

IntechOpen

Coding Theory

Advances and Applications in Informatics,
Data Analysis, and Cryptography

*Edited by Jonathan Blackledge
and A.F.M. Shahen Shah*



Coding Theory - Advances and Applications in Informatics, Data Analysis, and Cryptography

*Edited by Jonathan Blackledge
and A.F.M. Shahen Shah*

Published in London, United Kingdom

Coding Theory – Advances and Applications in Informatics, Data Analysis, and Cryptography
<http://dx.doi.org/10.5772/intechopen.1003419>
Edited by Jonathan Blackledge and A. F. M. Shahen Shah

Contributors

A. F. M. Shahen Shah, A. F. M. Suaib Akhter, Amor Smida, Brad Midgley, İbrahim Özen, Issam El Gmati, Jaouhar Fattahi, Jonathan Blackledge, Jonathan Blackledge, Muhammet Ali Karabulut, Raj Kumar Goswami, Ridha Ghayoula, Robert Kingstone, Sahbi Bahroun, Satyanarayana Degala, Shahen Shah, Vladimir Nikolaevich Shashikhin, Waly Mohamed, Wided Amara

© The Editor(s) and the Author(s) 2025

The rights of the editor(s) and the author(s) have been asserted in accordance with the Copyright, Designs and Patents Act 1988. All rights to the book as a whole are reserved by INTECHOPEN LIMITED. The book as a whole (compilation) cannot be reproduced, distributed or used for commercial or non-commercial purposes without INTECHOPEN LIMITED's written permission. Enquiries concerning the use of the book should be directed to INTECHOPEN LIMITED rights and permissions department (permissions@intechopen.com).

Violations are liable to prosecution under the governing Copyright Law.



Individual chapters of this publication are distributed under the terms of the Creative Commons Attribution 4.0 License which permits commercial use, distribution and reproduction of the individual chapters, provided the original author(s) and source publication are appropriately acknowledged. If so indicated, certain images may not be included under the Creative Commons license. In such cases users will need to obtain permission from the license holder to reproduce the material. More details and guidelines concerning content reuse and adaptation can be found at <http://www.intechopen.com/copyright-policy.html>.

Notice

Statements and opinions expressed in the chapters are those of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published chapters. The publisher assumes no responsibility for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained in the book.

First published in London, United Kingdom, 2025 by IntechOpen
IntechOpen is the global imprint of INTECHOPEN LIMITED, registered in England and Wales, registration number: 11086078, 167-169 Great Portland Street, London, W1W 5PF, United Kingdom

For EU product safety concerns: IN TECH d.o.o., Prolaz Marije Krucifikse Kozulić 3, 51000 Rijeka, Croatia, info@intechopen.com or visit our website at intechopen.com.

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library

Coding Theory – Advances and Applications in Informatics, Data Analysis, and Cryptography

Edited by Jonathan Blackledge and A. F. M. Shahen Shah

p. cm.

Print ISBN 978-1-83634-035-5

Online ISBN 978-1-83634-034-8

eBook (PDF) ISBN 978-1-83634-036-2

If disposing of this product, please recycle the paper responsibly.

IntechOpen

intechopen.com

Built by scientists, for scientists



Explore all IntechOpen books

Meet the editors



Jonathan Blackledge holds a BSc in Physics, a Diploma of Imperial College, a Ph.D. in Theoretical Physics (London University), and a Doctorate in Mathematical Information Technology from the University of Jyväskylä. He has published over 300 research articles and books and has supervised numerous postgraduate students, including 78 Ph.D. research assistants and 28 postdoctoral research fellows. After a parallel career in the UK scientific civil service and academia, he was appointed to the Science Foundation Ireland's Stokes Professorship in 2008. He holds Professorships with several research institutes and universities worldwide, including the University of KwaZulu-Natal in South Africa, where he was previously the Deputy Vice Chancellor for Research. Current research interests include the applications of Artificial Intelligence for Post-Quantum Cryptography.



A. F. M. Shahan Shah is an Associate Professor with the Department of Electronics and Communication Engineering and Director of the AI and Next-generation Wireless Communication Laboratory at Yildiz Technical University, Türkiye. He received the Ph.D. degree in Electronics and Communication Engineering from Yildiz Technical University, Türkiye. His current research interests include wireless communication, artificial intelligence, 6G, blockchain, IoT, etc. He has been a senior member of IEEE since 2019. He is currently serving as Editor-in-Chief of ICCK Transactions on Mobile and Wireless Intelligence, and ICRRD Quality Index Research Journal, Editor of the Open Transportation Journal (Bentham) and Discover Vehicles (Springer), and Associate Editor of the Journal of Cyber Security Technology (Taylor & Francis).

Contents

Preface	XI
Section 1	
Foundations of Coding Theory	1
Chapter 1	3
Introductory Chapter: Coding Theory – Advances and Applications in Informatics, Data Analysis, and Cryptography <i>by Jonathan Blackledge and A.F.M. Shahen Shah</i>	
Chapter 2	21
Perspective Chapter: Trellis Coded Modulation – A Promising Error Correction Scheme for Bandwidth Constrained Channel <i>by Raj Kumar Goswami</i>	
Chapter 3	35
Perspective Chapter: Weight/Distance Distributions and Decoding Error Probabilities of Random Rank Metric Codes <i>by İbrahim Özen</i>	
Section 2	
Modern and AI-Based Encryption Systems	49
Chapter 4	51
Perspective Chapter: Encryption of Information Based on Controlled Chaotic Systems <i>by Vladimir Nikolaevich Shashikhin</i>	
Chapter 5	71
Perspective Chapter: Application of Evolutionary Computing for Generating Encryption Algorithms with AI-Based Code Generation <i>by Jonathan Blackledge, Robert Kingstone and Brad Midgley</i>	
Chapter 6	117
Optimized RSA-1024 FPGA Implementation on Artix-7 <i>by Ridha Ghayoula, Sahbi Bahroun, Wided Amara, Jaouhar Fattahi, Amor Smida, Issam El Gmati and Waly Mohamed I</i>	

Section 3	
Future Security and Emerging Paradigms	147
Chapter 7	149
Perspective Chapter: Postquantum Security	
<i>by Satyanarayana Degala</i>	
Chapter 8	167
Information Coding and the Search for Extraterrestrial Intelligence	
<i>by Jonathan Blackledge</i>	
Chapter 9	213
Quantum-Resilient Cryptographic Techniques for AI-Driven	
6G Communication	
<i>by A.F.M. Shahen Shah, A.F.M. Suaib Akhter and Muhammet Ali Karabulut</i>	

Preface

1. Introduction

The rapid evolution of modern technology has profoundly reshaped the landscape of communications and information security. In this era of digital transformation, where global connectivity underpins nearly every aspect of human activity, the importance of robust, efficient, and secure information exchange has never been greater. Cryptography and coding theory share a common mathematical foundation in algebra, number theory, and information theory. While coding theory focuses on ensuring reliable transmission and storage of data in the presence of noise and errors, cryptography is concerned with safeguarding data from adversarial threats, ensuring confidentiality, integrity, and authenticity. In the modern landscape of data-driven technologies, these two perspectives have become increasingly intertwined. Secure communications, distributed systems, and intelligent networks require both error resilience and cryptographic protection to operate effectively.

The volume of this book, “Coding Theory – Advances and Applications in Informatics, Data Analysis and Cryptography,” is organized into three complementary sections. The three sections are *Foundations of Coding Theory*, *Modern and AI-Based Encryption Systems*, and *Future Security and Emerging Paradigms*. Together, these sections offer a comprehensive perspective on both the theoretical foundations and emerging frontiers of coding theory and cryptography, bridging academia, research, and real-world applications.

2. The Dual Pillars of Coding Theory and Cryptography: Communication and Security

Coding theory and cryptography both secure digital communication. While coding theory adds redundancy to correct errors in transmission, cryptography removes patterns to make the message unreadable to others. They are often used together, with coding theory ensuring the integrity of an encrypted message as it travels through a noisy channel. Coding theory operates at the intersection of two critical domains: communications engineering and information security. Originally conceived to address problems of data reliability and error correction in transmission systems, the discipline has expanded to encompass a wide range of applications, including cryptography, secure storage, and data compression. Today, coding theory remains essential for ensuring the integrity, confidentiality, and accessibility of information across a range of platforms, from personal devices to global communication networks.

In communications engineering, coding techniques are indispensable for mitigating noise, reducing transmission errors, and optimizing bandwidth utilization. From Shannon’s groundbreaking work on channel capacity to contemporary

advancements in low-density parity-check codes and polar codes, coding theory has consistently driven the performance of wireless, optical, and satellite communication systems.

In the realm of information security, coding theory underpins encryption algorithms, authentication protocols, and digital watermarking. The need for secure communication has grown exponentially in response to the increasing prevalence of cyber threats and the widespread dissemination of sensitive data. As quantum computing looms on the horizon, coding theory will play an instrumental role in shaping post-quantum cryptographic systems to safeguard information in the quantum era.

3. Advances and Emerging Opportunities

The journey of coding theory and cryptography has been marked by remarkable innovations, many of which are explored in this book. These include:

- **Error-Correcting Codes:** From traditional Hamming and Reed-Solomon codes to modern turbo and polar codes, error correction remains a cornerstone of reliable communication.
- **Network Coding:** A paradigm that optimizes data flow in complex networks, enabling faster and more resilient transmission paths.
- **Quantum Codes:** Bridging classical coding theory with quantum mechanics, quantum codes promise to revolutionize secure communication and computation.
- **Applications in Machine Learning:** Coding theory has begun to intersect with artificial intelligence, contributing to efficient model training and robust data encoding techniques.
- **Cryptography:** From the history of cryptography to modern cryptographic algorithms. The primary goal is to provide readers with a deep understanding of how cryptographic methods are evolving to meet the growing demand for secure communication, privacy, and data protection across various sectors.

The book also addresses challenges that remain unresolved, such as the need for lightweight codes suitable for IoT devices, designing codes resilient to adversarial attacks, understanding the full implications of quantum advancements, cryptographic algorithms, and post-quantum cryptography.

4. Structure of This Book

This book is structured to cater to a diverse audience, including researchers, practitioners, graduate students, and industry professionals. This book is organized into three sections presented as follows:

1. *Foundations of Coding Theory:* The first section, *Foundations of Coding Theory*, presents fundamental concepts, mathematical models, and analytical techniques

that form the basis of coding theory. It explores classical and modern approaches to error correction, information transmission, and performance optimization in communication systems.

2. *Modern and AI-Based Encryption Systems*: The second section, *Modern and AI-Based Encryption Systems*, focuses on recent developments in cryptographic methods and intelligent security mechanisms. It highlights the integration of artificial intelligence and machine learning in encryption, authentication, and privacy-preserving systems, reflecting current trends and innovations in secure communications.

3. *Future Security and Emerging Paradigms*: The third section, *Future Security and Emerging Paradigms*, addresses next-generation security challenges and solutions. It discusses quantum-resistant cryptography, blockchain-based protocols, and novel frameworks that shape the future landscape of data protection and information security.

Together, these sections provide a comprehensive resource on techniques for secure and reliable information transmission and protection.

5. Acknowledgments

The creation of this book would not have been possible without the contributions of researchers, engineers, and educators whose work has enriched the field of coding theory and cryptography. We extend our deepest gratitude to the academic community, whose relentless pursuit of knowledge continues to inspire innovation and drive progress. Special thanks also go to practitioners who translate theoretical advances into real-world solutions, bridging the gap between academia and industry.

Finally, we acknowledge the readers of this book. Whether you are a student taking your first steps into the subject, a professional seeking deeper insights, or a researcher aiming to push the boundaries of the field, we hope this book serves as a valuable resource and a catalyst for further exploration.

6. Conclusion

Coding theory stands as a testament to the power of mathematics and engineering in addressing some of the most pressing challenges of our time. Its applications extend far beyond its origins, influencing fields as diverse as biology, economics, and artificial intelligence. As we look toward a future defined by unprecedented technological growth, the principles and innovations of coding theory will undoubtedly remain central to advancing communications engineering and information security. Similarly, cryptography continues to evolve as an indispensable tool for protecting our digital world, from securing online transactions to safeguarding our privacy in the age of big data.

We hope that this book provides a thorough understanding of the subject and inspires new ideas and solutions in the ever-evolving landscape of technology. By exploring

both coding theory and cryptography, we aim to demonstrate how these two fields are fundamentally linked in their shared mission of ensuring reliable and secure information exchange. It is hoped that the book will provide readers with a valuable collection of research articles on the fascinating world of coding theory and cryptography, where mathematics meets engineering to shape the future of information exchange and protection.

Jonathan Blackledge

School of Electrical and Electronic Engineering,
Technological University Dublin,
Dublin, Ireland

A.F.M. Shahen Shah

Department of Electronics and Communication Engineering,
Yildiz Technical University,
Istanbul, Türkiye

Section 1

Foundations of Coding Theory

Introductory Chapter: Coding Theory – Advances and Applications in Informatics, Data Analysis, and Cryptography

Jonathan Blackledge and A.F.M. Shahen Shah

1. What is Coding Theory?

Coding theory is a branch of applied mathematics and computer science that explores how information can be efficiently and securely represented, transmitted, and stored. At its core, coding theory addresses fundamental challenges in communication systems, such as ensuring data integrity, optimising data transmission, and safeguarding information against errors and unauthorised access. It achieves this through the design and analysis of codes—structured representations of information that enable error correction, data compression, and secure encryption.

1.1 The essence of coding theory

The primary goal of coding theory is to manage the imperfections of real-world communication channels and storage systems. Noise, interference, and hardware limitations can distort or corrupt data during transmission or storage. Coding theory provides the mathematical frameworks and algorithms to detect and correct these errors, ensuring that information remains accurate and reliable [1].

In addition to improving reliability, coding theory also optimises efficiency. Data compression techniques, such as Huffman coding and arithmetic coding, reduce the amount of data transmitted or stored without compromising its content. Similarly, cryptographic codes enhance information security by encoding messages in a way that prevents unauthorised access, ensuring confidentiality and integrity [2].

1.2 Applications of coding theory

Coding theory has far-reaching applications in numerous domains, including:

- i. *Communications Systems*: From wireless networks to satellite communications, coding techniques are essential for mitigating noise and maximising data throughput.
- ii. *Data Storage*: Error-correcting codes protect data in hard drives, DVDs, and solid-state drives, ensuring long-term reliability.

- iii. *Cryptography and Security*: Coding theory underpins encryption algorithms and protocols that safeguard sensitive data from cyberattacks.
- iv. *Multimedia*: Compression algorithms allow efficient storage and streaming of audio, video, and images.
- v. *Emerging Technologies*: Coding theory plays a critical role in quantum computing, blockchain, and machine learning, where data reliability and security are paramount.

1.3 Historical and modern perspectives

The roots of coding theory can be traced back to Claude Shannon's pioneering work on information theory, which defined the limits of reliable communication over noisy channels. Hamming's introduction of error-correcting codes [3] soon followed, establishing practical techniques for managing errors. Over time, coding theory has expanded to address more complex challenges, such as multi-user communication systems, distributed storage, and quantum information.

Modern advancements in coding theory include low-density parity-check (LDPC) codes [4] and polar codes [5], which are widely used in 5G networks. Innovations in network coding, fountain codes [6], and post-quantum cryptography [7] demonstrate how the field continues to evolve in response to technological demands.

1.4 Why coding theory matters

As the digital age progresses, the volume and importance of data continue to grow. Reliable and secure communication is the backbone of modern society, enabling everything from global internet connectivity to financial transactions and autonomous systems. Coding theory ensures that this communication is not only efficient but also resilient to the complexities of the real world.

Moreover, the interdisciplinary nature of coding theory bridges mathematics, engineering, and computer science, fostering innovation across multiple fields. By providing a robust theoretical foundation and practical tools, coding theory continues to drive progress in communications, security, and beyond.

Thus, coding theory is more than just an academic discipline; it is a vital technology shaping the modern world. By addressing fundamental issues of reliability, efficiency, and security, coding theory enables the seamless flow of information that underpins contemporary life. Whether solving today's challenges or anticipating tomorrow's, coding theory remains an indispensable part of our technological journey.

2. Coding theory in communications engineering

Coding theory is a cornerstone of modern communications engineering, enabling the reliable and efficient transmission of information across noisy and imperfect channels. By designing and applying mathematical codes, engineers ensure that data can be accurately transmitted, even in the presence of interference, distortion, or errors. From the earliest telecommunication systems to today's high-speed wireless networks, coding theory has consistently pushed the boundaries of what is achievable

in communication. Thus, coding theory has a pivotal role in communications engineering, its foundational principles, key advancements, and its transformative impact on the field [8].

2.1 The need for coding theory in communications

In any communication system, the process of transmitting information is susceptible to errors due to noise, interference, and hardware limitations. Errors may arise from various sources, such as signal attenuation, thermal noise, or electromagnetic interference. These challenges necessitate mechanisms that can detect and correct errors without requiring retransmission, which may not always be feasible in real-time or high-latency systems.

Coding theory addresses this problem by introducing redundancy into the transmitted data. Through carefully designed codes, it becomes possible to:

- Detect errors that occur during transmission;
- Correct errors to recover the original information;
- Optimise bandwidth usage while maintaining data integrity.

2.2 Fundamental concepts

The theoretical framework of coding theory is built on several key concepts:

- Error Detection and Correction:* Error-detecting codes, such as parity checks, identify discrepancies in transmitted data. Error-correcting codes, such as Hamming codes, go a step further by reconstructing the original data [9].
- Channel Capacity:* Claude Shannon's groundbreaking work on information theory established the theoretical limit of data that can be reliably transmitted over a noisy channel. This limit, known as the channel capacity, serves as a benchmark for coding efficiency [10].
- Code Efficiency and Redundancy:* Efficient codes minimise the amount of additional data (redundancy) required to achieve reliable communication. Striking a balance between redundancy and error-correction capability is a central challenge in coding theory [11].

2.3 Key advancements in coding theory for communications

Over the decades, coding theory has evolved to address increasingly complex communication requirements. Notable advancements include:

- Linear Block Codes:* These codes, such as Hamming and Reed-Solomon codes, laid the foundation for systematic error correction. Reed-Solomon codes, for example, are widely used in storage devices and CDs due to their ability to correct burst errors [12].
- Convolutional Codes:* Used extensively in wireless communications, convolutional codes encode data streams for continuous error correction. They are

often combined with decoding algorithms, such as the Viterbi algorithm, for efficient error recovery [13].

- iii. *Turbo Codes*: Introduced in the 1990s, turbo codes revolutionised coding theory by achieving performance close to the Shannon limit. They use iterative decoding methods to provide excellent error correction, making them ideal for satellite and deep-space communications [14].
- iv. *Low-Density Parity-Check (LDPC) Codes*: These modern codes provide high performance for high-speed data applications, including 5G networks and Wi-Fi standards. The sparse matrix representation of these code allows for efficient encoding and decoding [15].
- v. *Polar Codes*: Recently adopted for 5G, polar codes are the first to achieve channel capacity for certain channels. Their structured nature simplifies implementation in high-throughput systems [16].

2.4 Applications in communications engineering

Coding theory has become integral to various aspects of communications engineering:

- i. *Wireless Networks*: Coding techniques improve the reliability and throughput of cellular networks, enabling seamless video streaming, voice calls, and data exchange.
- ii. *Satellite and Space Communication*: Error correction codes are essential for overcoming the significant delays and noise in space communication.
- iii. *Optical Fibre Communications*: Advanced coding schemes ensure high data rates and low error probabilities in optical networks.
- iv. *Internet of Things (IoT)*: Lightweight codes are critical for resource-constrained IoT devices, ensuring secure and efficient data transmission.
- v. *Quantum Communication*: Quantum error-correcting codes enable secure communication in quantum networks, addressing challenges unique to quantum systems.

2.5 The future of coding theory in communications

The ongoing development of communication systems, such as 6G networks and quantum communication, presents new challenges and opportunities for coding theory. Emerging areas include:

- i. *Post-Quantum Cryptography*: Designing codes that resist quantum-based attacks to ensure long-term data security [17].
- ii. *Network Coding*: Optimising data flow in distributed systems to enhance robustness and efficiency [18].

- iii. *Machine Learning Integration*: Leveraging artificial intelligence to design and decode codes dynamically for improved performance [19].

As the demand for faster, more secure, and energy-efficient communication grows, coding theory will remain a critical driver of innovation. Coding theory is therefore an indispensable element of communications engineering, enabling the reliable and efficient exchange of information in an increasingly interconnected world. Its principles and techniques have shaped the development of modern communication systems, ensuring that data remains accurate and secure even in challenging environments. As technology continues to advance, coding theory will undoubtedly play a central role in addressing the complexities of future communication systems, reinforcing its significance in both theory and practice.

3. Coding theory in information security

In an era where digital information drives economies, powers industries, and underpins societal functions, safeguarding data has become paramount. Cyber threats and unauthorised access to sensitive information pose significant risks to individuals, organisations, and nations. Coding theory, a mathematical discipline that originated to address errors in data transmission, has evolved to play a pivotal role in the domain of information security. By designing robust codes, coding theory underpins many cryptographic systems, authentication protocols, and data protection mechanisms, ensuring confidentiality, integrity, and resilience against cyberattacks. Coding theory therefore has a range of contributions in the field of information security, highlighting its foundational principles, applications, and emerging relevance in the age of quantum computing [20].

3.1 The role of coding theory in information security

At its core, coding theory addresses the efficient and reliable representation of data. While traditionally focused on correcting errors during transmission, its principles have found critical applications in securing information. The three primary objectives of information security are:

- i. *Confidentiality*: Ensuring that data is accessible only to authorised individuals;
- ii. *Integrity*: Preventing unauthorised alterations to data;
- iii. *Availability*: Guaranteeing that data remains accessible to authorised users when needed.

Coding theory contributes to these goals by:

- Encrypting data to prevent unauthorised access (confidentiality).
- Detecting tampered data using error-detecting codes (integrity).
- Encoding information for resilience against attacks and disruptions (availability).

3.2 Key applications of coding theory in information security

1. *Error Detection and Authentication*: Error-detecting codes, such as cyclic redundancy checks (CRC), are widely used to verify the integrity of data during transmission or storage. Additionally, coding techniques contribute to message authentication codes (MACs), which ensure that messages have not been altered in transit.
2. *Cryptography*: Cryptographic systems leverage coding theory to secure communications. Examples include:
 - i. *Public-Key Cryptography*: Lattice-based codes and error-correcting codes are employed in encryption schemes, particularly in post-quantum cryptography, which aims to resist attacks by quantum computers [21].
 - ii. *Symmetric-Key Cryptography*: Codes such as Advanced Encryption Standard (AES) incorporate substitution and permutation operations that have roots in coding principles [22].
3. *Steganography and Data Hiding*: Coding theory enhances steganographic techniques, where data is hidden within other data (e.g., embedding encrypted messages in images). Error-correcting codes ensure the hidden message can be accurately extracted despite noise or modifications [23].
4. *Digital Signatures*: Secure digital signatures, used for verifying the authenticity of electronic documents, rely on coding-based hash functions. These ensure that even minor changes to a document result in vastly different hashes, preventing forgery [24].
5. *Secure Storage Systems*: Redundancy introduced by error-correcting codes protects data stored on physical media, such as hard drives or cloud servers. Codes such as Reed-Solomon are used in RAID systems and distributed storage networks to recover data lost due to device failures or tampering [25].
6. *Post-Quantum Cryptography*: The advent of quantum computers threatens traditional cryptographic systems. Coding theory underpins several post-quantum cryptographic approaches, including McEliece and Niederreiter encryption schemes [26], for example, which rely on the difficulty of decoding random linear codes.

3.3 Advancements and challenges

1. *Quantum-Safe Cryptography*: Quantum computers, with their ability to solve problems such as integer factorisation efficiently, threaten many current cryptographic systems. Coding theory plays a crucial role in designing quantum-resistant algorithms, such as lattice-based cryptography and code-based cryptography. These systems use the inherent complexity of decoding large codes to ensure security [27].
2. *Lightweight Codes for IoT*: The proliferation of Internet of Things (IoT) devices necessitates secure yet lightweight cryptographic systems. Coding theory

contributes by developing efficient error-correcting and encryption codes that minimise computational and energy requirements [28].

3. *Resilience Against Side-Channel Attacks*: Side-channel attacks exploit physical implementations of cryptographic systems, such as power consumption or timing information. Coding techniques help design algorithms that resist such attacks by obfuscating operations and minimising exploitable patterns [29].
4. *Data Privacy and Homomorphic Encryption*: Coding theory enables homomorphic encryption, which allows computations to be performed on encrypted data without decrypting it. This ensures data privacy in applications, such as secure voting, confidential cloud computing, and privacy-preserving machine learning [30].

3.4 The future of coding theory in information security

As cyber threats grow, more sophisticated coding theory will remain at the forefront of innovation in information security. Areas of future development include:

- i. *Post-Quantum Era*: Developing practical and scalable post-quantum cryptographic systems to secure critical infrastructure.
- ii. *Blockchain and Distributed Ledgers*: Enhancing the security and scalability of blockchain systems through coding-based consensus mechanisms and error correction.
- iii. *AI and Security*: Integrating coding theory with machine learning to detect anomalies and predict potential security breaches.
- iv. *Biometric Security*: Using error-correcting codes to improve the accuracy and reliability of biometric systems, such as fingerprint or facial recognition.

Coding theory has thus, evolved far beyond its origins in error correction, becoming a linchpin of modern information security. By ensuring the confidentiality, integrity, and availability of data, it addresses the growing challenges posed by cyber threats and technological advancements. As the digital landscape continues to expand and threats become more complex, coding theory will remain an indispensable tool in safeguarding the information systems that underpin our connected world.

4. What is cryptography?

Since ancient times, people have sought to communicate secretly while also attempting to decipher the information exchanged by others. Traditionally, these missions were carried out by spies and couriers. In some cases, a coded language was used to write messages for added security. However, real protection often relied on keeping the courier anonymous.

With the rise of modern civilisation and the rapid expansion of computer networks, the speed of information flow has significantly increased. At the same time, new problems have emerged. For example:

- i. Most networks are not secure, allowing potential interception of messages.
- ii. This insecurity has made it necessary to encode messages using data encryption methods.

The study of these coding languages and associated methods is known as *Cryptology*, which is broadly divided into two fields:

1. *Cryptography*—the art of creating codes and safeguarding information.
2. *Cryptanalysis*—the science of breaking codes and deciphering protected data.

4.1 Origins and applications

The term *Cryptography* originates from the Greek word *kryptos*, meaning ‘hidden’. In this respect, the construction of the word is based on the following:

- Prefix ‘crypt-’ = hidden or vault
- Suffix ‘-graphy’ = writing

Cryptography refers to techniques that safeguard information and communications through the use of codes, ensuring that only the intended recipients may access and interpret the data. The term ‘cryptograph’, as distinct from ‘cryptogram’, first appeared in the nineteenth century in Edgar Allan Poe’s story *The Gold-Bug* [31]. Historically, cryptography primarily referred to *encryption*: transforming plaintext (ordinary information) into ciphertext (incomprehensible format) [32]. The reverse process, *decryption*, converts ciphertext back into plaintext.

In engineering, cryptography involves secure information and communication methods based on: (i) mathematical principles and (ii) rule-based computations (algorithms). These algorithms are used for:

- Cryptographic key generation.
- Digital signatures and verification.
- Data privacy protection.
- Secure internet browsing.
- Safe communications (e.g., credit card transactions, email)

Another related area which specifically embraces the meaning of ‘crypt’ is ‘Information Hiding’. This approach exploits the techniques of ‘Steganography’, a term that comes from the Greek words *steganos* (covered/hidden) and *graphein* (to write). Its literal meaning is ‘hidden writing’.

Steganography is the practice of hiding a secret message, file, or piece of data inside another medium (such as an image, audio file, video, or even text), so that the very existence of the hidden information is concealed.

Unlike cryptography, which makes a message unreadable to outsiders, steganography makes the message invisible. The plaintext or the ciphertext is hidden in another data type.

Hiding or ‘camouflaging’ encrypted data in this way provides another level of security. This is because it may not be obvious to a potential attacker that there is valuable information (encrypted or otherwise) embedded in the data that has been intercepted [33].

4.2 On the evolution of cryptography

Cryptography has progressed from simple manual methods to sophisticated mathematical algorithms, largely driven by the growing need for secure communication in an increasingly digital and interconnected world. Each historical period has introduced new challenges and innovations, gradually shaping the field into its modern form. **Table 1** presents a brief overview of the evolution of cryptography.

Year	Feature	Description	Impact
Ancient times.	Classical ciphers	Used simple substitution and transposition ciphers (e.g., Caesar cipher, Scytale).	Basic security, easily broken with frequency analysis.
1467	Polyalphabetic cipher	Vigenère cipher introduced, improving security over mono-alphabetic ciphers.	More resistant to frequency analysis.
1917	Mechanical encryption	The Enigma machine used for military encryption during WWI and WWII.	Provided strong encryption but was eventually broken.
1949	Modern cryptography theory	Claude Shannon formalised cryptographic security principles.	Laid the foundation for modern encryption techniques.
1976	Public-key cryptography	Diffie-Hellman key exchange introduced.	Allowed secure key exchange over insecure channels.
1977	Symmetric encryption standard.	DES adopted by NIST.	Standardised encryption but became weak over time.
1983	Public-key encryption	RSA algorithm developed.	Strong encryption but computationally expensive.
1993	Hash functions	MD5 widely used for data integrity.	Later found to be vulnerable to collisions.
1997	Hash strength improvement	Secure hash algorithm (SHA-1) introduced for stronger cryptographic hashing.	Later replaced due to vulnerabilities.
2000	Symmetric encryption upgrade	AES replaces DES.	Stronger encryption, widely used today.
2004	Improved hashing algorithm.	SHA-2 (SHA-256, SHA-512) introduced.	Stronger hashing, used in blockchain and digital signatures.
2010	Elliptic curve cryptography (ECC)	Provides strong security with smaller key sizes	Used in mobile devices and secure communications.
2016	Quantum-resistant cryptography	Research on post-quantum cryptography (PQC) begins.	Preparing for threats from quantum computers.
2020s	Blockchain and zero-knowledge proofs	Cryptography used in decentralised finance and privacy applications.	Enhances security without revealing sensitive data.
Future (2030+)	PQC adoption.	Implementation of quantum-safe algorithms.	Ensures security against quantum computing threats.

Table 1.
The evolution of cryptography.

1. Early origins

The origins of cryptography can be traced back to ancient Egypt around 2000 BCE, where hieroglyphics were used to conceal information. The full meaning of these intricate pictograms was restricted to a select few, ensuring controlled access to knowledge.

In classical antiquity, Julius Caesar (100–44 BCE) introduced one of the earliest known substitution ciphers. Distrusting his messengers, Caesar devised a technique—now known as the ‘Caesar cipher’—where each letter in a message was replaced by another letter at a fixed position ahead in the Roman alphabet. This method provided a simple yet effective way (for the time) to secure military communications. The ‘fixed position’ was, in effect, the ‘key’ to decrypting the message.

2. Pre-twentieth century

Before the twentieth century, cryptography primarily relied on simple substitution and transposition ciphers. While adequate for their time, these systems became increasingly vulnerable as literacy and cryptanalysis techniques advanced.

3. Mid-twentieth century

The mid-twentieth century marked a turning point with the development of mechanical cipher machines (e.g., the Enigma) and the introduction of the theoretical foundations for modern cryptography pioneered by mathematicians and cryptographers, such as Claude Shannon [10]. This period also laid the groundwork for computational methods in code-breaking and encryption.

4. Late twentieth century

With the rise of digital communication, more robust encryption standards were introduced, including:

- Data Encryption Standard (DES).
- Rivest–Shamir–Adleman (RSA).
- Advanced Encryption Standard (AES).

These algorithms provided stronger levels of data security, suitable for electronic communications and financial transactions, for example.

5. Twenty-first century and beyond

Modern cryptography continues to evolve with an emphasis on:

- Stronger and more efficient algorithms.
- Quantum-resistant encryption methods to counter future threats from quantum computing.

4.3 The future of cryptography in emerging technologies

Cryptography must continuously evolve to preserve *security, privacy, and effectiveness* in digital systems as technology advances and cyberthreats grow more sophisticated. Several transformative forces—including *artificial intelligence (AI)*, *decentralised security models*, *quantum computing*, and *novel encryption methods*—are expected to shape the future of cryptography.

4.3.1 Cryptography in the era of 6G

The advent of *6G networks*, offering ultra-high speeds, extremely low latency, and massive connectivity, will revolutionise communication. However, this new generation of networks will also introduce unique security challenges.

Since 6G will heavily rely on *AI and machine learning* for network optimisation, issues surrounding *data privacy, trust, and secure deployment* will become increasingly critical. Cryptography will play a central role in ensuring resilience against these challenges.

4.3.2 Post-quantum cryptography and neural networks

One of the most promising directions in contemporary research is *post-quantum cryptography*, which seeks to protect systems against the computational power of future quantum computers. Within this field, researchers are exploring *Neural Network (NN)-based cryptography* as a potential framework. Although research is still in its early stages, steady progress is being made across multiple fronts.

Key areas of interest include:

- *Public-key cryptography and secret sharing* as foundational tools for secure communication.
- *NN-based distributed systems* optimised for real-world use, particularly those that remain efficient on low-capacity devices.

4.3.3 Quantum-augmented neural networks

Another emerging area of exploration is the *integration of quantum computing with neural networks*. This involves constructing quantum circuits for neural networks and executing them on simulators or online quantum machines. Such approaches may lead to entirely new paradigms of cryptographic security by leveraging quantum principles in conjunction with machine learning techniques [34].

4.3.4 Outlook

As digital ecosystems become more *interconnected and intelligent*, cryptography will remain indispensable to securing trust and privacy. The convergence of *6G, AI, quantum computing, and advanced cryptographic techniques* promises both significant opportunities and formidable challenges, making ongoing research and innovation vital to the future of secure communications.

5. Summary: Interplay between cryptography and coding theory

Cryptography and coding theory share a deep and common mathematical foundation in disciplines, such as algebra, number theory, and information theory. Although they pursue distinct goals, their purposes often intersect.

Coding theory is primarily concerned with ensuring the reliable transmission and storage of data in the presence of noise, interference, or potential errors. In contrast, cryptography focuses on protecting information against adversarial threats, thereby guaranteeing confidentiality, integrity, and authenticity.

In today's data-driven world, the interplay between these two fields has become increasingly critical, as modern digital infrastructures require both resilience against errors and robust safeguards against malicious attacks.

5.1 Applications in communications engineering

The dual role of coding theory highlights its transformative impact across communications engineering and information security. In communications technology, coding techniques mitigate the effects of noise, channel imperfections, and data loss through mechanisms, such as error correction and data compression. Notable innovations include:

- *Linear Block Codes*—foundational techniques for reliable transmission.
- *Convolutional Codes*—widely applied in early digital communication systems.
- *Low-Density Parity-Check (LDPC) Codes*—enabling efficient error correction in modern systems.
- *Polar Codes*—applied in next-generation wireless networks, such as 5G and beyond.

These methods have revolutionised wireless communication, satellite systems, and optical networks, enabling unprecedented levels of efficiency, reliability, and scalability in global data transmission.

5.2 Applications in information security

Beyond communication systems, coding theory plays an equally pivotal role in securing information. Error-detecting and error-correcting codes underpin robust data storage, while cryptographic frameworks incorporate coding principles to develop secure encryption, authentication, and key-exchange mechanisms. Emerging areas of application include:

- *Post-quantum Cryptography*—algorithms resistant to quantum-based attacks.
- *Steganography*—concealing hidden information within ordinary data.
- *Homomorphic Encryption*—enabling secure computation on encrypted data.

The rise of quantum computing has further expanded the relevance of coding theory, driving research into secure, scalable solutions that ensure long-term data protection.

By enhancing both reliability and security, coding theory has become indispensable in the modern digital ecosystem. Its influence spans diverse sectors, including telecommunications, cloud computing, financial systems, electronic warfare, and critical infrastructure protection, positioning it as a central theme in the development of secure and resilient information technologies.

6. Conclusions and future directions

Coding theory has long served as a cornerstone of both communications engineering and information security, providing essential solutions to the challenges of data transmission and protection.

In *communications*, coding theory ensures the reliable flow of information across noisy and imperfect channels, enabling the rapid evolution of technologies, such as 5G and next-generation wireless systems.

In *information security*, it underpins encryption, authentication, and error correction, thereby safeguarding sensitive data in an increasingly interconnected and data-driven world.

Looking ahead, the significance of coding theory will only intensify. Emerging paradigms—including quantum computing, the Internet of Things (IoT), and cloud-based infrastructures—demand more sophisticated coding solutions to manage growing complexity. At the same time, innovations in artificial intelligence (AI) and machine learning are beginning to intersect with coding theory, opening possibilities for novel approaches to error detection, adaptive compression, and secure communication protocols. Through this convergence, coding theory will remain central to ensuring both reliability and security in digital systems (e.g., [35, 36]).

6.1 The future of cryptography

In parallel, the future of cryptography will be shaped by the need to confront quantum threats, safeguard privacy, and adapt to increasingly intricate technologies. The field is evolving from traditional encryption methods toward advanced paradigms, such as post-quantum algorithms, quantum cryptography, and AI-driven security techniques.

Cryptography will continue to underpin digital trust by enabling secure communication, privacy preservation, and technological innovation in a hyper-connected society.

6.2 Cryptography in 6G networks

Cryptography is expected to play a defining role in the deployment of 6G networks. With their ultra-high speeds, minimal latency, and vast device connectivity, 6G infrastructures will require security mechanisms far beyond conventional encryption. These security issues will include:

- *Quantum-safe cryptography* to ensure resilience against quantum-based threats.
- *AI-enhanced security frameworks* to support adaptive, intelligent defence mechanisms.
- *Privacy-preserving computation* to guarantee trust and data protection across diverse user cases.

These advanced methods will ensure that 6G networks remain secure, resilient, and trustworthy in the face of evolving challenges.

Together, coding theory and cryptography form the dual pillars of modern information systems, advancing reliability, security, and trust in the technologies that drive global progress.

