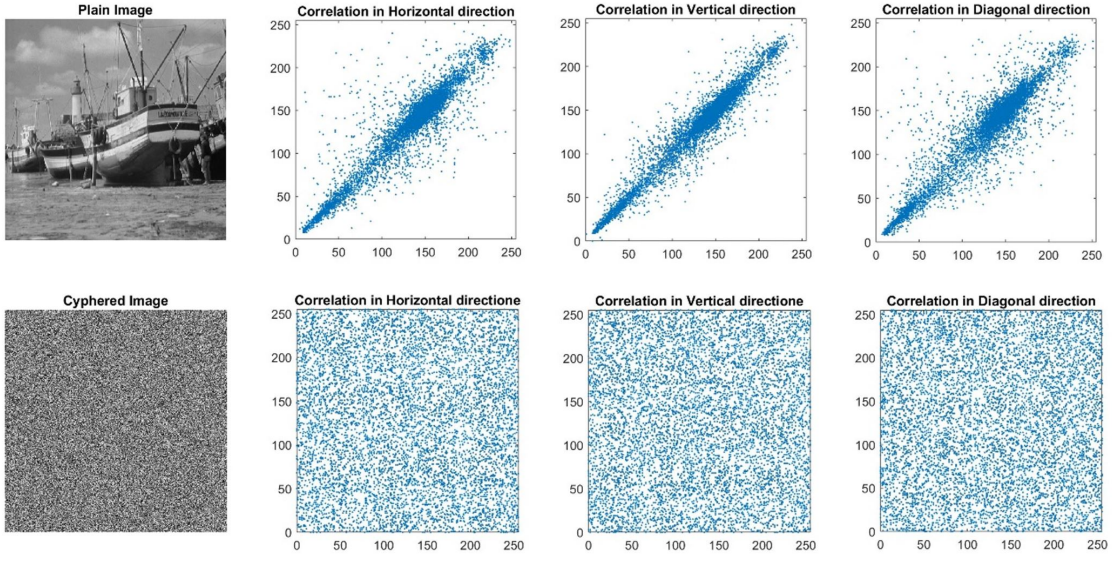


Figure 6.11 – Correlation results in 3 directions of grey image 'Boat'

Diffusion performance

A cryptosystem with good diffusion properties requires a high level of resistance to a chosen-plaintext attack. Difference analyses are applied between ciphered images encrypted by a certain number of plain images, which are of one-bit difference from each other. To resist these attacks, the cryptosystem must be sensitive to plaintext. When it comes to image encryption, this sensitivity means that a small change in the plaintext will lead to totally different ciphered images.

For each tested image, we have generated 100 different plain images with only one-bit differences at random pixel positions. The following tests are employed to assess the diffusion property of the proposed system.

(1) Hamming distance

To measure the sensitivity to the plain image, the calculation of Hamming distance is employed. Theoretically speaking, a one-bit change in the plaintext should lead to changes to 50% of the ciphertext for a well-designed cipher scheme. The Hamming distance (HD) of two ciphered images encrypted is calculated using equation (5.17), and the mean HD over the 100 cyphered images is given in Table. 6.4. It can be seen from Table.6.4 that all the HD values are close to 50%.

(2) NPCR and UACI results

We also employed here two commonly used indicators Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) to evaluate the sensitivity of the cryptosystem to the changes of the plain image. The former assesses the change rate of the number of pixels in two cipher images, whereas the latter, UACI, measures the average intensity of the differences between the plain and cipher images. The calculations of these two indicators are given below,

$$\begin{aligned} \mathrm{NPCR} &= \frac{1}{M_1 \times M_2 \times M_3} \times \sum_{u=1}^{M_1} \sum_{\nu=1}^{M_2} \sum_{w=1}^{M_3} D\left[u, \nu, w\right] \times 100\% \\ D\left[u, \nu, w\right] &= \begin{cases} 0, \text{ if } C_1\left[u, \nu, w\right] = C_2\left[u, \nu, w\right] \\ 1, \text{ if } C_1\left[u, \nu, w\right] \neq C_2\left[u, \nu, w\right] \end{cases} \end{aligned} \tag{6.12}$$

$$\mathrm{UACI} = \frac{1}{M_1 \times M_2 \times M_3 \times 255} \times \sum_{u=1}^{M_1} \sum_{\nu=1}^{M_2} \sum_{w=1}^{M_3} |C_1 - C_2| \times 100\% \tag{6.13}$$

In equation (6.12) and (6.13), C_1, C_2 represent two ciphered images encrypted from

one plain image with only one-bit difference in a random pixel; $M_1 \times M_2 \times M_3$ is the size of the image; (u, v, w) stands for the coordinate of the pixel (u-th row, v-th column and w-th color plan).

Table. 6.4 gives out the average NPCR and UACI of the 100 plain and ciphered images with a one-bit difference for all eight tested images. Knowing that the optimal values for NPCR under $\alpha = 0.05$ and UACI are equal to 99.6094% and 33.4635%, the results prove that the cryptosystem is sensitive to the changes of plain image, which indicates the high resistance to differential cryptanalysis and has good diffusion properties.

Table 6.4 – Statistical and performance analysis

Image	**Size**	**Mean NPCR**	**Mean UACI**	**Mean** χ^2_{exp}	**Mean HD(%)**	**Mean Entropy**
Baboon	256*256*3	33.4780	99.610	259.3507	50.0141	7.9991
Lena	512*512*3	33.4606	99.6105	255.8031	50.0139	7.9998
Goldhill	512*512*3	33.4590	99.6094	254.6217	49.9879	7.9998
Pepper	512*512*3	33.4606	99.6083	256.0509	49.9981	7.9998
Airfield	512*512*1	33.4611	99.6078	252.9924	49.9889	7.9993
Boat	512*512*1	33.4632	99.6091	255.0193	50.0007	7.9994
Black	512*512*1	33.4579	99.6080	252.2983	49.9986	7.9993
White	512*512*1	33.4493	99.6090	251.9953	50.0097	7.9993

Key space analysis

The secret keys of our proposed cryptosystem are composed of the following variables: 2 fractional orders for the Chen and Lu system (α_c and α_l); one set of control parameters and initial conditions for each of the fractional 3D systems ((a_c, b_c, c_c) and $X_1(0) = [x_{11}(0), x_{12}(0), x_{13}(0)]$ for Chen, and (a_l, b_l, c_l) and $X_2(0) = [x_{21}(0), x_{22}(0), x_{23}(0)]$ for Lu system); one initial condition for the FGDHL map ($X_g(0)$); 4 sets of parameters and intial conditions for the skew tent maps; and one variable tr defining the length of the FCPRNG output signal transient which will be cut off (to increase the complexity). The ranges of these variables for the FCPRNG are given in the Table. 6.5. With all these parameters, the encryption scheme has 24 different components in its secret keys. Since the default precision of the calculation is 10^{-15} for MATLAB, which we use, the key space of the encryption scheme can be calculated as being equal to 4.27×10^{358}, which is much larger than the required key space 2^{128} for a secure cryptosystem [Özkaynak, 2018b]. In terms of the key space, our proposed FCPRNG outperforms the generator discussed in

Table 6.5 – Keys and their ranges

Keys	**Ranges**	**Keys**	**Ranges**	**Keys**	**Ranges**	**Keys**	**Ranges**
$x_{11}(0)$	[-15,15]	$x_{12}(0)$	[-15,15]	$x_{13}(0)$	[0,30]	$x_{21}(0)$	[-15,15]
$x_{22}(0)$	[-15,15]	$x_{23}(0)$	[0,30]	$\mathrm{Xst}_1(0)$	(0,1)	p_1	(0,1)
$\mathrm{Xst}_2(0)$	(0,1)	p_2	(0,1)	$\mathrm{Xst}_3(0)$	(0,1)	p_3	(0,1)
$\mathrm{Xst}_4(0)$	(0,1)	p_4	(0,1)	$\mathrm{X}_g(0)$	(0,0.3)	α_c	[0.75,1]
α_l	[0.75,1]	a_c	[35,40]	b_c	[1.5,3.5]	c_c	[23,28]
a_l	[35,40]	b_l	[3,8]	c_l	[20,25]	tr	[1000,1500]

[Tutueva, 2020] and [Tutueva, 2019], which use adaptive Zaslavasky map and Chirikov map with key space of 2^{212} and 2^{159}, respectively. The acquired key space is also greater than almost all the encryption schemes reported in [Gayathri, 2016].