Author details

Jonathan Blackledge^{1*} and A.F.M. Shahan Shah²

1 School of Electrical and Electronic Engineering, Technological University Dublin, Dublin, Ireland

2 Department of Electronics and Communication Engineering, Yildiz Technical University, Istanbul, Türkiye

*Address all correspondence to: jonathan.blackledge59@gmail.com

IntechOpen

© 2025 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] McEliece RJ. The theory of information and coding. A mathematical framework for communication. In: Encyclopedia of Mathematics and its Applications. Vol. 3. London UK: Addison-Wesley Publishing Company; 1977. ISBN 10: 0201135027/ISBN 13: 9780201135022
- [2] Howard M, LeBlanc D. Writing Secure Code. 2nd ed. New Jersey, USA: Microsoft Press; 2003. ISBN 10: 0735617228/ISBN 13: 9780735617223
- [3] Huffman WC. Fundamentals of Error-Correcting Codes. Cambridge UK: Cambridge University Press; 2010. ISBN 10: 0521131707/ISBN 13: 9780521131704
- [4] Franceschini M, Ferrari G, Raheli R. LDPC Coded Modulations, Signals and Communication Technology. Heidelberg, Germany: Springer; 2009. DOI: 10.1007/978-3-540-69457-1
- [5] Gazi O. Polar Codes a Non-Trivial Approach to Channel Coding. Springer Topics in Signal Processing. Heidelberg, Germany: Springer; 2019. DOI: 10.1007/978-981-13-0737-9
- [6] Blake IF. Essays on. Coding Theory (2 - Coding for Erasures and Fountain Codes). Cambridge, UK: Cambridge University Press; 2024. DOI: 10.1017/9781009283403
- [7] Bernstein DJ, Buchmann J, Dahmen E. Post-Quantum Cryptography. Heidelberg, Germany: Springer; 2008. ISBN 10: 3540887016/ISBN 13: 978-3540887010
- [8] Cobley P, Schulz PJ. Theories and Models of Communication. Berlin, Boston: De Gruyter Mouton; 2013. DOI: 10.1515/9783110240450
- [9] Clark GC. Error-Correction Coding for Digital Communications (Applications of Communications Theory). Heidelberg, Germany: Springer; 2024. ISBN 10: 0306406152/ISBN 13: 9780306406157
- [10] Shannon CE. A mathematical theory of communication. The Bell System Technical Journal. 1948;27:379-423, 623-656. [Online] Available from: <https://people.math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf> [Accessed: December 18, 2024]
- [11] Hailman JP. Coding and Redundancy: Man-Made and Animal-Evolved Signals. Cambridge, USA: Harvard University Press; 2008. ISBN 10: 780674027954/ISBN 13: 978-0674027954
- [12] Wicker SB. Reed-Solomon Codes and their Applications. New Jersey, USA: Wiley; 1999. ISBN 10: 0780353919/ISBN 13: 978-0780353916
- [13] Dholakia A. Introduction to convolutional codes with applications. In: Springer International Series in Engineering and Computer Science. Heidelberg, Germany: Springer; 1994. DOI: 10.1007/978-1-4615-2712-1
- [14] Yuan J, Vucetic B. Turbo Codes: Principles and Applications, the Springer International Series in Engineering and Computer Science. Heidelberg, Germany: Springer; 2000. ISBN 13: 9780792378686
- [15] Lin S, Li DJ. Fundamentals of Classical and Modern Error-Correcting Code (Chapter 11: Introduction to Low-Density Parity-Check Codes, 406-463). Cambridge, USA: Cambridge University Press; 2022. DOI: 10.1017/9781009067928

- [16] Giar P. High-Speed Decoders for Polar Codes. Heidelberg, Germany: Springer International Publishing; 2018. ISBN: 9783319866994
- [17] Cheon JH, Johansson T. Post-quantum cryptography, 13th international workshop, PQCrypto 2022, virtual event, September 28-30, 2022. In: Proceedings, Lecture Notes in Computer Science. Heidelberg, Germany: Springer; 2022. DOI: 10.1007/978-3-031-17234-2
- [18] Greferath M, Pavcevic MO, Silberstein N, Vazquez-Castro MA. Network Coding and Subspace Designs. Switzerland: Springer Cham; 2018. DOI: 10.1007/978-3-319-70293-3
- [19] Alkhateeb A, Rueda L. Machine Learning Methods for Multi-Omics Data Integration. Switzerland: Springer Cham; 2024. DOI: 10.1007/978-3-031-36502-7
- [20] LaPierre R. Introduction to Quantum Computing. Switzerland: Springer Cham; 2021. DOI: 10.1007/978-3-030-69318-3
- [21] Wang X, Xu G, Wang M, Meng X. Mathematical Foundations of Public Key Cryptography. Florida, USA: CRC Press; 2020. ISBN: 9780367575434
- [22] Ramkumar M. Symmetric Cryptographic Protocols. Switzerland: Springer Cham; 2016. DOI: 10.1007/978-3-319-07584-6
- [23] Mayer J. Steganography - the Art of Hiding Information. London, UK: IntechOpen; 2024. DOI: 10.5772/intechopen.1001493. ISBN: 978-0-85466-901-1
- [24] Katz J. Digital Signatures, Springer-Verlag. USA: New York Inc.; 2010. ISBN: 9780387277110
- [25] Yang K, Jia X. Security for Cloud Storage Systems. New York, USA: Springer; 2014. DOI: 10.1007/978-1-4614-7873-7
- [26] Yadev SP, Singh R, Yadav V, Al-Turjman F, Kumar SA. Quantum-Safe Cryptography Algorithms and Approaches: Impacts of Quantum Computing on Cybersecurity, De Gruyter, Berlin, Germany. 2023. ISBN:9783110798364
- [27] ETSI (European Telecommunications Standards Institute). Quantum Safe Cryptography and Security: An Introduction, Benefits, Enablers and Challenges, ETSI White Paper No. 8. ETSI 06921 Sophia Antipolis CEDEX, France. ISBN: 979-10-92620-03-0. [Online]. Available from: <https://www.etsi.org/images/files/etsiwhitepapers/quantumsafewhitepaper.pdf> [Accessed: December 19, 2024]
- [28] Batra N, Goyal S. IoT Fundamentals with a Practical Approach. Florida, USA: CRC Press; 2024. ISBN: 9781032309705
- [29] Gui Y. Secure Cryptographic Designs Resilient to Side-Channel Attacks. Charlotte, USA. ISBN: 979-8-7282-2919-3. [Online]: The University of North Carolina at Charlotte; 2021. DOI: 10.5555/AAI28417730 [Accessed: December 20, 2024]
- [30] Lauter K, Dai W, Laine K. Protecting Privacy through Homomorphic Encryption. Berlin, Germany: Springer; 2021. DOI: 10.1007/978-3-030-77287-1
- [31] Whalen T. The code for gold: Edgar Allan Poe and cryptography. Representations. 1994;46(46):35-37
- [32] Kahn D. The Codebreakers. New York, USA: Macmillan; 1967. ISBN: 0-684-83130-9

[33] Yahya A. Steganography Techniques for Digital Images. Switzerland AG: Springer Nature; 2019. ISBN: 9783030132637

[34] Meraouche I, Dutta S, Tan H, Sakurai K. Neural networks-based cryptography: A survey. IEEE Access. 2021;**9**:124727-124740

[35] Radhakrishnan S, Naduvath S. Coding Theory - Recent Advances, New Perspectives and Applications. London, UK: IntechOpen; 2022. DOI: 10.5772/intechopen.92539. [Online] Available from: <https://www.intechopen.com/books/10522> [Accessed: December 19, 2024]

[36] Harkut D, Kasat KN. Coding Theory Essentials. London, UK, London, UK: IntechOpen; 2023. DOI: 10.5772/intechopen.104071. [Online] Available from: <https://www.intechopen.com/books/12064> [Accessed: December 19, 2024]

Perspective Chapter: Trellis Coded Modulation – A Promising Error Correction Scheme for Bandwidth Constrained Channel

Raj Kumar Goswami

Abstract

Error correction coding and modulation are integral components in digital communication systems, with modulators encoding information onto carrier signals for efficient transmission, while demodulators decipher received signals amidst noise and distortions. This encoding and decoding interplay forms the foundation of reliable communication, with channel encoding crucial for fortifying data against transmission errors, and decoding ensuring data integrity by rectifying errors. However, a trade-off exists between code rate reduction for error robustness and increased redundancy necessitating wider bandwidth. Trellis Coded Modulation (TCM) emerges as innovative, seamlessly combining encoding and modulation to achieve significant coding gain while maintaining bandwidth efficiency, exemplifying the delicate balance in pursuit of communication efficiency.

Keywords: trellis coded modulation, convolutional coding, maximum likelihood detection, AWGN, fading channel, rayleigh channel, rician channel, BER

1. Introduction

In the intricate landscape of constructing digital communication systems, error correction coding and modulation stand as fundamental pillars, each contributing distinct yet interconnected functionalities. Modulators, as the name suggests, are primarily tasked with encoding information onto carrier signals, a process crucial for efficient transmission and optimal bandwidth utilization. Their focus lies in imbuing data with the necessary attributes to traverse communication channels effectively. Conversely, demodulators serve as the counterpart, entrusted with the challenging task of deciphering received signals, and navigating through the maze of noise and distortions inherent in transmission. This intricate combination of encoding and decoding forms the backbone of reliable communication.

Central to this framework is channel encoding, a pivotal process wherein information undergoes transformation into a specific code, fortifying it against the dangers of

transmission errors. This fortification becomes indispensable as data traverses communication channels, encountering various obstacles during transmission. Decoding, the complementary process, entails not only retrieving the original data from encoded signals but also rectifying any errors that may have crept in during transmission, so as to ensuring data integrity.

In the relentless pursuit of efficiency, a strategic trade-off emerges i.e., the reduction of code rate to enhance robustness against errors, albeit at the cost of increased redundancy within the code, necessitating a broader bandwidth. It is within this delicate balance that Trellis Coded Modulation (TCM) emerges as a beacon of innovation. TCM seamlessly intertwines encoding and modulation, marrying the virtues of both to achieve substantial coding gain while steadfastly maintaining bandwidth efficiency.

2. Background

The seamless integration of error correction coding methods with compatible modulation schemes has long been a cornerstone of communication system design. Well before the advent of TCM, the concept of employing multilevel modulation for symbols undergoing convolutional encoding had already gained recognition. This approach, outlined in [1], laid the groundwork for enhancing the robustness of transmitted signals against errors. By expanding the signal set, not only does the encoding process introduce necessary redundancy vital for error correction, but it also serves to diminish the spacing between signal points while keeping the average energy constant. This crucial phenomenon is illustrated in **Figure 1**, where the impact of signal set expansion on the spacing between signal points is clearly explained. Such integration of encoding and modulation not only bolsters the resilience of communication

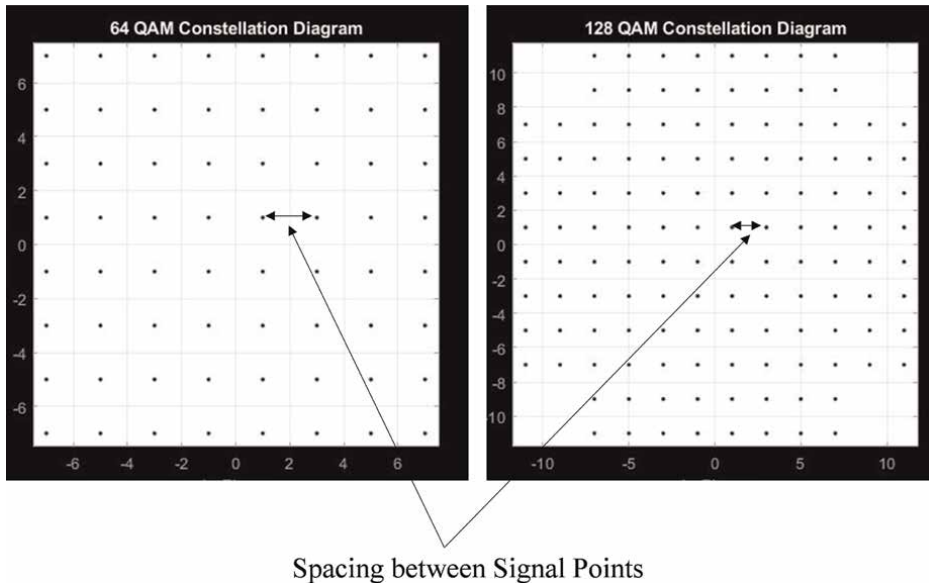


Figure 1.
Expanding a signal set results in a decrease in the separation between signal points.

systems but also optimizes spectral efficiency, thereby paving the way for more reliable and efficient data transmission in the ever-evolving landscape of digital communication.

The reduction in the gap between signal points resulting from the enlargement of the signal set can inadvertently contribute to an increase in the error rate, as documented in [2]. This occurrence arises from the heightened susceptibility of closely spaced signal points to interference and noise, thereby compromising the fidelity of transmitted data. However, to mitigate this detrimental effect and maintain the integrity of communication in bandwidth-limited environments, innovative coding methods such as TCM have emerged. TCM stands as an effective coding technique designed to counteract the harmful consequences of signal set expansion while simultaneously achieving coding gain, as explained in [3]. This coding gain denotes the improvement in communication performance attained through error correction coding, translating to a higher signal-to-noise ratio and enhanced data reliability. Importantly, TCM accomplishes this feat without sacrificing precious bandwidth resources, making it an indispensable tool for realizing reliable high-data-rate communication across channels constrained by bandwidth limitations. Thus, TCM represents a pivotal advancement in the realm of digital communication, offering a potent solution to the perennial challenge of achieving robust and efficient data transmission in bandwidth-constrained scenarios.

The attraction of TCM is particularly pronounced in terrestrial radio communications, where the availability of power and spectrum is inherently limited and immensely valuable. In such a domain, maximizing the efficiency of data transmission becomes paramount, necessitating innovative solutions that can adeptly navigate the constraints imposed by these scarce resources. At the heart of TCM lies a fundamental breakthrough—the conceptual fusion of convolutional encoding and modulation into a unified and cohesive process, as opposed to treating them as discrete entities. This paradigm shift marks a departure from conventional approaches, where encoding and modulation were typically considered as separate tasks.

By seamlessly integrating convolutional encoding with modulation, TCM unleashes a potent synergy that yields notable benefits in terms of effectiveness and efficiency within the communication system. This integrated approach capitalizes on the complementary strengths of both encoding and modulation, thereby enhancing the overall robustness and reliability of communication. The unification of these traditionally distinct processes not only streamlines the transmission system but also facilitates more effective utilization of available resources, thereby optimizing performance.

Central to the efficacy of TCM is its ability to strengthen noise resistance compared to uncoded transmission schemes. By imbuing the modulated signals with inherent redundancy through convolutional encoding, TCM fortifies them against the harmful effects of noise and interference encountered during transmission. This stimulated flexibility enables TCM to maintain signal integrity and fidelity even in the presence of adverse channel conditions, ensuring reliable communication over terrestrial radio links.

One of the remarkable aspects of TCM is its ability to achieve significant improvements in communication reliability without the need for expanding signal bandwidth or increasing transmission power. This represents a significant departure from traditional methods, where enhancing system performance often necessitated resource-intensive measures such as broader bandwidth allocations or heightened transmission power levels.

The evaluation of TCM typically revolves around quantifying the coding gain, it provides relative to an unencoded signal. Coding gain serves as a crucial metric for

assessing the effectiveness of TCM in mitigating the effects of channel noise and enhancing communication reliability. Essentially, coding gain measures the improvement in system performance achieved through the utilization of TCM compared to transmitting data without encoding.

By analyzing the coding gain, researchers and engineers can gain insights into the extent to which TCM enhances the robustness of communication systems. This metric encapsulates the tangible benefits conferred by TCM in terms of noise resilience and error correction capabilities. Moreover, it provides a standardized means of evaluating and comparing different TCM implementations, enabling informed decision-making in system design and optimization efforts.

The ability of TCM to deliver substantial coding gain without requiring additional bandwidth or transmission power underscores its efficacy as a versatile and resource-efficient coding technique. This capability positions TCM as a compelling solution for enhancing communication reliability across a diverse range of applications and scenarios, from terrestrial radio networks to satellite communications and beyond.

In the intricate design process of TCM, meticulous attention is paid to enhancing the minimum free distance by establishing a cohesive link between the chosen modulation scheme and the configuration of the trellis. This strategic approach ensures that the TCM system is proficient at optimizing the separation between individual transmitted symbols, thereby strengthening the code's error correction capabilities.

The translation of symbols from the output of the encoder to the input of the modulator within TCM is a distinct process. Unlike straightforward mappings, this involves partitioning the signal constellation into discrete signal sets, from which symbols are subsequently allocated. This division of the signal constellation is a key element enabling efficient encoding and modulation simultaneously, thus enhancing error correction capabilities and overall system performance, as elucidated in [4].

Within the framework of TCM, the choice of signal set is also governed by the output of the encoder, while systematic input is introduced to designate the specific symbol within the chosen signal set. This dual-step procedure, comprising the selection of signal sets and the mapping of symbols, is meticulously orchestrated to ensure an optimized arrangement for encoding and modulation. By harmonizing these processes, TCM achieves effective error correction and maximizes system efficiency.

The seamless integration of encoding and modulation within TCM pivots on the strategic partitioning of the signal constellation and the careful allocation of symbols. This holistic approach not only enhances error correction capabilities but also ensures efficient utilization of available resources, thereby elevating the performance and reliability of communication systems in various scenarios and environments.

Within the realm of convolutional codes, pairwise errors represent a prominent and impactful error type that can significantly undermine the reliability of the decoding process. These errors occur when the decoder erroneously selects a sequence different from the one originally transmitted by the sender, leading to potentially severe consequences. Pairwise errors in convolutional code decoding often stem from inaccuracies in state transitions, where the decoder misinterprets the received signal due to noise or other channel impairments.

To mitigate the impact of pairwise errors, a crucial step involves optimizing the Euclidean distance between diverging and emerging branches within the convolutional code decoding process, as highlighted in [5]. This optimization entails strategically allocating constellation points to these branches based on their respective Euclidean distances. By integrating modulation with the convolutional encoder in this manner, the encoding process becomes attuned to the characteristics of the

modulation scheme, thereby emphasizing the separation between signal points and enhancing error correction capabilities.

In the context of TCM, the detection process during decoding relies on soft decision rather than hard decisions. Unlike binary determinations based solely on the received signal, soft-decision decoding evaluates the probabilities associated with various symbol values, taking into account both noise and channel conditions. This approach enables the decoder to make more informed decisions, effectively distinguishing between different symbol possibilities and improving the accuracy of the decoding process. By incorporating soft-decision decoding, TCM enhances its resilience to noise and other impairments, thereby further bolstering its error correction capacities and overall reliability in communication systems.

The adoption of soft-decision decoding within TCM fosters robust decoding, particularly in scenarios where the received signal is susceptible to noise or interference. In contrast to hard-decision demodulation, which involves making binary conclusions about received symbols, soft-decision decoding enables a more critical assessment by considering the probabilities associated with different symbol values. This approach prevents the irretrievable loss of information that often accompanies hard-decision demodulation, thereby preserving valuable insights into the reliability or confidence levels associated with the received symbols.

In a coded scheme, employing hard-decision demodulation prior to decoding can result in a detrimental loss of information, ultimately leading to a decline in the signal-to-noise ratio (SNR). This loss occurs because hard-decision demodulation discards crucial data regarding the reliability of the received symbols, making it challenging for the decoder to accurately reconstruct the original information. Consequently, the performance of the system may suffer, as the ability to rectify errors and conduct successful decoding operations becomes compromised.

However, in fading channel scenarios with soft-decision decoding, particularly when employing the maximum likelihood criterion, the optimal sequence decoder's decision rule is based on the Euclidean distance. By considering this distance metric, the decoder can effectively evaluate the likelihood of different symbol sequences given the received signal and channel conditions. This sophisticated decision-making process enables the decoder to make informed choices, thereby improving the accuracy and reliability of the decoding operation even in challenging communication environments characterized by fading channels and noise. Soft-decision decoding enhances the robustness and performance of TCM systems, ensuring reliable communication in adverse conditions.

The soft-decision decoder in TCM employs a sophisticated approach to select the sequence with the highest likelihood of being the transmitted one by assessing the Euclidean distances between received symbols and potential transmitted symbols. By considering these distances, the decoder can accurately differentiate between possible transmitted sequences, taking into account the effects of fading, noise, and other channel anomalies. This reliance on the Euclidean distance metric enables the decoder to make informed decisions regarding the most probable transmitted sequence, even in challenging communication environments. By accurately separating the likely transmitted sequence from the received signal, the decoder enhances the reliability and efficiency of the decoding process within TCM systems.

Moreover, when coupled with a suitable interleaver, TCM offers additional coding advantages over uncoded schemes, particularly in fading channels. The interleaver helps to spread errors in time, mitigating the impact of burst errors commonly encountered in such channels. However, it is crucial to adhere to well-defined design criteria, as outlined

in [6], during the code creation process. These criteria ensure that the TCM scheme is optimized for the specific communication scenario, maximizing its effectiveness in combating channel impairments, and achieving robust performance. Thus, by carefully considering design principles and leveraging advanced decoding techniques like soft-decision decoding, TCM emerges as a powerful tool for reliable and efficient data transmission in fading channels and other challenging communication environments.

3. Design

Figure 2 provides a comprehensive block diagram describing the fundamental architecture of TCM scheme as applied within a fading multipath channel, as referenced in [7]. The process initiates with input bits, which are subjected to encoding via a convolutional encoder, resulting in the generation of a series of signals denoted as S_i . Each signal in this series corresponds to a n -dimensional vector selected from the $2^{(n+1)}$ signal-set, with the index ‘ i ’ denoting the present time index, as elucidated in [8]. This diagram serves as an illustrative representation of the intricate flow of information and signal manipulation within the TCM scheme, emphasizing the sequential stages of encoding and modulation. The utilization of complex notation allows for the representation of every signal within the TCM scheme as a point within a complex plane. This notation facilitates the visualization and analysis of signal characteristics and interactions, aiding in the comprehension of the scheme’s operation and performance within the context of a fading multipath channel. By presenting the architecture in a clear and structured manner, **Figure 2** offers valuable insights into the underlying mechanisms and processes governing TCM operation, thereby enhancing understanding and facilitating further exploration and analysis.

The visual representation provided by the block diagram offers an accessible means to comprehend and analyze the signals within the TCM scheme. Additionally, the coded signals can undergo interleaving—a crucial procedure aimed at reorganizing them to mitigate burst errors. Interleaving plays a pivotal role in enhancing the collective error correction capacities of the system, thereby bolstering the effectiveness of the TCM scheme in the face of channel irregularities, as emphasized in [9].

Before undergoing modulation and transmission across the channel, it is imperative to subject the signal to pulse shaping. This process is designed to counteract Inter-Symbol Interference (ISI), a phenomenon that arises due to the overlapping of

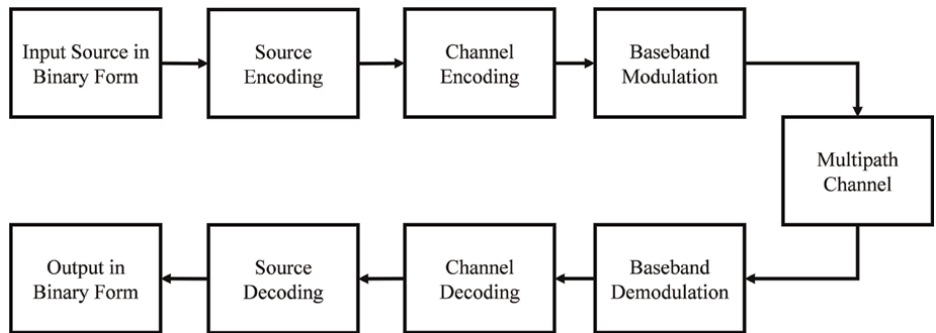


Figure 2.
System block diagram.

symbols in the temporal domain. By implementing pulse shaping techniques, the transmitted signal acquires a well-defined temporal profile, thereby reducing the likelihood of symbol overlap and diminishing ISI effects.

Despite these preparatory measures, the transmitted signal inevitably encounters the influence of Additive White Gaussian Noise (AWGN), which is an inherent characteristic of communication channels. This noise factor introduces randomness into the received signal and can significantly impact its overall quality. Consequently, it becomes imperative for communication systems to employ robust error correction techniques and modulation schemes capable of mitigating the adverse effects of noise and ensuring reliable signal reception, even in the presence of AWGN.

Upon reception, the received signal undergoes demodulation and quantization to prepare for the subsequent decoding phase. Additionally, a channel estimator may be employed to assess the channel's gain, providing valuable insights into its current state. By incorporating channel gain estimation, the decoder gains the ability to dynamically adapt its decoding strategy to accommodate the fluctuating channel conditions, thus enhancing the overall performance and reliability of the system.

The sequence $r_1 = (r_1, r_2, \dots, r_l)$ serves as the input for the TCM decoder, which employs Maximum Likelihood (ML) decoding, as outlined in [10]. Through ML decoding methods, the TCM decoder endeavors to identify the most probable transmitted sequence by analyzing the received sequence of symbols. By assessing the likelihood of different transmitted sequences, the decoder aims to accurately reconstruct the originally transmitted data, thereby mitigating any errors that may have occurred during the transmission process or through the channel.

By employing sophisticated decoding techniques such as ML decoding, the TCM decoder maximizes its ability to recover the transmitted data accurately, even in the presence of noise and other channel impairments. This robust decoding approach ensures the reliability and integrity of the communication system, contributing to seamless and dependable data transmission across a variety of communication channels and scenarios.

Figure 3 illustrates discrete time model that corresponds to system shown in **Figure 2**. This model provides a representation of received signal at the specific time index 'i', which can be formulated as follows:

$$r_i = c_i \cdot s_i + n_i \quad (1)$$

In the eq. (1), n_i represents complex Gaussian noise characterized by a mean of zero and a variance of $N_0/2$. Meanwhile, c_i signifies the complex channel gain, also conforming to a Gaussian distribution (complex) with σ_c^2 as variance.

$$c_i = a_i \cdot e^{j\phi_i} \quad (2)$$

Further, an alternative expression for c_i is

In the eq. (2), a_i denotes the amplitude, and ϕ_i represents the phase.

Presuming coherent detection at the receiver, the receiver undertakes compensation for channel's phase shift and therefore, eq. (1) can be rewritten as:

$$r_i = a_i s_i + n_i \quad (3)$$

In eq. (3), a_i signifies amplitude of noise.

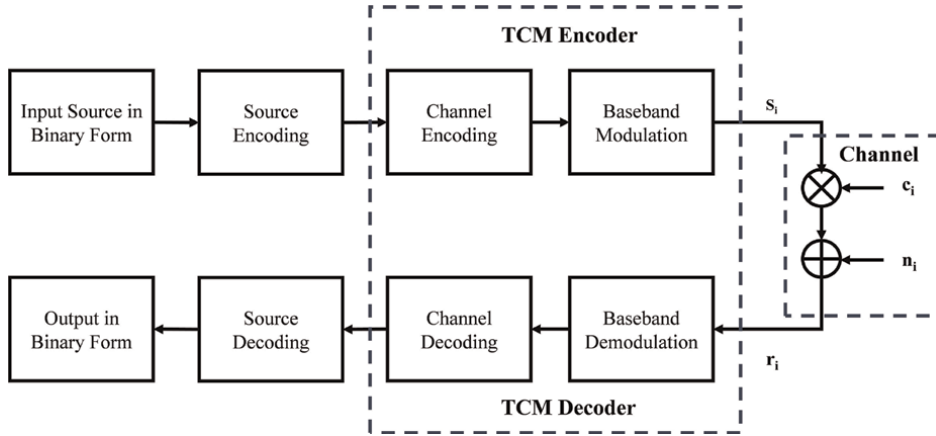


Figure 3.
The system model in **Figure 2** represented in baseband.

4. Implementation

The foundation for designing and building TCM codes for fading channels will be discussed in this section. These design rules act as a roadmap for creating TCM codes that can excel in both performance and resilience within the context of fading channels. These rules have been derived from analogous guidelines previously proposed for TCM codes designed for fading channels [11, 12] having a $2/3$ rate. Expanding upon these, the rules for designing rate $n/n + 1$, $2^{(n+1)}$ state TCM codes have been worked out in order to take care of the errors that are introduced by fading channels. Ungerboeck utilized heuristics to design an 8-state, Rate $2/3$ TCM scheme in respect of fading channels [7].

The signals pertaining to the shifts between states across successive stages are symbolized through a $2^{(n+1)} \times 2^{(n+1)}$ matrix. Each entry within the matrix corresponds to a signal linked with the transition from i^{th} state in k^{th} stage to j^{th} state in $(k + 1)^{\text{th}}$ stage within the trellis. This matrix offers an all-inclusive representation of the signal connections and shifts intrinsic to the TCM scheme, facilitating the scrutiny and shaping of the system's architecture. Further, within the matrix, elements of i^{th} row represent signals associated with paths originating from i^{th} state, whereas elements of j^{th} column represent signals associated with paths converging at j^{th} state. This configuration distinctly portrays the flow of signals and the interconnections among distinct states within the TCM scheme.

By employing set partitioning techniques, the $2^{(n+1)}$ signal set can be effectively separated into two distinct subsets based on their minimum intra-set distance. This division streamlines the arrangement and evaluation of the signal set, facilitating the identification and utilization of specific subsets within the TCM framework. The following set of rules has been introduced to assign signal points to elements within a matrix.

- a. Initial rule dictates that a signal is restricted to appear once only within a designated column or row.
- b. The second rule dictates that within a $2^{(n+1)}$ -State, Rate $n/(n + 1)$ code, not all transitions are feasible due to the constraint of having only 2^n paths originating from each state. Therefore, a signal can be connected to a transition pathway between two states solely if the Least Significant Bit of originating state's label

corresponds to the identical bit, $y \in \{0, 1\}$, as the Most Significant Bit of the target state's label.

- c. Corresponding to a given y value, all associated signals must exclusively be selected from either subset.

5. Performance analysis

The performance analysis considers the scenario where fading amplitudes are independent statistically and channels is memoryless [13]. Probability of error event plays a pivotal role in assessing the performance of the system means lower error event probabilities correlate with less errors, and *vice-versa*. Assuming detection is coherent, Channel State Information is ideal and fading for every symbol is independent, the upper limit for pairwise error probability during symbol sequence decoding in a Rician Channel scenario may be formulated as follows:

$$P_2(S_l, \hat{S}_l) \leq \prod_{i=1}^l \frac{(1+K)}{1+K+\frac{1}{4N_0}} \frac{e^{\left(-\frac{K}{1+K+\frac{1}{4N_0}} \frac{|(S_i - \hat{S}_i)|^2}{4N_0}\right)}}{|(S_i - \hat{S}_i)|^2} \quad (4)$$

In scenarios characterized by high SNR, the above equation simplifies to:

$$P_2(S_l, \hat{S}_l) \leq \prod_{i \in \eta} \frac{(1+K)e^{-K}}{\left(\frac{1}{4N_0}\right)^2 |(S_i - \hat{S}_i)|^2} \quad (5)$$

The symbol η signifies instances where S_i and $\hat{S}_i$ are unequal. Consequently, eq. (5) can be stated as below. Effective length of the error event $(S_l, \hat{S}_l)$ is denoted as l_η .

$$P_2(S_l, \hat{S}_l) \leq \frac{((1+K)e^{-K})^{l_\eta}}{\left(\frac{1}{4N_0}\right)^{l_\eta} d_P^2(l_\eta)} \quad (6)$$

$d_P^2(l_\eta)$ denotes square of product distance between signals S_i and $\hat{S}_i$ when these are unequal along error event pathway. This distance is:

$$d_P^2(l_\eta) = \prod_{i \in \eta} |(S_i - \hat{S}_i)|^2 \quad (7)$$

Error event occurs when the trajectory of received data veers away from the intended course as established by design. This concept is visually depicted in **Figure 4**, where the intended trajectory spans from s_1 to s_2 and extends to s_l , whereas the trajectory followed by received symbols manifests as $\hat{s}_1$, $\hat{s}_2$, and onward up to $\hat{s}_l$.

By encompassing all sequences those have been transmitted and combining all error events' probabilities across varying lengths l , extending from 1 to infinity, upper limit and upper bound can be formulated as:

$$P_e \leq \sum_{l=1}^{\infty} \sum_{S_l} \sum_{\hat{S}_l \neq S_l} P(S_l) P_2(S_l, \hat{S}_l) \quad (8)$$

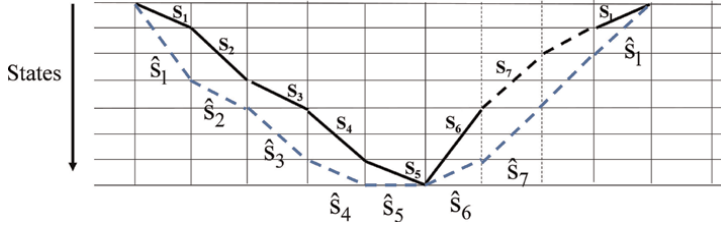


Figure 4.
Error event with a length of l .

In above equation, $P(S_l)$ denotes Apriori Probability of transmitting symbol. At high SNRs, $P_2(S_l, \hat{S}_l)$ can be substituted into eq. (8). As a result, upper bound pertinent to Rician fading channel can be articulated in following manner:

$$P_e \leq \sum_{l_\eta} \sum_{d_P^2(l_\eta)} \alpha(l_\eta, d_P^2(l_\eta)) \frac{((1+K)e^{-K})^{l_\eta}}{\left(\frac{1}{4N_0}\right)^{l_\eta} d_P^2(l_\eta)} \quad (9)$$

Eq. (9) signifies that average number of code sequences having an effective length l_η and a squared product distance of $d_P^2(l_\eta)$ is represented by $\alpha(l_\eta, d_P^2(l_\eta))$. When operating at high SNRs, error event is affected by l_η and $d_P^2(l_\eta)$. For sake of simplification, L has been adopted to symbolize minimum effective length, and $d_P^2(L)$ to denote associated squared product distance. Minimal effective length is regarded as effective length of code, so as to enable the approximate probability of error event in the following manner:

$$P_e \approx \alpha(L, d_P^2(L)) \frac{((1+K)e^{-K})^L}{\left(\frac{1}{4N_0}\right)^L d_P^2(L)} \quad (10)$$

Regarding the Rayleigh fading channel, in which K equals 0, eq. (10) can be further streamlined as:

$$P_e \approx \frac{\alpha(L, d_P^2(L))}{\left(\frac{1}{4N_0}\right)^L d_P^2(L)} \quad (11)$$

In context of an AWGN channel, by taking $K = \infty$, we may reformulate P_e as shown below [11]:

$$P_e \approx \frac{1}{2} N(d_{free}) \operatorname{erfc} \left(\sqrt{\frac{d_{free}^2}{4N_0}} \right) \quad (12)$$

Here, d_{free} denotes value of code's unrestricted Euclidean distance.

The investigation has focused on eq. (10) in relation to K (Rician parameter), with an acknowledgment of special cases in which K is either 0 or ∞ . To measure the designed TCM scheme's effectiveness, the achieved results can be compared with those of the uncoded modulation.

6. Results and conclusion

The analyses conducted reveal a substantial enhancement resulting from the implemented technique. In direct comparison to an uncoded modulation scheme, the developed approach has yielded a notable boost in SNR, surpassing many decibels (dB). This augmentation in SNR signifies the system's heightened resilience against noise and interference encountered during transmission processes. Furthermore, the integration of the TCM coding scheme has facilitated the attainment of elevated data transfer rates while maintaining robust reliability. Through these findings, it becomes evident that the designed technique not only reinforces the system's capacity to combat external disturbances but also enhances its efficiency in transmitting data with increased speed and dependability.

The significance of this enhancement becomes even more pronounced when considering communication channels prone to fading phenomena, wherein signal strength fluctuates due to multipath propagation or environmental factors. The TCM coding scheme imparts greater resilience to these fading effects within the communication system. Consequently, performance is noticeably improved, allowing for the transmission of data at higher rates even amidst challenging channel conditions. By mitigating the impact of fading, the designed technique ensures more consistent and reliable data transmission, thereby enhancing the system's effectiveness in real-world scenarios where environmental variables can significantly affect communication quality. **Figures 5–8** show the gain that have been achieved by utilizing the TCM scheme discussed in this chapter [12, 14–16].

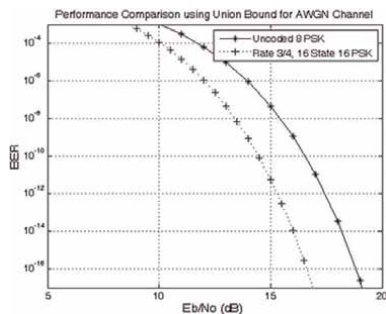


Figure 5.
Rate 3/4, 16-States, 16-PSK.

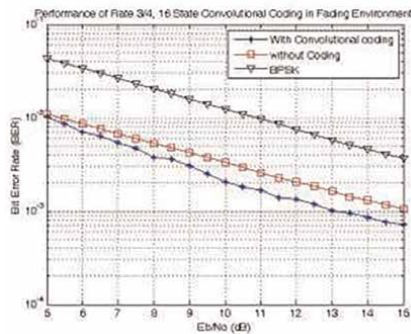


Figure 6.
Rate 3/4, 16 State 16-QAM.

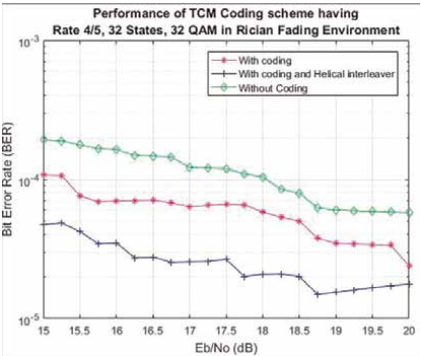


Figure 7.
Rate 4/5, 32-States, 32-QAM.

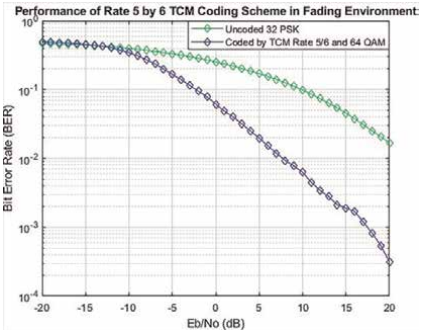


Figure 8.
Rate 5/6, 64 States 64-QAM.

Acknowledgements

I hereby confirm the use of ChatGPT only for paraphrasing the text and that too only for improving the English language from grammatical point of view and not for generating any technical content.

Author details

Raj Kumar Goswami
Gayatri Vidya Parishad College of Engineering for Women (Autonomous),
Visakhapatnam, India

*Address all correspondence to: rajkumargoswami@gmail.com

IntechOpen

© 2024 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Biglieri E, Divsalar D, Mclane PJ, Simon MK. Introduction to Trellis-Coded Modulation with Applications. New York: Maxwell Macmillan International Editions; 1991
- [2] Jafari H et al. Generalized closed-form expression for the bit error probability of MPSK with phase noise. *Wireless Personal Communications*. 2023;**130**: 2441-2450
- [3] Hosseinpour A, Montasari R. Modeling and Analyzing trellis-coded modulation on power line communication systems. *International Journal of Strategic Engineering*. 2022;**5**:1-10. DOI: 10.4018/IJoSE.292443
- [4] Baharuddin AH, Muharam M, Luthfi A, Angraini R. Performance analysis of trellis coded modulation and diversity combining on Wireless Channel. *IOP Conference Series: Materials Science and Engineering*. 2021; **1041**(1):012023. DOI: 10.1088/1757-899X/1041/1/012023
- [5] Arman H, Hadi-Vencheh A, Mavi RK, Khodadadipour M, Jamshidi A. Revisiting the interval and fuzzy TOPSIS methods: Is Euclidean distance a suitable tool to measure the differences between fuzzy numbers? *Complexity*. 2022;**2022**, Article ID 7032662:11. DOI: 10.1155/2022/7032662
- [6] Divsalar D, Simon MK. The Design of Trellis Coded MQAM for fading channels: Performance criteria. *IEEE Transactions on Communications*. 1988; **36**(9):1004-1012
- [7] Jamali HS, Le Ngoc T. Coded Modulation Techniques for Fading Channels. Boston: Kluwer Academic Publishers; 1994
- [8] Hanzo L, Ng SX, Keller T, Webb W. Quadrature Amplitude Modulation: From Basics to Adaptive Trellis-Coded, Turbo-Equalised and Space-Time Coded OFDM, CDMA and MC-CDMA Systems. 2nd ed. Chichester, United Kingdom: Wiley-IEEE Press; 2004. pp. 746-748
- [9] Li R, Tian J, Bian X, Li M. CRCSI: A generic block Interleaver for the next generation terrestrial broadcast systems. *Applied Sciences*. 2022;**12**:2025. DOI: 10.3390/app12042025
- [10] Blackburn S. Maximum likelihood decoding for multilevel channels with gain and offset mismatch. *IEEE Transactions on Information Theory*. 2015;**62**:1144-1149. DOI: 10.1109/TIT.2016.2521658
- [11] Schlegel CB, Pérez LC. Trellis and Turbo Coding. Hoboken, NJ: John Wiley & Sons Inc., Publication; 2004
- [12] Goswami RK. Design of 32 states, rate 4/5, 32 QAM, TCM code for Fading Channel. *IOSR Journal of Electronics and Communication Engineering*. 2021; **16**(5, Ser. I):01-08
- [13] Alamouti SM, Kallel S. Adaptive trellis-coded multiple-phase-shift keying for Rayleigh fading channels. *IEEE Transactions on Communications*. 1994; **42**:2305-2314
- [14] Goswami R, Rao GR, Rao GSB. Design of rate 3/4, 16-states, 16-PSK, trellis-coded modulation code for AWGN channel. *Intelligent Engineering Informatics, Advances in Intelligent Systems and Computing*. 2018;**695**: 571-580. DOI: 10.1007/978-981-10-7566-7_57
- [15] Goswami RK, Rao GS. Rate 3/4 TCM code having 16 state with 16 QAM for

fading channel. Journal of Emerging Technologies and Innovative Research. Apr 2019;6(4):316-319

[16] Goswami RK, Nambari S, Kandula SR, Roya PP, Sharief AH. Rate 5/6 TCM code having 64 states with 64 QAM for fading channel. International Journal of Electrical and Electronics Research. 2024;12(2):375-382

Perspective Chapter: Weight/ Distance Distributions and Decoding Error Probabilities of Random Rank Metric Codes

İbrahim Özen

Abstract

Rank metric codes have applications in network coding, post-quantum cryptography, and distributed storage systems. We review the basics of rank metric codes and then summarize the cases where the distance/weight distributions are determined by the basic parameters. The distance distributions of maximum rank distance (MRD) and dually quasi-MRD codes are obtained by their basic parameters. Weight distributions of codes enable us to obtain information on their decoding error probabilities. In this chapter, we obtain the expressions of expectations of the distance and weight distributions of random rank metric codes. Using these expectations we obtain estimations for the expected decoding error probabilities of random codes, both in general and linear cases.

Keywords: random rank metric codes, rank distance distribution, rank weight distribution, decoding error probabilities, network coding

1. Introduction

1.1 Notations and basic definitions

A rank metric code is a subset of the vector space $\mathbb{F}_q^{m \times n}$, where $\mathbb{F}_q$ is a finite field with q elements. We will assume in this chapter $m \geq n$, but the same arguments will apply for the other case. Any matrix in the code will be called a codeword.

If a rank metric code $\mathcal{C} \subset \mathbb{F}_q^{m \times n}$ has M elements, we will refer to this code as an (n, m, M) code. We will also consider linear codes. If $\mathcal{C} \subset \mathbb{F}_q^{m \times n}$ is a linear subspace of dimension k , then $\mathcal{C}$ will be called an $[n, m, k]$ linear code. The rank distance d_R between two matrices $C, D \in \mathbb{F}_q^{m \times n}$ is defined by the rank of their difference

$$d_R(C, D) = \text{rank}(C - D). \quad (1)$$

The minimum d of the pairwise distances of distinct words in $\mathcal{C}$ will be called the minimum distance of the code. If $\mathcal{C}$ is linear, the minimum distance equals to the minimum of the ranks (weights from now on) of nonzero codewords. If the minimum

distance is d , then we will call $\mathcal{C}$ an (n, m, M, d) or $[n, m, k, d]$ code depending on whether the code is nonlinear or linear, respectively.

We define the rate of an (n, m, M, d) code $\mathcal{C}$ with $R(\mathcal{C}) = \frac{\log M}{mn}$. All logarithms will be evaluated to the base q . The relative minimum distance is given by $\delta(\mathcal{C}) = \frac{d}{n}$. If the code is linear and has parameters $[n, m, k, d]$, its rate is given by $R(\mathcal{C}) = \frac{k}{mn}$ and its relative minimum distance equals $\delta(\mathcal{C}) = \frac{d}{n}$.

If $\mathcal{C}$ is an $[n, m, k, d]$ linear code in $\mathbb{F}_q^{m \times n}$, we define the dual of $\mathcal{C}$ with

$$\mathcal{C}^\perp = \left\{ D \in \mathbb{F}_q^{m \times n} : \langle C, D \rangle = \text{Trace}(CD^T) = 0 \right\}, \quad (2)$$

where $\text{Trace}(A) = \sum_{i=1}^m A_{ii}$ for any $m \times m$ matrix.

We refer the reader to the surveys [1–4] each of which concentrate in different aspects of the rank metric codes.

1.2 A motivational example: Network coding

Let us consider conveying information on the so called butterfly network in **Figure 1**. We have two packets A and B entering in the network and we want both packets to be received by the nodes s_1 and s_2 . The left part of the figure shows how network throughput suffers when we allow only forwarding in the nodes. When the packets A and B arrive at the intermediate node i_1 , it can forward only one of the packets. If it transmits the package A , the sink node s_2 receives both A and B but the node s_1 receives only A twice. We have a similar situation if i_1 transmitted B .

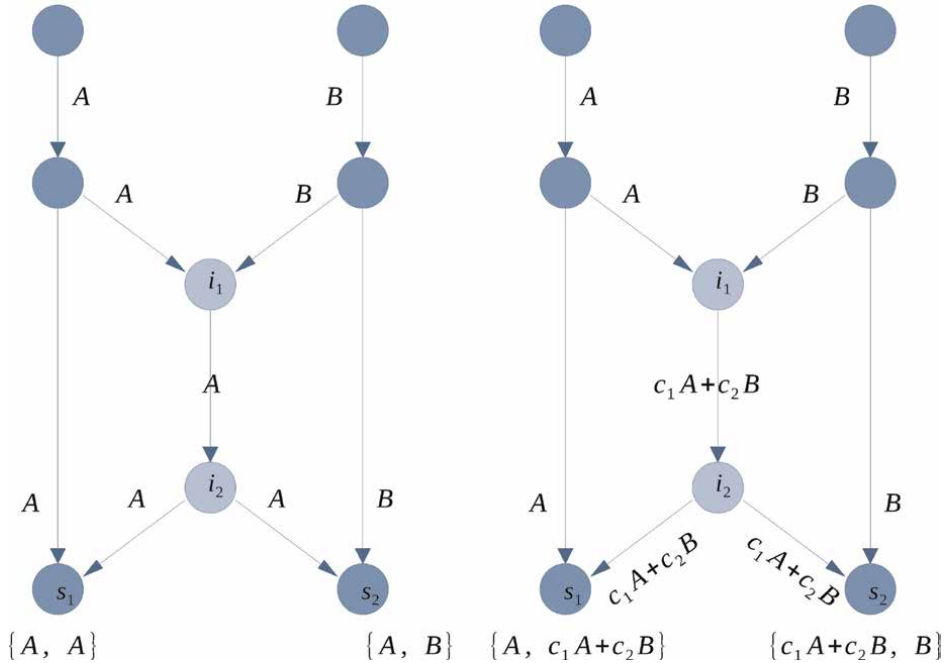


Figure 1.
Butterfly network.

If we let the nodes perform encoding by sending linear combinations of the received packets instead of just forwarding, the nodes s_1 and s_2 receive $\{A, c_1A + c_2B\}$ and $\{c_1A + c_2B, B\}$, respectively. If the coefficients c_1 and c_2 are chosen carefully, both nodes can obtain $\{A, B\}$.

Kötter and Kischang defined operator channels suitable for the above consideration [5]. An operator channel for a given positive integer N is a channel with input and output alphabet consisting of all subspaces of $\mathbb{F}_q^N$, denoted by $\mathcal{P}(N)$. In this channel, the channel input V and the output U are related as

$$U = (U \cap V) \oplus E, \quad (3)$$

where $(U \cap V) \cap E = \{\mathbf{0}\}$. If we assume $\dim(V) = m$, $\dim(U \cap V) = m - l$ and $\dim(E) = \gamma$ then we say that the channel committed l deletion and γ insertion errors. For subspaces $U, V \in \mathcal{P}(N)$, we say that V is (l, γ) reachable from U if Eq. (3) holds for l and γ .

We define the subspace distance between two subspaces $U, V \in \mathcal{P}(N)$ as

$$d_S(U, V) = \dim(U + V) - \dim(U \cap V). \quad (4)$$

If V is (l, γ) reachable from U with above notation, then

$$d_S(U, V) = (m - l + \gamma + m) - 2(m - l) = l + \gamma. \quad (5)$$

Further, we define constant dimension subspace codes. For a positive integer n , the set of all n dimensional subspaces in $\mathcal{P}(N)$ is denoted by $G(N, n)$. A constant dimension subspace code is a nonempty subset of $G(N, n)$.

Now, let A be an $n \times m$ matrix over $\mathbb{F}_q$, with $n \leq m$. The lifting of A to $G(n + m, n)$ is the row space of the concatenated matrix $[I_n | A]$, where I_n is the identity matrix. If we start with an (n, m, M, d_C) rank metric code $\mathcal{C} \subset \mathbb{F}_q^{m \times n}$ and take liftings of A^T , for all $A \in \mathcal{C}$ to $G(n + m, n)$, we have a constant dimension subspace code $\mathcal{C}_S \subset G(n + m, n)$ with $|\mathcal{C}_S| = M$ and minimum distance $d_{C_S} = 2d_C$.

This briefly summarizes the motivation to study rank metric codes as a tool in network coding.

2. Rank distribution of rank metric codes

2.1 Preliminaries

In this subsection, we will review basic formulas for rank metric codes. Let m, n be positive integers with $m \geq n$ and q be a prime power. We denote the number of matrices with rank w in $\mathbb{F}_q^{m \times n}$ with $N(w)$. This number is given by

$$N(w) = \begin{bmatrix} n \\ w \end{bmatrix} \alpha(m, w), \quad (6)$$

where

$$\alpha(m, w) = \begin{cases} \prod_{i=0}^{w-1} (q^m - q^i), & \text{for } 1 \leq w \leq m, \\ 1, & \text{for } w = 0, \quad m \geq 0, \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

and

$$\begin{bmatrix} n \\ w \end{bmatrix} = \frac{\alpha(n, w)}{\alpha(w, w)}. \quad (8)$$

The number $\begin{bmatrix} n \\ w \end{bmatrix}$ is called a Gaussian binomial and it gives the number of w dimensional subspaces of an n dimensional vector space over $\mathbb{F}_q$. This term is bounded as follows

$$q^{w(n-w)} \leq \begin{bmatrix} n \\ w \end{bmatrix} < K_q^{-1} q^{w(n-w)}, \quad \text{for all } 0 \leq w \leq n, \quad (9)$$

where $K_q = \prod_{i=0}^{\infty} (1 - q^{-i})$ [6].

We also define the volume of a ball of radius t around any vector:

$$B(t) = \sum_{w=0}^t N(w). \quad (10)$$

We have the following bounds for $N(w)$ and $B(t)$

$$q^{(m+n-1)w-w^2} \leq N(w) \leq q^{(m+n)w-w^2+\sigma(q)}, \quad (11)$$

$$q^{(m+n)t-t^2} \leq B(t) \leq q^{(m+n)t-t^2+\sigma(q)}, \quad (12)$$

where $\sigma(q) = \frac{-1}{\ln(q)} \sum_{j=1}^{\infty} \ln(1 - q^{-j})$ [6, 7].

Let two matrices M and $N \in \mathbb{F}_q^{m \times n}$, which are at rank distance w be given. The number of matrices on the intersection of two spheres, one centered at M with radius u and one centered at N having radius s , is called the intersection number $J(u, s, w)$. The formula of the intersection number is given in terms of q -Krawtchouk polynomials and can be found in the work of Gadouleau and Yan [8].

These numbers satisfy the following properties [9]:

$$N(d)J(u, s, d) = N(u)J(d, s, u), \quad (13)$$

$$\sum_{u=0}^n J(u, s, d) = N(s). \quad (14)$$

For asymptotic values of functions we will write $f(n) \doteq g(n)$ when

$$\lim_{n \rightarrow \infty} \frac{\log(f(n)/g(n))}{n^2} = 0. \quad (15)$$

2.2 MRD and quasi-MRD codes

Del Sarte showed that any (n, m, M, d) rank metric code satisfies the inequality $M \leq q^{m(n-d+1)}$, (for $m \geq n$) [10].

We call codes achieving this bound as *maximum rank distance* (MRD) codes.

The distance average enumerator polynomial of an (n, m, M) rank metric code $\mathcal{C}$ is defined by

$$E_{\mathcal{C}}(z) = \sum_{j=0}^n a_j z^j = \frac{1}{M} \sum_{C, D \in \mathcal{C}} z^{\text{rank}(C-D)}. \quad (16)$$

The coefficients $\{a_j : 0 \leq j \leq n\}$ give the distance averages of $\mathcal{C}$. Namely, they give the averages over the elements $C \in \mathcal{C}$, the number of codewords at distance j from C .

For linear codes, we define the weight enumerator polynomial by

$$W_{\mathcal{C}}(z) = \sum_{j=0}^n A_j z^j = \sum_{C \in \mathcal{C}} z^{\text{rank}(C)}. \quad (17)$$

The set $\{A_j : 0 \leq j \leq n\}$ is the weight distribution of the code and $A_j = |\{C \in \mathcal{C} : \text{rank}(C) = j\}|$.

The distance average enumerator $E_{\mathcal{C}}(z) = \sum_i a_i z^i$ of an MRD rank metric code is obtained from its parameters m, n , and d . The coefficients are given by [10]

$$a_i = \begin{bmatrix} n \\ i \end{bmatrix} \sum_{j=0}^{i-d} (-1)^j q^{\binom{j}{2}} \begin{bmatrix} i \\ j \end{bmatrix} (q^{m(i+1-j-d)} - 1), \text{ for all } d \leq i \leq n. \quad (18)$$

Delsarte also showed that for every given $n, m, d \leq n$, there exists an $[n, m, k = m(n-d+1), d]$ linear MRD code, a result which is contrary to the counter part in the Hamming metric.

Another result for MRD linear codes is that the dual of an MRD code is MRD [10]. So if a code $\mathcal{C}$ is a linear MRD code, the weight distribution of the code and its dual are determined by their fundamental parameters.

To generalize this result, de la Cruz et al. defined the notion of quasi-MRD codes [11]. Namely, if $\mathcal{C}$ is an $[n, m, k, d]$ linear code, it is called quasi-MRD if $m \nmid k$ and the rank deficiency defined below is 0

$$\text{Rdef}(\mathcal{C}) = n - \left\lfloor \frac{k}{m} \right\rfloor - d + 1. \quad (19)$$

The authors show that for any set of parameters $1 \leq d \leq n \leq m$ and $1 \leq k < nm$ such that $m \nmid k$, there exists an $[n, m, k, d]$ quasi-MRD code. Unlike MRD codes, there exist quasi-MRD codes whose dual is not quasi-MRD. If the dual of a quasi-MRD code $\mathcal{C}$ is also quasi-MRD, then $\mathcal{C}$ is called dually quasi-MRD (DQMRD). The authors prove that, the weight distribution of an MRD or a DQMRD code $\mathcal{C}$ with parameters $[n, m, k, d]$ can be obtained from their parameters as follows

$$A_i = \begin{bmatrix} n \\ i \end{bmatrix} \sum_{j=0}^{i-d} (-1)^j q^{\binom{j}{2}} \begin{bmatrix} i \\ j \end{bmatrix} (q^{k-m(n+j-i)} - 1), \text{ for all } d \leq i \leq n. \quad (20)$$

3. Random rank-metric codes

3.1 General case

An (n, m, M) random rank metric code is a multiset $\mathcal{C} = \{C_1, C_2, \dots, C_M\} \subset \mathbb{F}_q^{m \times n}$. For each $i \in \{1, 2, \dots, M\}$, C_i s are chosen randomly with their entries distributed independently and identically over $\mathbb{F}_q$.

We will make use of the following characteristic functions to obtain the expectations of the number of codeword pairs at distance w

$$\zeta_{i,j,w} = \begin{cases} 1, & \text{if } d(C_i, C_j) = w, \\ 0, & \text{otherwise.} \end{cases} \quad (21)$$

Clearly, expectations of these random variables are the probabilities of the events they describe

$$\mathbb{E} \zeta_{i,j,w} = \Pr(d(C_i, C_j) = w). \quad (22)$$

Similar characteristic functions are used in the work of Loidreau [7]. The aim there is to estimate the probabilities of having a word within a ball to obtain the Gilbert-Varshamov type bounds for rank metric codes.

For our purposes, the polynomial below is called the distance distribution polynomial of $\mathcal{C}$

$$D_{\mathcal{C}}(z) = \sum_{j=0}^n b_j z^j = \sum_{i=1}^M \sum_{j=1}^{i-1} z^{\text{rank}(C_i - C_j)}. \quad (23)$$

For each j , b_j gives the number of unordered distinct codeword pairs at rank distance j . The relation between the $E_{\mathcal{C}}(z)$ given in Eq. (16) and $D_{\mathcal{C}}(z)$ is as follows

$$E_{\mathcal{C}}(z) = 1 + \frac{2}{M} D_{\mathcal{C}}(z). \quad (24)$$

Proposition 1 Let $\mathcal{C} \subset \mathbb{F}_q^{m \times n}$ be a random code. For any given i , the probability that $\text{rank}(C_i - C_j) = w$ is given by

$$\Pr(\text{rank}(C_i - C_j) = w) = \mathbb{E}(\zeta_{i,j,w}) = \frac{N(w)}{q^{mn}}. \quad (25)$$

Proof: Let $C \in \mathbb{F}_q^{m \times n}$ be given. Clearly all $D \in \mathbb{F}_q^{m \times n}$ such that $\text{rank}(C - D) = w$, for any $0 \leq w \leq n$, is in one-to-one correspondence with rank w matrices. The correspondence can be given by

$$\Phi_M : \mathcal{A}_w(C) = \{D : \text{rank}(C - D) = w\} \rightarrow B = \left\{ U \in \mathbb{F}_q^{m \times n} : \text{rank}(U) = w \right\} \quad (26)$$

$$D \rightarrow C - D \quad (27)$$

So for any given C_i in a random code $\mathcal{C}$, the number of vectors C_j with $\text{rank}(C_i - C_j) = w$ can be chosen among $N(w)$ vectors in $\mathbb{F}_q^{m \times n}$.

Now we can calculate the expected distance distribution of a random code.

Theorem 1 The expectation of the distance distribution of a random (n, m, M) code $\mathcal{C}$ is given by

$$\mathbb{E}D_{\mathcal{C}}(z) = \sum_{i=0}^n \mathbb{E}(b_i) z^i = \frac{M(M-1)}{2q^{mn}} \sum_{w=0}^n N(w) z^w. \quad (28)$$

In particular, for any $C \in \mathcal{C}$, the expectation of the number of codewords at rank distance w to C is given by

$$\mathbb{E}A_w(C) = \frac{(M-1)}{q^{mn}} N(w). \quad (29)$$

Proof: Expectation of the distance distribution in Eq. (23) of a random code is given by

$$\mathbb{E}D_{\mathcal{C}}(z) = \sum_{i=1}^M \sum_{j=1}^{i-1} \mathbb{E} z^{\text{rank}(C_i - C_j)} = \sum_{w=0}^n \sum_{i=1}^M \sum_{j=1}^{i-1} \mathbb{E}(\zeta_{i,j,w}) z^w \quad (30)$$

$$= \sum_{w=0}^n \sum_{i=1}^M (i-1) \frac{N(w)}{q^{mn}} z^w = \sum_{w=0}^n \left(\frac{M(M-1)}{2} \right) \frac{N(w)}{q^{mn}} z^w \quad (31)$$

$$= \frac{M(M-1)}{2q^{mn}} \sum_{w=0}^n N(w) z^w. \quad (32)$$

To get $\mathbb{E}A_w(C_i)$ for any i , we sum up the expectations of $\zeta_{i,j,w}$ s as follows

$$\mathbb{E}A_w(C_i) = \sum_{j>i} \mathbb{E}(\zeta_{j,i,w}) + \sum_{j<i} \mathbb{E}(\zeta_{i,j,w}) = \frac{(M-1)}{q^{mn}} N(w). \quad (33)$$

3.2 Linear codes

A random linear $[n, m, k]$ code $\mathcal{C}$ is the linear span of an ordered multiset $\{F_0, F_1, \dots, F_{k-1}\} \subset \mathbb{F}_q^{m \times n}$, where each F_l is chosen randomly with their entries identically and independently distributed over $\mathbb{F}_q$. Let $\{\alpha_0 = \mathbf{0}, \alpha_1, \dots, \alpha_{q^k-1}\}$ be any labeling of the vector space $\mathbb{F}_q^k$. Then, we can name the elements of the code as $\mathcal{C} = \{C_0 = \mathbf{0}, C_1, C_2, \dots, C_{q^k-1}\}$, where $C_i = \sum_{j=0}^{k-1} \alpha_{ij} F_j$ and $\alpha_i = (\alpha_{i,0}, \alpha_{i,1}, \dots, \alpha_{i,k-1})$.

Let ξ_{iw} be the characteristic function of the event that $\text{rank}(C_i) = w$ for any $0 \leq i \leq q^k - 1$ and $0 \leq w \leq n$,

$$\xi_{iw} = \begin{cases} 1, & \text{if } \text{rank}(C_i) = w, \\ 0, & \text{otherwise.} \end{cases} \quad (34)$$

Clearly, we have

$$\mathbb{E}(\xi_{iw}) = P(\text{rank}(C_i) = w) = \begin{cases} 1, & \text{if } i = w = 0, \\ 0, & \text{if } i = 0, w > 0, \\ \frac{N(w)}{q^{mn}}, & \text{if } i > 0 \text{ and } w \geq 0. \end{cases} \quad (35)$$

We can calculate the expectation of the weight distribution of a random linear code, using the characteristic functions, as follows.

Theorem 2 The expectation of the weight distribution of a random linear $[n, m, k]$ code $\mathcal{C}$ is given by

$$\mathbb{E} W_{\mathcal{C}}(z) = \sum_{j=0}^n \mathbb{E}(A_j) z^j = 1 + \frac{(q^k - 1)}{q^{mn}} \sum_{w=0}^n N(w) z^w. \quad (36)$$

Moreover, for any $C \in \mathcal{C}$, the expected number of words at distance w from C is given by

$$\mathbb{E} A_w(C) = \frac{(q^k - 1)}{q^{mn}} N(w). \quad (37)$$

Proof: We have $W_{\mathcal{C}}(z) = \sum_{w=0}^n \sum_{i=0}^{q^k-1} \xi_{iw} z^w$.

$$\mathbb{E} W_{\mathcal{C}}(z) = \sum_{w=0}^n \sum_{i=0}^{q^k-1} \mathbb{E}(\xi_{iw}) z^w \quad (38)$$

$$= 1 + \sum_{w=0}^n \sum_{i=1}^{q^k-1} \frac{N(w)}{q^{mn}} z^w \quad (39)$$

$$= 1 + \frac{(q^k - 1)}{q^{mn}} \sum_{w=0}^n N(w) z^w. \quad (40)$$

The number of codewords at distance w from any $C \in \mathcal{C}$ is equal to the number of codewords of rank w by linearity. So we have

$$\mathbb{E} A_w(C) = \mathbb{E} A_w(\mathbf{0}) = \frac{(q^k - 1)}{q^{mn}} N(w), \text{ for any } w. \quad (41)$$

The Gilbert-Varshamov type bounds for rank metric codes are given in the work of Loidreau [7]. Below, we will obtain estimations on the minimum distances of random rank metric codes, using the expectations of the variables b_j and A_j s, obtained in Theorems 1 and 2, respectively. We will adopt the methods and the notations of Barg et al. [12], for the estimations of the minimum distances.

3.3 Gilbert-Varshamov type bounds for the random rank metric codes

Let us define

$$\delta_{GV}(R) = \frac{\alpha + 1}{2} - \sqrt{\left(\frac{\alpha - 1}{2}\right)^2 + \alpha R}. \quad (42)$$

Theorem 3 Let $\alpha = m/n$ be fixed.

1. For $0 \leq R < 1$ and any $\varepsilon > 0$, the probability that the relative minimum distance δ of a random (n, m, M) code $\mathcal{C}$ is less than $\delta_{GV}(2R) - \varepsilon$ goes to zero exponentially as $n \rightarrow \infty$.
2. For $0 \leq R < 1$ and any $\varepsilon > 0$, the probability that the relative minimum distance δ of a random linear $[n, m, k]$ code $\mathcal{C}$ is less than $\delta_{GV}(R) - \varepsilon$ goes to zero exponentially as $n \rightarrow \infty$.

Proof: For the first part, we have by Theorem 1, the expectation of the number of unordered pairs of codewords at rank distance d is given by

$$\mathbb{E}(b_d) = \frac{M(M-1)}{2q^{mn}} N(d). \quad (43)$$

We can estimate the expectation $\mathbb{E}(b_d)$ using the bound in Eq. (11).

$$\mathbb{E}(b_d) = \frac{M(M-1)}{2q^{mn}} N(d) \quad (44)$$

$$\doteq q^{(m+n)d - d^2 - mn + 2\log(M)}. \quad (45)$$

The exponent can be put in the following form

$$(m+n)d - d^2 - mn + 2\log(M) \quad (46)$$

$$= -n^2 \left[\delta^2 - (\alpha + 1)\delta + \alpha - 2 \frac{\log(M)}{n^2} \right] \quad (47)$$

$$\doteq -n^2 [\delta^2 - (\alpha + 1)\delta + \alpha - 2\alpha R]. \quad (48)$$

The roots of the quadratic equation above are

$$\delta_1 = \frac{\alpha + 1}{2} - \sqrt{\left(\frac{\alpha - 1}{2}\right)^2 + 2\alpha R} \quad \text{and} \quad (49)$$

$$\delta_2 = \frac{\alpha + 1}{2} + \sqrt{\left(\frac{\alpha - 1}{2}\right)^2 + 2\alpha R}. \quad (50)$$

We have $\delta_1 > 0$ and the exponent of the estimation is negative outside the interval $[\delta_1, \delta_2]$. Since we have

$$Pr(b_d \geq 1) \leq \mathbb{E}(b_d), \quad (51)$$

and $\mathbb{E}(b_d)$ is exponentially small for $\delta = d/n < \delta_{GV}(2R) - \varepsilon$, the probability that $\mathcal{C}$ has codeword pairs at distance d tends to zero as $n \rightarrow \infty$.

Similar estimations of $\mathbb{E}(A_j)$ s given in Theorem 2 gives the bound in item 2. We will need the following Theorems on the minimum distance of random codes. Theorem 4 ([7], Propositions 6,8)

1. For any random (n, m, M) code with $0 \leq R < 1/2$ and for any sufficiently small constant ϵ , we have

$$\Pr(d/n \geq \delta_{GV}(2R) + \epsilon) \xrightarrow{n \rightarrow \infty} 0, \quad (52)$$

exponentially fast.

2. For any random $[n, m, k]$ linear code with $0 \leq R < 1$ and for any sufficiently small constant ϵ , we have

$$\Pr(d/n \geq \delta_{GV}(R) + \epsilon) \xrightarrow{n \rightarrow \infty} 0, \quad (53)$$

exponentially fast.

Another Gilbert-Varshamov type bound is given in the work of Gadouleau and Yan [13] as follows:

$$R(\mathcal{C}) \geq (1 - \delta)(1 - \alpha). \quad (54)$$

4. Decoding error probabilities of random codes in bounded distance decoders

In this section, we will obtain estimations for the expected decoding error probabilities of random codes. We will use Gadouleau and Yan's formula for the error probabilities. The discussion on the decoding error probability is summarized from their work [14].

Definition 1 A rank symmetric channel over $\mathbb{F}_q^{m \times n}$ is a communication channel whose inputs and outputs are matrices in $\mathbb{F}_q^{m \times n}$ and all errors with the same rank are equiprobable.

Let $\mathcal{C} \subset \mathbb{F}_q^{m \times n}$ be a code with minimum distance d . Given a matrix $M \in \mathbb{F}_q^{m \times n}$, a bounded distance decoder with decision radius $t = \lfloor \frac{d-1}{2} \rfloor$ for $\mathcal{C}$, returns the codeword D when $\text{rank}(D - M) \leq t$ if such a codeword exists, and declares a failure otherwise. A decoding error occurs if the returned codeword is not the transmitted vector.

Decoder error probability (DEP) is the probability that the decoder produces a codeword that is not the transmitted word. Let $C \in \mathcal{C}$ be transmitted and M be received. If $u = \text{rank}(C - M)$ and $u \leq t$, the decoder correctly decodes M to C . If $t < u < d - t$, the received word M does not fall into a decision ball around any codeword and a decoder failure occurs. A decoder error happens only for $u \geq d - t$.

We can obtain the decoder error probability for $\mathcal{C}$, by summation over error occurrences, conditioned on sent codewords and the error ranks. Let us denote by $P(\mathcal{C}, u, t)$, the decoder error probability, conditioned on C is sent and a vector at rank distance u from C is received, on a rank symmetric channel. Then, the probability $P(\mathcal{C}, t)$ that there will be a decoding error using $\mathcal{C}$ on a bounded distance decoder is given by

$$P(\mathcal{C}, t) = \sum_{C \in \mathcal{C}} \sum_{u=d-t}^n P(\mathcal{C}, u, t) \Pr(C, u), \quad (55)$$

where $\Pr(C, u)$ is the probability that $C \in \mathcal{C}$ is transmitted and the additive error on the rank symmetric channel has rank u [15].

The proposition below shows how distance distribution $A_w(C) = \{D \in \mathcal{C} : \text{rank}(C - D) = w\}$ is involved in DEP. We refer to the work of Gadouleau and Yan [14] for its proof.

Proposition 2 [14] Let $\mathcal{C} \subset \mathbb{F}_q^{m \times n}$ be a code with minimum distance d and let $C \in \mathcal{C}$. The decoder error probability $P(C, u, t)$, conditioned on C is sent and a vector at rank distance u from C is received on a rank symmetric channel, is given by

$$P(C, u, t) = \frac{1}{N(u)} \sum_{w=d}^n A_w(C) \sum_{s=0}^t J(u, s, w), \quad \text{for } u \geq d - t. \quad (56)$$

Recall that the error probability formula in Eq. (55) of a code is conditioned on the rank of the error vector. By the symmetry requirement of the rank symmetric channel, every vector with the same rank has the same probability of being the error. For any vector with rank u , let us fix the probability of that vector being the error, as s_u . For each $0 \leq u \leq n$, the probability that an error of rank u occurs is $p_u = N(u)s_u$ and we have

$$\sum_{u=0}^n N(u)s_u = 1. \quad (57)$$

Below, we will obtain estimations for the expectation of the decoding error probability of a random code. We will consider the random code as a code with distance/weight distribution as in Eqs. (28) and (36). The formulas of Eqs. (29) and (37) will be particularly useful.

Theorem 5 Let $\mathcal{C}$ be a random (n, m, M) code with rate $0 \leq R < 1/2$. Let all codewords be equally probable and let all errors of rank u occur with probability s_u . The expectation of the decoding error probability $\mathbb{E}P(\mathcal{C})$ of $\mathcal{C}$ satisfies

$$\frac{(M-1)B(t_0)}{q^{mn}} \left(1 - p_0 - \dots - p_{d_0+t_0-1}\right) \leq \mathbb{E}P(\mathcal{C}) \leq \frac{(M-1)B(t_0)}{q^{mn}} \left(1 - p_0 - \dots - p_{d_0-t_0-1}\right), \quad (58)$$

where $t_0 = \lfloor \frac{d_0-1}{2} \rfloor$, $d_0 = \lceil n\delta_{GV}(2R) \rceil$, and $p_u = N(u)s_u$.

Proof: Let $\mathcal{C}$ be a random (n, m, M) code. We know by Eq. (29) that for any codeword $C \in \mathcal{C}$ and w , expected number of codewords at distance w to C is

$$\mathbb{E}A_w(C) = \frac{(M-1)}{q^{mn}} N(w). \quad (59)$$

Moreover, by Theorems 3 and 4, with high probability, the relative minimum distance of $\mathcal{C}$ is given by

$$\delta_{GV}(2R) = \frac{\alpha+1}{2} - \sqrt{\left(\frac{\alpha-1}{2}\right)^2 + 2\alpha R}. \quad (60)$$

Let us define $d_0 = \lceil n\delta_{GV}(2R) \rceil$ and $t_0 = \lfloor \frac{d_0-1}{2} \rfloor$. The expectation of the conditional probability can be expressed as follows

$$\mathbb{E}P(C, u, t) = \frac{1}{N(u)} \sum_{w=d_0}^n \mathbb{E}A_w(C) \sum_{s=0}^{t_0} J(u, s, w) \quad (61)$$

$$= \frac{1}{N(u)} \sum_{w=d_0}^n \frac{(M-1)}{q^{mn}} N(w) \sum_{s=0}^{t_0} J(u, s, w) \quad (62)$$

$$= \sum_{w=d_0}^n \frac{(M-1)}{q^{mn}} \sum_{s=0}^{t_0} J(w, s, u). \quad (63)$$

The last equality follows from the identity in Eq. (13). The error probability takes the following form

$$\mathbb{E}P(C) = \sum_{u \geq d_0 - t_0} \sum_{C_i \in \mathcal{C}} \frac{(M-1)}{q^{mn}} \sum_{w=d_0}^n \sum_{s=0}^{t_0} J(w, s, u) \frac{N(u)s_u}{M} \quad (64)$$

$$= \sum_{u \geq d_0 - t_0} \frac{(M-1)}{q^{mn}} \sum_{w=d_0}^n \sum_{s=0}^{t_0} J(w, s, u) N(u)s_u. \quad (65)$$

We can bound the error expectation from above by changing the limits of the index w as follows:

$$\mathbb{E}P(C) \leq \sum_{u \geq d_0 - t_0} \frac{(M-1)}{q^{mn}} \sum_{w=d_0-2t_0}^n \sum_{s=0}^{t_0} J(w, s, u) N(u)s_u. \quad (66)$$

By the identity (14), we have

$$\sum_{w \geq d_0 - 2t_0} \sum_{s=0}^{t_0} J(w, s, u) = \sum_{s=0}^{t_0} N(s) = B(t_0), \quad (67)$$

for any $u \geq d_0 - t_0$. (Contribution of the values in the range $0 \leq w \leq d_0 - 2t_0 - 1$ is zero.) The expected probability is bounded as follows:

$$\mathbb{E}P(C) \leq \frac{(M-1)B(t_0)}{q^{mn}} \sum_{u \geq d_0 - t_0}^n N(u)s_u. \quad (68)$$

Finally, we have

$$\mathbb{E}P(C) \leq \frac{(M-1)B(t_0)}{q^{mn}} (1 - p_0 - p_1 - \dots - p_{d_0 - t_0 - 1}), \quad (69)$$

where $p_u = N(u)s_u$ as defined above.

To obtain the lower bound for the expectation of the error probability, we change the limits of the index u in Eq. (65) as follows

$$\mathbb{E}P(\mathcal{C}) \geq \sum_{u \geq d_0 + t_0} \frac{(M-1)}{q^{mn}} \sum_{w=d_0}^n \sum_{s=0}^{t_0} J(w, s, u) N(u) s_u. \quad (70)$$

Similar to above, we have for any $u \geq d_0 + t_0$,

$$\sum_{w \geq d_0} \sum_{s=0}^{t_0} J(w, s, u) = \sum_{s=0}^{t_0} N(s) = B(t_0). \quad (71)$$

(Again the contribution of the cases $0 \leq w \leq d_0 - 1$ is zero.) So we obtain the lower bound for the expectation of the error probability

$$\mathbb{E}P(\mathcal{C}) \geq \frac{(M-1)B(t_0)}{q^{mn}} \sum_{u \geq d_0 + t_0}^n N(u) s_u \quad (72)$$

$$\geq \frac{(M-1)B(t_0)}{q^{mn}} (1 - p_0 - p_1 - \dots - p_{d_0 + t_0 - 1}). \quad (73)$$

The following theorem gives the expectation of the error probability of a random $[n, m, k]$ linear code. We omit the proof since it follows the same arguments.

Theorem 6 Let $\mathcal{C}$ be a random linear $[n, m, k]$ code with $0 \leq R < 1$. Let all codewords be equally probable and let all errors of rank u occur with probability s_u . The expectation of the decoding error probability $\mathbb{E}P(\mathcal{C})$ of $\mathcal{C}$ satisfies

$$\frac{(q^k - 1)B(t_0)}{q^{mn}} (1 - p_0 - \dots - p_{d_0 + t_0 - 1}) \leq \mathbb{E}P(\mathcal{C}) \leq \frac{(q^k - 1)B(t_0)}{q^{mn}} (1 - p_0 - \dots - p_{d_0 - t_0 - 1}), \quad (74)$$

where $t_0 = \lfloor \frac{d_0 - 1}{2} \rfloor$, $d_0 = \lceil n\delta_{GV}(R) \rceil$ and $p_u = N(u) s_u$.

Author details

İbrahim Özen

Department of Mathematics, Marmara University, İstanbul, Turkey

*Address all correspondence to: iozen@marmara.edu.tr

IntechOpen

© 2025 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Bartz H, Holzbaur L, Liu H, Puchinger S, Renner J, Wachter-Zeh A. Rank-metric codes and their applications. *Foundations and Trends® in Communications and Information Theory*. 2022;**19**(3):390-546
- [2] Sheekey J. 13. MRD codes: Constructions and connections. In: Schmidt K, Winterhof A, editors. *Combinatorics and Finite Fields: Difference Sets, Polynomials, Pseudorandomness and Applications*. Vol. 23. Berlin, Boston: de Gruyter; 2019. pp. 255-286
- [3] Gorla E. Rank-metric codes. arXiv: 1902.02650 [cs.IT]. 2019
- [4] Gorla E, Ravagnani A. Codes endowed with the rank metric. In: *Network Coding and Subspace Designs*. Cham: Springer; 2018. pp. 3-23
- [5] Koetter R, Kschischang FR. Coding for errors and erasures in random network coding. *IEEE Transactions on Information Theory*. 2008;**54**(8): 3579-3591
- [6] Gadouneau M, Yan Z. On the decoder error probability of bounded rank-distance decoders for maximum rank distance codes. *IEEE Transactions on Information Theory*. 2008;**54**(7): 3202-3206
- [7] Loidreau P. Asymptotic behaviour of codes in rank metric over finite fields. *Designs, Codes and Cryptography*. 2014; **71**:105-118. DOI: 10.1007/s10623-012-9716-0
- [8] Gadouneau M, Yan Z. Bounds on covering codes with the rank metric. *IEEE Communications Letters*. 2009; **13**(9):691-693. DOI: 10.1109/LCOMM.2009.090447
- [9] Brouwer AE, Cohen AM, Neumaier A. Distance-Regular Graphs. *Ergebnisse der Mathematik und Ihrer Grenzgebiete 3*. 1989
- [10] Delsarte P. Bilinear forms over a finite field, with applications to coding theory. *Journal of Combinatorial Theory, Series A*. 1978;**25**:226-241
- [11] de la Cruz J, Gorla E, López HH, Ravagnani A. Weight distribution of rank-metric codes. *Designs, Codes and Cryptography*. 2018;**86**:1-16
- [12] Barg A, Forney D. Random codes: Minimum distances and error exponents. *IEEE Transactions on Information Theory*. 2002;**48**(9):2568-2573
- [13] Gadouneau M, Yan Z. Properties of codes with the rank metric. In: *Proceedings of Globecom. San Francisco: GLO-BECOME*; 2006
- [14] Gadouneau M, Yan Z. On the decoder error probability of bounded rank distance decoders for rank metric codes. In: *2009 IEEE Information Theory Workshop*. New York City: IEEE; 2009. pp. 485-489
- [15] Gadouneau M, Yan Z. Preprint: On the Decoder Error Probability of Rank Metric Codes and Constant-Dimension Codes. DOI: 10.48550/arXiv.0812.2379

Section 2

Modern and AI-Based Encryption Systems

Perspective Chapter: Encryption of Information Based on Controlled Chaotic Systems

Vladimir Nikolaevich Shashikhin

Abstract

The paper deals with the problem of ensuring the safety of storage, processing, and transmission of information based on the cryptographic method using chaotic systems. The encryption algorithm is built on the basis of multidimensional mapping with specific properties of dependence on the initial data and transitivity for any pair of open sets. Algorithms for encrypting black and white and color images and text messages are considered. To strengthen the cryptographic resistance of the proposed algorithm to statistical and differential analysis, it is proposed to use a chaotic system with variable characteristics. The change in the characteristics, and above all the entropy and Lyapunov characteristic exponents, is achieved by introducing feedback on the phase vector of the system. Modal control methods were used to calculate the feedback coefficients in a chaotic mapping. Various examples of the use of chaotic mappings for image and text encryption are given.

Keywords: encryption of images, text encryption, chaotic mappings, control of the spectrum of Lyapunov characteristic exponents, statistical and differential analyses

1. Introduction

The problem of ensuring the security of information transmission is now becoming more and more urgent in connection with the increasing flow of information transmitted over open communication lines. Reliable encryption methods in the field of information storage and transmission are necessary in various areas: medical information systems, confidential video conferencing, and state and military communication systems.

Cryptographic methods stand out among various methods of protecting information and ensuring its integrity. However, traditional encryption algorithms, for example, AES and DES, were developed without taking into account the specific requirements for image encryption [1, 2]: a large amount of memory occupied, limited processing, and transmission time. Therefore, it became necessary to create new encryption algorithms based on the use of nonlinear functions [3, 4].

The development of encryption algorithms based on nonlinear functions is proceeding, among other things, along the path of applying methods and algorithms from other fields. The principle of phishing is considered in Ref. [5]. An algorithm based on cellular

automata and the operation of DNA sequences is presented in Ref. [6]. The use of geometric Clifford algebras with hypercomplex numbers (quaternions and octonions), which reduce the encryption time, is considered in Refs. [7, 8]. Encryption algorithms are also being built using fractal sorting matrices—square matrices of non-repeating integers [9]. In Ref. [10], an encryption method is considered based on the existence of periodic disturbances for nonlinear mappings, which lead to stabilization of cycles of a certain period and the output of the system to a regular mode. Information is encrypted using a one-to-one correspondence of text characters and stable cycles of perturbed display.

One of the promising directions in modern cryptography is the development and research of data encryption algorithms based on dynamic chaos [11, 12]. Such properties of chaotic systems as the exponential divergence of trajectories, ergodicity, and randomization are useful in the development of encryption schemes for digital images [13, 14]. Modern approaches to encryption use various chaotic maps and algorithms based on the composition of two maps that implement the operation of randomization and entanglement [15–18]. In Ref. [19], an algorithm for encryption of permutation-diffusion type images using the Kent mapping is considered [20]. An image encryption scheme using one-time keys based on linked chaotic systems, which allows to improve security indicators, is presented in Ref. [21]. A block encryption algorithm based on the use of several chaotic systems is proposed in Ref. [22]. Encryption based on hyperchaotic mappings is considered in Ref. [23].

The chapter discusses an algorithm for encrypting information matrices based on chaotic mapping, which simultaneously implements the operation of mixing and entanglement. To increase the cryptographic strength of the algorithm, a procedure for changing the characteristics of chaotic mapping (entropy and Lyapunov characteristic exponents) based on the introduction into the feedback system is proposed. A procedure for changing the spectrum of Lyapunov characteristic exponents of chaotic mapping using the modal control method has been developed.

2. Mathematical formulation of the problem

System model with chaotic dynamics. A nonlinear differential equation with a given initial state is considered as an evolutionary operator for the implementation of the encryption algorithm

$$\dot{x}(t) = F(x(t)), \quad x(t_0) = x_0, \quad (1)$$

where $x(t) \in P \subseteq R^n$ is the phase vector of the system; P is the phase space of the system; t is time, and $F : R^n \rightarrow R^n$ is a vector function with components $f_i(x(t))$, $i = \overline{1, n}$.

Among the set of nonlinear dynamical systems $S = \{P, F\}$, we will consider systems with chaotic mapping, for which the following conditions are met.

1. The mapping f has a significant dependence on the initial data or is sensitive (if there exists such a number $\delta > 0$ that for any $\varepsilon > 0$ and any point $x' \in X$ there is a point $x'' \in X$ and a number $m \in M$ such that $\rho(x', x'') < \varepsilon$, but $\rho(f^{(m)}(x'') - f^{(m)}(x')) \geq \delta$).
2. The mapping f is transitive (for any pair U, V of open sets, there is such $m \geq 0$ that $f^{(m)}(U) \cap V \neq \emptyset$).

Black and white image model. Let the raster model of the original image having a rectangular shape be represented by the following mapping

$$I : [a, b] \times [c, d] \rightarrow L(\mathbb{R}^{N \times M}), \quad (2)$$

where $L(\mathbb{R}^{N \times M})$ is the space of numerical matrices of dimension $N \times M$.
 Values N, M are related to pixel grid sizes

$$\Delta^{WH} = \{(i, j) : i = \overline{1, N} = [W], j = \overline{1, M} = [H]\}, \quad (3)$$

where $[*]$ is the whole part of the number.

With the help of sampling and quantization operations, the description of images is reduced to a set of samples, which can be represented in the form of a matrix

$$I = (I_{ij})_{i,j=1}^{N,M} \in L(\mathbb{R}^{N \times M}), \quad (4)$$

elements of which are implementations on a discrete grid of continuous functions of two variables $I : (0, W) \otimes (0, H) \rightarrow R$. Elements of these matrices, when encoding the pixel intensity with an eight-bit code, take integer values from the interval $[0; 255]$.

Color image model. In terms of bitmap graphics, a color image *IMG* is a set of two-dimensional elements – pixels $p(v, z)$, which are characterized by the coordinates $v \in \overline{1, N}$ and $z \in \overline{1, M}$. The numbers N and n determine the height and width of the image, respectively. Also, each pixel is an indivisible object described by three properties—the intensity of red r , blue b , and green g (using the RGB color model), which takes values from the range $[0, 2^d]$, where d is the number of bits of the intensity representation of each color.

Thus, the color image model can be represented as a function of five variables:

$$IMG = \varphi(v, z, r, b, g), \quad (5)$$

or a matrix

$$IMG = \begin{bmatrix} p_{11}(r, b, g) & p_{12}(r, b, g) & \dots & p_{1M}(r, b, g) \\ p_{21}(r, b, g) & p_{22}(r, b, g) & \dots & p_{2M}(r, b, g) \\ \dots & \dots & \dots & \dots \\ p_{N1}(r, b, g) & p_{N2}(r, b, g) & p_{11}(r, b, g) & p_{NM}(r, b, g) \end{bmatrix} \quad (6)$$

Text message model. Plaintext is a probabilistic model in the form of a sequence of signs $c_1, c_2, \dots, c_i, \dots, c_m$, in which each sign appears with a probability $p(c_i)$, and the probability of the appearance of this plaintext is

$$p(c_1, c_2, \dots, c_i, \dots, c_m) = \prod_{i=1}^m p(c_i). \quad (7)$$

Thus, open text is considered as an implementation of a stationary ergodic process with a discrete time and a finite number of states.