Key sensitivity tests

To resist the brute force attack, a cryptosystem must have a sufficiently large keyspace and be highly sensitive to changes in the secret key. 50 different sets of secret keys with only a one-bit difference for each of the 24 different key components are employed to encrypt the images in order to evaluate the sensitivity of our proposed encryption scheme to the changes in the secret keys. The mean NPCR, UACI, and avalanche (HD) results for each secret key component are given for the colored image "Baboon" in Table. 6.6.

It can be observed that the NPCR, UACI, and Hamming distance results for tested color images are all close to their optimal value 99.6094%, 33.4635%, and 50, respectively. This shows that a slight change to the secret key of the cryptosystem will impact the encryption of the image, which confirms the cryptosystem's resistance to brute-force attacks.

Time consumption

All the simulations have been conducted in MATLAB R2018b on a computer of Intel (R) Core (TM) i7-6700 CPU in Windows 10 Professional, 64-bit operating system with 3.40GHz processor, 32 GB RAM. The computational time of the proposed encryption scheme for several images with different sizes is given in Table.6.7.

Since the encryption time of an image cryptosystem is influenced by many factors, it is highly unlikely to get explicit comparison results when comparing the running time in different environments directly. Therefore, we also adopted the Encryption Throughput

Table 6.6 – Key sensitivity analysis of different keys for image Baboon

Secret Key	UACI	NPCR	HD	Secret Key	UACI	NPCR	HD
$x_{11}(0)$	33.4703	99.6105	49.9997	$x_{12}(0)$	33.4765	99.6096	50.0053
$x_{13}(0)$	33.4550	99.6113	49.9957	$x_{21}(0)$	33.4535	99.6120	49.9975
$x_{22}(0)$	33.4766	99.6087	50.0037	$x_{23}(0)$	33.4635	99.6127	50.0006
$\mathbf{Xst}_1(0)$	33.4656	99.6083	49.9912	p_1	33.4772	99.6110	49.9918
$\mathbf{Xst}_2(0)$	33.4498	99.6086	50.0016	p_2	33.4742	99.6138	50.0041
$\mathbf{Xst}_3(0)$	33.4533	99.6106	50.0006	p_3	33.4470	99.6081	49.9915
$\mathbf{Xst}_4(0)$	33.4520	99.6093	50.0059	p_4	33.4533	99.6088	49.9988
$\mathbf{X}_\mathbf{g}(0)$	33.4620	99.6139	49.9927	α_c	33.4641	99.6072	49.9928
α_l	33.4709	99.6074	49.9979	a_c	33.4637	99.6111	49.9952
b_c	33.4715	99.6075	49.9988	c_c	33.4693	99.6106	50.0047
a_l	33.4740	99.6086	49.9997	b_l	33.4657	99.6108	50.0010
c_l	33.4751	99.6111	49.9983	tr	33.4626	99.6094	49.9944

Table 6.7 – Time consumption for several images

Image	Size	Encryption time(s)	ET(MBps)	NCpB
Baboon	256*256*3	22.2238	0.008	384322.7
Airfield	512*512*1	33.4181	0.008	433431.7
Lena	512*512*3	160.3160	0.005	693010.14
LenaGrey	512*512*1	31.8463	0.008	415411.29
LenaGrey Ref([Luo, 2018])	512*512*1	-	0.035	95367.43
LenaGrey Ref([Qiao, 2019])	512*512*1	-	0.045	77385.32

(ET) and Number of needed Cycles per Byte (NCpB) to evaluate the encryption speed of the proposed cryptosystem. The calculation formula for ET and NCpB are given in equations (6.14) and (6.15) respectively, and the results are given in Table.6.7.

$$ET = \frac{Image_{size}(Byte)}{Encryption_{Time}(second)} \tag{6.14}$$

$$NCpB = \frac{CPUspeed(Hertz)}{ET(Byte/second)} \tag{6.15}$$

Compared to the encryption schemes given in the table, the computational time is relatively more significant than the other cryptosystems illustrated in [Luo, 2018] and [Qiao, 2019]. The generation of the FCPRNG outputs partly contributes to the greater time consumption, which the pie chart in Fig.6.12 explains. The pie chart displays the

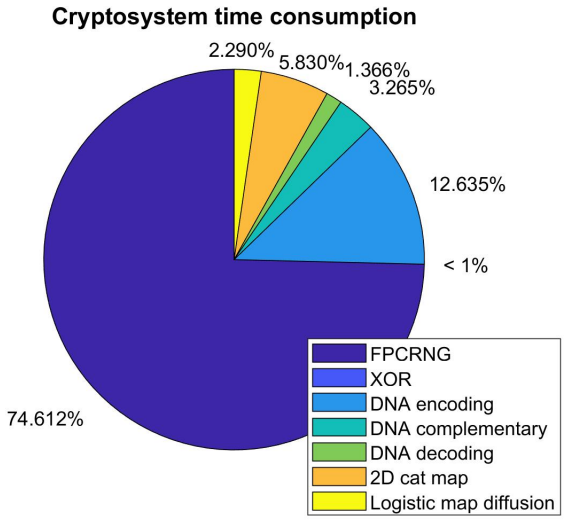


Figure 6.12 – Percentage of time consumption during each process

time consumption percentage of each scheme component for the encryption of grey images 'Lena' with size 512*512. It can be observed that the encryption process only takes a quarter of the whole cryptosystem running time. In the meantime, around three-fourths of the time (23.7615s out of 31.8463s) is spent on the generation of the pseudo-random numbers using the proposed FCPRNG.

Nevertheless, the merits of the proposed cryptosystem are significant. It is logical to remark that high security gained from more complex schemes often requires longer computational time, and there is a trade-off between the two aspects with respect to the envisaged applications. Therefore, our proposed cryptosystem can be used for security transmission whenever time consumption is not prioritized. In addition, it can be used for secure storage of information such as medical files in hospitals, personal data (such as fingerprints), confidential business documents and is suitable for the use of the home office.

6.3.3 Comparative analysis

In this part, we compared the performances of our proposed cryptosystem with other image encryption algorithms tested on benchmark images in terms of their diffusion and confusion performances. Different works encrypting same benchmark grey and colored images 'Boat' and 'Lena' have been compared. Among them, three have employed DNA-

Table 6.8 – Comparison on confusion property

Image	Cryptosystem	HD(%)	Entropy	Correlation coefficient		
				Horizontal	Vertical	Diagonal
Lena Grey 512*512	**Proposed**	**49.9977**	7.9993	**0.0013**	**-0.00008**	**0.0011**
	Ref [Zhao, 2020]	-	7.9977	0.0015	-0.00011	-0.0022
	Ref [Wu, 2018]	-	**7.9994**	0.0032	0.0016	0.0023
	Ref [Qiao, 2020]	50.0079	7.9993	0.0018	0.0001	0.0017
Boat 512*512	**Proposed**	**50.0007**	**7.9994**	**-0.00005**	**-0.00004**	**-0.00009**
	Ref [Qiao, 2020]	49.9978	7.9993	0.00047	0.00252	0.00124
	Ref [Wu, 2018]	-	7.9994	0.0003	0.0034	0.0011
	Ref [Wang, 2021]	-	7.9965	0.0024	0.0007	0.0040

based encryption schemes ([Wu, 2018], [Chai, 2019b] and [Wang, 2021]). The other three ([Qiao, 2020], [Zhao, 2020] and [Huang, 2018]) introduced the cryptosystems implementing chaotic systems with different encryption schemes and structures.

The features (well-recognized by the cryptography community) examined for the comparison of the confusion property are the hamming distance, the entropy test, and the correlation coefficients. The diffusion property (against the chosen-plaintext attack) which requires the cryptosystem to be highly sensitive to even one bit change in the plain image or in the secret key is analysed through NPRC and UACI.