The open text is converted into an information matrix, which is a two-dimensional array of elements $q(i, j)$, which are characterized by the coordinates $i \in \overline{1, N}$ and

$j \in \overline{1, M}$. The numbers N and n determine the number of rows and the number of columns of the information matrix, respectively. Each element is a binary number q from the gap $[0, 255]$, which corresponds to the code of the symbol of the plaintext alphabet $A(a_1, a_2, \dots, a_j, \dots, a_n)$. The set of elements q can be represented as a function of three variables: $Q = \varphi(i, j, q)$,

or a matrix of the following form

$$Q = \begin{bmatrix} q_{11} & q_{12} & \dots & q_{1M} \\ q_{21} & q_{22} & \dots & q_{2M} \\ \dots & \dots & \dots & \dots \\ q_{N1} & q_{N2} & \dots & q_{NM} \end{bmatrix}, \quad (8)$$

Criteria for resistance to cryptanalysis. The task of encryption using chaotic mappings is reduced to the formation of an information matrix $IMG^{(l)}$ obtained by encryption using l times of chaotic mapping $F^{(l)}$ to the original information matrix $IMG^{(0)}$:

$$IMG^{(0)} \xrightarrow{F^{(l)}} IMG^{(l)}, \quad (9)$$

where $IMG^{(0)}$ is determined by formula (2), and the mapping F — by formula (1).

The encryption algorithm must have cryptographic resistance $S = \{s_1, s_2\}$, where S_1 is resistance to statistical attacks, S_2 is resistance to differential attacks:

$$S_1 = \left\{ \begin{array}{l} \rho_C(p_i^c, p_{i+1}^c) \rightarrow 0 \\ H(m^C) \rightarrow 8 \end{array} \right\}, \quad S_2 = \left\{ \begin{array}{l} NPCR \rightarrow 100\% \\ UACI \rightarrow 33\% \end{array} \right\}, \quad (10)$$

where $\rho_C(p_i^c, p_{i+1}^c)$ is the correlation coefficient of values between the neighboring pixels horizontally, vertically, and diagonally of the information matrix; $H(m^C)$ is the information entropy; $NPCR$ is the percentage of near pixel change rate, and $UACI$ is the unified averaged changed intensity.

The pairwise correlation between two adjacent pixels horizontally, vertically, and diagonally of the source and encrypted information matrix is calculated by the formula

$$\rho(u_i, v_{i+1}) = \frac{\sum_{i=1}^{N \times M} (u_i - \bar{U})(v_{i+1} - \bar{V})}{N \times M} \sqrt{\frac{\sum_{i=1}^{N \times M} (u_i - \bar{U})^2}{N \times M} \frac{\sum_{i=1}^{N \times M} (v_{i+1} - \bar{V})^2}{N \times M}}, \quad (11)$$

$$\bar{U} = \frac{\sum_{i=1}^{N \times M} u_i}{N \times M}, \quad \bar{V} = \frac{\sum_{i=1}^{N \times M} v_i}{N \times M}, \quad (12)$$

where u_i, v_{i+1} is the intensity of the i -the pixel and the adjacent pixel,

$U = \{u_1, u_2, \dots, u_i, \dots, u_{N \times M}\}$, $V = \{v_1, v_2, \dots, v_i, \dots, v_{N \times M}\}$ are a number of pixel intensity values in the information matrix and a number of intensity values of neighboring pixels.

Information entropy is given by

$$H(m_s) = \sum_{s=0}^{2^n-1} P(m_s) \log_2(1/P(m_s)), \quad (13)$$

where $P(m_s)$, the probability of belonging to the intensity level $s \in [0 ; 255]$, is calculated

$$P(m_s) = \frac{m_s}{N \times M}. \quad (14)$$

To assess the resistance of the algorithm to differential analysis, the following is calculated:

- percentage of changed pixels (NPCR)

$$\Pi(I_1, I_2) = \frac{\sum_{i=1}^N \sum_{j=1}^M D(i, j)}{N \times M} \times 100\%, \quad (15)$$

where

$$D(i, j) = \begin{cases} 1, & \text{if } e_1(i, j) = e_2(i, j) \\ 0, & \text{if } e_1(i, j) \neq e_2(i, j) \end{cases} \quad \forall i = \overline{1, N}, \quad \forall j = \overline{1, M}, \quad (16)$$

I_1 —the initial information matrix, I_2 —the initial information matrix in which the level of one pixel is changed; $\hat{E}_1^{(p)} \hat{E}_2^{(p)}$ —encrypted information matrices corresponding to the initial information matrices I_1, I_2 ; $\hat{e}_1^{(p)}(i, j), \hat{e}_2^{(p)}(i, j)$ —the value of the pixel with the number (i, j) in the matrices $\hat{E}_1^{(p)}$ and $\hat{E}_2^{(p)}$;

- average change in pixel intensity (UACI)

$$C(E_1, E_2) = \frac{1}{N \times M} \sum_{i=1, j=1}^{N, M} \frac{|e_1(i, j) - e_2(i, j)|}{255} \times 100\%. \quad (17)$$

Control of the chaotic system spectrum. A mathematical model of a chaotic system in the synthesis of an encryption algorithm is an inhomogeneous differential equation with a control action

$$\dot{x}(t) = F(x(t)) + Bu(t), \quad x(t_0) = x_0. \quad (18)$$

The Lyapunov spectrum of the original nonlinear system (18)

$$\sigma(F) = \{\chi_i(F), i = \overline{1, n}\} \quad (19)$$

consists of n various descendingly ordered Lyapunov characteristic exponents $\chi_1(F) \geq \chi_2(F) \geq \dots \geq \chi_n(F)$.

The task of controlling the Lyapunov spectrum is to determine feedback over the phase vector of a nonlinear system

$$u(t) = K^* x(t) \quad (20)$$

such that the closed nonlinear system

$$\dot{x}(t) = F(x(t)) + BK^* x(t), \quad x(t_0) = x_0 \quad (21)$$

has a spectrum

$$\sigma(F + BK^*) = \{\chi_i(F + BK^*), i = \overline{1, n}\}, \quad (22)$$

equal to the required spectrum

$$\sigma(G) = \{\chi_i(G), i = \overline{1, n}\}. \quad (23)$$

The strength of the encryption algorithm when using systems with chaotic dynamics depends on the value of the senior (positive) Lyapunov characteristic exponent. Therefore, the task of increasing resistance to various kinds of attacks is reduced to determining the control parameters (20), in which the senior Lyapunov characteristic index (22) will increase in the closed-loop system (21). The feedback coefficient further allows expanding the “key size” of the encryption algorithm.

3. Algorithm for encrypting a black-and-white image

Chaotic mapping. In the process of image encryption, Rössler’s three-dimensional chaotic mapping is used as a model of the nonlinear system (1) [24].

$$\begin{cases} \dot{x}_1 = -(x_2 + x_3) \\ \dot{x}_2 = x_1 + ax_2 \\ \dot{x}_3 = b + x_3(x_1 - c). \end{cases} \quad (24)$$

This mapping, at the values of the parameters $a = 0.2, b = 0.2, c = 5.7$, has a spectrum of Lyapunov characteristic exponents equal to

$$\sigma(F) = \{\chi_1(F) = 0.1016; \chi_2(F) = 0.0922; \chi_3(F) = -5.6953\}, \quad (25)$$

and the trajectory in the three-dimensional phase space, which has the form shown in **Figure 1**.

The spectrum of Lyapunov characteristic exponents (25) and a strange attractor indicate the presence of chaotic dynamics in the system (24).

Encryption algorithm. The grayscale image encryption algorithm $N \times M$ using nonlinear mapping (24) has the following steps on one round of encryption.

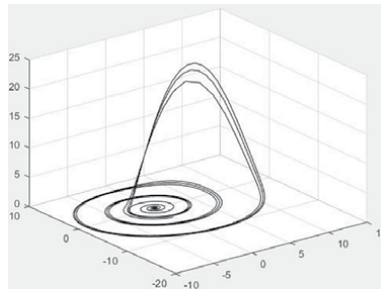


Figure 1.
Trajectory of the Rössler system.

Step 1. A raster model of the original image (3) with a matrix of the form (4) is formed on the basis of the original image

$$I = (I_{ij})_{i,j=1}^{N,M} \in L(\mathbb{R}^{N \times M}), \quad (26)$$

where i is the pixel number in the vertical row; j is the pixel number in the horizontal row; I_{ij} is the brightness value of the pixel with the number i, j ; N is the number of rows; and M is the number of columns of the pixel matrix determined by the grid dimensions (3).

Step 2. For some initial state $x(0) = (x_1(0), x_2(0), x_3(0))$ determined by the point of exit of the trajectory of the chaotic system (24) to the strange attractor, three sequences are generated

$$\begin{aligned} X_1 &= \{x_1(t_1), \dots, x_1(t_k), \dots, x_1(t_s)\}, \\ X_2 &= \{x_2(t_1), \dots, x_2(t_k), \dots, x_2(t_s)\}, \quad s = N \times M. \\ X_3 &= \{x_3(t_1), \dots, x_3(t_k), \dots, x_3(t_s)\}. \end{aligned} \quad (27)$$

In the system of Eq. (27), sequences X_1 and X_2 define the mixing of pixels, and sequence X_3 defines the scattering (brightness changes) of pixels.

Step 3. A chaotic matrix of the encrypted image of the first round is formed using the third equation of the system (27)

$$E^{(1)} = \begin{bmatrix} e_{1,1}^{(1)} & e_{1,2}^{(1)} & \dots & e_{1,M}^{(1)} \\ e_{2,1}^{(1)} & e_{2,2}^{(1)} & \dots & e_{2,M}^{(1)} \\ \dots & \dots & \dots & \dots \\ e_{N,1}^{(1)} & e_{N,2}^{(1)} & \dots & e_{N,M}^{(1)} \end{bmatrix} \in \mathbb{R}^{N \times M}, \quad (28)$$

where $e_{ij}^{(1)} = [x_3(t) \bmod 255]$ is the intensity of the gray pixel with the row number $i = [x_1(t) \bmod M]$ and the column number $j = [x_2(t) \bmod N]$; $[*]$ is the whole part of the number.

Further, repeating steps 2–3 above the image matrix (28) for p rounds, we obtain an encrypted image $\hat{E}^{(p)}$ of the form

$$\hat{E}^{(1)} = (\hat{e}_{ij}^{(1)})_{i,j}^{N,M} \in \mathbb{R}^{N \times M}. \quad (29)$$

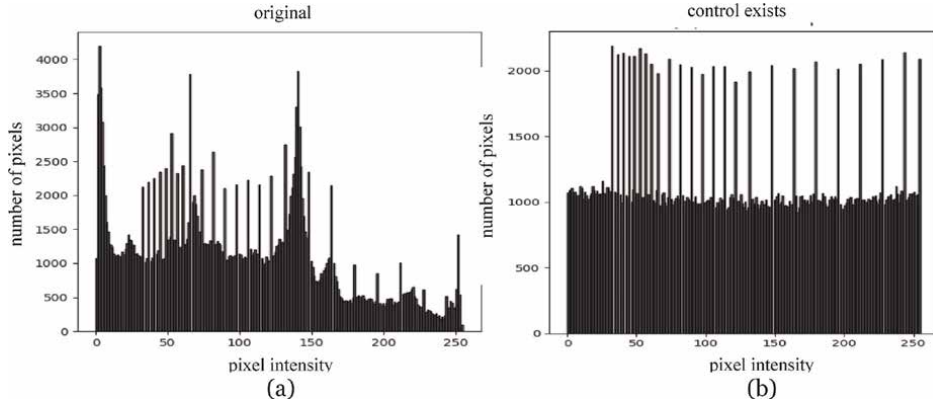
The number of rounds is determined by the required cipher strength indicators.

Evaluation of the algorithm's robustness is carried out using statistical and differential cryptanalysis [25]. The gray-scale Lena image size 512×512 is used as the source image.

The distribution of pixels by gray intensity values for the original image is shown in **Figure 2a**, and for the encrypted one in **Figure 2b**.

The results of calculating the resistance criteria according to formulas (11)–(17) after two rounds of encryption using a chaotic display without a control and with a control action are shown in **Table 1**.

Control of Lyapunov characteristic exponents of chaotic mapping. To increase the durability of the proposed image encryption algorithm, a control action is introduced

**Figure 2.**

a—Histogram of the original image, b—histogram of the encrypted image.

Chaotic mapping	Leading LCE	Picture	Correlation coefficient			Entropy	NPCR %	UACI %
			Horiz.	Vert.	Diag.			
—	—	original	0.0293	0.0263	0.0653	7.55	—	—
Without control	0.1016	encrypted	0.000175	0.000192	0.000380	7.76	99.54	33.46
With control	0.2862	encrypted	0.000101	0.000186	0.000117	7.96	99.55	33.61

Table 1.

Criteria for the strength of the encryption algorithm.

into a system with chaotic dynamics to increase the positive Lyapunov characteristic exponent and information entropy. The control is sought in the form of linear feedback on phase coordinates (20).

The solution to the problem of changing the spectrum of characteristic indicators of a nonlinear system is based on the Grobman-Hartman theorem [26]. Any system in the neighborhood of a hyperbolic singularity is locally topologically equivalent to its linear approximation. Thus, the behavior of a nonlinear system in the vicinity of a hyperbolic singular point is similar to the behavior of a linearized system. The special point is hyperbolic if the Jacobian matrix in it does not have eigenvalues on the imaginary axis. A change in the characteristic exponents of the linearized system, which coincide with the real part of the eigenvalues of the Jacobian matrix, entails a change in the characteristic exponents of the nonlinear system.

The desired eigenvalues of the Jacobian matrix of the linearized system can be provided using the modal regulator synthesis technique based on linear matrix inequalities [27].

Synthesis of control of a nonlinear system. Let, at the singular point x^* of system (18) in the absence of control, the Jacobi matrix be equal to J^* , and the vector of its eigenvalues be equal to v^* . Let us set the vector of the desired eigenvalues $\bar{v}^*$ of the Jacobian $\bar{J}^*$ of the nonlinear system in the form

$$\bar{v}_i^* = v_i^* + \alpha \operatorname{Re}(v_i^*), \quad (30)$$

where α is the coefficient selected according to the schedule.

In order for the Jacobian $\bar{J}^*$ of the system (21) to have the given eigenvalues (23), we choose the control in the form

$$u(t) = Kx(t), \quad (31)$$

then the Jacobian of the system with this control will be equal to

$$\bar{J}^* = J^* + BK, \quad (32)$$

where $K \in \mathbb{R}^{m \times n}$ is the feedback coefficient, which is found by the method of modal control synthesis.

To select the coefficient α , a graph of the dependence of the senior characteristic indicator of a nonlinear system with control on the coefficient α is plotted.

The Rössler system, at the values $a = 0.2$, $b = 0.2$, $c = 5.7$ of the parameters, has two special points

$$\begin{aligned} x1^* &= (0.0070 \quad -0.0351 \quad 0.0351), \\ x2^* &= (5.6930 \quad -28.4649 \quad 28.4649), \end{aligned} \quad (33)$$

and the Jacobian matrix is

$$J = \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ x_3 & 0 & x_1 - c \end{bmatrix}. \quad (34)$$

Eigenvalues of the Jacobian matrix calculated at singular points:

$$v(J(x1^*)) = \begin{cases} v_1 = 0.0970 + 0.9952i \\ v_2 = 0.0970 - 0.9952i, \\ v_3 = -5.6870 \end{cases} \quad (35)$$

$$v(J(x2^*)) = \begin{cases} v_1 = 5.4280i \\ v_2 = -5.4280i, \\ v_3 = 0.1930 \end{cases} \quad (36)$$

and the feedback coefficient in a nonlinear system is

$$K^* = (1.0426 \quad -0.4943 \quad -22.8945). \quad (37)$$

The spectrum (22) of the nonlinear system (21) with synthesized control is equal to

$$\sigma(F + bK^*) = \{\chi_1 = 0.2862; \chi_2 = 0.0026; \chi_3 = -3.8931\}. \quad (38)$$

The higher indicator of the closed system is 2.82 times higher than the higher indicator of the original system, which indicates an increase in chaos in the system.

Properties of an encryption algorithm based on controlled chaos. The strength of the encryption algorithm based on a chaotic feedback system is evaluated according to the same criteria of statistical and differential cryptanalysis that were used when testing

the algorithm without control. The results of testing the encryption algorithm with control are given in **Table 1**.

From the above test results, it follows the improvement of all compared criteria of cryptographic strength when using an encryption algorithm with a control action compared to an algorithm without control.

4. Encryption of color images

Hyperchaotic system. We will consider the encryption of a color image using an example of the Sprott system [28], which can be recorded in the form

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = x_5 \\ \dot{x}_5 = 3.9(x_1^2 - 1) - 9.17x_2 - 3.94x_3 - 7.068x_4 - x_5 \end{cases} \quad (39)$$

The singular points of (39) have coordinates $O_1(1; 0; 0; 0; 0)$, $O_2(-1; 0; 0; 0; 0)$, and its Lyapunov characteristic exponents are

$$\lambda_1 = 0.2211; \quad \lambda_2 = 0.0053; \quad \lambda_3 = -0.2309; \quad \lambda_4 = -0.3073; \quad \lambda_5 = -0.6051. \quad (40)$$

The projections of the phase portrait of the Sprott system are shown in **Figure 3**.

The type of projections of the phase portrait and the spectrum of Lyapunov characteristic exponents indicate the presence of chaotic dynamics in the system (39).

Encryption algorithm based on the Sprott system. The color image encryption algorithm $N \times M$ with a size of using nonlinear mapping (39) has the following steps on one round.

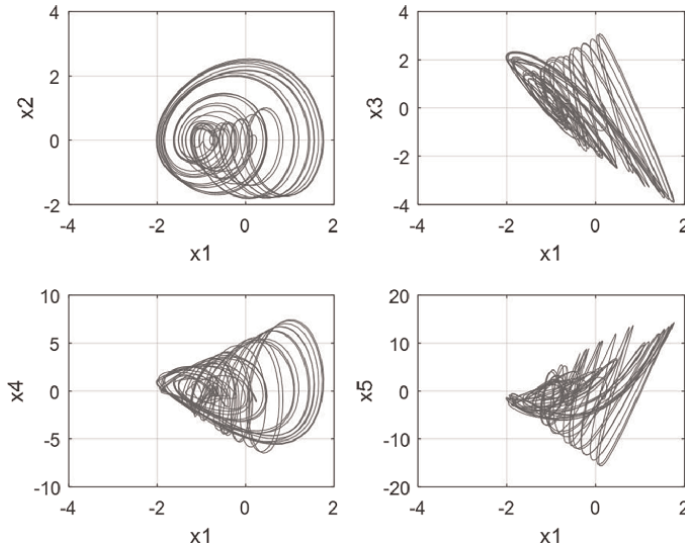


Figure 3.
The projections of the phase portrait of the Sprott system.

Step 1. A raster model of the original image (6) is formed on the basis of three matrices

$$IMG_c = \begin{bmatrix} p_{11}(c) & \dots & p_{1M}(c) \\ \dots & \dots & \dots \\ p_{N1}(c) & \dots & p_{NM}(c) \end{bmatrix}, \quad (41)$$

where $p_{vz}(c)$, $c \in \{r, b, g\}$ is the brightness value of the pixel in red, blue, and green.

Step 2. For some initial state

$$x(0) = (x_1(0), x_2(0), x_3(0), x_4(0), x_5(0)), \quad (42)$$

determined by the point of exit of the trajectory of the hyperchaotic system (39) to the strange attractor, five sequences are generated

$$\begin{aligned} X_1 &= \{x_1(t_1), \dots, x_1(t_k), \dots, x_1(t_s)\}, \\ X_2 &= \{x_2(t_1), \dots, x_2(t_k), \dots, x_2(t_s)\}, \\ X_3 &= \{x_3(t_1), \dots, x_3(t_k), \dots, x_3(t_s)\}, \\ X_4 &= \{x_4(t_1), \dots, x_4(t_k), \dots, x_4(t_s)\}, \\ X_5 &= \{x_5(t_1), \dots, x_5(t_k), \dots, x_5(t_s)\}. \end{aligned} \quad (43)$$

$s = N \times M$

In the system of Eq. (43), the sequences X_1 and X_2 define the mixing of pixels, and the sequences X_3, X_4, X_5 define the scattering (brightness changes of three colors) of pixels.

Step 3. Three chaotic matrices of the encrypted image of the first round for three colors are formed using the third, fourth, and fifth Eq. (43)

$$IMG_c^{(1)} = \begin{bmatrix} p_{11}(c)^{(1)} & \dots & p_{1M}(c)^{(1)} \\ \dots & \dots & \dots \\ p_{N1}(c)^{(1)} & \dots & p_{NM}(c)^{(1)} \end{bmatrix} \in \mathbb{R}^{N \times M}, \quad (44)$$

where $p_{vz}(c)^{(1)} = [x_k(t) \bmod 255]$, $k = 3, 4, 5$ is the intensity of the three colors of the pixels with the row number $v = [x_1(t) \bmod N]$ and the column number $z = [x_2(t) \bmod M]$; $[*]$ is the whole part of the number.

Further, applying steps 2–3 to images (44) for l rounds, we obtain an encrypted image $IMG_c^{(l)}$

$$IMG_c^{(l)} = \begin{bmatrix} p_{11}(c)^{(l)} & \dots & p_{1M}(c)^{(l)} \\ \dots & \dots & \dots \\ p_{N1}(c)^{(l)} & \dots & p_{NM}(c)^{(l)} \end{bmatrix} \in \mathbb{R}^{N \times M}. \quad (45)$$

The number of rounds is determined by the required cipher strength indicators.

We apply the considered encryption algorithm based on hyperchaotic mapping (39) to an image of 300x300 pixels, which is shown in **Figure 4**, and the encryption result determined by the matrix (45)—in **Figure 5**.

The histograms of the color intensity distribution are shown in **Figure 6**.



Figure 4.
Source image.



Figure 5.
Encrypted image.

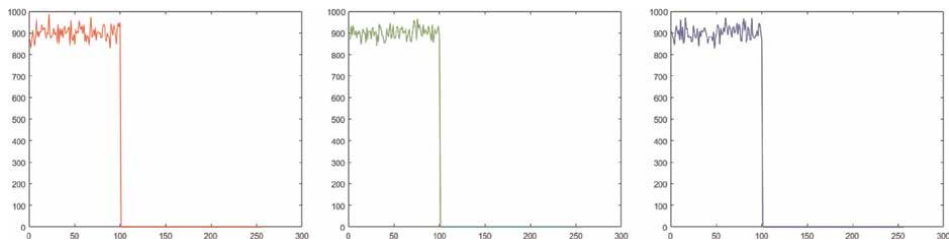


Figure 6.
Red, green, and blue intensity distribution histograms.

Evaluation of the strength of the encryption algorithm. The proposed color image encryption algorithm is tested according to the criteria of resistance to statistical and differential cryptanalysis.

Statistical cryptanalysis. The assessment of the algorithm's resistance to static cryptanalysis is reduced to the calculation of the following indicators: vertical, horizontal, and diagonal correlation coefficients between adjacent image pixels, as well as information entropy.

The results of calculating the criteria for the strength of the encryption algorithm using hyperchaotic mapping according to formulas (8)–(14) are shown in **Table 2**.

Increasing the strength of the algorithm. For nonlinear dynamic systems that are used in image encryption, one can set and solve the problem of increasing chaos. Since one of the key indicators of randomness is the Lyapunov characteristic exponents, by introducing feedback control, one can increase the value of the highest positive Lyapunov characteristic exponent and achieve the desired spectrum. The Lyapunov

Hyperchaotic mapping		Without increasing chaos	With increasing chaos
Static cryptanalysis	ρ_{zop}^r	0.0189	0.0021
	ρ_{sep}^r	0.0201	0.0011
	ρ_{duaz}^r	0.0428	0.0014
	ρ_{zop}^g	0.0156	0.0015
	ρ_{sep}^g	0.0196	0.0019
	ρ_{duaz}^g	0.0231	0.0022
	ρ_{zop}^b	0.0299	0.0009
	ρ_{sep}^b	0.0153	0.0005
	ρ_{duaz}^b	0.0461	0.0014
	$H(m^r)$	6.8456	7.9734
	$H(m^g)$	7.5464	7.9649
	$H(m^b)$	7.7990	7.9689
Differential cryptanalysis	$NPCR^r$	96.1312	99.6468
	$NPCR^g$	97.5709	99.6213
	$NPCR^b$	97.3912	99.6545
	$UACI^r$	31.4599	33.0589
	$UACI^g$	32.3490	33.1415
	$UACI^b$	32.1267	33.1828

Table 2.
Evaluation of the strength of the encryption algorithm.

characteristic exponents depend on the eigenvalues of the Jacobian matrix of the nonlinear system. Following the algorithm presented in Ref. [27], we synthesize feedback that will make the system (39) more chaotic and, consequently, will enhance the strength of the encryption algorithm.

Let us introduce the control action into the Sprott system (39)

$$u(t) = Kx(t) \in \mathbb{R}^m, \quad m = 5, \quad (46)$$

then it will take the form

$$\dot{x}(t) = F(x(t)) + Bu(t), \quad x(0) = x_0, \quad (47)$$

where $B \in \mathbb{R}^m$ is the matrix of control inputs.

The Jacobian matrix of system (47) calculated at the singular point O_1 is equal to

$$J = \partial F / \partial x|_{x^*} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 7.8 & -9.17 & -3.94 & -7.068 & -1 \end{pmatrix}. \quad (48)$$

The required eigenvalues of the Jacobian matrix $\tilde{J}$ are selected by the following formula:

$$\bar{v}(\tilde{J}) = v(J) + \alpha \cdot \text{Re}(v(J)) + \beta \cdot \text{Im}(v(J)), \quad (49)$$

where $v(J)$ are the eigenvalues of the Jacobian matrix of the original nonlinear system; α is a coefficient that affects the shift of the eigenvalues along the real axis; β is a coefficient that affects the shift of the eigenvalues along the imaginary axis. When solving the chaos amplification problem, it is necessary to increase the largest positive eigenvalue by choosing $\alpha > 0$ and $\beta > 0$.

By choosing $\alpha = 1$, $\beta = 0$, we determine the control, where the feedback coefficient calculated using the Sylvester equation

$$K = GP^{-1}, \quad P: \tilde{J}P + P\tilde{J} = BG \quad (50)$$

is equal to

$$K = (6.6296; \quad 0.8229; \quad -1.1969; \quad -0.8502; \quad -0.4567). \quad (51)$$

With this control, the spectrum of the nonlinear system (47) has the form:

$$\lambda_1 = 0.4494; \quad \lambda_2 = -0.1228; \quad \lambda_3 = -0.3581; \quad \lambda_4 = -0.6129; \quad \lambda_5 = -0.8124. \quad (52)$$

The higher Lyapunov characteristic exponent is approximately doubled compared to the previous value $\lambda_1 = 0.2211$.

For the system (47) with feedback, the stability criteria were calculated, which are given in **Table 2**. According to **Table 2**, the encryption algorithm with chaos amplification not only meets the criteria of strength, but also shows better indicators of statistical and differential cryptanalysis compared to the algorithm without chaos amplification.

5. Encryption of the text

Chaotic mapping. Text encryption is considered on the example of a three-dimensional continuous Rössler system (24) with parameters equal to $a = 0, 2$; $b = 0, 2$; $r = 5, 7$. Special points of system (24) with the specified parameters have coordinates

$$O_1 = (-14.299 \quad 0.708 \quad -286.694), \quad (53)$$

$$O_2 = (-0.001 \quad 0 \quad -0.014), \quad (54)$$

and its characteristic exponents are equal to

$$\chi_1 = 0.061; \quad \chi_2 = 0.018; \quad \chi_3 = -5.443. \quad (55)$$

The algorithm for encrypting and decrypting text is determined by the following steps. Step 1. The text is converted into a square information matrix of the form (8)

$$Q = (q_{ij})_{i,j=1}^{N,M} \in \mathbb{N}^{N \times M}, \quad (56)$$

where i is the number of the element in the vertical row; j is the number of the element in the horizontal row; q_{ij} is the value of the element in the position i, j ; N is the number of rows; and M is the number of columns of the symbol matrix. In the

matrix Q , each element corresponds to one character of the text. The dimension of the matrix is calculated based on the length of the text, rounded up, and the missing elements are filled with spaces.

Step 2. Based on the key, an array of pseudo-random numbers is formed

$$E = \{\varepsilon_0, \varepsilon_1, \dots, \varepsilon_c, \dots, \varepsilon_s | s = N \times M\}, \quad (57)$$

which is used to set the initial conditions for the third component of the chaotic system. Here $\varepsilon_c \in \mathbb{R}$ and takes values from the interval $[0;1]$.

Step 3. A vector of initial conditions for a chaotic system is formed

$$X(t_0) = (x_1(t_0) = i = 1, \quad x_2(t_0) = j = 1, \quad x_3(t_0) = \varepsilon_0) \quad (58)$$

and the process of integrating Eq. (24) begins. Here, $x_1(t_0) = i = 1$, $x_2(t_0) = j = 1$, the initial conditions for the first and second components, are determined by the position of the first element of the information matrix. The values of the first and second components of the state vector of the system (24) at the time t_1 determine the new position of the information matrix element: $i^1 = [x_1(t_1) \bmod M]$ —row number and $j^1 = [x_2(t_1) \bmod N]$ —column number. Thus, the elements of the information matrix are mixed during the first round. The new value of the element is determined by the sum of the third component of the chaotic system and the pseudo-random number $q_{i,j}^{(1)} = [(x_3(t_1) + \varepsilon_0) \bmod AL]$, which ensures the scattering of the elements. Here AL is the power of the plaintext alphabet.

Step 4. As a result of applying the operations of the third step to each element of the initial information matrix, the information matrix of the first round of encryption is calculated

$$Q^{(1)} = \begin{bmatrix} q_{1,1}^{(1)} & q_{1,2}^{(1)} & \dots & q_{1,M}^{(1)} \\ q_{2,1}^{(1)} & q_{2,2}^{(1)} & \dots & q_{2,M}^{(1)} \\ \dots & \dots & \dots & \dots \\ q_{N,1}^{(1)} & q_{N,2}^{(1)} & \dots & q_{N,M}^{(1)} \end{bmatrix} \in \mathbb{R}^{N \times M}. \quad (59)$$

To calculate the next point of the trajectory of the chaotic system for each element of the information matrix, two coordinates of the current element in the initial information matrix and the next number from the pseudo-random sequence are selected as the vector of the initial conditions.

Step 5. Steps 3 and 4 are repeated l times, l is the number of rounds of encryption.

The decryption process is similar to the encryption algorithm, but the elements of the information matrix are taken in reverse order and the summation with a pseudo-random number is replaced by its subtraction.

Thus, text encryption is essentially reduced to the generation of a chaotic system under given initial conditions of three sequences

$$\begin{aligned} X_1 &= \{x_1(t_1), \dots, x_1(t_k), \dots, x_1(t_s)\}, \\ X_2 &= \{x_2(t_1), \dots, x_2(t_k), \dots, x_2(t_s)\}, \quad s = N \times M. \\ X_3 &= \{x_3(t_1), \dots, x_3(t_k), \dots, x_3(t_s)\}. \end{aligned} \quad (60)$$

The sequences X_1 and X_2 define the mixing of elements, and the sequence X_3 together with the sequence of pseudo-random numbers defines the scattering (code changes) of elements.

Evaluation of the quality of the encryption and decryption algorithm. One of the most important properties of any encryption algorithm is the rate of information loss after decryption.

To measure the quality of encryption, an indicator of the percentage of changed elements of the original matrix and the information matrices obtained after decryption is used.

The quality of text reproduction is also determined by element-by-element comparison of the source text and the text obtained after decryption. The criterion is the percentage of matching characters of both texts.

Experiments to assess the durability and quality of the algorithm were carried out on the text from [29]. The text fragment consists of 3865 characters and is shown in **Figure 7**.

The histograms of the source text and the encrypted information matrix are shown in **Figure 8**.

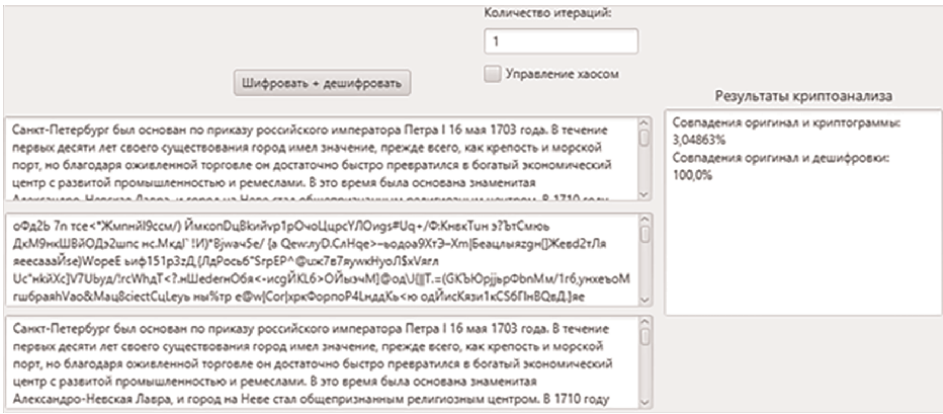


Figure 7.
Fragment of the source and decrypted text.

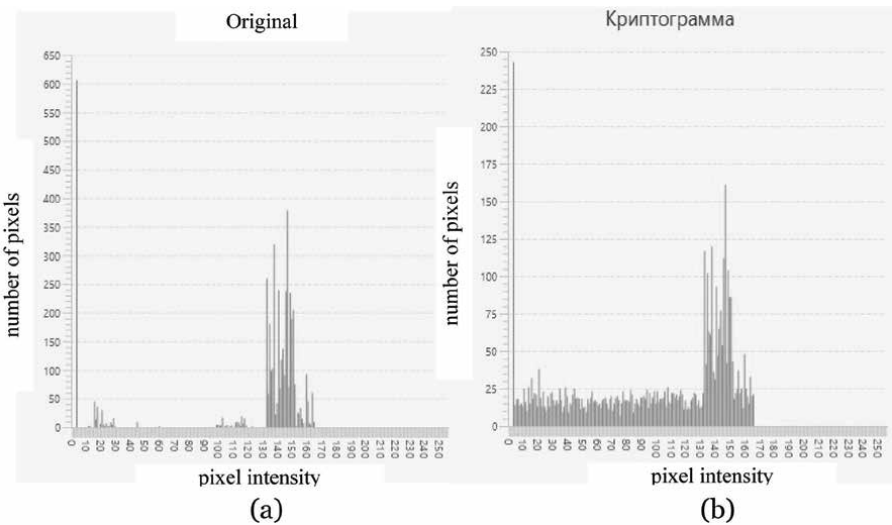


Figure 8.
a—Histogram of plain text; b—histogram of an encrypted information matrix.

Criteria for the strength		Information matrix	
		Initial	Encrypted
Correlation coefficient	horiz.	0.071	0.003
	vertic.	0.038	0.021
	diagon.	-1.4×10^{-5}	-6.7×10^{-4}
Entropy		4.647	6.873
Percentage of elements changed, %		—	97.361
Average change in intensity, %		—	23.653

Table 3.
Criteria for the strength of the algorithm.

The results of calculating the criteria for the stability of the algorithm using the Rössler system are shown in **Table 3**.
The use of a chaotic system makes it possible to implement simultaneous mixing and scattering of the elements of the information matrix. Computational experiments confirm the resistance of the developed algorithm to statistical and differential analysis and the coincidence of the source and decrypted text by 100%.

6. Conclusion

The algorithms for encrypting information matrices associated with images in grayscale, color images, and text messages are presented, based on the use of appropriate chaotic mappings that implement simultaneous mixing and scattering. To increase the cryptographic strength of the algorithm, a method for synthesizing feedback on the phase vector of a nonlinear system is proposed, which provides an increase in the highest Lyapunov characteristic exponent responsible for the degree of randomness. The synthesis technique is based on the modal control method applied to nonlinear chaotic systems.
The use of the proposed algorithms is considered on the example of the chaotic Rössler system and the hyperchaotic Splott system. The test results confirmed the increase in the resistance of the proposed encryption algorithm to statistical and differential analysis due to an increase in the Lyapunov characteristic exponent, which is achieved by introducing feedback into the chaotic system used for encryption.

Author details

Vladimir Nikolaevich Shashikhin
Peter the Great St. Petersburg Polytechnic University, St. Petersburg,
Russian Federation

*Address all correspondence to: shashihin@bk.ru

IntechOpen

© 2024 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Sidorenko AV, Shakinko IV, Sidorenko YV. Image encryption algorithm using two-dimensional chaotic maps. *System Analysis and Applied Informatics*. 2016;**2**:44-49
- [2] Chen P. A fast image encryption based on chaotic map and lookup table. *Nonlinear Dynamics*. 2015;**79**(3): 2121-2131
- [3] Sidorenko AV, Mulyarchuk KS. Data encryption based on discrete chaotic systems and maps. *Reports of BSUIR*. 2013;**71**(1):62-67
- [4] Seyedzadeh SM, Norouzi B, Mirzakuchaki S. RGB image encryption algorithm based on Choquet fuzzy integral. *The Journal of Systems and Software*. 2014;**97**(11):128-139
- [5] Leonov GA, Kuznetsov NV, Mokaev TN. Homoclinic orbits and self-excited and hidden attractors in a Lorenz-like system describing convective fluid motion. *The European Physical Journal Special Topics*. 2015; **224**(8):1421-1458
- [6] Chai X, Gan Z, Yang K. An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and DNA sequence operations. *Signal Processing: Image Communication*. 2017;**52**:6-19
- [7] Kuznetsova KS. Hardware-oriented algorithm of the quaternion cryptosystem. *Technical Sciences*. 2018; **202**(8):182-190
- [8] Chukanov SN. Signal transmission with encryption by geometric algebra method. *Bulletin of Voronezh State University. Series. System analysis and information technology*. 2020;**3**:25-31
- [9] Xian Y. Fractal sorting matrix and its application on chaotic image encryption. *Information Sciences*. 2021;**547**: 1154-1169
- [10] Loskutov AY. Using chaotic mappings for information protection. *Bulletin of the Moscow University. Series 3. Physics. Astronomy*. 2008;**2**: 15-19
- [11] Gulyaev YV, Belyaev RV, Vorontsov GM, et al. Information technologies based on chaotic dynamics for transmission, processing, storage and protection of informatics. *Information Technologies, Rensit*. 2018;**10**(2):279-312
- [12] Behnia S, Akhshani A, Mahmodi H. Applications of tripled chaotic maps in cryptography. *Chaos, Solitons & Fractals*. 2009;**40**(1):505-519
- [13] Jaryal S, Marwaha C. Comparative analysis of various image encryption techniques. *International Journal of Computational Intelligence Research*. 2017;**13**(2):273-286
- [14] Li C, Luo G, Qim K. An image encryption scheme based on chaotic map. *Nonlinear Dynamics*. 2017;**87**(3): 127-136
- [15] Xu L, Li Z, Li J. A novel bit-level image encryption based on chaotic map. *Optics and Lasers in Engineering*. 2016;**78**(3):17-25
- [16] Kalso A, Ghebleh M. A robust chaotic algorithm for digital image. *Communications in Nonlinear Science and Numerical Simulation*. 2014;**19**(6): 1898-1908
- [17] Sidorenko AS, Shishko MS. Image encryption based on chaotic dynamics

with elements of a genetic algorithm. Informatics. 2018;**29**(1):95-100

[18] Kaur M, Singh D, Sun K. Color image encryption using non-dominated sorting generated algorithm with local chaotic search based 5D chaotic map. Future Generation Computer Systems. 2020; **107**(6):333-350

[19] Wong K-W, Kwok BS-H, Yuen C-H. An efficient diffusion approach for chaos-based image encryption. Chaos, Solitons and Fractals. 2009;**41**:2652-2663

[20] Wang W, Si M, Pang Y, Ran P. A encryption algorithm based on combined chaos in body area networks. Computers and Electrical Engineering. 2018;**65**: 282-291

[21] Dong C. Color image encryption using one-time keys and coupled chaotic systems. Signal Processing: Image Communication. 2014;**29**:628-640

[22] Liu H, Kadir A, Gong P. A fast color image encryption scheme using one-time S-boxes based on complex chaotic system and random noise. Optics Communication. 2015;**338**:340-347

[23] Volos C, Maaita JO, Vaidyanathan S. A: Novel four-dimensional hyperchaotic four-wing system with a saddle-focus equilibrium. IEEE Transactions on Circuits and Systems II: Express Briefs. 2017;**64**(3):339-343

[24] Rossler OE. An equation for continuous chaos. Physics Letters. 1976; **57**(5):397-398

[25] Sidorenko AV, Shakinko IV, Sidorenko YV. Algorithm for encrypting images using two-dimensional chaotic mappings. System Analysis and Applied Informatics. 2016;**2**:44-49

[26] Grobman D. Homeomorphism of systems of differential equations. Reports of the Academy of Sciences of the USSR. 1959;**128**(5):880-881

[27] Shashihin VN, Turulin AV, Budnik CV. Image encryption algorithm based on controlled chaotic maps. Computing, Telecommunications and Control. 2021;**14**(1):7-21

[28] Sprott JC, Rowlands G. Improved correlation dimension calculation. International Journal of Bifurcation and Chaos. 2001;**11**:1861-1880

[29] Historical information about the city. In: Economic Forum in St. Petersburg. St. Petersburg, Russia: Roscongress; 2024. Available from: <https://forumspb.com/o-sankt-peterburge/history>

Perspective Chapter: Application of Evolutionary Computing for Generating Encryption Algorithms with AI-Based Code Generation

Jonathan Blackledge, Robert Kingstone and Brad Midgley

Abstract

Symbolic regression, a data-driven approach that models mathematical expressions to fit given data, has emerged as a powerful tool across diverse fields, including cryptography. Its capability to uncover underlying relationships and generate concise, interpretable equations establishes it as an innovative method for designing encryption algorithms. By leveraging symbolic regression, it is possible to develop encryption schemes that are not only secure but also efficient and adaptable to evolving computational challenges. This approach facilitates the automated exploration of novel algorithmic structures, reveals hidden efficiencies, and enhances cryptographic robustness against emerging threats. Serving as a bridge between machine learning and mathematical innovation, symbolic regression opens new frontiers in the realm of secure communication. This study explores the use of symbolic regression to construct a potentially unlimited number of encryption algorithms. The process involves modeling and simulating real random number streams obtained from various sources. In this work, random number streams are sourced from Random.org, while the symbolic regression process is implemented using the TuringBot application. The encryption algorithms (based on nonlinear functions) derived from this method are then translated into codes (specifically Python computer programs) using an AI-based code generator. For this purpose, the OpenAI ChatGPT system is employed to generate code in the presentation of an example case study—the generation of strong passwords from weak but memorable passwords.