The results with respect to benchmark images, namely grey and colored 'Lena', and 'Boat', respectively, are given in Table.6.8 and Table.6.9. It can be noticed that our proposed cryptosystem achieves similar satisfactory encryption performance with respect to all the above-listed works regarding the evaluated metrics for both diffusion and confusion properties. To further analyze these characteristics, we also marked the values which are closest to the ideal values in bold cases in the comparison tables. For the confusion property, our proposed algorithm possesses closer HD and most of the correlation coefficients for the grey 'Lena' image. For benchmark 'Boat' image, our proposed cryptosystem outperforms the other works in Table.6.8 for all the given metrics. For the diffusion property, we compared the plaintext and key sensitivity employing NPRC and UACI results for grey and colored 'Lena' images. As announced in section 5.2.5, the optimal values for NPRC and UACI are 99.6094% and 33.4635%, respectively. Our proposed scheme achieved better performances concerning the grey 'Lena' image encryption. As for the colored 'Lena' image, the superiority of our proposed cryptosystem is not significant, but it still possesses the advantage of acquiring greater keyspaces while exhibiting equivalent secure encryption performances.

Table 6.9 – Comparison on diffusion property

Image	**Cryptosystem**		**Plaintext sensitivity**		**Key sensitivity**	
			NPRC(%)	UACI(%)	NPCR(%)	UACI(%)
LenaGrey 512*512	**Proposed**		**99.6107**	**33.4471**	**99.6093**	33.4483
	Ref [Qiao, 2020]		99.6080	33.4925	99.6100	**33.4763**
	Ref [Zhao, 2020]		99.6184	33.5793	-	-
	Ref [Wang, 2021]		99.6066	33.4977	-	-
LenaRGB 512*512*3	**Proposed**		99.6105	33.4606	**99.6095**	33.4544
	Ref [Qiao, 2020]		**99.6097**	33.4573	99.6062	33.4672
	Ref [Huang, 2018]	R	99.6093	33.4678	99.6089	33.4589
		G	99.6099	33.4577	99.6089	33.4598
		B	99.6090	**33.4608**	99.6085	**33.4624**
	Ref [Chai, 2019b]	R	99.60	33.56	-	-
		G	99.60	33.45	-	-
		B	99.61	33.49	-	-

6.4 Conclusion

In this chapter, we proposed a secure image encryption cryptosystem innovatively. The cryptosystem consists of an efficient pseudo-random number generator consisting of three different fractional chaotic systems and a block cipher based on DNA encoding and decoding.

The employed FCPRNG is a more laborated version of the generator proposed in Chapter 5. The non-uniform grid numerical calculation method with randomly altered grid space controlled by two skew-tent maps (Grid 2 discussed in Chapter 4) is adopted for the employed 3D fractional chaotic systems. Much greater keyspace size is gained through the use of the fractional chaotic systems and the combination of three systems, which can be of great use to resist brute-force attacks. The confusion process of the cipher is composed of the DNA encoding, the permutation in the DNA level carried out by a modified 2D map, and the DNA decoding. A 32-bits discrete logistic map is employed to achieve the diffusion of the scheme.

The statistical analysis of the designed FCPRNG and the security analysis of the image cipher demonstrated that the design of the cryptosystem is indeed secure and can successfully resist most of the known attacks. Though the time consumption is relatively more significant compared to some other encryption schemes, due to its overwhelming significant keyspace, the cryptosystem can be implemented in the applications such as secure image storage where the encryption time is not the primary consideration.

Conclusion and perspective

This thesis focuses on fractional chaotic dynamics and the design of the Fractional Chaotic Pseudo-Random Number Generator (FCPRNG). The proposed FCPRNG is implemented to construct secure and reliable chaos-based cryptosystems.

In chapter 1, we introduced some basics of chaotic dynamics and chaos-based cryptography. The excellent features of the chaotic system, such as random-like behavior and sensitivity to initial conditions, make the nonlinear systems possessing chaotic behavior a great candidate for cryptosystem design. The state of the art of chaos-based cryptosystem and CPRNG design were analyzed. Existing problems in the current literature were stipulated. We orientated readers' attention to the relatively novel and less discussed cryptosystems design implementing the fractional chaotic system. The use of DNA computing in existing image encryption algorithms was also discussed.

As another crucial part of our thesis, we introduced the fundamentals of fractional calculus and fractional chaotic systems in Chapter 2. Different characterizations for fractional derivatives were illustrated. The stability of the fractional nonlinear system and the necessary conditions for fractional systems to be chaotic were demonstrated. Two fractional chaotic systems derived from the classical integer order Chen and Lu system were presented, which was later employed for our FCPRNG and cryptosystem design. From the aspect of secure cryptosystems design, the introduced fractional derivatives act as extra parameters and enhance the security of the cipher against brute-force attacks.

To implement the fractional chaotic systems for digital implementation, such as for FCPRNG design, the systems must be numerically approximated or calculated. Therefore, in Chapter 3, we gave the numerical calculation methods for both fractional chaotic maps and multi-dimensional fractional chaotic systems. We used the piecewise-argument calculation method for the fractional generalized double-humped logistic map employed for our FCPRNG. Two numerical calculation methods based on Grunwald and Caputo characterizations were studied and adopted to calculate the states of the fractional Chen and Lu system. The impacts of using these two different numerical approaches for distinct fractional derivative characterizations were analyzed.

In Chapter 4, we proposed a non-uniform grid calculation method based on the ABM

corrector predictor method discussed in Chapter 3. Two different non-uniform grids were studied. For the first non-uniform grid, one skew-tent map was used to vary the calculation step size for each system state. The grid space takes one of the values among five values ranging from 0.001 to 0.005. For the second method, two skew tent maps were used to work together and assigned a grid space among the five possible values for each calculation step. We employed the proposed grids to solve the fractional Chen and Lu system numerically. The simulation carried out showed that with our proposed non-uniform grid calculation method, a greater chaotic range of fractional derivative order was acquired.

Applying the proposed calculation method with a first non-uniform (Gird 1), we proposed for the very first time in Chapter 5 a secure FCPRNG integrating three different fractional chaotic systems, namely the fractional chaotic Chen, fractional chaotic Lu systems, and fractional double-humped logistic map. Based on the FCPRNG, a new stream cipher was proposed and tested. The conducted experiment results verified the security of the stream cipher.

A new secure image cryptosystem based on a confusion-diffusion structure adopting proposed FCPRNG and DNA encoding and decoding was designed in Chapter 6. The keystream of the cryptosystem is provided by a FCPRNG similar to the one in Chapter 5 but with a larger key space size by employing the second non-uniform grid (Grid 2) for the calculation of fractional Chen and Lu systems. The DNA encoding and decoding, together with a modified cat map consist of the confusion layer of the proposed block cipher. A logistic map with finite precision was employed as the diffusion module. The proposed image encryption cryptosystem possesses great confusion and diffusion properties, which allows to resist common attacks.

Perspectives of future work

Based on what we have achieved in this thesis, several perspectives to which future work can be oriented are given in the following.

Firstly, our work is achieved entirely using MATLAB. Due to the characteristics and configuration of the software itself, the computational power and efficiency are relatively low. Therefore, though we have proved that our designed FCPRNG and proposed image encryption algorithm have excellent performances in terms of security, the numerical calculation of fractional chaotic systems on the platform leads to relatively heavier com-

putational time. In addition, the analysis performance of the cryptosystem stays at a theoretical level. Hence, implementing the cryptosystem in other programming environments and further hardware implementations can be expected.

Secondly, we have studied two different numerical calculation methods based on Grunwald and Caputo characterizations. The impacts on the chaoticity of multi-dimensional fractional chaotic systems employing the methods have been discussed and compared. Regarding this topic, other existing approximation methods should be investigated to find the most efficient and appropriate method for FCPRNG application. Besides, in this thesis, we only considered the fractional systems with one single fractional derivative order for their system equations. The cases of incommensurate fractional orders should also be taken into consideration.

Thirdly, we have analyzed the chaotic performance of fractional chaotic Chen and Lu systems by applying our proposed non-uniform grid calculation method. However, it can be completed with other analyses, especially for the change of chaoticity of the systems relating to the system parameters. A parameter sweeping is in demand for a more in-depth understanding of this topic. We chose fractional Chen and Lu systems since they are the most discussed fractional chaotic systems when it comes to chaos-based cryptosystem design. Nevertheless, other fractional chaotic systems should also be investigated to find the ones that are suitable for FCPRNG design, providing perhaps greater chaotic parameters and fractional order ranges.

Apart from these, in this work, we adopted two 3D fractional chaotic systems calculated using two different non-uniform grids and one fractional chaotic discrete map and designed two FCPRNGs possessing good randomness and statistical properties. The fractional chaotic systems and map are integrated parallelly by performing XOR operations. Other grid choices and FCPRNG structures could be proposed and tested for simpler, more efficient FCPRNG design.

Last but not least, we focussed our research interest on adopting the FCPRNG for symmetric cryptosystem design in this thesis. However, as a PRNG, there are many other fields where PRNG is employed and works as a crucial component, such as numerical simulations, communication systems, electronic games, and control theory. Applying FCPRNG to these applications can also be an exciting research direction.

BIBLIOGRAPHY

Abraham, L. and N. Daniel (2013). Secure image encryption algorithms: A review. *International journal of scientific & technology research*, vol. 2, no. 4, pages: 186–189.

Adleman, L. M. (1994). Molecular computation of solutions to combinatorial problems. *science*, vol. 266, no. 5187, pages: 1021–1024.

Ahmad, M., M. N. Doja, and M. M. S. Beg (2021). Security analysis and enhancements of an image cryptosystem based on hyperchaotic system. *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 1, pages: 77–85.

Ahmed, H., H. Kalash, and O. Faragallah (Mar. 2007). An Efficient Chaos-Based Feedback Stream Cipher (ECBFSC) for image encryption and decryption. *Informatica (Slovenia)*, vol. 31, pages: 121–129.

Akhavan, A., A. Samsudin, and A. Akhshani (2017). Cryptanalysis of an image encryption algorithm based on DNA encoding. *Optics & Laser Technology*, vol. 95, pages: 94–99.

Alvarez, G. and S. Li (2006). Some basic cryptographic requirements for chaos-based cryptosystems. *International journal of bifurcation and chaos*, vol. 16, no. 08, pages: 2129–2151.

Arfan, M., K. Shah, A. Ullah, M. Shutaywi, P. Kumam, and Z. Shah (2021). On fractional order model of tumor dynamics with drug interventions under nonlocal fractional derivative. *Results in Physics*, vol. 21, pages: 103783.

Bagley, R. L. and R. Calico (1991). Fractional order state equations for the control of viscoelasticallydamped structures. *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pages: 304–311.

Basalto, N., R. Bellotti, F. De Carlo, P. Facchi, and S. Pascazio (2005). Clustering stock market companies via chaotic map synchronization. *Physica A: Statistical Mechanics and its Applications*, vol. 345, no. 1-2, pages: 196–206.

Bassham, L. E., A. L. Rukhin, J. Soto, J. R. Nechvatal, M. E. Smid, E. B. Barker, S. D. Leigh, M. Levenson, M. Vangel, D. L. Banks, N. A. Heckert, J. F. Dray, and S. Vo (2010). *SP 800-22 Rev. 1a. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. Technical report. Gaithersburg, MD, USA.

Biham, E. (1991). Cryptanalysis of the chaotic-map cryptosystem suggested at EUROCRYPT'91. In: *Proceedings of the Workshop on the Theory and Application of of Cryptographic Techniques*, Springer, pages: 532–534.

Billings, L. and E. M. Bollt (2001). Probability density functions of some skew tent maps. *Chaos, Solitons & Fractals*, vol. 12, no. 2. Chaos in Ecology, pages: 365–376. URL: https://www.sciencedirect.com/science/article/pii/S0960077999002040.

Boccara, N. and N. Boccara (2004). *Modeling complex systems*. vol. 1. Springer.

Boriga, R., A. C. Dăscălescu, and I. Priescu (2014). A new hyperchaotic map and its application in an image encryption scheme. *Signal Processing: Image Communication*, vol. 29, no. 8, pages: 887–901.

Cafagna, D. and G. Grassi (2003). New 3D-scroll attractors in hyperchaotic Chua's Circuits Forming a Ring. *Int. J. Bifurc. Chaos*, vol. 13, pages: 2889–2903.

Cao, C., K. Sun, and W. Liu (2018). A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map. *Signal Processing*, vol. 143, pages: 122–133.

Caputo, M. (Nov. 1967). Linear Models of Dissipation whose Q is almost Frequency Independent—II. *Geophysical Journal International*, vol. 13, no. 5, pages: 529–539. URL: https://doi.org/10.1111/j.1365-246X.1967.tb02303.x.

Cartwright, M. L. and J. E. Littlewood (1947). On Non-Linear Differential Equations of the Second Order: II. The Equation $\ddot{y} + kf(y, \dot{y} + g(y, k) = p(t) = p_1(t) + kp_2(t); k > 0, f(y) \geqq 1$. *Annals of Mathematics*, vol. 48, no. 2, pages: 472–494. URL: http://www.jstor.org/stable/1969181 (visited on 09/13/2022).

Chai, X., Z. Gan, Y. Chen, and Y. Zhang (2017). A visually secure image encryption scheme based on compressive sensing. *Signal Processing*, vol. 134, pages: 35–51.

Chai, X., X. Fu, Z. Gan, Y. Lu, and Y. Chen (2019a). A color image cryptosystem based on dynamic DNA encryption and chaos. *Signal Processing*, vol. 155, pages: 44–62.

— (2019b). A color image cryptosystem based on dynamic DNA encryption and chaos. *Signal Processing*, vol. 155, pages: 44–62.

Chen, G. and X. Dong (1998). *From chaos to order : methodologies, perspectives and applications*. vol. 24. World Scientific.

Chen, G., Y. Mao, and C. K. Chui (2004). A symmetric image encryption scheme based on 3D chaotic cat maps. *Chaos, Solitons & Fractals*, vol. 21, no. 3, pages: 749–761. URL: https://www.sciencedirect.com/science/article/pii/S0960077903006672.

Chen, G., Y. Chen, and X. Liao (2007). An extended method for obtaining S-boxes based on three-dimensional chaotic Baker maps. *Chaos, solitons & fractals*, vol. 31, no. 3, pages: 571–579.

Chen, L., B. Ma, X. Zhao, and S. Wang (2017). Differential cryptanalysis of a novel image encryption algorithm based on chaos and Line map. *Nonlinear Dynamics*, vol. 87, no. 3, pages: 1797–1807.

Chua, L., M. Komuro, and T. Matsumoto (1986). The double scroll family. *IEEE Transactions on Circuits and Systems*, vol. 33, no. 11, pages: 1072–1118.

Danca, M.-F. and N. Kuznetsov (2018). Matlab code for Lyapunov exponents of fractional-order systems. *International Journal of Bifurcation and Chaos*, vol. 28, no. 05, pages: 1850067.

Dataiku (Aug. "2021). *Mary Lucy Cartwright: The Inspired Mathematician Behind Chaos Theory*. urlhttps://www.historyofdatascience.com/mary-lucy-cartwright-the-inspired-mathematician-behind-chaos-theory.

Deng, W. and C. Li (2005). Chaos synchronization of the fractional Lü system. *Physica A: Statistical Mechanics and its Applications*, vol. 353, pages: 61–72.

Deng, W. (2007). Short memory principle and a predictor–corrector approach for fractional differential equations. *Journal of Computational and Applied Mathematics*, vol. 206, no. 1, pages: 174–188.

DEVANEY, R. L. (1986). An Introduction to Chaotic Dynamical System. *Benjamin/Cummings*.

Diab, H. (2018). An efficient chaotic image cryptosystem based on simultaneous permutation and diffusion operations. *IEEE access*, vol. 6, pages: 42227–42244.