Keywords: cryptography, data encryption, symbolic regression, evolutionary computing, AI-based code generation

1. Introduction

The field of modern cryptography plays a pivotal role in safeguarding the confidentiality, integrity, and authenticity of digital information. As cyberthreats grow in complexity, traditional encryption methods face escalating challenges, necessitating the development of adaptive and innovative solutions.

A promising avenue of research combines evolutionary computing and symbolic regression to design the personalised encryption algorithms. These advanced computational techniques enable the creation of secure, efficient, and customisable encryption systems tailored to specific user requirements and computational environments.

1.1 The need for personalised encryption

Standard encryption algorithms, such as advanced encryption standard (AES) [1] and Rivest-Shamir-Adleman (RSA) [2], have long been the foundation of data security. However, these algorithms adopt a universal approach, applying the same cryptographic techniques across diverse systems and applications. As security demands become increasingly sophisticated and varied, there is a growing need for personalised encryption solutions that consider unique system constraints, computational capabilities, and individual privacy preferences.

Personalised encryption transcends traditional approaches by integrating contextual factors such as data type, user behaviour, resource limitations (e.g. processing power, hardware capabilities), and evolving threat landscapes. Conventional cryptographic techniques are typically static and lack the adaptability required for such dynamic scenarios. In this context, evolutionary computing and symbolic regression can excel by exploring vast solution spaces to meet the specific requirements.

1.2 Evolutionary computing and its role

Evolutionary computing (EC) refers to a family of optimisation techniques inspired by the principles of natural selection. These include genetic algorithms, genetic programming, and evolutionary strategies [3]. EC operates by iteratively improving candidate solutions using a fitness function that evaluates their performance. For encryption algorithm development, EC can generate cryptographic functions optimised for criteria such as speed, memory efficiency, and resistance to specific attacks.

The EC process involves creating a population of potential encryption solutions, encoding them as modifiable representations. Operations like crossover (combining traits from parent solutions), mutation (introducing random variations), and selection (retaining the most fit solutions) enable the population to evolve iteratively. Over successive generations, increasingly optimal encryption algorithms emerge.

A major advantage of EC is its ability to navigate extensive solution spaces, often uncovering unconventional yet effective encryption schemes. Defining fitness functions that prioritise security, efficiency, or other criteria, EC can produce bespoke algorithms tailored to specific use cases and constraints.

1.3 Symbolic regression: Bridging the gap

Symbolic regression, a subfield of evolutionary computing, focuses on identifying mathematical models or symbolic expressions that best represent a dataset. Unlike conventional regression methods that fit parameters to predefined models, symbolic regression explores the model space itself, producing novel and interpretable expressions [4].

In the context of encryption, symbolic regression enables the discovery of mathematical functions governing cryptographic processes such as key generation, transformation, and message scrambling. These functions may encompass a wide range of

operations, from standard techniques like modular arithmetic to more intricate nonlinear transformations. For instance, symbolic regression can identify functions that optimise encryption and decryption procedures while introducing unpredictability, complicating reverse engineering by attackers.

Symbolic regression also offers transparency, as the resulting encryption algorithms are interpretable. This facilitates debugging, auditing, and improving the security of these systems. Moreover, it enhances the customisation potential of encryption algorithms, enabling them to adapt to specific user or application requirements while maintaining high levels of security and computational efficiency.

1.4 Personalised encryption with evolutionary computing and symbolic regression

The integration of evolutionary computing and symbolic regression provides a robust framework for designing personalised encryption algorithms. By harnessing the adaptability of EC and the modelling capabilities of symbolic regression, this approach enables the creation of encryption systems tailored to diverse requirements. Key applications include:

1. *Customisation for data types and usage contexts:* Different data formats (text, images, and video) and applications (cloud storage, mobile devices, and IoT) demand distinct security measures. Evolutionary computing can optimise encryption algorithms for these diverse scenarios, while symbolic regression ensures that the underlying models are mathematically appropriate for specific data characteristics.
2. *Resource-conscious optimisation:* Computational environments vary significantly, from resource-constrained IoT devices to high-performance cloud servers. Evolutionary algorithms can evolve encryption schemes that balance security and efficiency according to the system's computational capabilities. Symbolic regression can derive compact, efficient functions suitable for low-power environments.
3. *Continuous security evolution:* Static algorithms may become vulnerable to advanced cryptographic attacks. Evolutionary computing enables ongoing adaptation, evolving encryption algorithms to counter emerging threats. Symbolic regression contributes novel mathematical structures that enhance resilience to attacks such as brute force or side-channel analysis.
4. *Personalised security preferences:* Users and organisations often prioritise different aspects of encryption, such as speed versus security. Evolutionary computing and symbolic regression allow algorithms to be fine-tuned to align with these preferences, delivering solutions that meet specific security and performance objectives.

1.5 Challenges and future directions

Despite its potential, the application of evolutionary computing and symbolic regression in cryptography presents challenges. One significant issue is the risk of overfitting, where an evolved encryption algorithm performs well on training data but

fails under real-world conditions or against novel attack strategies. Robust testing and validation procedures are essential to address this concern.

Additionally, evolutionary algorithms can be computationally intensive, especially when exploring large and complex solution spaces. Advances in parallel computing and optimisation techniques are crucial to improving the practicality of evolving encryption algorithms for real-world applications.

Ensuring the long-term security of personalised encryption algorithms is another challenge. Cryptographic threats evolve rapidly, necessitating dynamic solutions. The integration of evolutionary techniques provides a pathway for continuous algorithm enhancement, maintaining robustness against emerging attacks.

1.6 Discussion

The use of evolutionary computing and symbolic regression to develop personalised encryption algorithms represents a transformative approach to cryptography. These methods enable the creation of adaptive, efficient, and user-specific encryption systems, addressing the limitations of traditional cryptographic techniques. As the need for secure, scalable, and customisable encryption solutions grows, evolutionary computing and symbolic regression are likely to play an increasingly important role in shaping the future of cryptographic innovation.

In this work, we demonstrate a practical methodology for developing personalised encryption algorithms using the TuringBot application. These algorithms can subsequently be implemented in software with the assistance of tools such as OpenAI's ChatGPT for code generation. This approach provides a flexible toolkit for users to create customised encryption solutions, generalising principles from post-quantum cryptography. In this context, we present a case study illustrating how an evolutionary algorithm can transform a memorable but weak password into a strong password—an arbitrarily long string.

2. Encryption systems

Consider the following schematic representation of a symmetric encryption system:

$$C = E_K(P) \Rightarrow P = D_K(C) \quad (1)$$

where $\Rightarrow$ implies the associated process of decryption.

Here, P represents the plaintext, which is processed using an encryption algorithm denoted by E to generate the ciphertext C . This process critically depends on a key, K . Coupled with a corresponding decryption algorithm, D , the plaintext can be recovered by the recipient of the ciphertext, provided they possess the same key, K .

The symbol $\Rightarrow$ conventionally denotes the transmission of the ciphertext over an insecure channel or network. However, it could also represent the uploading of encrypted data to the cloud, for instance. In either scenario, the creator and sender of the ciphertext (Alice) and its recipient (Bob) must know two essential pieces of information:

1. The encryption/decryption algorithm, and
2. The key used to encrypt/decrypt the data with that algorithm.

In principle, and with respect to symmetric encryption, both the encryption/decryption algorithm and the key(s) must be exchanged between Alice and Bob before secure communication can occur. However, the conventional paradigm assumes that Alice and Bob already have prior access to the algorithm(s) and that such algorithms are well-established and likely publicly scrutinised. This assumption aligns with the *Kerckhoff–Shannon Principle*, which states [5]:

‘A cryptosystem should be secure even if everything about the system, except the key, is public knowledge’

This idea was succinctly phrased by Claude Shannon as [6]:

‘The enemy knows the system.’

Under this principle, Alice and Bob only need to exchange the key, making the key exchange the most critical issue in the implementation of all symmetric encryption systems.

In a report on the secret history of Hut 8—the Naval Section at Bletchley Park (Station X) in the UK from 1941 to 1945—it is stated [7]:

‘The continuity of breaking Enigma ciphers was undoubtedly an essential factor in our success, and it does appear to be true to say that if a key has been broken regularly for a long time in the past, it is likely to continue to be broken in the future, provided that no major change in the method of encryption takes place.’

This statement refers to the famous Enigma encryptor used by German armed forces during the Second World War. It highlights a fundamental security issue: potential weaknesses in applying the Kerckhoff-Shannon principle. At the same time, it raises a practical concern. While it may be assumed that Alice and Bob have the technical ability to generate and share a key (given appropriate training in creating strong keys and access to a key-exchange algorithm), they will not necessarily possess the expertise to develop a robust, key-dependent (or otherwise) encryption algorithm. Historically, such algorithms have been created by specialists in cybersecurity.

However, with the recent advancements in artificial intelligence (AI), this landscape is changing. AI has the potential to democratise the development of encryption algorithms, making it possible for non-experts to create secure, customised cryptographic systems. This shift introduces new opportunities and challenges, and it is central to the work discussed here.

3. Encryption algorithms

An encryption algorithm must be designed to have a number of properties that provide cryptographic strength which, on a generic basis, are taken to include randomness, unpredictability, and complexity. More specifically, these properties include ensuring that the cypher is uniformly distributed (with no bias towards any number in the cypher stream, thereby providing a ‘maximum entropy cypher’), a high positive Lyapunov exponent (ensuring that the stream becomes random within a few iterations), a high cycle length, and good diffusing properties so that different keys (involving a change of a single element) produce different cyphers in which any and all elements of the cypher stream have an equal likelihood of changing their value [8].

Encryption algorithms typically rely on the design of a pseudo random number generator (PRNG) to produce a cypher. There are many approaches to doing this, but one of the key elements of the work report here is that a PRNG can be generated using symbolic regression and then coded using an AI code generating app (to be discussed later). However, once the PRNG has been developed, there is the issue of how it is then used to encrypt the plaintext. In other words, with reference to the encryption/decryption representation given by Eq. (1), what process should be applied to design the algorithms denoted by E and D .

Consider the case when a PRNG has output a one-dimensional numerical array of random numbers or ‘noise’ denoted by $n_i \equiv (n_1, n_2, \dots, n_N)$, where N is the length of the array (typically equal to the length of the plaintext array for which the encryption process is required). The array is typically taken to consist of integer elements, but floating-point elements might also be considered depending of the specific application.

Let p_i denote the plaintext array and c_i denote the ciphertext array. How should we combine n_i and p_i to construct c_i (encrypt) and then recover p_i from c_i (decrypt)? There are a number of possible answers to this question. Assuming that p_i and n_i are $\geq 0 \forall i$, these possibilities include the following:

i. **Additive:** $c_i = p_i + n_i \Rightarrow p_i = c_i - n_i$.

ii. **Multiplicative:** $c_i = p_i n_i \Rightarrow p_i = c_i / n_i, n_i \neq 0 \forall i$.

iii. **Exponentiation:** $c_i = p_i^{n_i} \Rightarrow p_i = c_i^{1/n_i}, n_i \neq 0 \forall i$

iv. **Phase-only encryption:** $c_i = p_i \exp(jn_i) \Rightarrow p_i = c_i \exp(-jn_i)$
where $j = \sqrt{-1}$

v. **Convolutional coding:**

$$c_i = n_i \otimes p_i \equiv \sum_{k=1}^N n_{i-k} p_k \Rightarrow p_i = m_i \otimes c_i$$

where

$$m_i \otimes n_i = \delta_i = \begin{cases} 1, & i = 1; \\ 0, & i > 1. \end{cases}$$

is the Kronecker delta. The problem is then to find the function m_i to obtain the decrypt. In this context, decryption becomes an inverse problem—the deconvolution problem.

If the arrays are taken to be composed of decimal integers (which might require converting the elements of a cypher composed of floating-point numbers to integers), then they can be written in binary form. The process of adding the cypher and plaintext arrays then becomes equivalent to an XOR process. In this case, if we construct binary strings of length $M \neq N$, denoted by $p \in \{0, 1\}^M$, $n \in \{0, 1\}^M$ and $c \in \{0, 1\}^M$, then

$$c = n \oplus p \Rightarrow p = n \oplus c \quad (= n \oplus n \oplus p = p)$$

where $\oplus$ denotes the XOR (modulo-2 addition) operator.

In terms of a conventional plaintext composed of a string of alphanumeric characters and other symbols associated with the 7-bit or 8-bit ASCII codes, the decimal integers are typically determined by these codes. To convert a numerical cypher back into ASCII symbols, the remainder function is used. Thus, in the case of an additive cypher using 7-bit ASCII code, we consider

$$c_i = (p_i + n_i) \bmod(128) \Rightarrow p_i = (c_i - n_i) \bmod(128)$$

where the mod function is the modulo operator which returns the remainder (in this case, the remainder of $p_i + n_i$ or $c_i - n_i$ divided by 128). This encryption method is generally known as the 'Vernam cypher'.

The ciphertext does not necessarily need to be transformed into ASCII characters before transmission and/or storage. Furthermore, if the ciphertext consists of floating-point values, such a transformation is not possible. This is the case when encryption of floating-point digital signals is required such as audio files, for example, or any other digital signal where p_i is taken to be an array whose elements are of type float (with a specified precision). In this case, decryption must be undertaken to the same floating-point precision as the process of encryption.

Convolution coding and decoding can be accomplished by processing the discrete spectra of the array through application of a discrete Fourier transform (implemented using a fast Fourier transform) and the convolution theorem. Thus, consider the case where we encrypt the spectrum of the plaintext as follows:

$$C_i = \frac{N_i}{|N_i|^2} P_i, \quad |N_i| > 0 \forall i$$

where C_i , N_i , and P_i are the (complex) spectra of c_i , n_i , and p_i , respectively.

Decryption is then easily achieved by simply multiplying the spectrum of the ciphertext with the complex conjugate of the spectrum of the cypher, i.e.

$$N_i^* C_i = \frac{N_i^* N_i}{|N_i|^2} P_i = P_i$$

The encryption methods presented in this section are a limited selection of numerous other approaches that can be considered. Moreover, they can be extended to consider the encryption/decryption of multidimensional arrays. However, in some cases, it is important to condition the algorithms in such a way to avoid numerical overflow, that is the error that occurs when the value of a number exceeds the upper or lower limit allowed by the chosen data type, especially in the case of exponentiation, for example.

Each of the methods (i) - (v) given above requires a PRNG to compute the cypher n_i . The PRNG maybe an entirely separate process and/or an integral part of the algorithm. This is discussed in the following section.

4. Conventional pseudo-random number generation

We are interested in constructing an algorithm *Algol* that produces a key-dependent cypher, that is

$$Algol(Key) \rightarrow n_i$$

where $\rightarrow$ denotes that the key-dependent algorithm ‘generates’ the cypher n_i .

Many cryptographic systems are based on a series of so-called round transformations, which are relatively simple and produce pseudo random number streams using pseudo random number generator (PRNG).

A PRNG is a function or an algorithm that produces a sequence of numbers from a relatively short seed (initial conditions: password, plaintext) based on some iteration function. The earliest such algorithms were based on a combination of modular arithmetic coupled with a prime number. Modular arithmetic is a system of arithmetic for integers, which considers the remainder. In modular arithmetic, numbers ‘wrap around’ upon reaching a given fixed quantity (this given quantity is known as the modulus) to leave a remainder. Modular-based functions tend to behave more erratically than conventional functions, especially when they are ‘coupled’ with a prime number.

The reason for considering a prime number in combination with modular arithmetic is because a prime number is divisible only itself or 1. Thus, application of a prime number in a iterative process involving modular arithmetic tends to produce an output consisting of random numbers. However, all such algorithms have an intrinsic cycle length which is when the random numbers start to repeat after a certain number of iterations. Thus, such algorithms have to be ‘tuned’ to provide random number streams that have optimal properties for the process of encrypting data.

An early algorithm of this type dates back to the 1950s (e.g. [9, 10]) and is compounded in the following iteration known as the linear congruential method:

$$n_{i+1} = (an_i + b) \bmod(P), \quad i = 1, 2, \dots, N - 1$$

where P is a prime number, and a and b are constants (which can include the case when $b = 0$). The cycle length of this iteration is determined by the value P , and thus, a large prime number is desirable. The ‘quality’ of this iterator, in terms of the random number sequence that it generates, depends on the specific values of a , b , and P that are chosen. Choosing such parameters and testing the result(s) is part of the ‘tuning process’, a process that applies to all iterators and is discussed further later on. In principle, this includes the range of values of the key n_1 —an integer number used to initiate the iteration.

The relative simplicity and numerical efficiency associated with such a linear iteration naturally led to further developments based on the application of prime numbers coupled with modular arithmetic. The reasons for this are that early micro-processors for which encryption schemes were initially designed could operate very effectively as integer processors. Consequently, the early developments of PRNGs were based on using this approach, which led to notable PRNGs for cryptography such as the following:

- The Blum-Blum-Shub generator [11]:

$$n_{i+1} = n_i^2 \bmod(PQ)$$

where P and Q are prime numbers

- The Blum-Micali generator [12]:

$$n_{i+1} = P^{n_i} \bmod(Q)$$

- The RSA (Rivest, Shamir, and Adleman) generator [13]:

$$n_{i+1} = n_i^e \bmod(PQ)$$

where e is a relative prime of $(P-1)(Q-1)$, i.e. e and $(P-1)(Q-1)$ have no common factors except for 1.

There are numerous other encryptions methods based on the use of modular arithmetic with or without the use of prime numbers, for example the RC4 and AES algorithms and the Diffie-Helman method for key exchange [14]. However, the RSA generator has a special place in the history of cybersecurity. This is because it is the basis for one of the most commonly used asymmetric (public-private key) encryption algorithms.

The generator has a symmetry associated with the encryption and decryption processes which is compounded in the following result [13]:

$$c_i = p_i^e \bmod(PQ) \rightarrow p_i = c_i^d \bmod(PQ)$$

where

$$de = \bmod[(P-1)(Q-1)] \quad (2)$$

In this case, the prime numbers P and Q and the co-prime $e < PQ$ need be distributed to Alice and Bob in such a way that they are unique to Alice and Bob on the condition that d exists through a known solution to Eq. (2). In this context, e, PQ represents a public key, and d is a private key that is unique to Alice or Bob.

To implement such a scheme requires an appropriate infrastructure to be established by a trusted third party who is 'business' is to distribute values of e, PQ and d to its clients—a public key infrastructure (PKI). Note, however, that in order to solve for the decryption key d , P , and Q need to be known. Thus, the security of this approach is based on the computational difficulty of finding P and Q from their product PQ (prime number factorisation) which is known (publicly).

Introduced in 1977, the RSA algorithm is a widely used cryptographic system based on the mathematical challenge of prime factorisation. RSA relies on the generation of two large prime numbers, whose product forms a modulus used in both the public and private keys. The security of RSA hinges on the fact that, for classical computers, factorising a large composite number into its prime constituents is computationally infeasible within a reasonable timeframe. The computational time required to factorise PQ grows exponential with the size of the number PQ . This intractability ensures that, even if an adversary possesses the public key, they cannot feasibly derive the private key.

5. Post-quantum security

The advent of quantum computing represents a transformative shift in computational capabilities, enabling the efficient resolution of problem classes previously deemed intractable by classical methods. While this progress holds immense promise across various domains, it poses a severe threat to contemporary cryptographic systems, particularly those founded on the computational difficulty of prime factorisation. Among these, the RSA algorithm—a cornerstone of public-key cryptography—is especially vulnerable to quantum attacks. The field of post-quantum

security has thus emerged as a critical area of research, focusing on the development of cryptographic methods that are resistant to the capabilities of quantum computing.

RSA's security is fundamentally threatened by quantum computing. Introduced in 1994, Shor's algorithm [15] efficiently factors large integers in polynomial time. For a quantum computer, the computational time required to factorise $N = PQ$ grows only logarithmically with the size of the number. This renders RSA effectively obsolete in a post-quantum world, where sufficiently advanced quantum computers could decrypt RSA-secured communications with minimal effort.

5.1 Quantum computing and Shor's algorithm

Shor's algorithm can exploit the principles of quantum mechanics, including superposition and entanglement, to address the factorisation problem exponentially faster than any classical algorithm. The algorithm efficiently identifies the periodicity of mathematical functions, translating directly to factorisation in the context of RSA. For instance, given a large composite number N , Shor's algorithm can determine its prime factors in $O\left[(\log N)^3\right]$ time on a quantum computer, compared to the exponential time required by classical approaches [16].

The implications of this breakthrough are profound. Cryptographic systems relying on RSA for secure communication—including Internet protocols such as HTTPS and digital signature schemes—become vulnerable to quantum adversaries. Current RSA key sizes, typically ranging from 2048 to 4096 bits, are designed to resist classical attacks but provide no effective defence against quantum attacks.

5.2 Securing the future

Post-quantum cryptography (PQC) aims to develop cryptographic systems that are secure against both classical and quantum computing threats. Unlike RSA, which is based on the hardness of factorisation, PQC schemes are founded on mathematical problems believed to be resistant to quantum attacks.

Prominent candidates include:

- i. *Lattice-Based cryptography*: Algorithms such as learning with errors (LWE) and Ring-LWE rely on the difficulty of solving lattice problems, which remain computationally hard for both classical and quantum computers. These algorithms are valued for their efficiency and broad applicability [17].
- ii. *Code-based cryptography*: Systems like the McEliece cryptosystem leverage the hardness of decoding random linear codes. Although they require larger key sizes, these schemes have withstood decades of cryptanalysis, demonstrating strong resilience [18].
- iii. *Hash-based cryptography*: Hash-based signature schemes, such as the eXtended Merkle Signature Scheme (XMSS), rely on the security of cryptographic hash functions, which are regarded as quantum-resistant [19].
- iv. *Multivariate polynomial cryptography*: These schemes depend on the difficulty of solving systems of nonlinear equations over finite fields, another problem area considered resistant to quantum attacks [20].

- v. *Isogeny-based cryptography*: Leveraging the computational difficulty of finding isogenies between elliptic curves, these systems provide compact key sizes and are a promising area of research in post-quantum cryptography [21].

5.3 Transitioning to a post-quantum world

The transition to post-quantum cryptographic systems presents significant challenges. First, cryptographic protocols must be rigorously standardised and tested for security and efficiency. Organisations such as the National Institute of Standards and Technology (NIST) are leading efforts to evaluate and standardise post-quantum algorithms, with several candidates currently in the final stages of evaluation [22].

Second, practical deployment necessitates compatibility with existing systems. This involves updating cryptographic libraries and ensuring seamless integration with diverse applications and hardware. Hybrid approaches, where post-quantum algorithms are deployed alongside classical ones, are anticipated to facilitate compatibility and ease the transition.

Finally, the computational overheads associated with some post-quantum algorithms pose performance challenges. Many candidate algorithms, while secure, have larger key sizes and higher computational requirements, which need to be optimised for widespread adoption.

5.4 The imperative for preparedness

The rise of quantum computing demands a proactive approach to safeguarding digital security. RSA, once a cornerstone of secure communications, is rendered vulnerable by the advent of quantum capabilities, particularly through Shor's algorithm.

Post-quantum cryptography offers a viable path forward, leveraging mathematical problems believed to be resistant to quantum attacks. Collaborative efforts such as *Open Quantum Safe*, established in 2016, exemplify the drive to develop and implement quantum-resistant tools [23].

Preparing for the post-quantum era is not merely a theoretical exercise but a practical necessity to ensure the integrity and confidentiality of digital communications for future generations.

6. Cryptography using chaos

Traditional cryptographic techniques often rely on mathematical algorithms and computational complexity. The integration of chaos theory into cryptography introduces a novel approach that leverages the inherent unpredictability and sensitivity to initial conditions of chaotic systems—commonly known as the 'butterfly effect' [24, 25].

Chaos-based cryptography exploits the complex, aperiodic, and deterministic behaviour of chaotic systems to generate encryption keys, design secure communication protocols, and enhance data security. These systems provide a rich source of random-like sequences that are highly sensitive to initial parameters, making them ideal for cryptographic applications.

This approach offers several potential advantages, including resistance to traditional cryptanalytic methods, high key sensitivity, and efficient implementation in digital systems. Chaos-based cryptography is particularly relevant for securing data in

resource-constrained environments, such as the Internet of Things (IoT), where lightweight and robust security solutions are essential [26].

While the demand for secure and efficient cryptographic solutions grows, chaos theory offers a promising alternative to conventional methods, pushing the boundaries of secure communication systems. However, computer implementations of chaotic systems exhibit behaviour that differs significantly from modular arithmetic-based systems. Such implementations often require high-precision floating-point arithmetic, making them computationally demanding. This challenge limited their adoption until the advent of floating-point microprocessors in the 1990s.

Chaos-based systems can face some significant limitations, such as short cycle lengths and trivial pattern generation, which can render them unsuitable for robust data encryption. Additionally, while their output may exhibit chaotic properties, the distributions are often non-uniform, a statistical inconsistency with the requirements of cryptographic strength. As a result, chaos-based cyphers must be rigorously evaluated, modified, and, if necessary, post-processed in order to ensure their suitability for encryption.

6.1 Chaotic iterators for cryptography

A simple chaotic iterator can be described by:

$$n_{i+1} = f(n_i), \quad n_1 \equiv \text{key}, \quad i = 1, 2, \dots, N \quad (3)$$

where N is the desired cypher length, and f is a strictly nonlinear function that may depend on one or more parameters. The key may typically be a floating-point number between 0 and 1 (non-inclusively).

Eq. (3) is an example of an iterative function system (IFS), which typically outputs floating-point numbers rather than integers. It represents a generic chaotic map that can be extended to coupled systems (e.g. a Lorenz system [24]). This study focuses on such systems and their potential applications in cryptography.

To be used as a cypher generator, its cryptographic strength must be assessed. The iteration must generate a sequence of floating-point numbers which include the following properties:

- *High sensitivity to initial conditions:* The sequence must be ultra-sensitive to the key value (the initial condition), which, for generality, should be treated as a floating-point number.
- *Uniform distribution:* The generated array must be strictly uniformly distributed.
- *Long cycle length:* The sequence should have a sufficiently long cycle length to avoid repetition, at least within the string length of a plaintext.
- *Chaotic behaviour:* The system should have a positive and relatively large Lyapunov exponent, indicating rapid transition to fully chaotic behaviour for relatively few iterations.

6.2 Designing nonlinear functions

Identifying suitable nonlinear functions f in Eq. (3) can follow two main approaches:

- *Using existing nonlinear models*: Draw on functions developed from research in nonlinear dynamics, such as the ‘Matthews map’, which is a modified version of the Verhulst process [27].
- *Inventing functions*: Develop novel functions tailored for cryptographic purposes, where physical modelling is secondary to ensuring robustness in encryption.

In either case, the cryptographic strength of the resulting system must be rigorously evaluated. This involves addressing the following key issues:

- *Structural stability*: Ensure that the iterator is structurally stable, meaning that it exhibits consistent cycle lengths across all initial conditions. Many pseudo-chaotic systems lack this property and are therefore unsuitable.
- *Unpredictability*: Establish criteria that guarantee the computational unpredictability of the chaotic output for any initial condition.
- *One-way compatibility*: Demonstrate that the mapping characterised by the function f in Eq. (3) is non-invertible, ensuring that the output cannot be reversed to deduce the key.
- *Asymmetric cryptography*: Investigate whether chaotic systems can support trapdoor functions, which are fundamental to asymmetric cryptography. Currently, chaos theory is primarily applicable to symmetric cryptography, as no trapdoor transformations have been identified.

6.3 Challenges and opportunities in chaos-based cryptography

Chaos-based encryption faces several challenges compared to conventional methods:

- *Computational inefficiency*: Chaotic systems require floating-point arithmetic, which is more computationally demanding than modular arithmetic.
- *Short cycle lengths*: Chaotic systems often exhibit shorter cycle lengths compared to conventional cyphers.

However, chaos-based systems offer a unique advantage: the potential for generating an unlimited number of algorithms [26]. This capability allows for multi-algorithmic solutions, where algorithms can be publicly shared in accordance with the Kerckhoff-Shannon principle, as required [28]. In this context, chaos-based methods are ‘open-quantum safe’, as they do not rely on prime numbers or modular arithmetic for security.

Most chaotic maps require pre- and post-processing to achieve cryptographic strength [29]. The continuous development of such maps, however, presents practical challenges and risks of human error. To address this, recent advances in machine learning (ML) have enabled the automation of algorithm generation. Evolutionary computing (EC), in particular, shows promise for automating the creation and evaluation of chaos-based cryptographic algorithms. This study explores the application of EC combined with real-world noise sources to develop robust and innovative encryption systems [30].

7. Evolutionary computing

Evolutionary computing, a subset of artificial intelligence (AI), is inspired by the principles of natural evolution and genetic selection. It employs algorithms to simulate the processes of mutation, crossover, and selection to solve complex optimisation and search problems. Among its varied applications, symbolic regression stands out as a powerful technique for discovering mathematical models and relationships directly from raw data. Unlike traditional regression methods, which assume a predefined model structure, symbolic regression searches for both the structure and parameters of the model, making it a flexible and adaptable tool for addressing real-world problems.

In this section, we explore the principles of evolutionary computing, the mechanics of symbolic regression, their integration, and applications across diverse fields. We also discuss the advantages, limitations, and future directions of this approach in a rapidly evolving computational landscape. First, we consider a brief history of the subject.

7.1 A brief history of evolutionary computing

The origins of evolutionary computing can be traced back to the mid-twentieth century when scientists began exploring analogies between natural evolution and problem-solving. In the 1950s, pioneers such as Alan Turing speculated about ‘learning machines’ that could adapt and evolve over time [31]. However, it was John Holland’s work in the 1960s that laid the formal foundation for evolutionary algorithms. At the University of Michigan, Holland developed genetic algorithms (GAs), which use the concepts of selection, crossover, and mutation to optimise solutions [32].

7.1.1 Emergence of evolutionary programming and strategies

Parallel to Holland’s work, Lawrence J. Fogel introduced evolutionary programming in 1962, focusing on evolving finite-state machines for prediction and control tasks [33]. Around the same time, Ingo Rechenberg and Hans-Paul Schwefel in Germany developed Evolution Strategies (ES) to address engineering optimisation problems [34]. Unlike GAs, which often used binary encoding, ES emphasised real-valued parameters and self-adaptive mutation rates, making them suitable for continuous optimisation tasks.

7.1.2 Integration and growth: 1970s and 1980s

By the 1970s, evolutionary computing had gained recognition as a distinct field. The publication of Holland’s seminal book, *Adaptation in Natural and Artificial Systems* [32], introduced genetic algorithms to a broader audience and established a formal framework for their implementation. The 1980s witnessed the rise of genetic programming (GP), popularised by John Koza, which extended GAs to evolve computer programs. GP found applications in symbolic regression, automated software design, and beyond [35].

7.1.3 Modern era: Computational power and hybrid approaches

The 1990s and 2000s marked a turning point driven by advances in computational power and high-performance computing systems. Evolutionary algorithms were integrated with techniques such as neural networks, fuzzy logic, and particle swarm

optimisation, giving rise to hybrid systems capable of addressing complex problems [36]. Developments such as the covariance matrix adaptation evolution strategy (CMA-ES) [37] further advanced continuous optimisation, and open-source frameworks like Distributed Evolutionary Algorithms in Python (DEAP) democratised access to evolutionary tools.

7.1.4 Evolutionary computing in the twenty-first century

Today, evolutionary computing is integral to AI and optimisation science [38]. Its applications span autonomous systems, industrial design, financial modelling, and drug discovery. Advances in quantum computing and symbolic regression have opened new frontiers, enabling evolutionary algorithms to tackle previously intractable problems [39]. The field reflects continuous innovation and interdisciplinary collaboration, promising transformative solutions for future challenges, typically in the form of Python ‘solutions’ [40].

7.2 Principles of evolutionary computing

Evolutionary computing derives its core concepts from Darwinian evolution, particularly ‘survival of the fittest.’ It uses a population-based approach, where potential solutions are represented as individuals in a population. These individuals evolve over successive generations, guided by an objective function that evaluates their performance.

Key components of evolutionary computing include [41, 42]:

- i. *Representation*: Solutions are encoded as chromosomes or genotypes, typically as strings or trees.
- ii. *Fitness function*: This function evaluates how well a solution performs relative to the given problem.
- iii. *Genetic operators*:
 - *Mutation*: Introduces random changes to solutions to maintain genetic diversity;
 - *Crossover*: Combines parts of two parent solutions to produce offspring;
 - *Selection*: Chooses individuals based on their fitness to propagate to the next generation.
- iv. *Termination criteria*: The algorithm stops when a satisfactory solution is found or a predefined computational limit is reached.

7.3 Symbolic regression

Symbolic regression is a data-driven approach to model discovery that identifies mathematical expressions describing relationships between input and output variables. Unlike standard regression methods, symbolic regression does not require the

user to assume a specific functional form. Instead, it searches over a space of potential models to find the best fit for the data [43, 44]. Key aspects include:

- i. *Representation of models*: Models are represented as expression trees, where nodes correspond to mathematical operators (e.g. $+$, $-$, $\times$, $\div$) and functions (e.g. $\sin$, $\cos$, $\exp$, $\log$), while leaves correspond to variables or constants.
- ii. *Search space*: The search space includes all possible mathematical functions, expressions, and combinations thereof within a specified complexity limit.
- iii. *Fitness evaluation*: Fitness is based on criteria such as accuracy (e.g. mean squared error) and simplicity (e.g. number of nodes in the expression tree).
- iv. *Parsimony*: Balancing accuracy and simplicity prevents overfitting.

7.4 Integration of evolutionary computing and symbolic regression

Evolutionary computing provides a robust framework for symbolic regression. Genetic programming (GP) is particularly well-suited for this task [35]. Key components of GP-based symbolic regression include:

- i. *Initial population*: Randomly generated mathematical expressions form the initial population.
- ii. *Fitness function*: Each expression is evaluated for its ability to model the target data accurately while maintaining simplicity.
- iii. *Genetic operations*: Mutation and crossover evolve the population. For instance, mutation replaces a subtree with a randomly generated expression, while crossover combines subtrees from two parents.
- iv. *Evolutionary process*: For over successive generations, the population converges towards optimal or near-optimal models.

7.5 Applications of symbolic regression in evolutionary computing

Applications of evolutionary computing using symbolic regression include:

- i. *Scientific discovery*: Uncovering fundamental equations in physics, chemistry, and biology.
- ii. *Engineering*: System identification, control system design, and fault diagnosis.
- iii. *Finance and economics*: Modelling financial time series, detecting market patterns, and optimising trading strategies.
- iv. *Health care*: Discovering biomarkers, modelling patient outcomes, and understanding disease dynamics.
- v. *Environmental science*: Modelling climate patterns, predicting pollution levels, and optimising resource usage.

7.6 Advantages, limitations, and future directions

Advantages of evolutionary symbolic regression include flexibility, transparency, automation, and robustness in handling noisy and incomplete data. However, limitations such as computational cost, potential overfitting, and scalability challenges persist. Future directions include:

- i. *Hybrid approaches*: Combining symbolic regression with machine learning techniques.
- ii. *Parallel computing*: Exploiting modern hardware architectures.
- iii. *Domain-specific enhancements*: Incorporating domain knowledge to guide searches.
- iv. *Quantum computing*: Exploring quantum algorithms for scalability.
- v. *Real-time applications*: Developing methods for dynamic systems.

Symbolic regression remains a powerful tool for uncovering interpretable models, driving innovation in science, engineering, and beyond.

8. Overview of current evolutionary computing systems

Evolutionary computing systems have evolved significantly in recent years, leveraging advancements in algorithms, computational power, and integration with other AI techniques. These systems are employed across a range of applications, from optimisation and machine learning to scientific discovery and game development. This overview highlights some of the prominent evolutionary computing systems currently available, categorised by their key features and user cases.

1. *Genetic algorithms (GAs)*. Genetic algorithms are one of the most widely used evolutionary computing systems. Inspired by natural selection, GAs work by evolving a population of candidate solutions to a problem. The key features include:
 - i. Binary or real-valued representation of solutions.
 - ii. Operations like selection, crossover, and mutation to evolve solutions.
 - iii. Fitness functions to guide evolution.
 - iv. Notable implementations include:
 - Distributed Evolutionary Algorithms in Python (DEAP): A Python library for implementing GAs with flexible and customisable operators;
 - GA Toolbox in MATLAB: A comprehensive tool for building and analysing GA models.
 - v. Applications include:
 - Optimisation problems (e.g. scheduling, resource allocation);
 - Engineering design and control systems.

2. *Genetic programming (GP)*. Genetic programming extends GAs to evolve computer programs or mathematical expressions, making it suitable for tasks like symbolic regression and automated software generation. The main features of GP are as follows:

- i. Representation as tree structures.
- ii. Evolution of program syntax and semantics.
- iii. Notable implementations:
 - Eureqa: A symbolic regression tool that uses GP to find mathematical relationships in data;
 - Karoo GP: An open-source GP framework in Python for machine learning and regression tasks;
 - TuringBot: Predicts a target variable as a function of one or more input variables, discovering clear and interpretable mathematical formulas that capture the nuanced relationships in data.
- iv. Applications include:
 - Scientific discovery;
 - Feature engineering in machine learning.

3. *Evolution strategies (ES)*. Evolution strategies focus on optimising real-valued parameters and are particularly effective for continuous optimisation problems. Its key features are as follows:

- i. Emphasis on mutation and selection, with less reliance on crossover.
- ii. Use of self-adaptive mutation rates.
- iii. Notable implementations include:
 - Covariance Matrix Adaptation Evolution Strategy (CMA-ES): A state-of-the-art ES for solving nonlinear, non-convex optimisation problems.
 - PyCMA: A Python implementation of CMA-ES.
- iv. Example applications:
 - Robotics control;
 - Neural network hyperparameter optimisation.

4. *Differential evolution (DE)*. Differential evolution is a population-based optimisation algorithm particularly suited for multi-modal and non-differentiable objective functions. The key features of this system are as follows:

- i. Simple and efficient mutation and recombination strategies.
- ii. Robust handling of nonlinear constraints.
- iii. Notable implementations:
 - SciPy differential evolution module: A well-integrated DE implementation for Python users;
 - DEoptim: An R package for solving global optimisation problems using DE.
- iv. Applications:
 - Multi-objective optimisation;
 - Parameter tuning in machine learning.

5. *Particle swarm optimisation (PSO)*. Inspired by the social behaviour of flocks of birds and schools of fish, PSO is a swarm intelligence technique for solving optimisation problems. Key features include:

- i. Individuals (particles) adjust their positions in the search space based on personal and group experiences.
- ii. Simple and computationally efficient.
- iii. Notable implementations:
 - PySwarm: A Python library for implementing PSO;
 - SwarmOps: A library for various swarm intelligence algorithms, including PSO.
- iv. Applications:
 - Function optimisation;
 - Neural network training.

6. *Ant colony optimisation (ACO)*. ACO is inspired by the foraging behaviour of ants and is effective for combinatorial optimisation problems. Key features include:

- i. Use of pheromone trails to guide the search process.
- ii. Iterative improvement of solutions.
- iii. Notable implementations:
 - ACOPY: A Python library for building ACO models;
 - ACO Toolkit for Java: A framework for developing ACO algorithms.

iv. Applications:

- Travelling salesman problem;
- Network routing.

7. *Multi-objective evolutionary algorithms (MOEAs)*. MOEAs are designed to solve optimisation problems involving multiple conflicting objectives. Some of the key features of MOEA are as follows:

- i. Generation of Pareto-optimal solutions.
- ii. Balancing trade-offs between objectives.
- iii. Notable implementations:

- Non-dominated sorting genetic algorithm (NSGA-II/NSGA-III) which is widely used for multi-objective optimisation;
- Platypus: A Python library for MOEAs, supporting NSGA-II, MOEA/D, and other algorithms.

iv. Applications:

- Engineering design optimisation;
- Economic modelling.

8. *Hybrid evolutionary systems*. Hybrid systems integrate evolutionary algorithms with other computational techniques, such as machine learning and fuzzy logic, to enhance performance. Such systems include the following principal features:

- i. Combination of global search (evolutionary) and local search (gradient-based) methods.
- ii. Integration with neural networks for adaptive learning with the following notable implementations:

- Neuro-Evolution of Augmenting Topologies (NEAT): Combines evolutionary algorithms with neural network topology optimisation;
- TensorFlow with evolutionary algorithms: Custom implementations for evolving neural network weights and structures.

iii. Applications include:

- Deep learning architecture optimisation;
- Dynamic system modelling.

9. *Frameworks and platforms*. Several comprehensive platforms provide tools for implementing various evolutionary algorithms and customising them for specific applications. They include the following:

- i. Distributed Evolutionary Algorithms in Python (DEAP) which supports multiple types of evolutionary algorithms, including GAs, GP, and ES.
- ii. Inspire: A platform focused on industrial optimisation tasks using evolutionary methods.
- iii. Evolutionary Computation in Java (ECJ): A scalable Java-based framework for evolutionary algorithms.

The availability of diverse evolutionary computing systems reflects their versatility and growing importance in solving complex problems across industries. From classical approaches like genetic algorithms and particle swarm optimisation to advanced multi-objective and hybrid systems, evolutionary computing continues to push the boundaries of optimisation and AI. As computational capabilities expand, these systems are poised to address even more sophisticated challenges in science, engineering, and beyond.

9. Code generation using OpenAI ChatGPT

The prototype software provided in this article is generated using ChatGPT. For this reason, a short introduction to this approach is now provided.

9.1 Overview of ChatGPT

OpenAI ChatGPT [45] and other developing systems such as MS Copilot, Google's Gemini, for example, provide a different approach to using existing pieces of code, obtained from a specific source, or some manual-type 'object search' of open-source code or otherwise. Thus, software engineers must become fluent with such systems, whether it be for cybersecurity or otherwise, now and in the future. This is because systems of this type are going to change the face of software engineering, e.g. [46, 47].

What may have, in the past, taken weeks, months or even years to produce an app, working with a team of software engineers, focusing on specific aspects of a project (requiring the associated programme/project management), can now be accomplished with just one software engineer working in conjunction with AI programming pair systems.

OpenAI is a research organisation and AI laboratory focusing on the development and advancement of artificial intelligence (AI) technologies. Established in December 2015, OpenAI's mission revolves around ensuring that artificial general intelligence (AGI) is beneficial to all. They aim to conduct research in a collaborative and open manner, sharing their findings to accelerate progress in AI.

One of the notable projects by OpenAI is generative pre-trained transformer (GPT), an architecture that has significantly impacted natural language processing (NLP) tasks. GPT uses a transformer-based model, which is a type of neural network architecture known for its effectiveness in handling sequential data like text.

ChatGPT, also known as GPT-3.5, is one of the iterations within the GPT series. It is a large-scale language model that excels in understanding and generating human-like text based on the input it receives. Trained on a diverse range of Internet data, ChatGPT can comprehend and generate text in a conversational manner, allowing it to hold discussions, answer questions, compose stories, provide information, and perform various language-based tasks.

The architecture of ChatGPT is based on deep learning, that is a deep neural network composed of multiple layers of attention mechanisms. This enables it to capture complex patterns and relationships within the input text. It is pre-trained on a massive corpus of text from diverse sources, allowing it to learn a wide range of language patterns, semantics, and syntactic structures. In this context, it also performs as an Internet search engine.

ChatGPT's capabilities are a result of its training on vast amounts of data and fine-tuning on specific tasks or domains. Through this process, it can generate contextually relevant and coherent responses, understand context in conversations, and provide accurate information based on its training data. However, while ChatGPT demonstrates remarkable language understanding and generation abilities, it is essential to note that it has limitations. The model generates responses based on patterns in its training data and might produce inaccurate or biased information if the input is misleading or ambiguous. Additionally, it lacks real understanding or consciousness and operates solely based on statistical patterns learned from its training data. Nevertheless, launched on 30 November 2022, by January 2023, ChatGPT had become what was then the fastest-growing consumer software application in history, gaining over 100 million users in a few months.

It is arguable that ChatGPT is the first practically usable AI system to come to market and has set the scene for further future developments by OpenAI and competitor products in the future. These include Google's Gemini announced on 6 December 2023 and is at present positioned as a principal contended to OpenAI ChatGPT Version 4. Developed by Google DeepMind, Gemini is a family of multi-modal large language models based on the application of a deep learning paradigm.

The development of deep learning systems over the past 10 years is the key to the current growth in AI systems. This has been driven by the considerable increase in processing speeds coupled with core memory. One of the reasons for this has been the development of processors for the games industries and the dedicated graphics processing required to provide the quality and diversity of real-time image synthesis.

Shallow learning models typically require feature engineering, wherein domain knowledge is used to select and extract relevant features from the input data. While these algorithms can be interpretable and computationally less intensive, they may struggle with capturing intricate patterns in complex datasets. They are best suited for AI systems that involve routine but specialist and limited data.

Deep learning is a subset of machine learning that employs neural networks with multiple layers (often ranging from tens to hundreds or even thousands of layers). These networks are designed to automatically learn hierarchical representations of data through these multiple layers, where each layer extracts increasingly abstract features from the input data.

Deep learning architectures, such as convolutional neural networks for image recognition, recurrent neural networks for sequence data, and transformers like ChatGPT for natural language processing, have shown remarkable success in various domains. Deep learning models excel at learning representations from raw data, eliminating the need for manual feature engineering and data processing algorithms development. They can handle large volumes of data and learn intricate patterns, leading to state-of-the-art performance in many complex tasks. However, deep learning models often require massive amounts of data for training and can be computationally intensive, which may pose challenges in terms of resource requirements and computational costs.

9.2 Code generation

Computer code generation using ChatGPT, for example, represents a transformative approach to software development, leveraging advanced natural language processing (NLP) capabilities. ChatGPT, based on OpenAI's GPT architecture, is a sophisticated large language model trained on vast amounts of text data, including programming languages and software documentation. Its ability to understand and generate human-like responses allows it to assist developers in creating, debugging, and optimising code. These capabilities include:

- i. *Code writing*: ChatGPT can generate code snippets, functions, and even complete programs in various programming languages based on user prompts or natural language descriptions.
- ii. *Debugging*: It identifies errors in code and suggests fixes, streamlining the debugging process.
- iii. *Unit testing*: Providing tests to check on the I/O, performance, and compatibility of a module.
- iv. *Documentation*: The model can write clear and concise documentation, helping to explain complex code logic or APIs; this includes commenting code after it has been developed by a programmer.
- v. *Learning aid*: ChatGPT serves as a tool for learning programming by providing step-by-step examples and explanations.
- vi. *Optimisation*: It suggests improvements to existing code for better efficiency or readability.
- vii. *Translation*: It will translate from one form of source code to another for a range of programming languages.
- viii. *Object orientation*: Conversion of code based on a modular programming paradigm to provide an object-oriented 'solution'.
- ix. *Formal analysis*: Application of formal analysis to provide fault-tolerant code, for example.

In regard to code generation and/or translation, ChatGPT can generate code in over 20 languages, including Python, C/C++/C#, Java and JavaScript, Ruby, PHP, Swift, TypeScript, Kotlin, Go, R, MATLAB, Rust, HTML/CSS, structured query language (SQL), Shell scripting (Bash), Perl, Dart, Lua, Scala, Lisp, and assembly languages (x86, ARM). However, the complexity and accuracy of the generated code may vary based on the task, language nuances, and specific requirements. Translation is also constrained by the compatibility of available libraries across different programming languages and the equivalence of their functionalities.

In regard to a wide variety of applications, the following are notable:

- i. *Rapid prototyping*: Developers can quickly generate prototype code to test ideas or concepts.

- ii. *Educational use:* Students and beginners use it to learn coding basics or tackle challenging problems.
- iii. *Automation:* Routine coding tasks, such as generating boilerplate code or repetitive structures, can be automated.
- iv. *Collaboration:* ChatGPT acts as an AI partner, offering creative solutions to coding challenges.

While ChatGPT is a powerful tool, its responses are not always flawless and hence have limitations. It might produce syntactically correct but logically flawed code or provide solutions that lack context-specific considerations. Verification by human experts remains essential (at least for the present).

For instance, when translating code that uses multidimensional arrays, differences in how array elements are stored in memory may not always be properly considered. This can lead to inconsistencies occurring in multidimensional array processing between different programming languages as each language usually has a default storage scheme.

Nonetheless, ChatGPT and other emerging AI systems have considerable potential in terms of the future management of software engineering projects in general and the development of specific apps including the translation and upgrading of legacy code.

The integration of ChatGPT with development tools, IDEs, and cloud platforms could further enhance productivity, making it a valuable asset in modern software engineering workflows. Continuous advancements in AI training and model capabilities promise an even greater impact on the future of coding. In summary, computer code generation using ChatGPT combines AI-driven efficiency with human creativity, empowering developers to work faster, learn effectively, and innovate with greater confidence.

10. Cypher generation using evolutionary computing

Evolutionary computing can be used to generate cryptographic cyphers by applying biologically inspired techniques such as genetic algorithms or evolutionary strategies. These methods evolve a population of candidate cypher keys or transformation rules over successive generations to maximise a fitness function—typically based on criteria like confusion, diffusion, and resistance to cryptanalysis. By introducing mutation, crossover, and selection, the algorithm can explore a vast space of cypher configurations, potentially discovering novel and robust encryption schemes. This approach is particularly useful in generating nonlinear components such as S-boxes or in optimising existing cypher structures for enhanced security and performance. In this section, we consider the use of the TuringBot to implement this approach in practice.

10.1 On the TuringBot

TuringBot.com is an innovative AI-powered platform designed to streamline the process of algorithm creation and optimisation. By leveraging cutting-edge artificial intelligence, TuringBot enables users to generate, test, and refine algorithms for a variety of applications, such as data analysis, automation, and machine learning, without requiring extensive programming expertise.

Its user-friendly interface and advanced capabilities make it an invaluable tool for professionals, researchers, and students seeking efficient solutions to complex computational problems. TuringBot stands out as a versatile and accessible resource in the ever-evolving landscape of artificial intelligence and technology.

TuringBot.com is a symbolic regression software tool designed to automatically discover analytical formulas that best fit a given dataset. Symbolic regression differs from traditional regression techniques by not assuming a predefined model structure; instead, it searches the space of possible mathematical expressions to find the one that best explains or 'fits' the data. TuringBot was developed to make this process efficient, intuitive, and accessible, especially for those without a background in machine learning or advanced data science.

Launched in the late 2010s, TuringBot was created in response to the growing need for interpretable artificial intelligence models. While most machine learning tools rely on complex neural networks or ensemble models that are difficult to understand and verify, TuringBot emphasised simplicity, transparency, and mathematical interpretability.

By combining evolutionary algorithms with equation simplification techniques, TuringBot is able to produce human-readable formulas that describe complex datasets. This makes it particularly valuable in scientific, engineering, and academic contexts, where understanding the model structure is as important as prediction accuracy.

The software is available for Windows and Linux and comes with a straightforward graphical user interface, allowing users to load data, configure parameters, and generate formulas with minimal setup. TuringBot also offers command-line integration for advanced users and supports exporting results for further analysis.

As of the mid-2020s, TuringBot has gained a user base across various domains, from physics and finance to biology and control systems. It continues to evolve with improvements in computational performance and integration with modern workflows. The name 'TuringBot' pays homage to Alan Turing, reflecting the tool's focus on combining algorithmic intelligence with human-understandable outputs.

In an era increasingly focused on explainable AI, TuringBot stands out as a lightweight, focused solution for data modelling grounded in classical mathematical reasoning. In this context, the system uses symbolic regression to evolve a formula that represents a simulation of the data provided subject to a root mean square error (RMSE) between the data and the formula together with other 'solution information' and 'search options'.

In this work, we are interested in using the system to simulate real noise, that is to derive a formula that can be used as a key-dependent, or otherwise, cypher-generating algorithm.

A range of mathematical operations and functions are available including basic operations (addition, multiplication, and division), trigonometric, exponential, hyperbolic, logical, history, and 'other' functions. While all such functions can be applied, their applicability is problem-specific. Moreover, the TuringBot formula requires translation to a specific programming language. This requirement necessitates attention regarding compatibility with the mathematical libraries that are available to implement the formula in a specific language and the computational time required to compute such functions.

There is the issue of how many data points should be used for the evolutionary process itself. In this context, it is noted that the demo version used in the case example provided later, only allows a limited number of data points to be used, that is

quoting from the demo version (2024): ‘Only the first 50 lines of an input file are considered’.

It is therefore important to note that we are not aiming to produce a near-perfect simulation of noise (obtained from an independent source as discussed in the following section). Instead, we are interested in using the system to produce a nonlinear formula which provides a unique algorithm for generating random numbers and regenerates them, ideally, subject to the same key. To do this, we resort to using real random data obtained from natural noise sources as an input to the system. In effect, we use TuringBot to simulate real (natural) noise.

10.2 Natural noise sources

Regarding the application of TuringBot for generating cyphers, we require data that is based on a recording of real noise, ideally, white noise that is uniformly distributed.

Natural noise (e.g. environmental noise which is available from a range of sources) is highly dimensional with infinitely many states and independent variables. However, in general, the entropy of such noise may not be maximal because of its self-organisational properties and correlations. This is because many noise sources have random self-affine properties, which is a fundamental property of many forms of natural noise. True random number generators (TRNGs), also known as hardware random number generators, use circuitry to gather entropy from a hardware source and process it into a random bit stream.

For example, the Araneus Alea II [48] uses a reverse biased semiconductor junction to generate white noise. It is a USB stick which plugs into a USB port and is powered by it. The noise is amplified and digitised, and the output bits processed to produce a random bit stream that is effectively free of bias and correlation. Such systems are designed for cryptographic applications like key generation and one-time pads which are not reproducible.

There is an important historical example of the use of natural noise. This relates to the SIGSALLY (Green Hornet) encryptor developed by AT & T Bell Labs used by Prime Minister Winston Churchill and President Franklin Roosevelt for 1-to-1 transatlantic communications from 1942 to 1945. The ‘encryptor’ was based on the addition of noise to voice signals to produce an output (analogue) signal with minimal a signal-to-noise ratios, scrambling the speech signals over all frequency bands. The source of the noise was derived from a vacuum tube by recording the output of the tube (given no input signal) on a phonograph record.

The result was a recording of electronic noise which was used to mask voice signals through the addition of the recorded noise. The technology required to apply this approach in practice involved the time registration associated with the addition and subtraction of the noise source in a two-way sense. The distribution of these noise sources (i.e. the recorded media) was strictly controlled for obvious reasons. However, provided this ‘control’ was not comprised, the system represented a one-time pad and was effectively impossible to attack successfully.

10.3 Internet-based noise sources: Random.org

Random.org [49] is a widely recognised online platform that generates true random numbers by harnessing atmospheric noise as its randomness source. Unlike pseudo-random number generators, which rely on algorithms and are inherently

deterministic, Random.org provides randomness that is both unpredictable and unbiased, making it suitable for applications requiring high levels of security.

Since its inception in 1998, Random.org has been used for diverse purposes, including cryptography, scientific experiments, lotteries, and games. The platform offers a range of services, such as random number generation for integers, sequences, and even binary strings. Additionally, it provides specialised tools like random list shuffling, coin flipping, and card shuffling.

The service has gained trust and recognition for its integrity and transparency, as it allows users to validate the randomness of its outputs. Furthermore, the systems premium options include APIs for developers, enabling seamless integration into software and systems requiring true random data.

10.4 Conceptual overview of the approach

In order to provide the reader with a conceptual overview of the approach, we refer to **Figure 1** which shows a flow diagram for the basic components associated with the generation of a cypher using evolutionary computing. In this work, Random.org and TuringBot.com are used to facilitate a source of random numbers (from real noise) and a formula that approximates the random number stream, respectively.

The accuracy of the simulation depends on the number of iterations that are performed in TuringBot which in turn determines the computational time used to evolve the nonlinear function. The function is then iterated to allow a key to be applied, which forms the initial condition of the iterator. The nonlinear function then provides a key-dependent sequence of random numbers. However, to ensure that the sequence is uniformly distributed, post-processing is required to produce a cypher that can be used to undertake the encryption process.

10.5 Cryptographic strength

Once the cypher has been generated, it may be tested for its cryptographic strength, tests that all cypher-generating functions must pass. As illustrated in **Figure 2**, these include the following:

1. The generation of a uniformly distributed cypher, that is the cypher function is post-processed to generate an output with a completely uniform histogram.
2. The power spectral density function of the cypher is uniformly distributed, thereby making a spectral attack redundant.

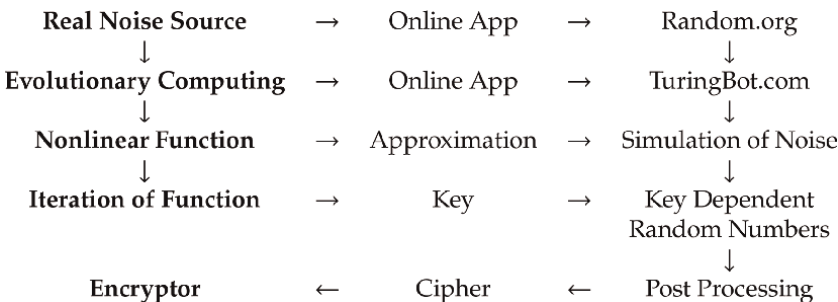


Figure 1.
Schematic (flow diagram) of the basic components associated with the generation of a cypher using evolutionary computing with Random.org and TuringBot.com.

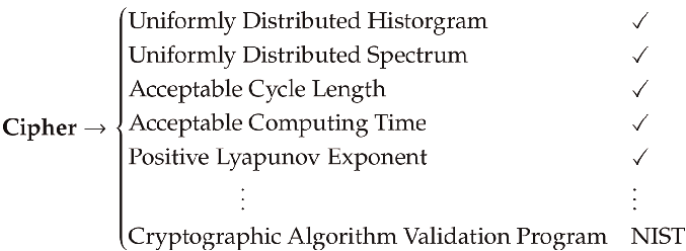


Figure 2. Simple schematic showing a sample of the ‘tests’ that can be applied evaluate the cryptographic strength’ of a cypher evolved through the simulation of real noise using the TuringBot. Each test that the cypher passes (indicated by a ✓) improves its strength, ultimately leading to possible submission to the Cryptographic Algorithm Validation Program of the Computer Security Resource Centre in the Information Technology Laboratory at the National Institute of Standards and Technology (NIST).

- 3. The autocorrelation of the cypher is a (Kronecker) delta function, indicating that there are no correlations below the length of the cypher that is set for encrypting a plaintext, thereby indicating that the cycle length is larger than any and all plaintext(s).
- 4. The computing time required for the floating-point computations (typically to double precession) is acceptable for the given hardware that will implement the encryption process.
- 5. The cypher has a large and positive Lyapunov exponent relative to a known algorithm with accepted cryptographic strength, for example the advanced encryption standard.

Finally, a cypher can be submitted for further independent testing and potential certification through, for example, the Cryptographic Algorithm Validation Program of the Computer Security Resource Centre in the Information Technology Laboratory at the National Institute of Standards and Technology (NIST) [50]. This step is not strictly necessary, and in a sense, defeats the whole point of the concept, by sharing cryptographic algorithms with others, algorithms that can be generated ‘on the fly’ in a timely and effective way. Nevertheless, for some potential commercial applications, certification may be an accepted, if unnecessary requirement.

These tests must be understood in the context of an infinite and truly random string, which has no statistical bias, a delta autocorrelated function and an infinite and uniform power spectrum (white noise). A high and strictly positive Lyapunov exponent is necessary because the iteration function that it is taken to characterise must generate chaotic trajectories within a few iterations.

10.6 Structural stability and conditions for unpredictability

For a cypher-generating algorithm to be considered structurally stable, it must exhibit (approximately) the same cycle length across all initial conditions. Additionally, it should ensure computational unpredictability for any and all initial conditions (i.e. keys).

In this context, there is no formal proof that an evolved nonlinear function will generate an unpredictable random number stream for every possible key. The tests

discussed previously do not guarantee the diffusive properties of a cypher; specifically, that the output remains structurally stable.

Ideally, such an algorithm would exhibit both a consistent cycle length and Lyapunov exponent for all keys. However, most known pseudo-chaotic systems do not possess these properties, and currently, no rigorous analytical test exists to assess them.

This represents a significant challenge. Without resolving it, we cannot ensure that a cryptosystem based on an evolved nonlinear function will consistently produce uncorrelated output sequences for arbitrary keys, for example floating-point numbers between 0 and 1 that have of variable number of decimal places.

10.7 Iterative vs. non-iterative functions

When using evolutionary computing to generate a cypher, structural stability is more likely to be achieved if a longer sequence of real noise is used to seed the evolutionary computing process. However, this comes at the cost of increased computational time. Additionally, incorporating a broader set of mathematical functions—such as those available in tools like TuringBot—can enhance the complexity of the evolved nonlinear function. But this is at the cost of increasing the computational demands required to generate the cypher.

These considerations highlight the importance of assessing whether an evolved nonlinear function is applied iteratively or otherwise. That is, the function can either be employed directly to generate a once-only cypher, or it can be applied iteratively in a key-dependent manner. In the former case, the function itself effectively serves as the key.

10.8 Example application

An example of using Random.org coupled with TurinBot to generate a cypher-generating algorithm is illustrated in **Figure 3**. In this example, an array consisting of 50 floating-point values in the range 0 to 1 was generated using Random.org as shown in **Figure 3** in the left hand side (LHS) column. Using this data, the system was conditioned to evolve a formula using only a limited range of the available functions, namely trigonometric, exponential, and hyperbolic functions.

The output considered was the result of an evolution after 44 iterations with a RMSE~0.0922. The selection of the mathematical functions used, and the solutions developed are presented here for illustrative purposes only. A full study of the function types and solution accuracy requirements, subject to input data of an arbitrary length, lies beyond the scope of this publication. In this context, the TuringBot ‘solution’ is given by.

$$0.432357 + (-0.498752 * \cos(0.594808 * (-4.10993 * \cosh(\text{row}) - (0.590558 + \tanh(\cos(\tan(\cos(\text{row} / (-0.00153462))) + \text{row} * \text{row}))))))$$

where row denotes the index of the array.

This formula represents a nonlinear function f given in Eq. (3). Iterating, we construct the following Python solution that returns a pseudo-random number stream seeded by a key—a floating-point number of arbitrary length in the range 0 to 1.

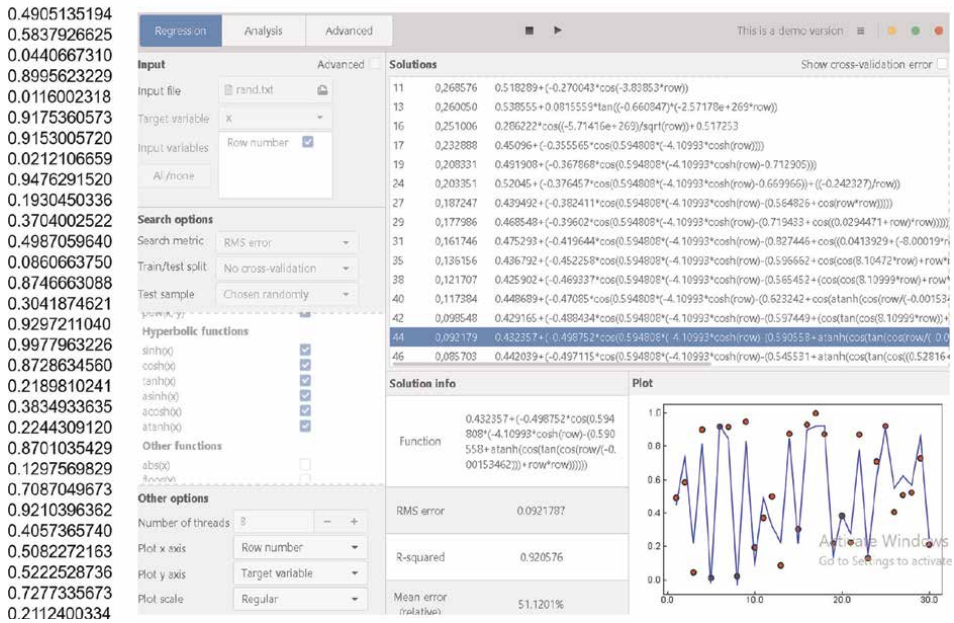


Figure 3.
Example application of a TuringBot simulation applied to a set of 50 floating-point fractional numbers obtained from Random.org shown in the column given to the LHS of the screenshot.

```
import numpy
# Define Evolutionary Computer
# Generated (ECG) Cipher function.
# Inputs: array length N and
# Key of type float in range (0,1).
def ECGCipher(N, Key):
    #Initialise numpy array
    row = numpy.empty(N)
    #Set initial value to key
    row[0] = Key
    # Iterate using TuringBot generate nonlinear function
    for i in range(0,N-1):
        row[i+1] = 0.432357 + (-0.498752 * math.cos(
            0.594808 * (-4.10993 * math.cosh(row[i]) -
            (0.590558 + math.atanh(math.cos(
            math.tan(math.cos(row[i] / (-0.00153462)))
            + row[i] * row[i])))))
    return(row)
```

In order to guarantee a cypher that is uniformly distributed, we post-process this function—see **Figure 1**. For example, the following Python function undertakes this operation by using `scipy.stats.rankdata` to compute the ranks of the input array, preserving the relative order of elements.

```
from scipy.stats import rankdata
def transform_to_uniform(input_array):
# Transform the statistical distribution
# of an input array into a uniform distribution.
# Parameters: input_array (list or np.ndarray), i.e.
# the input array with an arbitrary distribution.
# Returns: np.ndarray - An
# array with a uniform distribution.
# Convert the input array to a numpy array
    input_array = numpy.array(input_array)
# Compute ranks of the input array
# (1-based ranking)
    ranks = rankdata(input_array, method='average')
# Normalize ranks to range[ 0, 1]
# to get the CDF values
    cdf_values = (ranks - 1) / (len(input_array) - 1)
# Map the CDF values to a uniform distribution
    uniform_array = cdf_values
# This array is now uniformly distributed in[ 0, 1]
    return uniform_array
```

10.9 Example implementation: Securing data on the cloud

As cloud adoption proliferates, ensuring data security—especially confidentiality and integrity—has become paramount. Traditional encryption methods often rely on predetermined algorithms and static keys. However, evolutionary computing techniques provide a dynamic alternative for generating encryption keys and internal cypher components that adaptively evolve to improve robustness.

A notable approach involves the genetic algorithm (GA)-driven generation of PRNGs tailored for cloud cryptography. For instance, users may combine shift registers and genetic evolution to strengthen secrecy keys for cloud environments, demonstrating enhanced randomness and key-space expansion through fitness-based optimisation. Such data-derived key generation improves resistance to brute force and statistical attacks by continuously adapting, based on noise or entropy sources.

Another evolutionary model is to leverage GAs to optimise elliptic-curve cryptography (ECC) key generation. This evolutionary ECC (ECC-GA) approach reduces computation time and key size while maintaining secure levels of cryptographic strength; a hybrid method that enhances efficiency and allows flexible adaptation across cloud contexts.

Moreover, evolutionary optimisation of Substitution-boxes (S-boxes)—a core nonlinear component in symmetric cyphers—has the potential to dramatically improve nonlinearity, confusion, and diffusion metrics. These dynamically generated S-boxes are more resistant to cryptanalysis than static designs and can be regenerated per session or user, complicating attack models.

However, evolutionary algorithms introduce practical challenges: high computational cost, definition of secure fitness functions, and vulnerability to local optima. In cloud environments, these costs can be mitigated *via* scalable resources and 'Evolution as a Service (EaaS)'—a paradigm enabling encrypted outsourcing of evolutionary

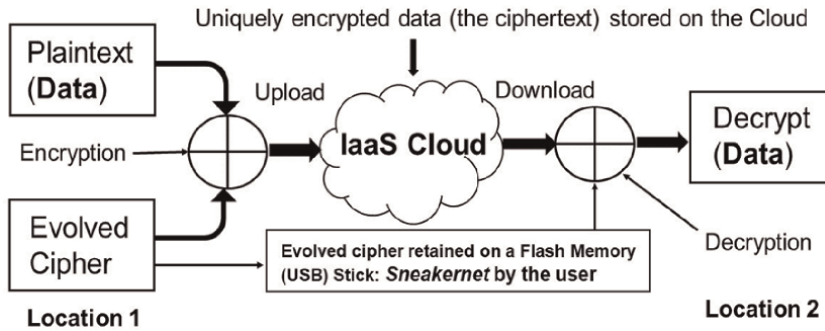


Figure 4. Schematic diagram illustrating data encryption (modulo-2 encryption denoted by $\oplus$) using a unique ciphertext obtained using EC. The encrypted data is stored on the cloud, and the user retains the cypher on a flash memory stick (as a 'sneakernet') to decrypt the data at the same (Location 1) or a different (Location 2) data access point where the encrypted data is downloaded from the cloud and decrypted.

computation while preserving user privacy. Evolutionary computation therefore offers promising avenues for enhancing cloud data security by generating robust keys, optimising cryptographic elements, and adapting to dynamic threats.

A primary concern in the use of the cloud for the storage of data has and continues to be the security of the data that is placed on the cloud. This includes protecting client confidentiality in legal and medical applications, for example. For this purpose, EC can be used to generate a unique cypher in the implementation of a one-time pad (OTP) for a 'one-to-cloud' (OTC) application. The basic scheme is illustrated in the schematic given in **Figure 4**, which shows how a bijective relationship between the plaintext and the OTP is established using (by way of an example) an exclusive-OR gate (XOR) to implement modulo-two encryption. The ciphertext output from the gate is uploaded to and stored in the Infrastructure as a Service (IaaS) Cloud.

In this OTC application, there is no key distribution problem (KDP) to be solved¹, a problem that typically accompanies OTP encryption. Encryption is applied locally by the cloud client who transports the OTP key to another location for decrypting the data after it has been downloaded from the cloud. This approach provides a solution to cloud security in which the ciphertext cannot be broken providing the approach is used correctly. The security relies on the approach of the user to managing the cypher which would typically be retained on a flash memory USB stick.

Traditional bi-directional OTP communication applications have a KDP because the key is shared between two parties. In this case, however, there is no KDP because only the client encrypts/decrypts the data. More importantly the OTP is based on a cypher that is unique to one user and one user alone. This eliminates the feasibility of successfully launching side-channel and other less-sophisticated attacks irrespective of the encryption used.

The encrypted data is completely unreadable and hence unbreakable, for adversaries who gain access illegally to a cloud server. For this purpose a large database of cyphers can be generated and stored in an air-gapped computer (one not connected to the Internet) at Location 1 as given in **Figure 4**. When required, a plaintext-size OTP is extracted from the OTP database and written to a flash drive, which is carried by

¹ A KDP is when a key is required to be distributed between two or more users

the client for decryption at another location—Location 2 as given in **Figure 4**. In both locations, a flash drive-type device is taken to be connected to a computer via a USB port but disconnected immediately after use.

The encryption algorithm(s), which may be key dependent or otherwise (as discussed in Section 10.7), is retained on a USB stick, for example. In the latter case, the USB stick housing the encryption software becomes the ‘key’. The data is then retrieved from the cloud as required. In this way, the data is secured using non-standard encryption. More importantly, it is encryption that has, ideally, been developed by and remains under the command and control of the user.

11. Case study: Strong password generation

For all the advancements in the field of cybersecurity, it is arguable that one of the principal failures lies in the simplest aspect, namely the generation, application, and management of suitable passwords.

A common issue we all encounter is generating passwords and then forgetting them from time to time. This challenge becomes even more daunting when considering the multiple passwords we need to create and manage for emails, social media, banking, and other accounts. Consequently, account login processes often include options such as ‘Forgot password?’ to facilitate password resets. These resets usually involve guidelines to ensure the new password is not related to any previously used passwords and incorporates characters that enhance its ‘strength.’

Passwords should be composed of a mix of upper- and lower-case letters combined with keyboard symbols to form an uncorrelated string that bears no resemblance to recognisable patterns such as names. This creates a strong password—essentially a cryptographically robust key—whose strength also depends on the length of the string.

Although we are all capable of generating such passwords and often receive guidance when doing so, the challenge lies in remembering them—especially when managing multiple passwords. As a result, users often resort to reusing the same password across different systems, typically compromising security. This practice significantly increases the risk of accounts being hacked.

In any organisation, weak passwords pose significant risks due to the volume of confidential information stored on their systems, ranging from personal data to institutional secrets. Another potential risk arises when passwords are cracked, enabling hackers to launch phishing campaigns.

In such scenarios, hackers gain control of an account and impersonate the account holder, sending links to trusted contacts and spreading spyware, viruses, and disinformation. To mitigate these risks, organisations increasingly emphasise the importance of creating strong passwords and fostering a culture of vigilance where individuals do not share their passwords and remain alert while entering them.

The lack of discipline associated with users creating and using strong passwords and maintains their security remains one of the primary weaknesses in cybersecurity. Understanding how hackers gain access to passwords and implementing countermeasures is therefore critical to protecting against predatory cybercriminals, [51].

In this section, we consider an approach to creating a strong password from a memorable password based on the application of evolutionary computing. This is accomplished using the Python programming language and ChaptGPT to generate the (Python) code.

11.1 Strong password generation using a memorable weak password

It is relatively simple to generate a strong password by randomising ASCII characters using a PRNG subject to a given string length (input by the user), which can then be output to the screen and/or written to a file. The strong password can then be copied from the file (or directly from the screen) and pasted into an application at the point of access as and when password verification is required. However, writing a password to a file is generally bad practice, especially if the file is retained.

In this context, the problem is how to remember the password that is produced, given that such a password might be one of many that have to be ‘remembered’. Retaining passwords on file so that they can be cut and pasted as required risks them being compromised if the storage device is compromised (stolen or hacked).

The file in which the passwords are maintained can itself be encrypted using a key that is easily remembered by the user. However, there is another approach which is to use a weak password that is easily remembered by the user, and, from this weak password, generate a corresponding strong password that can be regenerated multiple times. In this way, the strong password required for a given account can be generated from a weak user-defined and memorable password.

The Python program provided in Appendix A uses function `random` from the Python `random` module. The user is required to input a memorable password of their choice together with the (string) length of the new strong password that is required.

The program is an edited version of the code generated by an AI Bot. In this case, the AI app used was OpenAI ChatGPT V3.5 and was based on the following ‘Message ChatGPT’ prompt (‘Ask anything’):

‘Develop a Python program that allows the user to input a memorable password and the length required to output a corresponding strong password that is always the same for the same memorable password using the Python function `random`. and writing the result to the screen and to a file’.

The problem with this method is that the user must now remember both their weak password and the string length of the strong password required. This can be accomplished by including a number in the memorable weak password. For example, instead of using a weak password such a *myname*, one can now include a number, such as the day of one’s birthday, for example *myname23*, assuming that one is born on the 23 (of a given month of a given year). In this case, the maximum length of the strong password is 31 (the maximum number of days in a month—the month of January, for example). However, there are many other ways that can be ‘invented’ to include the length of the strong password required by ‘embedding’ the string length in a memorable but weak password. These inventions are ultimately at the discretion of the user.

11.2 Application of evolutionary computing

We consider the use of an EC generated formula to replace the Python function `random` (as given in Appendix A) in the generation of a strong password. This is based on the use of ChatGPT to generate Python code using the following ‘message’ (in the field ‘Ask anything’):

‘By applying iteration to the EC-generated formula

$$0.432357 + (-0.498752 * \cos(0.594808 * (-4.10993 * \cosh(\text{row}) - (0.590558 + \tanh(\cos(\tan(\cos(\text{row}/(-0.00153462)))) + \text{row} * \text{row}))))))$$

to produce a random number stream, generate Python code that inputs a memorable password and outputs a strong password of a specified length’.

The resulting code is provided in Appendix B.

12. Summary and conclusions

Chaos-based encryption using nonlinear functions has many disadvantages compared with conventional encryption algorithms. They are relatively computationally inefficient (requiring the application of floating-point arithmetic) and typically have low cycle lengths (compared to conventional cyphers). The principal advantage is that it provides the potential for ‘inventing’ an unlimited number of algorithms that can be used to produce multi-algorithmic encryption ‘solutions’.

Algorithms can of course be published so that the approach conforms to the Kerchhoff-Shannon principle as required, but in the knowledge that a new set of algorithms can be developed efficiently, evolutionary computing being used to automate algorithm generation.

This work has presented a novel approach to leveraging symbolic regression for the development of encryption algorithms, utilising AI-based code generation to explore and optimise cryptographic techniques. By harnessing symbolic regression, we have demonstrated how mathematical expressions can be evolved to create encryption mechanisms.

The methodology highlights the potential of AI-driven approaches in generating nonlinear and highly intricate nonlinear function, which can be tailored to meet the specific cryptographic requirements. The balance between complexity, performance, and security using this approach has not been considered in detail and remains to be studied. However, the illustrative examples provided of how the technique can be implemented to derive personalised cyphers underscores the viability of symbolic regression in discovering innovative encryption primitives that are resilient to known attacks while maintaining computational efficiency. Furthermore, the integration of AI-based code generation enables rapid prototyping and testing of these algorithms, streamlining the process of cryptographic development.

The proposed framework paves the way for several future research directions, including the exploration of hybrid encryption schemes, the use of quantum-resistant primitives, and the application of explainable AI to enhance the interpretability of the generated algorithms. Additionally, by incorporating advanced techniques like multi-objective optimisation, we can further refine the trade-offs between security, speed, and resource consumption.

Symbolic regression and AI-driven code generation offer a promising paradigm for advancing the state of cryptographic design, pushing the boundaries of what is achievable through traditional methods. This work lays the foundation for a new era of automated cryptographic development, driven by AI’s capacity to innovate and optimise.

In conclusion, it is clear that quantum computing is making legacy encryption methods redundant. Moreover, the future may be based on a democratisation of cryptography through AI based on:

- personalised encryption algorithms and key generators;
- DIY encryption methods and protocols.

Evolutionary computing offers a novel approach to creating numerous unique cyphers tailored for personalised authentication and data security. This is achieved by

extending the Fundamental Cryptographic Law, transforming it from ‘One Message, One Key’ to ‘One Message, One Key, One Cypher’.

Notably, due to the simplicity with which an unlimited number of cyphers can be generated using the methods discussed in this work, it is conceivable to replace the key itself with a cypher. This replacement can occur without requiring iterative processes, as quantified in Eq. (3), for example. Such an approach addresses concerns regarding the validation of conditions for unpredictability—specifically, the need for iterators to produce high-entropy random number streams with consistent cycle lengths across all possible initial conditions, i.e. floating-point numbers in a range between 0 to 1 with an arbitrary number of decimal places, subject to the bit precision of the computer where bit precision defines how finely and accurately a floating-point number can be represented in binary.

In the example presented in this work, the TuringBot solution used is relatively constrained, being developed for a limited-sized data array (50 elements) and a restricted set of functions. It is anticipated that the structural stability and unpredictability of a cypher derived from a TuringBot solution would improve significantly when applied to larger data arrays and when all available functions (and more) are utilised—a hypothesis that has yet to be rigorously investigated or validated. In this context, the example nonlinear function presented in this work is strictly for demonstrative purposes only.

Finally, the challenge of securely sharing such cyphers between Alice and Bob remains. The principle aligns with traditional key-sharing mechanisms, but in this scenario, a function of the form $ECGCipher(M, Key)$, as given in Section 10.8, must be exchanged. For this purpose, a quantum-secure version of a three-pass protocol, such as the one developed [52], can be employed.

In the case when a single user (i.e. Alice or Bob) is only interested in developing a personalised algorithm to encrypt their data before uploading it onto the cloud, for example, such an exchange is of course not necessary.

The aim of this publication has been to provide readers with a background to two forms of AI - EC and OpenAI ChatGPT—which can help them to generate their own encryption algorithms quickly and effectively. There are a number of issues that have yet to be investigated in detail. However, it is hoped that the material herein can provide readers with enough information for them to develop the approach further and implement it in a wide range of applications that are of practical significance. In terms of using AI to generate code for information security, it is an approach that is intended to optimise efficiency, accuracy, and most importantly, self-reliance.

In this context, it is worth noting that TuringBot is just one of several emerging applications in the field of genetic programming that can be used to evolve nonlinear functions for simulating real noise. Among these are various Python-based solutions that enable the development of a fully integrated Python program, eliminating the need to rely on an external application like TuringBot. One such example is *gplearn*, a Python library that implements genetic programming with an application programming interface inspired by and compatible with ‘scikit-learn’ [53].

Acknowledgements

Jonathan Blackledge acknowledges the support of Taighde Eireann—Research Ireland and the School of Electrical and Electronic Engineering, Technological University Dublin, Ireland.

Robert Kingstone is supported by the Faculty of Arts, Science and Technology, University of Wales, Wrexham, UK.

Brad Midgely acknowledges the support of the Department of Teaching and Learning, Barnsley College.

The authors acknowledge the use of OpenAI's ChatGPT (a generative artificial intelligence chatbot) in contributing to the composition and structure of this work, for proofreading the manuscript, and for assisting in the development of the Python code provided, and, as discussed in the appendices.

Conflict of interest

The authors declare no conflict of interest.

Appendix A: Python code for strong password generation

The ChatGPT-generated Python code provided in this appendix outputs a strong password with a user-defined length from a weak but memorable password.

Note that the code presented here has been free formatted in order that it fits within the page width associated with the formatting of this publication.

In order to correct for this after cutting and pasting the code in to a text file, say, the reader can use ChatGPT, to re-format and re-structure the code by asking the following:

'Format and correct the following Python code: ... '.

where ... is the code obtained from this appendix.

The code will then be formatted and corrected, with indentation, line breaks, and invalid quote characters, for example, replaced with standard ASCII quotes. The output generated can then be copied and pasted into a Python editor and run.

```
import random
import string
import hashlib

def generate_deterministic_password(memorable_password,
    length):
    if length < len(memorable_password):
        raise ValueError("Length must be at least the same
            number of characters as the memorable password.")

    # Use a hash of the memorable password to
    # seed the random generator with hash function sha
    seed = int(hashlib.sha256(memorable_password.encode()).
        hexdigest(), 16)
    random.seed(seed)

    # Generate extra characters deterministically
    extra_characters = ".join(random.choices(
        string.ascii_uppercase + string.digits +
        string.punctuation + string.ascii_lowercase,
        k=length - len(memorable_password)
    ))
```

```
# Combine memorable password and extra characters
combined = list (memorable_password +
extra_characters)
random.shuffle (combined)

return ".join(combined)

def main():
    print("Deterministic Password Generator")
    memorable_password =
    input("Enter a memorable password: ")
    length = int (input ("Enter the desired length of\
the strong password: "))

    try:
        strong_password = generate_deterministic_password
        (memorable_password, length)
        print (f"Strong password: { strong_password} ")

        # Save the strong password to a file
        with open ("deterministic_strong_password.txt",
"w") as file:
            file.write (f"Memorable password:
{ memorable_password} \n")
            file.write (f"Strong password:
{ strong_password} \n")

    print ("The strong password has been saved to
'deterministic_strong_password.txt'.")
    except ValueError as e:
        print (f"Error: { e} ")

if __name__ == "__main__":
    main()
```

Appendix B: Python code using an EC-generated cypher

The Python code given in this appendix is based on a modification of the code given in Appendix A where the function *random* is replaced with the example EC-generated cypher discussed in Section 10.8.

In order to run this code, it will need to be re-formatted and corrected, as discussed in Appendix A.