Diacu, F. and P. Holmes (1999). *Celestial encounters : the origins of chaos and stability*. vol. 22. Princeton university press.

Diethelm, K., N. J. Ford, and A. D. Freed (2002). A Predictor-Corrector Approach for the Numerical Solution of Fractional Differential Equations. *Nonlinear Dynamics*, vol. 29, pages: 3–22.

Dorcak, L. (1994). Numerical Models for the Simulation of the Fractional-Order Control systems. UEF-04-94, The Academy of Sciences, Inst. of Experimental Phsic.

El Assad, S. and M. Farajallah (2016). A new chaos-based image encryption system. *Signal Processing: Image Communication*, vol. 41, pages: 144–157.

El Raheem, Z. and S. Salman (2014). On a discretization process of fractional-order logistic differential equation. *Journal of the Egyptian Mathematical Society*, vol. 22, no. 3, pages: 407–412.

El-Sayed, A. (1996). Fractional-order diffusion-wave equation. *International Journal of Theoretical Physics*, vol. 35, no. 2, pages: 311–322.

Farah, M., R. Guesmi, A. Kachouri, and M. Samet (2020a). A new design of cryptosystem based on S-box and chaotic permutation. *Multimedia Tools and Applications*, vol. 79, no. 27, pages: 19129–19150.

Farah, M. B., R. Guesmi, A. Kachouri, and M. Samet (2020b). A novel chaos based optical image encryption using fractional Fourier transform and DNA sequence operation. *Optics & Laser Technology*, vol. 121, pages: 105777.

Farajallah, M., S. El Assad, and M. Chetto (2013). Dynamic Adjustment of the Chaos-Based Security in Real-Time Energy Harvesting Sensors. In: *Proceedings of the 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, pages: 282–289.

Farajallah, M., S. El Assad, and O. Déforges (2016). Fast and Secure Chaos-Based Cryptosystem for Images. *Int. J. Bifurc. Chaos*, vol. 26, pages: 1650021:1–1650021:21.

Farajallah, M., S. E. Assad, and O. Déforges (Apr. 2018). Cryptanalyzing an image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion. *Multimedia Tools and Applications*, vol. 77, no. 21, pages: 28225–28248. URL: https://hal.archives-ouvertes.fr/hal-01794015.

Farajallah, M., S. El Assad, and O. Deforges (2018). Cryptanalyzing an image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion. *Multimed Tools Appl*, vol. 77.

Feigenbaum, M. J. (1978). Quantitative universality for a class of nonlinear transformations. *Journal of statistical physics*, vol. 19, no. 1, pages: 25–52.

Feltekh, K., D. Fournier-Prunaret, and S. Belghith (2014). Analytical expressions for power spectral density issued from one-dimensional continuous piecewise linear maps with three slopes. *Signal Processing*, vol. 94, pages: 149–157. URL: https://www.sciencedirect.com/science/article/pii/S0165168413002132.

Ford, N. J. and A. C. Simpson (Apr. 2001). The numerical solution of fractional differential equations: Speed versus accuracy. *Numerical Algorithms*, vol. 26, no. 4, pages: 333–346. URL: https://doi.org/10.1023/A:1016601312158.

Fridrich, J. (1998). Symmetric ciphers based on two-dimensional chaotic maps. *International Journal of Bifurcation and chaos*, vol. 8, no. 06, pages: 1259–1284.

Gao, X., J. Yu, S. Banerjee, H. Yan, and J. Mou (2021). A new image encryption scheme based on fractional-order hyperchaotic system and multiple image fusion. *Scientific Reports*, vol. 11, no. 1, pages: 1–21.

Garasym, O., R. Lozi, and I. Taralova (2016). Robust PRNG based on homogeneously distributed chaotic dynamics. In: *Proceedings of the Journal of Physics: Conference Series*, vol. 692, no. 1, IOP Publishing, pages: 012011.

Garrappa, Roberto (2021). *Predictor-corrector PECE method for fractional differential equations*. MATLAB Central File Exchange, accessed June 23, 2022. URL: https://www.mathworks.com/matlabcentral/fileexchange/32918-predictor-corrector-pece-method-for-fractional-differential-equations.

Gayathri, J. and S. Subashini (2016). A survey on security and efficiency issues in chaotic image encryption. *International Journal of Information and Computer Security*, vol. 8, no. 4, pages: 347–381.

Goldreich, O. (2001). The Foundations of Cryptography - Volume 2: Basic Applications. In.

Guyeux, C., Q. Wang, and J. Bahi (2010). A Pseudo Random Numbers Generator Based on Chaotic Iterations. Application to Watermarking. In: *Proceedings of the WISM 2010, Int. Conf. on Web Information Systems and Mining*, China, pages: 202–211, URL: https://hal.archives-ouvertes.fr/hal-00563317.

Hadamard, J. (1898). Les surfaces à courbures opposées et leurs lignes géodesiques. *Journal de Mathématiques Pures et Appliquées*, vol. 4, pages: 27–73.

Hamza, R. (2017). A novel pseudo random sequence generator for image-cryptographic applications. *Journal of Information Security and Applications*, vol. 35, pages: 119–127.

Hardesty, L. (2010). *Explained: Linear and nonlinear systems*. Accessed: 2022-09-26. URL: http://https://news.mit.edu/2010/explained-linear-0226.

Heaviside, O., H. J. Josephs, and A. B. Bernard (1971). *Electromag netic theory*. Chelsea Publishing Company, New York.

Hellekalek, P. (1998). Good random number generators are (not so) easy to find. *Mathematics and Computers in Simulation*, vol. 46, no. 5, pages: 485–505. URL: https://www.sciencedirect.com/science/article/pii/S0378475498000780.

Holmgren, R. A. (2000). *A first course in discrete dynamical systems*. Springer Science & Business Media.

Hua, Z., S. Yi, and Y. Zhou (2018). Medical image encryption using high-speed scrambling and pixel adaptive diffusion. *Signal Processing*, vol. 144, pages: 134–144.

Huang, L., S. Cai, M. Xiao, and X. Xiong (2018). A simple chaotic map-based image encryption system using both plaintext related permutation and diffusion. *Entropy*, vol. 20, no. 7, pages: 535.

Huang, R., X. Liao, A. Dong, and S. Sun (2020). Cryptanalysis and security enhancement for a chaos-based color image encryption algorithm. *Multimedia Tools and Applications*, vol. 79, no. 37, pages: 27483–27509.

Ismail, S. M., L. A. Said, A. G. Radwan, A. H. Madian, and M. F. Abu-Elyazeed (2018). Generalized double-humped logistic map-based medical image encryption. *Journal of advanced research*, vol. 10, pages: 85–98.

Katugampola, U. N. (2011). A new approach to generalized fractional derivatives. *arXiv preprint arXiv:1106.0965*.

Keliher, L., H. Meijer, and S. Tavares (1999). Modeling linear characteristics of substitution-permutation networks. In: *Proceedings of the International Workshop on Selected Areas in Cryptography*, Springer, pages: 78–91.

Khan, M., T. Shah, and M. A. Gondal (2013). An efficient technique for the construction of substitution box with chaotic partial differential equation. *Nonlinear Dynamics*, vol. 73, no. 3, pages: 1795–1801.

Knudsen, L. R. and M. J. Robshaw (2011). *The Block Cipher Companion*. Springer-Verlag Berlin and Heidelberg GmbH & Co. K.

Kocarev, L. (2001). Chaos-based cryptography: a brief overview. *IEEE Circuits and Systems Magazine*, vol. 1, no. 3, pages: 6–21.

Kolmogorov, A. N. (1954). On conservation of conditionally periodic motions for a small change in Hamilton's function. *Proceedings of the USSR Academy of Sciences*, vol. 98, pages: 527–530.

— (1991). The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds Numbers. *Proceedings of the Royal Society A: Mathematical*, vol. 434, no. 1890, pages: 9–13.

Krasnobrizha, A., P. Rozycki, L. Gornet, and P. Cosson (2016). Hysteresis behaviour modelling of woven composite using a collaborative elastoplastic damage model with fractional derivatives. *Composite Structures*, vol. 158, pages: 101–111.