Note that the code can be further improved by asking ChatGPT to (for example):
‘Improve the formatting, structure, coherence and commentary of the following Python code and provide a header to explain its operation: ... *Code* ...’.

```
import math
```

```
def ec_formula(row):
    return 0.432357 + (-0.498752 * math.cos(
        0.594808 * (-4.10993 * math.cosh(row) -
        (0.590558 + math.atanh(math.cos(
        math.tan(math.cos(row/(-0.00153462)))+row*row))))))

def generate_random_number_stream(seed, length):
    stream=[ seed]
    for _ in range(length - 1):
        stream.append(ec_formula(stream[-1]))
    return stream

def generate_ec_strong_password(memorable_password, length):
    if length < len(memorable_password):
        raise ValueError("The length must be greater
        than or equal to the length of the memorable
        password.")

    # Convert memorable password into a numerical seed
    # by summing normalized ASCII values
    seed =
    sum(ord(char)/256 for char in memorable_password)

    # Generate a random number stream based on the seed
    # subject to a uniformly distributed transformation.
    random_stream=transform_to_uniform
    (generate_random_number_stream(seed, length)

    # Normalize the stream to printable ASCII characters
    strong_password="".join(
        chr(int(abs(value * 1000) % 94) + 33) \
        for value in random_stream
    )

    return strong_password

def main():
    print("EC-Based Password Generator with
    Random Stream")
    memorable_password =
    input("Enter a memorable password: ")
    length = int(input("Enter the desired
    length of the strong password: "))

    try:
        strong_password = generate_ec_strong_password
        (memorable_password, length)
        print(f"Strong password: {strong_password}")

        # Save the strong password to a file
        with open("ec_strong_password.txt",
```

```
"w") as file:
    file.write(f"Memorable password:
{ memorable_password} \n")
    file.write(f"Strong password:
{ strong_password} \n")

    print("The strong password has been saved to
'ec_strong_password.txt'.")
except ValueError as e:
    print(f"Error: {e} ")

if __name__ == "__main__":
    main()
```

Author details

Jonathan Blackledge^{1,2,3,4,5*}, Robert Kingstone⁶ and Brad Midgley⁷

1 School of Electrical and Electronic Engineering, Technological University Dublin, Ireland

2 Department of Computer Science, University of Western Cape, South Africa

3 Centre for Advanced Studies, Warsaw University of Technology, Poland

4 Chandigarh Engineering College, Chandigarh Group of Colleges, India

5 INTI International University, Malaysia

6 Faculty of Arts, Science and Technology, University of Wales, Wrexham, UK

7 Department of Teaching and Learning, Barnsley College University Centre, Barnsley, South Yorkshire, UK

*Address all correspondence to: jonathan.blackledge@tudublin.ie

IntechOpen

© 2025 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Daemen J, Rijmen V. The Design of Rijndael: AES - The Advanced Encryption Standard. Heidelberg, Germany. [Online]. Available from: <https://link.springer.com/book/10.1007/978-3-662-04722-4>; Springer-Verlag; 2002 [Accessed: 3 August 2025]
- [2] Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Communications of the Association for Computing Machinery (ACM). 1978;21(2):120-126. DOI: 10.1145/359340.359342 [Accessed: 13 August 2025]
- [3] Back T. Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University Press, Oxford, UK; 1996. Online. ISBN: 978019756092. [Online]. Available from: <https://academic.oup.com/book/40791> [Accessed: 14 August 2025]
- [4] Schmidt M, Lipson H. Distilling free-form natural laws from experimental data. Science. 2009;324(5923):81-85. DOI: 10.1126/science.1165893
- [5] Kerckhoffs A. La Cryptographie Militaire. Journal des sciences militaires [Military Science Journal] (in French). 1883;IX:161-191 [Online]. Available from: https://www.petitcolas.net/kerckhoffs/crypto_militaire_2.pdf [Accessed: 13 January 2025]
- [6] Shannon C. Communication theory of secrecy systems. Bell System Technical Journal. 1949;28(4):656-715. [Online]. Available from: <https://pages.cs.wisc.edu/rst/642-spring-2014/shannon-secrecy.pdf> [Accessed: 13 January 2025]
- [7] Hoare O. Enigma: Code breaking and the second world war - The true story through contemporary documents. In: Introduced and Selected by Oliver Hoare. Richmond, Surrey: UK Public Records Office; 2002
- [8] Blackledge JM, Ptitsyn N. On the applications of deterministic chaos for encrypting data on the cloud. In: Third International Conference on Evolving Internet, INTERNET 2011, 19-24 June, IARIA, Luxembourg. 2011. pp. 78-87. [Online]. Available from: <https://arrow.tudublin.ie/engscheleart/170/>. ISBN: 978-1-61208-008-6 [Accessed: 29 June 2025]
- [9] Thomson WE. A modified congruence method of generating pseudo-random numbers. The Computer Journal. 1958;1(2):83. DOI: 10.1093/comjnl/1.2.83
- [10] Rotenberg A. A new pseudo random number generator. Journal of the ACM. 1960;7(1):75-77. DOI: 10.1145/321008.321019. S2CID 16770825
- [11] Blum L, Blum M, Shub M. A simple unpredictable pseudo random number generator. Society for Industrial and Applied Mathematics (SIAM), Journal on Computing. 1986;15(2):364-383. [Online]. Available from: https://shub.ccny.cuny.edu/articles/1986-A_simple_unpredictable_pseudo-random_number_generator.pdf [Accessed: 17 January 2025]
- [12] Blum M, Micali S. How to generate cryptographically strong sequences for pseudo-random bits. SIAM Journal on Computing. 1984;13(4):850-864. [Online]. Available from: https://web.archive.org/web/20150224041435/http://www.csee.wvu.edu/~xinl/library/papers/comp/Blum_FOCS1982.pdf [Accessed: 14 January 2025]
- [13] Mollin RA. RSA and Public-Key Cryptography, Discrete Mathematics

and its Application. Florida, USA: Chapman and Hall/CRC; 2002 ISBN-10: 1584883383 / ISBN-13: 978-1584883388

[14] Diffie W, Hellman ME. New directions in cryptography. *IEEE Transactions on Information Theory*; (New York, USA: IEEE). 1976;**10**(6): 644-654. [Online]. Available from: <https://ee.stanford.edu/%7Ehellman/publications/24.pdf> [Accessed: 14 January 2025]

[15] Shor PW. Algorithms for quantum computation: Discrete logarithms and factoring. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Santa Fe, NM, USA. 20-22 Nov 1994. pp. 124-134. DOI: 10.1109/SFCS.1994.365700. [Online]. Available from: <https://ieeexplore.ieee.org/document/365700> [Accessed: 19 August 2025]

[16] Bernstein DJ, Buchmann J, Dahmen E. *Introduction to Post-Quantum Cryptography*. Berlin and Heidelberg, Germany: Springer-Verlag; 2009. ISBN: 9783642100192

[17] Regev O. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the Association for Computing Machinery (JACM)*. 2009;**56**(6):1-40. DOI: 10.1145/1568318.1568324 [Accessed: 19 August 2025]

[18] Bernstein DJ, Lange T, Peters C. Attacking and defending the McEliece cryptosystem. In: Buchmann JJ, Ding J, editors. *Post-Quantum Cryptography, PQCrypto 2008*. Lecture Notes in Computer Science. Vol. 5299. Heidelberg. [Online]: Springer, Berlin; 2008. DOI: 10.1007/978-3-540-88403-3_3 [Accessed: 12 January 2025]

[19] Buchmann J, Dahmen E, Hulsing A. XMSS – A practical forward secure

signature scheme based on minimal security assumptions. In: *Post-Quantum Cryptography (PQCrypto 2011)*. Lecture Notes in Computer Science. Vol. 7071. Berlin: Springer; 2011 [Online]. pp. 117-129. Available from: <https://huelsing.net/wordpress/wp-content/uploads/2013/05/mssgesamt.pdf> [Accessed: 16 March 2025]

[20] Ding J, Schmidt D. Rainbow, a new multivariable polynomial signature scheme. In: Ioannidis J, Keromytis A, Yung M, editors. *Applied Cryptography and Network Security, ACNS 2005*. Lecture Notes in Computer Science. Vol. 3531. Berlin, Heidelberg. [Online]: Springer; 2005. DOI: 10.1007/11496137_12 [Accessed: 16 August 2025]

[21] Jao D, De Feo L. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Yang BY, editor. *Post-Quantum Cryptography, PQCrypto 2011*. Lecture Notes in Computer Science. Vol. 7071. Berlin, Heidelberg. [Online]: Springer; 2011. DOI: 10.1007/978-3-642-25405-5_2

[22] Moody D et al. Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardisation Process. Gaithersburg, MD, USA: National Institute of Standards and Technology (NIST) Interagency/Internal Report (NISTIR) – 8309; 2020 [Online]. Available from: <https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8309.pdf> [Accessed: 14 January 2025]

[23] Linux Foundation, Open Quantum Safe, Open Quantum Safe Project, Linux Foundation's Post-Quantum Cryptography Alliance (PQCA). University of Waterloo; 2016. [Online]. Available from: <https://openquantumsafe.org/>. [Accessed: 14 January 2025]

[24] Lorenz EN. Deterministic non-periodic flow. *Journal of the*

Atmospheric Sciences. 1963;20(2):130-141. [Online]. Available from: <https://www.astro.puc.cl/~rparra/tools/PAPERS/lorenz1962.pdf> [Accessed: 4 March 2025]

[25] Toffoli T, Margolus N. Cellular Automata Machines: A New Environment for Modelling. Cambridge, Massachusetts, USA: MIT Press; 1987. DOI: 10.7551/mitpress/1763.001.0001. ISBN electronic: 9780262291019. [Online]. Available from: <https://direct.mit.edu/books/monograph/4258> [Accessed: 16 January 2025]

[26] Zhen P, Zhao G, Min L, Li X. A survey of chaos-based cryptography. In: Ninth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, Guangdong, China, 2014. 2014. pp. 237-244. DOI: 10.1109/3PGCIC.2014.69. [Online]. Available from: <https://ieeexplore.ieee.org/document/7024589> [Accessed: 26 January 2025]

[27] Matthews R. On the derivation of a chaotic encryption algorithm. Cryptologia. 1989;13(1):29-42. [Online]. DOI: 10.1080/0161-118991863745 [Accessed: 23 January 2025]

[28] Blackledge J, Ptitsyn N. Encryption using deterministic chaos. ISAST Transactions on Electronics and Signal Processing. 2010;4(1):6-17. [Online]. Available from: <https://arrow.tudublin.ie/engscheart2/15/> [Accessed: 14 January 2025]

[29] Blackledge J. Cryptography and steganography: New algorithms and applications. In: Lecture Notes, Centre for Advanced Studies. Warsaw, Poland: Warsaw University of Technology; 2011. DOI: 10.21427/D7ZS63. [Online]. Available from: <https://arrow.tudublin.ie/engscheart2/40/> [Accessed: 6 January 2025]

[30] Blackledge JM, Bezobrazov S, Tobin P, Zamora F. Cryptography using evolutionary computing. In: Irish Signal and Systems Conference ISSC2013. Letterkenny, Ireland: Letterkenny Institute of Technology; 2013. [Online]. Available from: <https://arrow.tudublin.ie/engscheart/197/> [Accessed: 16 January 2025]

[31] Turing AM. Computing machinery and intelligence. Mind. 1950;59:433-460. [Online]. Available from: https://www.cs.ox.ac.uk/activities/ieg/e-library/sources/t_article.pdf [Accessed: 19 August 2025]

[32] Holland JH. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, Complex Adaptive Systems. Cambridge, Massachusetts, USA: The MIT Press; 1992. DOI: 10.7551/mitpress/1090.001.0001. ISBN electronic: 9780262275552. [Online]. Available from: <https://direct.mit.edu/books/monograph/2574> [Accessed: 10 August 2025]

[33] Fogel DB. Evolutionary programming: An introduction and some current directions. Statistics and Computing. 1994;4:113-129 [Online]. DOI: 10.1007/BF00175356 [Accessed: 16 August 2025]

[34] Rechenberg IR. Evolutionsstrategie: Optimierung Technischer Systeme Nach Prinzipien der Biologischen Evolution, Frommann-Holzboog 15, Stuttgart, Germany. 1973. [Online]. Available from: <https://gwern.net/doc/reinforcement-learning/exploration/1973-rechenberg.pdf> [Accessed: 3 July 2025]

[35] Koza JR. Genetic programming as a means for programming computers by natural selection. Statistical Computing.

- 1994;4:87-112. Available from: <https://link.springer.com/article/10.1007/BF00175355> [Accessed: 23 March 2025]
- [36] Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia. Vol. 4. 1995. pp. 1942-1948. DOI: 10.1109/ICNN.1995.488968. [Online]. Available from: <https://ieeexplore.ieee.org/document/488968> [Accessed: 7 March 2025]
- [37] Hansen N, Muller SD, Koumoutsakos P. Reducing the time complexity of the De-randomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary Computation*. 2003;11(1): 1-18. DOI: 10.1162/10636560321828970. [Online]. Available from: <https://ieeexplore.ieee.org/document/6790790> [Accessed: 23 June 2025]
- [38] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*. 2002;6(2):182-197. DOI: 10.1109/4235.996017. [Online]. Available from: <https://ieeexplore.ieee.org/document/996017> [Accessed: 4 May 2025]
- [39] Fortin F et al. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*. 2012;13: 2171-2175. [Online]. Available from: <https://www.jmlr.org/papers/volume13/fortin12a/fortin12a.pdf> [Accessed: 16 March 2025]
- [40] Stephens T. Genetic Programming in Python. San Francisco, California, USA: GitHub Repository; 2023. [Online]. Available from: <https://github.com/trevorstephens/gplearn> [Accessed: 15 January 2025]
- [41] Back T, Fogel DB, Michalewicz Z. *Evolutionary Computation 1: Basic Algorithms and Operators*. Florida, USA. eBook ISBN9781315274638. [Online]: CRC Press; 2000. DOI: 10.1201/9781482268713 [Accessed: 14 March 2025]
- [42] Schwefel HP. *Evolution and Optimum Seeking*. New Jersey, USA. ISBN: 978-0-471-57148-3. [Online]. Available from: https://www.researchgate.net/publication/220690578_Evolution_and_Optimum_Seeking; Wiley; 1995 [Accessed: 19 August 2025]
- [43] Storn R, Price K. Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*. 1997;11(4):341-359. DOI: 10.1023/A:1008202821328
- [44] Dorigo M, Gambardella LM. Ant Colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*. 1997;1(1):53-66. DOI: 10.1109/4235.585892. [Online]. Available from: <https://ieeexplore.ieee.org/document/585892> [Accessed: 23 June 2025]
- [45] Open AI. ChatGPT: Chatbot Large Language Model Generative Pre-trained Transfer. San Francisco, California, USA; 2022. [Online]. Available from: <https://openai.com/index/chatgpt/>
- [46] Marar H. Advancements in software engineering using AI. *Computer Software and Media Applications*. 2024; 6(1):3906. DOI: 10.24294/csma.v6i1.3906. [Online] Available from: https://www.researchgate.net/publication/377918372_Advancements_in_software_engineering_using_AI [Accessed: 15 January 2025]
- [47] Sauvola J, Tarkoma S, Klemettinen M, et al. Future of software development

with generative AI. *Autom Software Engineering*. 2024;**31**(26):1-8. [Online]. Available from: <https://link.springer.com/article/10.1007/s10515-024-00426-z> [Accessed: 15 January 2025]

Available from: <https://gplearn.readthedocs.io/en/stable/> [Accessed: 6 July 2025]

[48] Araneus Alea II. True Random Number Generator, Araneus Information Systems, Finland. 2024. [Online] Available from: <https://www.araneus.fi/products/alea2/en/> [Accessed: 15 January 2025]

[49] Random.org. 2025 Randomness and Integrity Services Ltd, Dublin, Ireland. [Online] Available from: <https://www.random.org> [Accessed: 15 January 2025]

[50] Cryptographic Algorithm Validation Program. Gaithersburg, Maryland, USA: Computer Security Resource Centre, Information Technology Laboratory at the National Institute of Standards and Technology (NIST). US Department of Commerce; 2025. Available from: <https://csrc.nist.gov/projects/cryptographic-algorithm-validation-program> [Accessed: 27 June 2025]

[51] Walker D. *How Do Hackers Get your Passwords?* London, UK: ITPro; 2023. [Online] Available from: <https://www.itpro.com/security/34616/the-top-password-cracking-techniques-used-by-hackers> [Accessed: 15 January 2025]

[52] Blackledge JM et al. Phase-only digital encryption using a three-pass protocol. In: 30th Irish Signals and Systems Conference (ISSC 2019), June 2019. Maynooth, Ireland: Maynooth University; 2019. [Online] Available from: <https://arrow.tudublin.ie/engscheleart/297/> [Accessed: 16 May 2025]

[53] Stephens T. Gplearn's documentation. *Genetic Programming in Python, with a Scikit-Learn Inspired and Compatible API*: gplearn; 2025. [Online]

Optimized RSA-1024 FPGA Implementation on Artix-7

Ridha Ghayoula, Sahbi Bahrour, Wided Amara, Jaouhar Fattahi, Amor Smida, Issam El Gmati and Waly Mohamed I

Abstract

The design and hardware implementation of the RSA public key cryptographic algorithm on an Artix-7 FPGA are discussed in this chapter, with a focus on successfully utilizing this design implementation in securely processing medical imaging data. The system is designed in VHDL and is targeted for a high-speed and secure embedded solution for data encryption in sensitive healthcare based applications. For the sake of computational efficiency, we include both Montgomery multiplication algorithm and modular exponentiation from the library since they are critical for speeding up arithmetic operations for large integers as in RSA. These operations dramatically decrease computation execution time for encryption and decryption and the method is available for real-time embedded systems. The proposed design allows flexible message block sizes and can be adjusted to different security levels. This flexibility makes it applicable to various scenarios, ranging from encrypting tiny data packets to secure big medical images file (e.g. up to 1024 bits). Simulations and synthesis on the Artix-7 FPGA are used to show the practicability and efficiency of the design in speed, resources usage, and security. In the age of the importance of the security and privacy of medical information, we focus in this chapter on the fact that the combination of RSA cryptography, Montgomery modular arithmetic, and FPGA technology is strong and at.

Keywords: FPGA, RSA cryptographic system, Montgomery's algorithm, medical image application, modular exponentiation, embedded systems

1. Introduction

Asymmetric encryption was first presented by Whitfield Diffie and Martin Hellman in 1976 [1, 2] while providing a solution to the key exchange problem securely. What makes our system different is that it is simple to generate a key pair, but very hard to extract a private key from a public key.

In 1977, the RSA algorithm was proposed [3], marking one of the earliest implementations of asymmetric encryption that utilized key pairs to enable secure message communication through mathematical principles. Even today, this algorithm remains widely employed, particularly on the Internet for secure transactions and online commerce, forming the foundation of many modern encryption systems [4, 5].

The RSA algorithm works by initially generating two pairs of asymmetric keys: one for the sender, referred to as *Alice*, and one for the recipient, referred to as *Bob*. After each party obtains their respective key pairs, secure communication can commence. *Alice* retrieves *Bob*'s public key (typically exchanged beforehand or made publicly accessible) and encrypts her message with it. The encrypted message is then sent to *Bob*, who decrypts it using his private key, which remains confidential. This process eliminates the need for sensitive key exchanges, ensuring that only *Bob*'s private key can decrypt the message, thereby securing the communication [6–8].

The security of RSA hinges on its use of a one-way encryption and decryption function. Applying this function with the private key is straightforward, but attempting to reverse the process without the private key is extraordinarily complex. Consequently, RSA was one of the first asymmetric algorithms to guarantee security, not only during encryption but also throughout the key transmission process by utilizing public and private key pairs. Its robustness is attributed to the computational difficulty of factoring large prime numbers, coupled with the use of multiple one-way functions to safeguard message security. However, improper implementation or neglected security measures can open the door to potential exploits. While no major vulnerabilities have been discovered in the algorithm itself, improperly configured systems may still be susceptible to attacks. Each year, security researchers uncover flaws in various systems, though it remains rare to find attacks capable of completely compromising a widely used encryption algorithm.

In practice, asymmetric encryption algorithms are not always preferred due to the computational cost associated with encrypting and decrypting long messages. Instead, a hybrid approach is often employed, combining symmetric and asymmetric algorithms to harness the strengths of both. Symmetric algorithms offer faster encryption and decryption, while asymmetric algorithms ensure secure key transmission. This concept, known as hybrid cryptography, traces its origins to the Diffie-Hellman key exchange protocol, which pioneered the use of asymmetric encryption. Today, hybrid cryptography frequently relies on the symmetric AES algorithm due to its speed and reliability. AES is widely adopted, even by the US government for top-secret classifications, owing to its robustness and the absence of practical attacks against it thus far. Consequently, RSA is rarely used alone but rather as part of hybrid systems, such as OpenPGP or GnuPG.

RSA (Rivest-Shamir-Adleman) encryption is inherently asymmetric [3], employing a pair of keys: a public key for encryption and a private key for decryption. These keys are generated by an individual, traditionally named *Alice*, who wishes to receive confidential data. *Alice* shares her public key, allowing her correspondents (e.g., *Bob*) to encrypt messages sent to her. Only *Alice* can decrypt these messages with her private key. Additionally, *Alice* may use her private key to sign messages, enabling others to verify the signature with her public key. Crucially, it must be “computationally infeasible” to derive the private key from the public key, given the computational resources and known methods available at the time of the exchange, and for the duration the secret must remain protected. RSA is frequently used to securely transmit symmetric encryption keys, after which communication proceeds using the faster symmetric encryption [9–11].

From a theoretical standpoint, breaking RSA is closely tied to factoring the modulus n into its prime factors p and q . Knowing n , e , and d makes it relatively easy to derive p and q . However, attacks against RSA do not always rely on factorization [12]. In 1998, Dan Boneh [13] from Stanford University and Ramarathnam Venkatesan from Microsoft demonstrated that, for small exponents, breaking RSA is not equivalent to factoring n , suggesting alternative attack avenues.

Another notable attack, devised by Paul Kocher [14], involves timing side-channel attacks, where an adversary carefully measures the time taken by smart cards to decrypt messages, exploiting this information to deduce the secret key. Mitigation strategies involve introducing artificial delays to obscure timing information.

Furthermore, in 1994, Peter Shor [15] revealed that quantum computers could efficiently factorize large integers, posing a significant threat to RSA. Shor's algorithm reduces the time complexity of factoring n to a scale proportional to the key length, effectively rendering RSA insecure in the face of large-scale quantum computing.

RSA remains a cornerstone of modern cryptography, its efficiency and security are closely tied to proper implementation and the ongoing evolution of cryptographic research. The emergence of quantum computing, in particular, underscores the need for continued vigilance and the development of post-quantum cryptographic algorithms [16].

2. Methodology

Prime numbers have long intrigued mathematicians [17, 18]. Their allure lies in their deceptively simple definition—a prime number is a natural number with exactly two distinct divisors, 1 and itself—contrasted with their infinite nature and seemingly irregular distribution. These numbers hold significant importance in arithmetic, the branch of mathematics focused on integers, and have found crucial applications in modern technologies, especially in cryptography for securing information. As internet usage has expanded, so too has the necessity for protecting sensitive data, such as credit card numbers, during transmission. This is where the RSA algorithm becomes essential, leveraging a fundamental property of prime numbers [19–21].

Asymmetric cryptography, also known as public-key cryptography, relies on the concept of a one-way function to encode messages. Such a function should be easy to apply to any message but extremely difficult to invert, making it challenging to retrieve the original message from the encrypted one. This technique not only secures the communication of secret messages but also enables sender authentication [22].

The process can be likened to a two-lock box: when one lock is secured, the only way to open the box is by using the other lock. In this analogy, one key is public, meaning anyone can access it, while the other key remains private, known only to its owner. For instance, if Bob wishes to send a confidential message to Alice, he places his message in Alice's box and locks it with her public key. Only Alice, possessing the private key, can unlock the box and read the message. Conversely, if Alice wants to sign a message, she locks the box with her private key. Any recipient can then unlock it with her public key, verifying the message's authenticity and confirming Alice as the sender.

The objective of asymmetric cryptography is to construct this virtual “two-lock box.” The RSA system, widely used and increasingly relevant in computer communications, exemplifies this principle by capitalizing on prime numbers' unique properties.

RSA serves as a public-key mechanism for signing and authenticating messages: the sender encodes the message with their private key, and the recipient decodes it with the sender's public key, ensuring the sender's identity [23].

However, RSA is slower than traditional symmetric-key algorithms, making it inefficient for encoding large messages. As a result, RSA is often combined with conventional algorithms in a two-step process. First, RSA facilitates the secure

transmission of a third, shared secret key by encrypting it with the recipient's public key. Upon receipt, the recipient decrypts it using their private key. In the second step, the shared key is used with a symmetric algorithm to rapidly transmit large messages. Since symmetric algorithms can be 100 to 10,000 times faster than RSA, this hybrid approach is now common.

The RSA protocol relies on a fundamental arithmetic property that is surprisingly concise to prove. To appreciate this, one must understand key concepts about integers. Recall that a prime number is divisible only by itself and one, forming an infinite set: $\{2, 3, 5, 7, 11, 13, 17, 19, 23, 29 \dots\}$. These numbers are crucial in studying integers because every integer has a unique prime factorization (ignoring order) [24].

We say that p divides n if there exists an integer q such that $n = p \times q$. Two integers p and q are considered coprime if their greatest common divisor is 1. In such cases, Bézout's theorem guarantees the existence of integers m and n (one positive, one negative) satisfying $m \times p + n \times q = 1$. These results form the foundation of RSA's operation [25].

2.1 Theorem

The RSA theorem states that if two distinct prime numbers p and q are chosen, with

$$n = pq, \quad (1)$$

$$\varphi(n) = (p-1)(q-1), \quad (2)$$

and if integers e and d satisfy

$$ed \equiv 1 \pmod{\varphi(n)}, \quad (3)$$

then for any integer m we have

$$m^{ed} \equiv m \pmod{n}. \quad (4)$$

The proof relies on the fact that there exists an integer k such that

$$ed = 1 + k\varphi(n), \quad k \in \mathbb{Z}. \quad (5)$$

Working modulo p , if p divides m then

$$m^{ed} \equiv m \equiv 0 \pmod{p}, \quad (6)$$

otherwise $\gcd(m, p) = 1$ and, by Fermat's little theorem,

$$m^{p-1} \equiv 1 \pmod{p}. \quad (7)$$

Since $\varphi(n)$ is a multiple of $p-1$, there exists an integer t such that

$$ed = 1 + t(p-1), \quad (8)$$

which implies

$$m^{ed} = m^{1+t(p-1)} = m \cdot (m^{p-1})^t \equiv m \pmod{p}. \quad (9)$$

A similar argument holds modulo q , (so)

$$m^{ed} \equiv m \pmod{q}. \quad (10)$$

By the Chinese Remainder Theorem, we then conclude that

$$m^{ed} \equiv m \pmod{n}, \quad (11)$$

establishing that RSA encryption and decryption are indeed inverse operations.

2.2 Security of RSA

The security of the RSA algorithm is based on two conjectures. The first is that in order to break RSA and discover the private key, the number n must be factored. The second is that factoring is a hard problem, i.e., there is no fast algorithm (of polynomial complexity) to solve this question. Neither conjecture is proven; thus, it is possible that both are false. If this is indeed the case, then RSA is not a secure cryptographic algorithm.

The generation of quadruplets (p, q, e, d) with prime numbers p and q of a few tens or hundreds of digits is an easy and fast task, essential for real-world use of the cryptographic system. Primes of this size are easily obtained by probabilistic primality algorithms. One can then randomly choose e and check whether it is coprime with $(p-1)(q-1)$, which is an easy and fast operation using Euclid's algorithm. If e is coprime, the same algorithm provides the private key d simultaneously.

With modern techniques used to program RSA systems, doubling the key length multiplies the key generation time by 16 and the encoding/decoding time by 4. The use of RSA [26] is therefore said to be polynomial: the computation time grows as a polynomial function of the key length, whereas it is assumed that breaking RSA would require super-polynomial time. As computing power increases, the gap widens between the resources required to create and use longer keys and the power needed to break RSA. In other words, over time, RSA becomes more robust and secure. Confidence in RSA's security is not based on proof but rather on the repeated failure of all attempts to break the system over the past 25 years.

The proposed methodology in this work consists of the following steps:

- Study number theories applicable to RSA.
- Develop a mathematical model to compute RSA operations separately, connecting them through an advanced system bus based on the modular reduction system.
- Efficiently parallelize the mathematical solution using the Montgomery algorithm and modular exponentiation.
- Convert the solution to hardware modules described using VHDL. This phase forms the core of the work, with the VHDL code applied to the Xilinx Artix-7 FPGA in the next stage.
- Verify the hardware implementation using the Xilinx Vivado Synthesizer.
- Apply the implementation to the field of medical imaging.

2.3 New angles of attack

Theoretically, the situation remains challenging. As soon as n is factored into $p \times q$, d is immediately revealed. If we know n , e , and d , we can quickly find p and q . However, RSA's vulnerability is not necessarily tied to factoring n ; it is possible to break RSA through other means. Recent arguments suggest that such attacks should be seriously considered. In 1998, Boneh and Venkatesan established that for small exponents e , breaking RSA is not equivalent to factoring n .

Another clever attack, proposed by Paul Kocher, involves meticulously measuring the time taken by smart cards to decrypt messages, using these results to find the secret key. This attack can be prevented by distorting the apparent calculation time, for example by adding variable delays at the end of the decoding process.

Finally, in 1994, Peter Shor [15] demonstrated that with a quantum computer, one can factor integers efficiently, in a time proportional to the key length.

The comparison in **Table 1** highlights different cryptographic algorithms implemented on FPGAs, analyzing their approaches, resource usage, and results across several FPGA devices. Performance varies significantly, showcasing the evolution of FPGA-based cryptographic implementations and the trade-offs between performance and resource utilization. These comparisons provide valuable insights for selecting suitable algorithms and hardware configurations for cryptographic applications in FPGA environments.

Even if today the quantum computer is still in the realm of fiction, its development being estimated at a few decades, the progress of certain special services cannot be excluded. Finally, the RSA breakages already achieved are all linked to weaknesses in the implementation and not to the attack on the mathematical core. These are only practical observations drawn from the fact that all the recorded attempts to attack the RSA fail when the security conditions are respected. This observation does not preclude the idea that the theoretical risk associated with RSA is great, since the difficulty of factoring a number cannot be demonstrated, nor even that factoring is necessary to break RSA [3].

Several times in the history of espionage, an encryption system was considered inviolable by its users while an enemy service quietly read all the messages that fell

Reference	Algorithm	Year	Approach	Resources	Results
Ref. [27]	Montgomery	2016	Montgomery	Virtex-5XC5V	Slices = 8012
Ref. [21]	Booth's Algorithm	2015	Booth's Algorithm	Spartan3X	LUT = 1134
Ref. [28]	Square-and-Multiply	2014	Square-and-Multiply	Virtex5xc5vtx	Slices = 28,350
Ref. [29]	Montgomery	2011	Montgomery	Virtex-6	277.26 ms
Ref. [30]	Modular Exponentiation	2009	Modular Exponentiation	Virtex-II Pro	2.521 ms
Ref. [19]	Booth's Technique	2009	Booth's Technique	Virtex-II	< 15 and 50 ms
Ref. [31]	RSA IP	2020	RSA IP	xc7vx485	44 DSPs, LUT 210
Ref. [32]	VDFs	2020	VDFs	Virtex UltraScale	24,120 LUT (128)

Table 1.
Comparison with the state of the art.

into its hands. Thus, during the First World War, the French services decoded messages intended for German submarines and acted accordingly. During the Second World War, the code of the famous German Enigma machine was cracked. But the hundreds of people involved in this decryption remained silent for over 30 years. Today we know that powerful intelligence agencies are devoting considerable resources to decrypting the RSA. The risk that specialists somewhere know a way to break the RSA and remain silent must therefore be seriously considered. In the absence of results that mathematically prove the security of RSA, as with many other encryption methods, it is wise not to put too much faith in these methods and to combine systems that are accepted as secure [33–37].

2.3.1 Message encryption

If M is a natural integer strictly less than n representing a message, then the encrypted message will be represented by

$$C \cong M^e \pmod{n} \quad (12)$$

The natural number C being chosen to be strictly less than n .

2.3.2 Message decryption

To decipher C , we use d , the inverse of e modulo $(p-1)(q-1)$, and we find the clear message M by

$$M \cong C^d \pmod{n} \quad (13)$$

3. RSA algorithm

The complete system design is illustrated in **Figure 1**, presenting the Proposed Hardware Design Schematic Diagram of RSA.

The security of the RSA algorithm relies on two conjectures. The first conjecture asserts that breaking RSA and discovering the private key requires factoring the number n . The second conjecture states that factorization is a challenging problem, meaning there is no efficient algorithm with polynomial complexity to solve it. However, neither conjecture has been proven, leaving open the possibility that both could

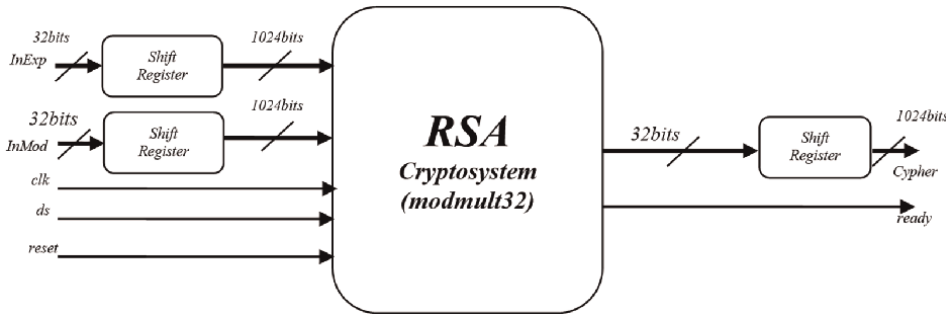


Figure 1.
Proposed hardware design schematic diagram of RSA CIPHER.

be false. If that were the case, RSA would not be considered a secure cryptographic algorithm [3, 38–42].

Generating quadruplets (p, q, e, d) with prime numbers p and q of dozens or even hundreds of digits is a straightforward and rapid task, crucial for the practical use of the cryptographic system. Large prime numbers can be efficiently obtained through probabilistic primality tests. Once p and q are selected, e can be chosen randomly, ensuring that e is coprime with $(p - 1)(q - 1)$. This can be verified quickly using Euclid's algorithm, which simultaneously determines the private key d when e satisfies the coprime condition.

With current RSA system programming techniques, doubling the key length results in a 16-fold increase in key generation time and a fourfold increase in encryption and decryption times.

The computational complexity of RSA is considered polynomial: the processing time grows polynomially with key length. On the other hand, it is hypothesized that breaking RSA would require algorithms with super-polynomial complexity. Some argue that as computational power increases, the gap will widen between the resources needed to generate and use longer keys and those required to break RSA. In essence, the longer the key, the more resilient and secure RSA becomes over time.

Confidence in RSA's security does not stem from formal proof but rather from the persistent failure over more than 25 years of all attempts to compromise the system.

3.1 Key generation algorithm

This is the original algorithm.

- Generate two large random primes, p and q , of approximately equal size such that their product $n = pq$ is of the required bit length, e.g. 1024 bits.
- Compute $n = pq$ and $\phi = (p - 1)(q - 1)$.
- Choose an integer e , $1 < e < \phi$, such that $\gcd(e, \phi) = 1$.
- Compute the secret exponent d , $1(n, e)$, such that $ed = 1 \bmod \phi$.
- The public key is (n, e) and the private key (d, p, q) . Keep all the values d, p, q and ϕ secret.

Sometimes the private key is written as (d, p, q) because you need the value of n when using d . Other times we might write the key pair as $((N, e), d)$.

4. Hardware implementation of RSA cryptography algorithm

FPGA implementations of RSA have attracted considerable attention as they address the inherent performance challenges of software-based cryptography [43]. These implementations exploit dedicated hardware resources—such as DSP slices, Block RAM, and configurable logic—to perform complex arithmetic operations like modular exponentiation and Montgomery modular multiplication with significantly higher speed and efficiency. In many designs, parallelism and pipelining are leveraged to process multiple operations concurrently, effectively reducing the overall latency

during encryption and decryption procedures. Moreover, the flexibility of FPGA architectures allows designers to fine-tune resource allocation and clock frequency, striking an optimal balance between throughput, power consumption, and hardware utilization. Detailed synthesis and testing on platforms such as Artix-7, Virtex-7, and Spartan series have shown that FPGA-based RSA can achieve real-time performance even for large key sizes like RSA-1024, thereby making it a robust solution for secure communications in both embedded systems and high-performance applications. The RSA Encryption/Decryption modules with key length 1024 are designed and implemented based on VHDL code. The Top Level Block Diagrams of RSA (Key Generator, Encryption, and Decryption) are shown in **Figure 1**.

5. Synthesis and implementation

The proposed work is to design and implement an RSA that could be used to assure security in many communication network applications. Our proposed solution is applied using FPGA Artix-7 technology along with the VHDL as a hardware description language. The implementation phase of the proposed architecture included hardware and software equipment's. Accordingly, following are the main component that has been used to accomplish the implementation and coding phase.

The synthesis results for the RSACYPHER-8BITS implementation show efficient usage of FPGA resources. The design utilizes only 1.17% of the available Look-Up Tables (LUTs), 0.45% of the Flip-Flops (FF), 17.14% of the Input/Output (IO) pins, and 3.13% of the global buffers (BUFG). These results indicate that the design is resource-efficient, requiring only a small portion of the available resources in the FPGA. For more details, refer to **Table 2**.

Table 3 presents the synthesis results for the RSACYPHER-16 BITS version. The FPGA resources used are 1.83% for the Look-Up Tables (LUT), 0.72% for Flip-Flops

Logic utilization	Used	Available	Utilization (%)
LUT	244	20,800	1.17
FF	187	4600	0.45
IO	36	210	17.14
BUFG	1	32	3.13

Table 2.
Synthesis results (RSACYPHER-8 BITS) - device utilization summary (estimated values).

Logic utilization	Used	Available	Utilization (%)
LUT	380	20,800	1.83
FF	299	4600	0.72
IO	68	210	32.38
BUFG	1	32	3.13

Table 3.
Synthesis results (RSACYPHER-16 BITS) - device utilization summary (estimated values).

Logic utilization	Used	Available	Utilization (%)
LUT	649	20,800	3.12
FF	523	4600	1.26
IO	132	210	62.86
BUFG	1	32	3.13

Table 4.
Synthesis results (RSACYIPHER-32 BITS) - device utilization summary (estimated values).

(FF), 32.38% for Input/Output (IO) pins, and 3.13% for Global Buffers (BUFG). This indicates that the usage of LUTs and FFs is low, but the design utilizes a significant portion of the available IO pins, likely due to the management of external signals.

Table 4 presents the results for the RSACYIPHER-32 BITS version. In this case, the LUT usage is 3.12%, FF usage is 1.26%, and IO usage is 62.86%. While the LUT and FF usage remains relatively low, it is evident that the usage of IO pins has significantly increased compared to the 16-bit version. This suggests that the 32-bit version requires more external connections or communication with other modules.

These results show a general trend where increasing the data width (from 8bits to 32 bits) leads to greater utilization of IO resources.

Table 5 presents the implementation resources and power consumption for the RSACYIPHER at different data bit widths (8 bits, 16 bits, and 32 bits). For the 8-bit version, the total on-chip power is 3.127 W, with a junction temperature of 39.9°C and a thermal margin of 60.1°C. For the 16-bit version, the power consumption increases to 5.136 W, the junction temperature rises to 49.5°C, and the thermal margin decreases to 50.5°C. Finally, the 32-bit version consumes 9.84 W of power, with a junction temperature of 72.0°C and a thermal margin of 28.0°C.

The effective junction-to-ambient thermal resistance (JA) remains constant at 4.8° C/W across all three versions, indicating that the thermal dissipation efficiency per watt of power consumed does not change. The power supplied for each version is 0 W, which suggests that the power is assumed to be supplied externally or not relevant to this data. Additionally, the confidence level in the thermal performance data is marked as low for all versions.

These results highlight the increasing power consumption and junction temperature as the data bit width increases, reflecting the higher computational demands of the larger bit-width versions.

Power	8 bits	16 bits	32 bits
Total On-Chip Power	3.127w	5.136 w	9.84 w
Junction temperature	39.9°c	49.5 °c	72.0 °c
Thermal margin	60.1°c (12.5w)	50.5°c (10.5w)	28.0°c (5.8w)
Effective JA	4.8 °c/w	4.8°c/w	4.8°c/w
Power supplied	0 w	0 w	0 w
Confidence level	Low	Low	Low

Table 5.
Implementation resources power- RSACYIPHER.

5.1 Description of Montgomery's algorithm

In 1985 Peter Montgomery published his studies on a way to calculate the modular product without the need to perform division by the modulus [17]. This feature facilitates hardware implementation, since modular multiplication is generally considered a complicated arithmetic operation due to the inherent operation of multiplication and division, so it is too expensive in terms of area. Division, so it is too expensive an operation in terms of implementation area. There are two approaches to calculating modular multiplication: The first performs the modulo operation after the multiplication and the second one during the multiplication. The modulo operation is accomplished by integer division in which only the remainder is needed for the computational the computational advantage. The ordinary modular multiplication algorithm for the calculation of $(A \times B \bmod M)$ takes the normal method of multiplication which method of multiplication which accumulates the digits of the products $A \times b_i$ and reduces the the modular nesting to keep the result. These reductions are achieved by subtracting the correct multiple of the modulus from the intermediate result. This reduction depends on the most significant bit of the significant bit of the operand. From these three algorithms, it can be seen that algorithm (1) requires more iterations than algorithm (2), and thus calculates a Montgomery modular multiplication with $n + 1$ clock cycles instead of n . cycles instead of n . Algorithm (1) is generally preferable for use in the systolic Ring structure, where the multiplier is constructed with identical elements [44]. Algorithm (3) is a modification of algorithm (2) in which the multiplicand A is shifted by by 1 bit to simplify the q_i calculation. This means that the result is a factor of 2, which is too large, and so the loop performs an additional time and subsequently algorithm (3) also requires $n + 1$ clock cycles to perform the calculations. In our work we have chosen algorithm (2) for the simple reason that it is the fastest and most adaptable to our application. Adaptable to our application. On the other hand, Montgomery's algorithm [44] reverses the order of processing the digits of the multiplicand, using the least significant bit of the intermediate result to perform an addition rather than a addition rather than subtraction. Each of the three algorithms shown below is slightly modified and lends itself to a slightly different hardware slightly different hardware architecture.