Lan, R., J. He, S. Wang, T. Gu, and X. Luo (2018). Integrated chaotic systems for image encryption. *Signal Processing*, vol. 147, pages: 133–145.

Laskin, N. (2000). Fractional market dynamics. *Physica A: Statistical Mechanics and its Applications*, vol. 287, no. 3-4, pages: 482–492.

L'ECUYER, P. and R. SIMARD (2007). TestU01 : AC library for empirical testing of random number generators. *ACM Transections on Mathmematical Software(TOMS)*, vol. 33.

Li, T.-Y. and J. A. Yorke (1975). Period Three Implies Chaos. *The American Mathematical Monthly*, vol. 82, no. 10, pages: 985–992. URL: `http://www.jstor.org/stable/2318254` (visited on 09/14/2022).

Li, Y., Y. Chen, and I. Podlubny (2009). Mittag–Leffler stability of fractional order nonlinear dynamic systems. *Automatica*, vol. 45, no. 8, pages: 1965–1969.

Li, Y., C. Wang, and H. Chen (2017). A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. *Optics and Lasers in Engineering*, vol. 90, pages: 238–246.

Li, H., L. Deng, and Z. Gu (2020). A robust image encryption algorithm based on a 32-bit chaotic system. *IEEE Access*, vol. 8, pages: 30127–30151.

Li, Z., C. Peng, W. Tan, and L. Li (2020). A novel chaos-based color image encryption scheme using bit-level permutation. *Symmetry*, vol. 12, no. 9, pages: 1497.

Liouville, J. (1832). Mémoire sur quelques Quéstions de Géometrie et de Mécanique, et sur un nouveau genre de Calcul pour résoudre ces Quéstions. *Journal de l'école Polytechnique*, pages: 1–69.

Liu, H. and X. Wang (2010). Color image encryption based on one-time keys and robust chaotic maps. *Computers & Mathematics with Applications*, vol. 59, no. 10, pages: 3320–3327. URL: `https://www.sciencedirect.com/science/article/pii/S0898122110001938`.

Liu, H., X. Wang, et al. (2012). Image encryption using DNA complementary rule and chaotic maps. *Applied Soft Computing*, vol. 12, no. 5, pages: 1457–1466.

Liu, L., S. Hao, J. Lin, Z. Wang, X. Hu, and S. Miao (2018). Image block encryption algorithm based on chaotic maps. *IET Signal Processing*, vol. 12, no. 1, pages: 22–30.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of atmospheric sciences*, vol. 20, no. 2, pages: 130–141.

Lorenz, E. (1972). *Predictability: does the flap of a butterfly's wing in Brazil set off a tornado in Texas?* American Association for the Advancement of Science.

Lu, J. and G. Chen (2002). A New Chaotic Attractor Coined. *Int. J. Bifurc. Chaos*, vol. 12, pages: 659–661.

Lu, J., G. Chen, X. Yu, and H. Leung (2004). Design and analysis of multiscroll chaotic attractors from saturated function series. *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 12, pages: 2476–2490.

Lu, J. and g. Chen (2006). A note on the fractional-order Chen system. *Chaos, Solitons & Fractals*, vol. 27, no. 3, pages: 685–688.

Luo, Y., R. Zhou, J. Liu, S. Qiu, and Y. Cao (2018). An efficient and self-adapting colour-image encryption algorithm based on chaos and interactions among multiple layers. *Multimedia Tools and Applications*, vol. 77, no. 20, pages: 26191–26217.

Luo, Y., S. Tang, J. Liu, L. Cao, and S. Qiu (2020). Image encryption scheme by combining the hyper-chaotic system with quantum coding. *Optics and Lasers in Engineering*, vol. 124, pages: 105836.

Lv, X., X. Liao, and B. Yang (2018). A novel pseudo-random number generator from coupled map lattice with time-varying delay. In: vol. 94, pages: 325–341.

Lynnyk, V., N. Sakamoto, and S. Čelikovský (2015). Pseudo random number generator based on the generalized Lorenz chaotic system. *IFAC-PapersOnLine*, vol. 48, no. 18. 4th IFAC Conference on Analysis and Control of Chaotic Systems CHAOS 2015, pages: 257–261. URL: `https://www.sciencedirect.com/science/article/pii/S2405896315023046`.

Ma, Y., C. Li, and B. Ou (2020). Cryptanalysis of an image block encryption algorithm based on chaotic maps. *Journal of Information Security and Applications*, vol. 54, pages: 102566.

Machado, J. T., V. Kiryakova, and F. Mainardi (2011). Recent history of fractional calculus. *Communications in nonlinear science and numerical simulation*, vol. 16, no. 3, pages: 1140–1153.

Matignon, D. (1996). Stability results for fractional differential equations with applications to control processing. In: *Proceedings of the Computational engineering in systems applications*, vol. 2, no. 1, Citeseer, pages: 963–968.

Matthews, R. (1989). On the derivation of a "chaotic" encryption algorithm. *Cryptologia*, vol. 13, no. 1, pages: 29–42.

Merrikh-Bayat, F., M. Afshar, and M. Karimi-Ghartemani (2009). Extension of the root-locus method to a certain class of fractional-order systems. *ISA Transactions*, vol. 48,

no. 1, pages: 48–53. URL: https://www.sciencedirect.com/science/article/pii/S0019057808000463.

Milnor, J. W. (1815). Attractor. *Scholarpedia*, vol. 1(11).

Mitchell, F. (1989). *Academic American encyclopaedia*. vol. 4. Danbury: Grolier Incorporated.

Monje, C. A., Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu-Batlle (2010). *Fractional-order systems and controls: fundamentals and applications*. Springer Science & Business Media.

Muhammad, Z. M. Z. and F. Özkaynak (2019). Security problems of chaotic image encryption algorithms based on cryptanalysis driven design technique. *IEEE Access*, vol. 7, pages: 99945–99953.

Niels, H. A. (1823). Solution de quelques problèmes à l'aide d'intégrales définies: *Magazin for Naturvidenskaberne*, pages: 55–68.

Odibat, Z. M., N. Corson, M. Aziz-Alaoui, and C. Bertelle (2010). Synchronization of chaotic fractional-order systems via linear control. *International Journal of Bifurcation and Chaos*, vol. 20, no. 01, pages: 81–97.

Oustaloup, A., J. Sabatier, P. Lanusse, R. Malti, P. Melchior, X. Moreau, and M. Moze (2008). An overview of the CRONE approach in system analysis, modeling and identification, observation and control. *IFAC Proceedings Volumes*, vol. 41, no. 2, pages: 14254–14265.

Overman, E. S. (1996). The new science of management: Chaos and quantum theory and method. *Journal of Public Administration Research and Theory*, vol. 6, no. 1, pages: 75–89.

Özkaynak, F. (2018a). Brief review on application of nonlinear dynamics in image encryption. *Nonlinear Dynamics*, vol. 92, no. 2, pages: 305–313.

— (Jan. 2018b). Brief review on application of nonlinear dynamics in image encryption. *Nonlinear Dynamics*, vol. 92, no. 2, pages: 305–313.

Ozkaynak, F. (2020). A novel random number generator based on fractional order chaotic Chua system. *Elektronika ir Elektrotechnika*, vol. 26, no. 1, pages: 52–57.

Patidar, V., N. K. Pareek, and K. K. Sud (2009a). A new substitution–diffusion based image cipher using chaotic standard and logistic maps. *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 7, pages: 3056–3075.

Patidar, V., N. K. Pareek, and K. K. Sud (2009b). A new substitution–diffusion based image cipher using chaotic standard and logistic maps. *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 7, pages: 3056–3075.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A*, vol. 185, pages: 71–110.

Petras, I., L. Dorcak, P. O'Leary, B. Vinagre, and I. Podlubny (2000). The modelling and analysis of fractional-order control systems in frequency domain. *arXiv preprint math/0008186*.

Petráš, I. (2011). *eq:FracProperty*. Springer Science & Business Media.

Podlubny, I. (1999). *Fractional Differential Equations*. Academic Press.

Poincaré, H. (1890). Sur le probleme des trois corps et les equations de la dynamique. *Acta mathematica*, vol. 13, no. 1, pages: A3–A270.

— (1900). Introduction. *Acta Mathematica*, vol. 13, no. 1-2, pages: 5–7.

— (2017). *The three-body problem and the equations of dynamics : Poincaré's foundational work on dynamical systems theory*. vol. 443. Springer.

Qiao, Z., I. Taralova, and S. El Assad (Dec. 2019). A robust pseudo-chaotic number generator for cryptosystem based on chaotic maps and multiplexing mechanism. In: *Proceedings of the International Conference for Internet Technology and Secured Transactions (ICITST'2019)*, ICITST 2019, London, United Kingdom, pages: [Qiao et al] 4 pages, URL: `https://hal.archives-ouvertes.fr/hal-02430572`.

Qiao, Z., S. El Assad, and I. Taralova (2020). Design of secure cryptosystem based on chaotic components and AES S-Box. *AEU - International Journal of Electronics and Communications*, vol. 121, pages: 153205.

Qiao, Z. (2021). *Nonlinear dynamics, applications to chaos-based encryption*. 2021ECDN0016. PhD thesis. URL: `http://www.theses.fr/2021ECDN0016/document`.

Radwan, A. G., S. K. Abd-El-Hafiz, and S. H. AbdElHaleem (2012). Image encryption in the fractional-order domain. In: *Proceedings of the 2012 International Conference on Engineering and Technology (ICET)*, pages: 1–6.

Rivest, R. L. (1991). Cryptography. In: *Handbook of Theoretical Computer Science*. Ed. by J. V. Leeuwen. vol. 1. Elsevier.

S., M. K. and B. Ross (1993). An Introduction to the Fractional Calculus and Fractional Differential Equations. In.

Sahari, M. L. and I. Boukemara (2018). A pseudo-random numbers generator based on a novel 3D chaotic map with an application to color image encryption. vol. 94, no. 1, pages: 723–744.

Salamon, D. A. (2004). The Kolmogorov-Arnold-Moser theorem. eng. *Mathematical Physics Electronic Journal [electronic only]*, vol. 10, pages: Paper No. 3, 37 p.–Paper No. 3, 37 p. URL: http://eudml.org/doc/128870.

Schöll, E. and E. Scholl (2001). *Nonlinear spatio-temporal dynamics and chaos in semiconductors*. no. 10. Cambridge University Press.

Shannon, C. E. (1949). Communication theory of secrecy systems. *The Bell system technical journal*, vol. 28, no. 4, pages: 656–715.

Silva, C. (1993). Shil'nikov's theorem-a tutorial. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 10, pages: 675–682.

Stinson, D. R. (2005). *Cryptography: theory and practice*. Chapman and Hall/CRC.

Tarasov, V. E. and V. V. Tarasova (2018). Macroeconomic models with long dynamic memory: Fractional calculus approach. *Applied Mathematics and Computation*, vol. 338, pages: 466–486.

Tavazoei, M. S. and M. Haeri (2007a). A necessary condition for double scroll attractor existence in fractional-order systems. *Physics Letters A*, vol. 367, no. 1, pages: 102–113. URL: https://www.sciencedirect.com/science/article/pii/S0375960107008237.

Tavazoei, M. and M. Haeri (2007b). Unreliability of frequency-domain approximation in recognising chaos in fractional-order systems. *IET Signal Processing*, vol. 1, no. 4, pages: 171–181.

Tavazoei, M. S. and M. Haeri (2008a). Chaotic attractors in incommensurate fractional order systems. *Physica D: Nonlinear Phenomena*, vol. 237, no. 20, pages: 2628–2637.

— (2008b). Limitations of frequency domain approximation for detecting chaos in fractional order systems. *Nonlinear Analysis: Theory, Methods & Applications*, vol. 69, no. 4, pages: 1299–1320.

Teh, J. S., M. Alawida, and Y. C. Sii (2020). Implementation and practical problems of chaos-based cryptography revisited. *Journal of Information Security and Applications*, vol. 50, pages: 102421.

Tutueva, A., D. Pesterev, A. Karimov, D. Butusov, and V. Ostrovskii (2019). Adaptive Chirikov Map for Pseudo-random Number Generation in Chaos-based Stream Encryption. In: *Proceedings of the 2019 25th Conference of Open Innovations Association (FRUCT)*, pages: 333–338.

Tutueva, A. V., E. G. Nepomuceno, A. I. Karimov, V. S. Andreev, and D. N. Butusov (2020). Adaptive chaotic maps and their application to pseudo-random numbers generation. *Chaos, Solitons & Fractals*, vol. 133, pages: 109615.

Wang, Y., K.-W. Wong, C. Li, and Y. Li (2012). A novel method to design S-box based on chaotic map and genetic algorithm. *Physics Letters A*, vol. 376, no. 6-7, pages: 827–833.

Wang, Z., X. Huang, and J. Zhou (June 2013). A Numerical Method for Delayed Fractional-Order Differential Equations: Based on G-L Definition. *Applied Mathematics & Information Sciences*, vol. 7, pages: 525–529.

Wang, X. and Q. Wang (2014). A novel image encryption algorithm based on dynamic S-boxes constructed by chaos. *Nonlinear Dynamics*, vol. 75, no. 3, pages: 567–576.

Wang, Y., P. Lei, H. Yang, and H. Cao (2015). Security analysis on a color image encryption based on DNA encoding and chaos map. *Computers & Electrical Engineering*, vol. 46, pages: 433–446. URL: https://www.sciencedirect.com/science/article/pii/S0045790615000981.

Wang, H., D. Xiao, X. Chen, and H. Huang (2018). Cryptanalysis and enhancements of image encryption using combination of the 1D chaotic map. *Signal processing*, vol. 144, pages: 444–452.

Wang, X., L. Feng, R. Li, and F. Zhang (2019). A fast image encryption algorithm based on non-adjacent dynamically coupled map lattice model. *Nonlinear Dynamics*, vol. 95, no. 4, pages: 2797–2824.

Wang, X., Y. Wang, X. Zhu, and C. Luo (2020). A novel chaotic algorithm for image encryption utilizing one-time pad based on pixel level and DNA level. *Optics and Lasers in Engineering*, vol. 125, pages: 105851. URL: https://www.sciencedirect.com/science/article/pii/S0143816619302143.

Wang, X. and Y. Su (2021). Image encryption based on compressed sensing and DNA encoding. *Signal Processing: Image Communication*, vol. 95, pages: 116246.

Watson, J. D. and F. H. Crick (1953). Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid. *Nature*, vol. 171, no. 4356, pages: 737–738.

Wei, X., L. Guo, Q. Zhang, J. Zhang, and S. Lian (2012). A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system. *Journal of Systems and Software*, vol. 85, no. 2, pages: 290–299.

Wen, H. (2014). A review of the Hénon map and its physical interpretations. In.

Wen, W., Y. Zhang, M. Su, R. Zhang, J.-x. Chen, and M. Li (2017). Differential attack on a hyper-chaos-based image cryptosystem with a classic bi-modular architecture. *Nonlinear Dynamics*, vol. 87, pages: 383–390.

Wen, W., K. Wei, Y. Zhang, Y. Fang, and M. Li (2020). Colour light field image encryption based on DNA sequences and chaotic systems. *Nonlinear Dynamics*, vol. 99, no. 2, pages: 1587–1600.

Wikipedia contributors (2022). *Nonlinear system — Wikipedia, The Free Encyclopedia*. [Online; accessed 6-June-2022]. URL: `https://en.wikipedia.org/w/index.php?title=Nonlinear_system%5C&oldid=1089335711`.

Wu J, i., L. Xiaofeng, and B. Yang (2018). Image encryption using 2D Hénon-Sine map and DNA approach. *Signal Process.*, vol. 153, pages: 11–23.

Wu, X., B. Zhu, Y. Hu, and Y. Ran (2017). A novel color image encryption scheme using rectangular transform-enhanced chaotic tent maps. *IEEE Access*, vol. 5, pages: 6429–6436.

Wu, J., X. Liao, and B. Yang (2018). Image encryption using 2D Hénon-Sine map and DNA approach. *Signal Processing*, vol. 153, pages: 11–23.

Xue, X., D. Zhou, and C. Zhou (2020). New insights into the existing image encryption algorithms based on DNA coding. *PLoS One*, vol. 15, no. 10, pages: e0241184.

Yang, F., J. Mou, J. Liu, C. Ma, and H. Yan (2020). Characteristic analysis of the fractional-order hyperchaotic complex system and its image encryption application. *Signal processing*, vol. 169, pages: 107373.

Zhang, G. and Q. Liu (2011). A novel image encryption method based on total shuffling scheme. *Optics communications*, vol. 284, no. 12, pages: 2775–2780.

Zhang, Q., L. Guo, and X. Wei (2013). A novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system. *Optik - International Journal for Light and Electron Optics*, vol. 124, no. 18, pages: 3596–3600. URL: `https://www.sciencedirect.com/science/article/pii/S0030402612009126`.

Zhang, W., K.-w. Wong, H. Yu, and Z.-l. Zhu (2013). An image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion. *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 8, pages: 2066–2080.

Zhang, Q., L. Liu, and X. Wei (2014). Improved algorithm for image encryption based on DNA encoding and multi-chaotic maps. *AEU-International Journal of Electronics and Communications*, vol. 68, no. 3, pages: 186–192.

Zhang, J., D. Hou, and H. Ren (2016). Image encryption algorithm based on dynamic DNA coding and Chen's hyperchaotic system. *Mathematical Problems in Engineering*, vol. 2016.

Zhao, C.-F. and H.-P. Ren (2020). Image encryption based on hyper-chaotic multi-attractors. *Nonlinear Dynamics*, vol. 100, no. 1, pages: 679–698.

Zhou, G., D. Zhang, Y. Liu, Y. Yuan, and Q. Liu (2015). A novel image encryption algorithm based on chaos and Line map. *Neurocomputing*, vol. 169, pages: 150–157.

Zhou, N., S. Pan, S. Cheng, and Z. Zhou (2016). Image compression–encryption scheme based on hyper-chaotic system and 2D compressive sensing. *Optics & Laser Technology*, vol. 82, pages: 121–133.

Zhu, C. and K. Sun (2018). Cryptanalyzing and Improving a Novel Color Image Encryption Algorithm Using RT-Enhanced Chaotic Tent Maps. *IEEE Access*, vol. 6, pages: 18759–18770.

Zhu, C., S. Li, and Q. Lu (2019). Pseudo-random Number Sequence Generator Based on Chaotic Logistic-Tent System. In: *Proceedings of the 2019 IEEE 2nd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE)*, pages: 547–551.

APPENDIX

Appendix A. Tests in the NIST test suite

Frequency test. This test examines whether the number of zeros and ones are approximately equal in the generated sequence as it should be for a truly random sequence. The other tests won't be proceeded if the sequence fails to pass this test.

Block frequency test. This test focus on the proportion of ones with a M-bits block. The occurence of '1' should be approximately equal to $\frac{M}{2}$ as in a random sequence.

Run test This test count the total number of uninterrupted subsequences of identical bits in the sequence. It evaluates whether the subsequences of ones and zeros oscillate too fast or slow, which does not fit the requirement for random sequence.

Long run test. The test evaluate whether the length of the longest run of ones is in accordance with the expected length for the random sequence within M-bit blocks. Typical three different M values are tested where it is equal to 8, 128 and 10^4.

Rank text. Typically, this test focus on the 32×32-bits matrices rearranged from the original sequence to evaluate the linear dependence among the fixed length substrings within the sequence.

Discret fourier transform test. This test applies discrete fourier transform to the sequence to detect periodic features that deviates from randomness assumption.

Non-overlapping Template Matching Test. The test searches for a specific m-bits pattern using an m-bits window in the tested sequence and count the number of its occurrences. The window slides one bit at a time if the pattern is not find, and starts from the end of the pattern otherwise.

Overlapping Template Matching Test. The number of the occurences of a specific pre-defined m-bit pattern is of interest to this test. The window of m-bit slides one bit at a time no matter the pattern is observed or not.

Universal statistical test. The test focus on the number of bits between matching patterns to detect whether the sequence can be significantly compressed. A random sequence should not be able to compress significantly without information loss.

Linear Complexity Test. This test tries to figure out whether the sequence is

complex enough to be considered as random by focusing on testing the length of a linear feedback shift register.

Serial Test. The test evaluates all possible overlapping m-bit patterns to find out if they have the same probability of appearance.

Approximate Entropy Test. The test aims to compare the frequency of overlapping blocks with two consecutive block size(m and m+1) against the expected result.

Cumulative Sums test. The test calculates the cumulative sum of adjusted digits (0 to -1, 1 to $+1$) of the partial sequences occurring in the tested sequence to see whether it is close to the expected value for a random sequence.

Random Excursions Test. The test verifies matching an observable random visits to the given state of the cycle with the expected one. The target is to indicate deviation from the low of visits distribution if it exists.

Random Excursions Variant Test. The test calculates the total number of visits to the given state at random wandering. If the number deviates from the theoretically expected number of total visits to the given state it indicates the defect.

Titre : Conception de générateur de nombres pseudo-aléatoires chaotiques fractionnaires et application au crypto-système d'images

Mot clés : Systèmes chaotiques fractionnaires, crypto-système basé sur le chaos, chiffrement de flux, chiffrement par bloc, générateur de nombres pseudo-aléatoires

Résumé : Dans cette thèse, nous avons utilisé des systèmes chaotiques pour concevoir des générateurs de nombres pseudo-aléatoires (PRNG) et appliqué ces derniers aux cryptosystèmes en raison de leurs caractéristiques prometteuses, telles que le caractère aléatoire et la sensibilité aux conditions initiales. Les systèmes chaotiques fractionnaires, bien que moins discutés que les cartes et systèmes chaotiques classiques d'ordre entier, possèdent une complexité inhérente qui apporte de la nouveauté, de la complexité et des clés secrètes supplémentaires à la conception Chaotic PRNG (CPRNG), qui à son tour améliore la sécurité du cryptosystème.

Cette thèse a étudié les différentes approches de calcul numérique pour les systèmes chaotiques fractionnaires. Une méthode de calcul utilisant une grille non uniforme avec deux compositions de grille différentes a été proposée pour résoudre numériquement les systèmes chaotiques fractionnaires 3D. Les CPRNG Fractionnaires (FCPRNG), qui répondent aux exigences aléatoires et statistiques, ont été conçus pour la première fois en utilisant trois systèmes chaotiques fractionnaires différents. De plus, un chiffrement par flux et un chiffrement par blocs basés sur des méthodes de codage et de décodage de l'ADN ont été proposés et étudiés à l'aide des FCPRNG conçus. Les deux schémas de chiffrements ont été vérifiés comme étant sûrs et fiables.

Title: Fractional chaotic pseudo-random number generator design and application to image cryptosystem

Keywords: Fractional chaotic system, chaos-based cryptosystem, stream cipher, block cipher, pseudo-random number generator (PRNG)

Abstract: Chaotic systems have been employed to design pseudo-random number generators (PRNG) and applied to cryptosystems due to their promising features, such as randomness and sensitivity to initial conditions. The fractional chaotic systems, though much less discussed than the classical integer order chaotic maps and systems, possess intriguing intricacy which can provide novelty, complexity, and extra secret keys to the Chaotic PRNG (CPRNG) design, which in turn enhance the security of the cryptosystem.

This thesis investigated different numerical calculation approaches for fractional chaotic systems. A non-uniform gird calculation method with two different grid compositions was proposed to solve the 3D fractional chaotic systems numerically. The Fractional CPRNGs (FCPRNG), which meet the randomness and statistical requirements, were designed for the first time employing three different fractional chaotic systems. In addition, a stream cipher and a block cipher based on DNA encoding and decoding methods were proposed and studied using the designed FCPRNGs. Both ciphers have been verified to be secure and reliable.