Algorithm 1: Monpro1 Algorithm.

```

1: procedure MONPRO1( $A, B, M$ ).
2:   Input: Integer  $A$  (multiplicand), integer  $B$  (multiplier with bit representation  $\{b_0, b_1, \dots, b_{n-1}\}$ ), modulus  $M$ .
3:   Output: Integer  $S_n$ , representing the final computed result.
4:    $S_{-1} \leftarrow 0$  ▷ Initialize the state variable to zero
5:    $M' \leftarrow M + \frac{1}{2}$  ▷ Compute an adjusted modulus to incorporate a correction factor
6:   for  $i = 0$  to  $n - 1$  do.
7:      $q_i \leftarrow S_{i-1} \bmod 2$  ▷ Extract the least significant bit of the current state
8:      $S_i \leftarrow \frac{S_{i-1}}{2} + q_i \cdot M' + b_i \cdot A$  ▷ Update state: perform a right-shift, add a modular
       correction if needed, and include the contribution of  $A$  when the  $i^{th}$  bit of  $B$  is 1
9:   end for.
10:  return  $S_n$  ▷ Return the final state after processing all bits
11: end procedure.
    
```

The pseudo-code for performing the 3 operations operations is given in algorithms (1), (2) and (3) where:

$$A = \sum_{i=0}^{k-1} a_i \times 2^i; B = \sum_{i=0}^{k-1} b_i \times 2^i \text{ and } M = \sum_{i=0}^{k-1} m_i \times 2^i$$

(with) $a_i, b_i, m_i \in \{0, 1\}$

Where M is a k -bit integer (ie: $0 < M < 2^k$),

$A, B < M$ (and)

n is equal to $k + 2$ and.

S is the remainder ($0 < S < 2^k$).

Monpro1 is an iterative algorithm that processes inputs A , B , and M , where B is handled bitwise (with each bit denoted as b_i) and the modulus is adjusted to $M' = M + \frac{1}{2}$ to incorporate a corrective factor. The algorithm initializes the state S_{-1} to 0, and then, for each iteration from $i = 0$ to $n - 1$, it performs the following computations in a single integrated process. First, the least significant bit of the previous state is extracted:

$$q_i = S_{i-1} \bmod 2, \quad (14)$$

and then the state is updated according to:

$$S_i = \frac{S_{i-1}}{2} + q_i \cdot M' + b_i \cdot A. \quad (15)$$

This approach combines a right-shift operation (division by 2), a conditional modular correction based on q_i , and the selective addition of A when b_i equals 1. By integrating modular reduction within the iterative process, it avoids costly division operations, making Monpro1 particularly well-suited for FPGA implementations in RSA cryptographic systems where efficient handling of modular arithmetic is essential for high performance.

Algorithm 2: Monpro2 Algorithm.

```

1: procedure MONPRO2( $A, B, M$ ).
2:   Input: Integer  $A$  (multiplicand), integer  $B$  (multiplier with bit representation  $\{b_0, b_1, \dots, b_{n-1}\}$ ), modulus  $M$ .
3:   Output: Integer  $S_{n-1}$ , representing the final computed result.
4:    $S_{-1} \leftarrow 0$  ▷ Initialize the accumulator state
5:   for  $i = 0$  to  $n - 1$  do.
6:      $q_i \leftarrow (S_{i-1} + b_i A) \bmod 2$  ▷ Compute the correction bit  $q_i$  as the modulo 2 sum
of the previous state and  $b_i A$ 
7:      $S_i \leftarrow \frac{S_{i-1} + q_i M + b_i A}{2}$  ▷ Update the state by adding  $q_i M$  and  $b_i A$  to  $S_{i-1}$ , then performing a right-shift (division by 2)
8:   end for.
9:   return  $S_{n-1}$  ▷ Return the final computed state after processing all bits
10: end procedure.

```

The Monpro2 algorithm begins by initializing an accumulator S_{-1} to 0. For each bit b_i of the multiplier B (from $i = 0$ to $n - 1$), it calculates a correction bit q_i using the equation

$q_i = (S_{i-1} + b_i A) \bmod 2$, which determines the parity of the sum of the previous state and the current weighted contribution $b_i A$. The algorithm then updates the state S_i with the formula $S_i = \frac{S_{i-1} + q_i M + b_i A}{2}$, effectively integrating modular reduction and multiplication through a right-shift operation. This efficient integration of arithmetic operations makes Monpro2 particularly suitable for hardware implementations, such as FPGA-based RSA systems, where resource optimization and processing speed are critical.

Algorithm 3: Monpro3 Algorithm.

```

1: procedure MONPRO3( $A, B, M$ ).
2:   Input: Integer  $A$  (multiplicand), integer  $B$  (multiplier with bit representation
       $\{b_0, b_1, \dots, b_{n-1}\}$ ), modulus  $M$ .
3:   Output: Integer  $S_n$ , representing the final computed result.
4:    $S_{-1} \leftarrow 0$  ▷ Initialize the accumulator to zero
5:    $A \leftarrow 2 \times A$  ▷ Double the multiplicand  $A$  to prepare for the modified
      accumulation
6:   for  $i = 0$  to  $n - 1$  do.
7:      $q_i \leftarrow S_{i-1} \bmod 2$  ▷ Extract the least significant bit of the previous state
8:      $S_i \leftarrow \frac{S_{i-1} + q_i M + b_i A}{2}$  ▷ Update the state by combining a right-shift, a modular
      correction, and the weighted contribution of  $A$ 
9:   end for.
10:  return  $S_n$  ▷ Return the final state after processing all bits
11: end procedure.
    
```

Monpro3 is an iterative algorithm that extends previous modular arithmetic approaches by first doubling the multiplicand A to enhance the weighted contribution in the subsequent computations. Starting with an initial state $S_{-1} = 0$, the algorithm processes each bit b_i of the multiplier B in sequence; at each iteration, it extracts the least significant bit $q_i = S_{i-1} \bmod 2$ from the current state, which is then used to adjust the state update via the term $q_i M$. The state S_i is updated using the formula

$$S_i = \frac{S_{i-1} + q_i M + b_i A}{2}, \quad (16)$$

which effectively combines a right-shift (division by 2), a modular correction based on q_i , and the selective addition of the doubled multiplicand A when b_i is 1. This integrated approach ensures efficient modular reduction during the iterative process, making Monpro3 particularly well-suited for hardware implementations in cryptographic applications, such as RSA on FPGA platforms.

As shown in the scalable architecture for Montgomery multiplication in **Figure 2**, the process involves the first stage of multiplication, typically focusing on the initial steps of the algorithm. **Figure 3** illustrates the second stage, where the architecture continues with further steps to complete the Montgomery multiplication, ensuring scalability and efficiency in larger computations.

5.1.1 The Monpro function

The MonPro function calculates the Montgomery product of the form:

$$\text{MonPro1}(A, B, M) = AB r^{-1} \bmod M \quad (17)$$

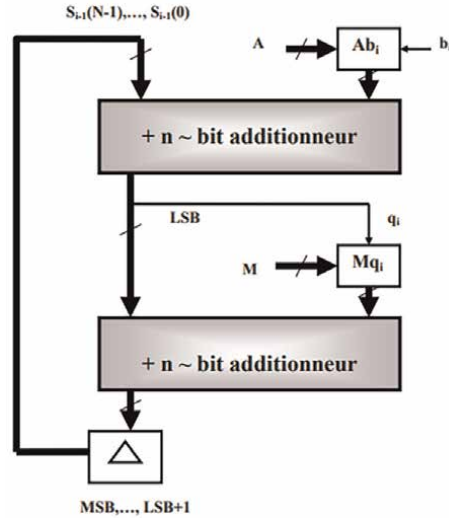


Figure 2.
Architecture Scalable pour la Multiplication de Montgomery (1).

Unfortunately, the extra factor r is taken into the calculation to produce the correct result. Here, r^{-1} is the inverse of $r(modM)$, ie: $r^{-1}r = 1(modM)$, where r is indicated by:

$$r = 2^n \quad (18)$$

The m -residue A of an integer $A < M$ is defined by:

$$A = ArmodM \quad (19)$$

The MonPro function can be used to convert an integer to its M -residue as follows:

$$Monpro(A, r, M) = Ar^2r^{-1} = ArmodM = A \quad (20)$$

The Montgomery product of 2 M -residues, A, B is equal to M -residue, S :

$$S = MonPro(A, B, M) = AB r^{-1}modM = ArBrr^{-1}modM = AB rmodM = \delta r^{-1}modM \quad (21)$$

Finally to convert S back to the ordinary integer form, δ :

$$\delta = Sr^{-1}modM = Monpro(1, S, M) \quad (22)$$

A prerequisite is that the modulus, M , must be a prime number to r (ie: $\gcd(M; r) = 1$). This is always the case in the RSA because M is the product of 2 large prime numbers ($p \times q$, and odd and since r is a power of 2. Since the operations of mod r and division r are inherently fast in binary systems since r is a power of 2, then Montgomery's algorithm is more faster and easier to compute than simple modular multiplication $ABModM$. However, the additional operations of converting the M -residue form, and pre-compiling $2^{2n}modM$ add extra computational steps. Thus, there

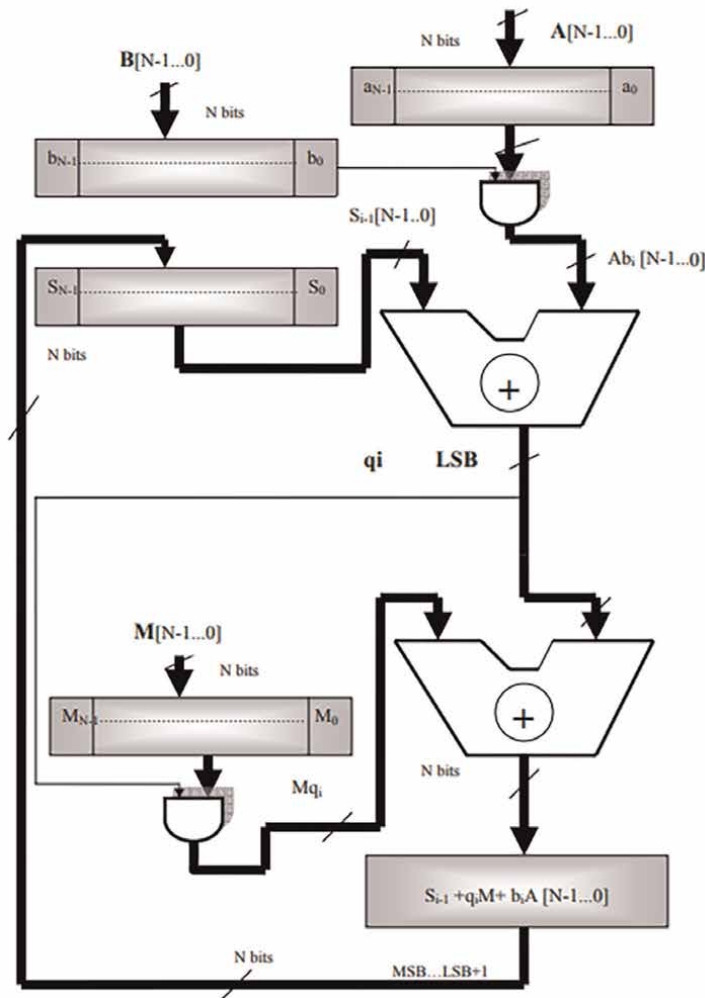


Figure 3.
Architecture Scalable pour la Multiplication de Montgomery (2).

is a disadvantage when only a simple modular multiplication needs to be performed. It is more appropriate when multiple multiplications of the same modulus have to be performed as in RSA's modular exponentiation algorithm.

Algorithm 4: Algorithm L-R (MonExp1).		
1: procedure MONEXP1(P, ε, M).		
2: $C_1 \leftarrow 2^{2n} \bmod M$	$\triangleright$ Compute the mapping constant C_1	
3: $P \leftarrow \text{Monpro}(C_1, P, M)$	$\triangleright$ Map P using C_1	
4: $R \leftarrow \text{Monpro}(C_1, 1, M)$	$\triangleright$ Initialize R by mapping 1	
5: for $i = k - 1$ downto 0 do .		
6: $R \leftarrow \text{Monpro}(R, R, M)$	$\triangleright$ Square R : compute $R^2 \bmod M$	
7: if $\varepsilon_i = 1$ (then)		
8: $R \leftarrow \text{Monpro}(R, P, M)$	$\triangleright$ Multiply: update R when the i^{th} bit ε_i is 1	

```

9:      end if.
10:   end for.
11:    $R \leftarrow \text{Monpro}(1, R, M)$      $\triangleright$  Re-map  $R$  to reverse the initial transformation
12:   return  $R$ .
13: end procedure.

```

5.2 Modular exponentiation

The Algorithm L-R, implemented as MonExp1, performs modular exponentiation using a left-to-right binary method combined with an initial mapping and a final re-mapping step. It starts by computing a constant $C_1 = 2^{2^n} \bmod M$ to transform the base P and to initialize the result R . The main loop iterates from $i = k - 1$ down to 0, where each iteration first squares the current result R (via $\text{Monpro}(R, R, M)$), and then conditionally multiplies it by the mapped base P if the corresponding bit ε_i of the exponent is 1. This combination of squaring and conditional multiplication, both executed through the efficient Monpro procedure, allows the algorithm to integrate modular multiplication and reduction seamlessly. Finally, a re-mapping.

Algorithm 5: Algorithm R-L (MonExp2).

```

1: procedure MONEXP2( $P, \varepsilon, M$ ).
2:    $C_1 \leftarrow 2^{2^n} \bmod M$      $\triangleright$  Compute the mapping constant  $C_1$ 
3:    $P \leftarrow \text{Monpro}(C_1, P, M)$      $\triangleright$  Map  $P$  using  $C_1$ 
4:    $R \leftarrow \text{Monpro}(C_1, 1, M)$      $\triangleright$  Initialize  $R$  by mapping 1
5:   for  $i = 0$  to  $k - 1$  do
6:     if  $\varepsilon_i = 1$  (then)
7:        $R \leftarrow \text{Monpro}(R, P, M)$      $\triangleright$  Multiply: update  $R$  when the  $i^{\text{th}}$  bit  $\varepsilon_i$  is 1
8:     end if.
9:      $P \leftarrow \text{Monpro}(P, P, M)$      $\triangleright$  Square: update  $P$  for the next bit
10:  end for.
11:   $R \leftarrow \text{Monpro}(1, R, M)$   $\triangleright$  Re-map  $R$  to reverse the initial transformation
12:  return  $R$ .
13: end procedure.

```

Algorithm R-L, implemented as MonExp2, performs modular exponentiation using a right-to-left binary method combined with an initial mapping and a final re-mapping step. It begins by computing a constant $C_1 = 2^{2^n} \bmod M$ to transform the base P and to initialize the result R . The main loop iterates from $i = 0$ to $k - 1$, where each iteration first checks if the i^{th} bit ε_i of the exponent is 1. If so, it updates R by multiplying it with the current P using the Monpro procedure. Regardless of the bit's value, P is then squared (via $\text{Monpro}(P, P, M)$) to prepare for the next bit. This approach ensures that both squaring and conditional multiplication operations are efficiently managed within the loop. Finally, a re-mapping step ($\text{Monpro}(1, R, M)$) reverses the initial transformation, yielding the final result R , which is equivalent to $P^\varepsilon \bmod M$. This method is particularly suitable for hardware implementations, such as FPGA-based cryptographic systems, where efficient modular arithmetic operations are essential. Modular exponentiation is performed by repeated modular

multiplication [44]. There are two common common algorithms that can be used: The binary L-R method and the binary R-L method. In algorithm (4), square and multiply are operations that must be performed sequentially, so the multiplications must be performed serially. be performed sequentially, and so multiplication must be performed serially. Means that square and multiply are operations that can be performed in the same multiplier, which is multiplier, which is simple to implement from a hardware point of view, thus optimizing the implementation area [44]. In algorithm (5), the square and the multiplied are independent operations, and can be executed in parallel. This means that at least 50% of the clock cycles are required to complete the operation. However, both physical multipliers are required to achieve the acceleration of the of the algorithm. Therefore, the area $\times$ speed products of the two algorithms are very similar. Similar. For our work, the R-L algorithm (algorithm (5)) will be used since the primary goal is to increase the baud rate. is to increase the bit rate of real-time exponentiation to encrypt and decrypt data. This work led to the development of an architecture for implementing the RSA cryptographic algorithm on a programmable FPGA circuit using Montgomery multiplication and modular Montgomery multiplication and modular exponentiation to ensure better hardware performance. This work enables 32-bit messages to be encrypted. The size of these messages can be up to 1024 bits to form an elementary block. The architecture of this work allows flexibility of application, as several blocks can be grouped together according to application rules, and it will be possible to encrypt even 2048-bit messages. These rules can include RNS base decomposition or other processes. Several improvements could be made to this architecture, for example, it would be possible to increase the size of the key or the encrypted message by implementing other methods. The architecture presented in this thesis is based on a description in VHDL language that can be implemented on an FPGA programmable circuit. So, if we wanted to make a faster version, or one dedicated to another architecture, we could easily integrate it with a faster ASIC technology. The major drawback of this architecture is its rigidity. We therefore set out to increase the modifiability of this architecture. On the other hand, FPGA circuits do not offer all the desired security aspects, since it is enough to analyze the contents of the associated ROM to go back to the schematic.

As illustrated in the block diagram of modular exponentiation in **Figure 4**, the operation involves calculating $x^e \bmod n$. The process starts with the input values of the base x , exponent e , and modulo n . Then, an exponentiation calculation is performed, typically using a fast exponentiation method, followed by a modulo reduction at each power step to avoid excessively large results. Finally, the result $x^e \bmod n$ is obtained as the output of the process.

6. RSA encryption/decryption of medical images

Encrypting medical images with the RSA algorithm, a public-key cryptography method, provides a robust solution for securing sensitive medical information. This process begins with the generation of an RSA key pair, a shared public key, and a confidential private key. Medical images are then converted into digital data and encrypted using the public key. This approach ensures that only those with the private key can access the original information, ensuring confidentiality and data security. RSA encryption also enhances the integrity of images by protecting them against unauthorized modifications. This measure, crucial for preserving patient

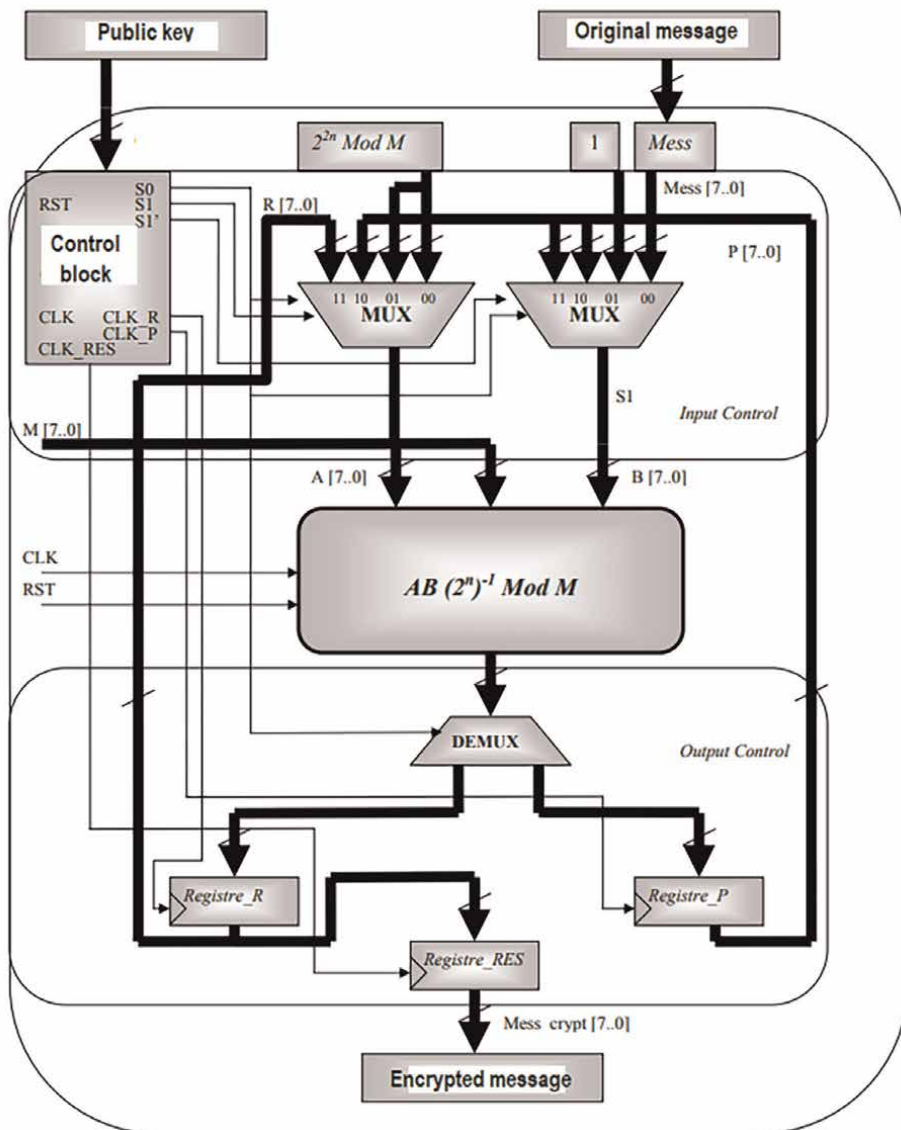


Figure 4.
Block diagram of modular exponentiation.

confidentiality, allows medical institutions to process and store images while adhering to strict data protection standards [45–50]. For medical images:

- Authentication of images with the insertion of information certifying the origin and attachment of an image to a given patient;
- Checking the integrity of images, by placing control information in the image, such as a digital signature, for example;
- The addition of metadata (data-hiding) to enrich the content of images by associating it with a semantic description of its content;

Another more comprehensive scenario combines both authentication and integrity checking of the images and seeks to establish a secure link between these images and the associated examination report. In information protection, the watermark appears as complementary to the mechanisms discussed previously because it associates the protection information and the image to be protected in a single entity: a marked image [51–54].

Another way to ensure confidentiality is to adapt the level of protection according to the application and the time available. It is in this part that we find selective encryption where users can apply proportional or adjustable security according to the level of protection desired according to the length of the key of the RSA algorithm. Many applications can make do with selective encryption; the images are then partially visible without completely revealing all the information. Selective encryption can be interesting in the case of medical images taken from a medical device and to be sent over the network in order to establish a remote diagnosis. In addition, the medical image acquisition device may be in an ambulance or in any other mobile vehicle, and in this case the transmission is carried out via wireless networks. For vital reasons, in this type of applications, the images must be transmitted quickly and surely, and in this case a selective encryption seems to be the best solution (time/security compromise) [55–62].

The **Figure 5** presented in this work illustrates the process of RSA 1024-bit encryption and decryption applied to a medical image. The interpretation of each figure is as follows:

- a. *Original image*: The first image shows the original medical image, which could represent an MRI, CT scan, or another type of medical imaging modality. This image is clear and contains recognizable structures or patterns that correspond to the underlying medical data. The pixel values in this image directly represent

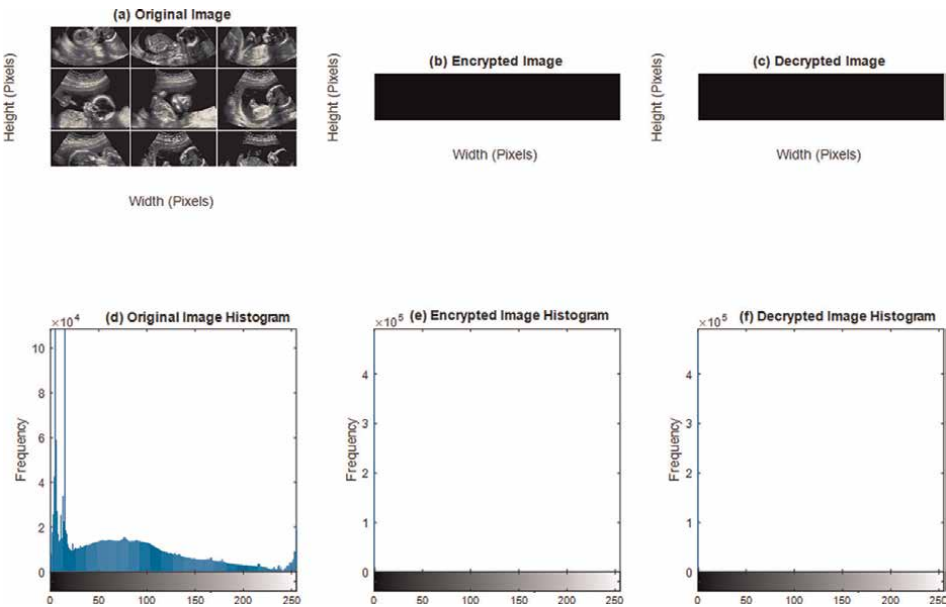


Figure 5.
(a) An original image and its encrypted Image (b) with RSA, with a key of 1024 bits (686x455 pixels).

the intensity of light at each point, providing a faithful representation of the original medical data.

- b. *Encrypted image*: This image represents the result of applying RSA encryption to the original medical image. After the encryption process, the image appears as a random collection of noise, with no discernible patterns or recognizable structures. The pixel values are scrambled using modular exponentiation, ensuring that the encrypted image cannot be interpreted without the corresponding private key. This randomization is a critical feature of the encryption process, which ensures the security and privacy of the image.
- c. *Decrypted image*: The decrypted image shows the result of applying the private RSA key to the encrypted image. The goal of the decryption process is to recover the original image. Ideally, the decrypted image should be identical to the original one, confirming that the encryption and decryption processes were successful. Minor discrepancies might appear due to the limitations of RSA when working with large images, but in this example, the decrypted image should closely resemble the original.
- d. *Original image histogram*: This histogram displays the distribution of pixel intensities in the original image. The x-axis represents the pixel intensity values (ranging from 0 to 255 for grayscale images), while the y-axis represents the frequency of each intensity level. The histogram of the original image provides insights into the image's contrast and brightness. A well-spread histogram suggests good contrast in the image, while a concentrated histogram indicates lower contrast.
- e. *Encrypted image histogram*: The histogram of the encrypted image should appear nearly uniform, reflecting the randomization of pixel values caused by the RSA encryption process. A flat or uniform histogram is expected because encryption scrambles the pixel intensities, making the image indistinguishable from random noise. This uniform distribution of intensities further ensures that the encrypted image cannot be easily analyzed or interpreted.
- f. *Decrypted image histogram*: The histogram of the decrypted image should closely resemble the histogram of the original image, confirming that the decryption process successfully restored the original pixel values. If the decrypted histogram shows significant deviation from the original, it may indicate an issue with the encryption or decryption process.

The figures demonstrate how RSA 1024-bit encryption transforms the original image into an unreadable form and how the decryption process can successfully recover the original image, both visually and in terms of pixel intensity distribution.

The original image **Figure 6** has a file size of 51.0 MB (with a 1.7 MB compressed version), which indicates that the image contains a significant amount of detailed information. With a resolution of 3797 x 4692 pixels, the image has a high pixel count, allowing for detailed visual representation, which is essential in medical imaging where fine details are crucial for analysis.

The image's physical dimensions are 32.3 x 39.6 cm or 12.7 x 15.6 inches when printed at 300 DPI (dots per inch). This resolution ensures that the image is suitable

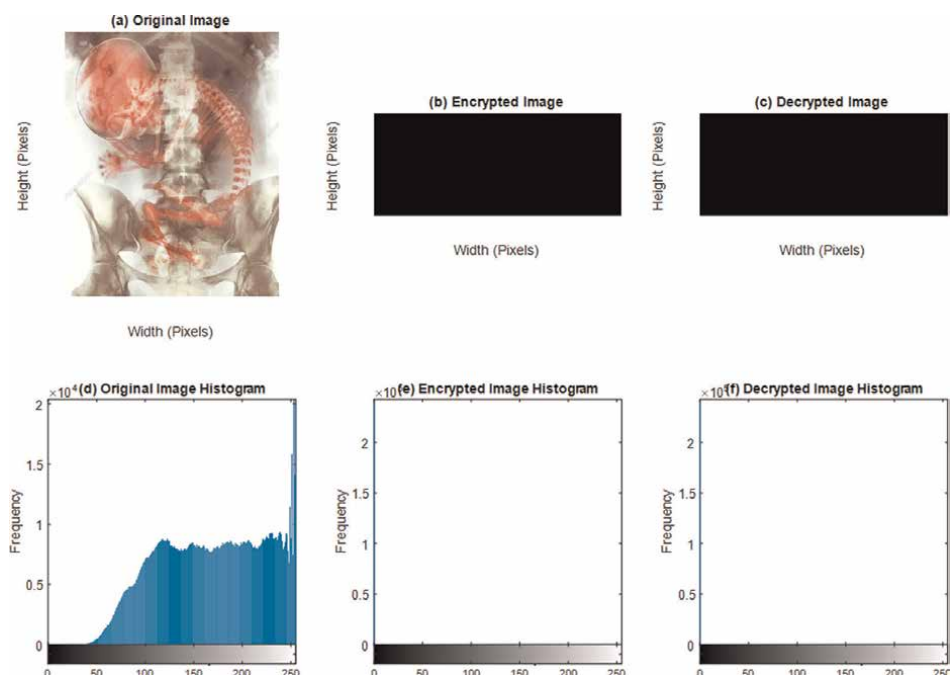


Figure 6.
 (a) An original image and its encrypted Image (b) with RSA, with a key of 1024 bits (1000x572 pixels).

for high-quality prints or digital displays, with 300 DPI being a standard for print quality. The high resolution and large size of the image make it well-suited for medical applications, where precision and clarity are vital for accurate interpretation and diagnosis.

These characteristics contribute to the richness of the image, making it suitable for RSA encryption and decryption processes, which aim to maintain the integrity and confidentiality of such high-resolution, medically relevant data.

The **Figure 7** presents ultrasound findings taken on postnatal day 2, demonstrating a heterogeneous complex mass with both solid echogenic and cystic anechoic components. The solid echogenic area, marked by the red solid circle, indicates a portion of the mass with dense tissue or structures that reflect ultrasound waves, appearing brighter on the image. The cystic anechoic component, identified by the red arrow, represents a fluid-filled space that does not reflect ultrasound waves, appearing darker on the image. Additionally, osseous elements and linear echogenic areas with posterior acoustic shadowing, shown by the yellow arrow, suggest the presence of bone or calcified tissue within the mass. The posterior acoustic shadowing occurs due to the dense nature of the bone, blocking ultrasound waves and creating a shadow behind it. These findings together indicate a complex mass with a combination of solid, cystic, and possibly calcified components. Further clinical evaluation may be required to determine the exact nature of the mass.

Figure 5 shows a comparison between an original image and its encrypted version using the RSA algorithm with a 1024-bit key. The original image has dimensions of 686x455 pixels, and the encrypted image is provided for comparison. This highlights how the encryption process with RSA secures the image content and affects its visual appearance. Similarly, **Figure 6** compares an original image with a 1024-bit RSA encryption, but with an image of 1000x572 pixels. It demonstrates how the RSA

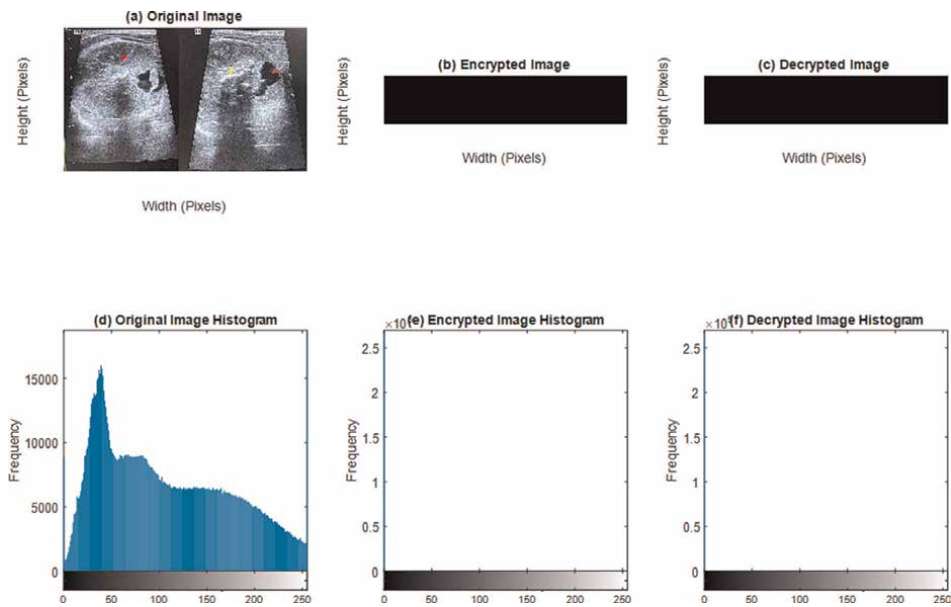


Figure 7. *Ultrasound findings taken on postnatal day 2 demonstrated a heterogeneous complex mass with solid echogenic (red solid circle) and cystic anechoic components (red arrow). Osseous elements and linear echogenic areas with posterior acoustic shadowing were also noted within the mass (yellow arrow).*

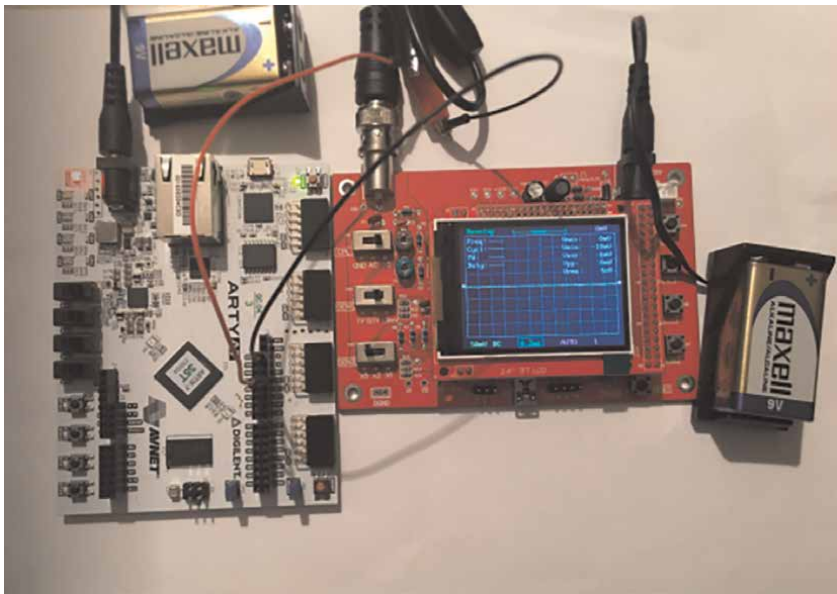


Figure 8. *Arty Artix-7 FPGA development board.*

encryption alters the image's appearance. **Figure 8** presents the Arty Artix-7 FPGA Development Board, which is likely the hardware platform used for implementing the RSA algorithm. This suggests that FPGA is used to accelerate the encryption and decryption processes in the experiment. Lastly, **Figure 9** shows the results of

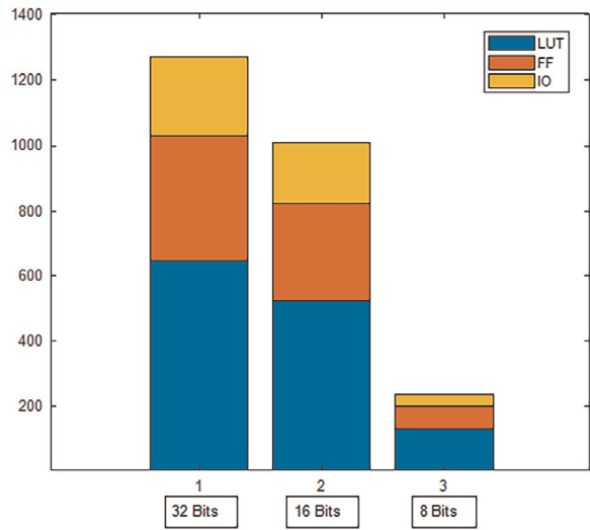


Figure 9.
Results of the implementation of RSA on 8, 16, and 32 bits.

implementing RSA on 8, 16, and 32 bits, likely comparing the performance, resource usage, or power consumption of RSA for different bit lengths, thereby illustrating the impact of bit size on the efficiency of RSA encryption and decryption operations.

In that session, we have formed the structure, which will be used or implemented for RSA Cryptography algorithm based on the Montgomery's algorithm and the modular exponentiation and we have defined the different blocks (stages) of RSA algorithm by VHDL language.

The resulting architectures for the blocks are efficient since we have followed a graphical mode which allows to cut some sub-blocks and an important function that has reduced the number of the programmable logical cells of the architecture of the algorithm.

reprogrammable logic cells in the FPGA system.

And at last, we can conclude that this architecture can be reformed to be suitable for more powerful hardware architecture that able to encrypt larger message.

7. Conclusion

A full hardware realization of the RSA cryptosystem was implemented in hardware on an Artix-7 FPGA in this chapter using Montgomery multiplication and modular exponentiation to improve the performance. The proposed structure in VHDL offers a possible method to obtain efficient and secure encryption for sensitive data, specifically in the case of medical images. The results also demonstrate that implementation of optimal arithmetic blocks into the FPGA brings a powerful solution to embedded systems that need to perform fast and secure protection of data [63]. Additionally, the design is both modular and flexible, with capabilities to accommodate different messages lengths such as in large medical images. While the performance of the architecture is adequate, refinements can be explored in the future including the enlarging the key size, the further strengthening of side-channel defenses and the design in ASCIC

for faster performance and better hardware efficiency. In summary, this research emphasizes the importance of hardware-based cryptographic approaches in the era of data security, particularly in healthcare industry.

Acknowledgements

The authors would like to thank all those who contributed to the development of this work. The authors also acknowledge the use of *ChatGPT (OpenAI)* for language refinement during the preparation of this manuscript.

Author details

Ridha Ghayoula^{1*}, Sahbi Bahroun², Wided Amara³, Jaouhar Fattahi⁴, Amor Smida⁵, Issam El Gmati⁶ and Waly Mohamed I⁵

1 Faculty of Engineering, University of Moncton, New Brunswick, Canada

2 Institut Supérieur d'Informatique de Tunis (ISI), Tunis, Tunisia

3 SysCom Laboratory, ENIT, University of Tunis El Manar, Tunis, Tunisia

4 Department of Computer Science and Software Engineering, Laval University, Quebec, Canada

5 Department of Medical Equipment Technology, College of Applied Medical Sciences, Al-Majmaah University, Al Majma'ah, Saudi Arabia

6 College of Engineering, Umm Al-Qura University, Al Gunfudha, Makkah, Saudi Arabia

*Address all correspondence to: ridha.ghayoula@umoncton.ca

IntechOpen

© 2025 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Diffie W, Hellman M. New directions in cryptography. *IEEE Transactions on Information Theory*. 1976;22(6):644-654. DOI: 10.1109/79.526899
- [2] Diffie W, Hellman ME. Multiuser cryptographic techniques. In: *AFIPS'76, National Computer Conference*, New York, New York, NY, USA: ACM (Association for Computing Machinery)/AFIPS; 1976. pp. 109-112. DOI: 10.1145/1499799.1499815
- [3] Rivest RL, Shamir A, Adlema L. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*. 1978;21(2):96-99. DOI: 10.1145/357980.358017
- [4] Sahu SK, Pradhan M. FPGA implementation of RSA encryption system. *International Journal of Computer Applications*. 2011;19(9):10-12. Available from: <https://citeseerx.ist.psu.edu/viewdoc/download?rep=rep1&type=pdf&doi=10.1.1.206.3453>
- [5] Al-Haija QA, Smadi M, Al-Ja'Fari M, Al-Shua'Ibi A. Efficient FPGA implementation of RSA coprocessor using scalable modules. *Procedia Computer Science*. 2014;34:647-654. DOI: 10.1016/j.procs.2014.07.092
- [6] Saini S, Lata K, Sharma A, Sinha GR. An FPGA implementation of the RSA algorithm using VHDL and a Xilinx system generator for image applications. *IOP Science*. 2021;1:1-29. Available from: <https://iopscience.iop.org/book/edit/978-0-7503-3919-3/chapter/bk978-0-7503-3919-3ch1.pdf>
- [7] Gaithuru JN, Bakhtiari M, Salleh M, Muteb AM. A comprehensive literature review of asymmetric key cryptography algorithms. In: *9th Malaysian Software Engineering Conf. (MySEC)*. Piscataway, NJ, USA: IEEE; 2015. pp. 236-244. Available from: <https://ieeexplore.ieee.org/abstract/document/7475227>
- [8] Gurpreet K, Vishal A. An efficient implementation of RSA algorithm using FPGA and big prime digit. *International Journal of Computer & Communication Engineering Research (IJCCER)*. 2013;1(4). Available from: <https://www.ijettjournal.org/archive/ijett-v68i1p204>
- [9] Shehata K, Hussien H, Yehia S. FPGA implementation of RSA encryption algorithm for E-passport application. In: *The International Conference on Electrical Engineering*. Vol. 1. Egypt: EKB Journals/Egyptian Knowledge Bank; 2014. p. 136. Available from: https://journals.ekb.eg/article_30363.html
- [10] Karakra A, Alsadeh A. A-RSA: Augmented RSA. In: *SAI Computing Conference (SAI)*, London, UK. Piscataway, NJ, USA: IEEE; 2016. DOI: 10.1109/SAI.2016.7556103
- [11] Musa U, Adebisi MO, Adigun O, Adebisi AA, Aremu CO. Hybrid cloud storage techniques using RSA and ECC. In: *International Conference on Science, Engineering and Business for Sustainable Development Goals (SEB-SDG)*. Piscataway, NJ, USA: IEEE; 2023. DOI: 10.1109/SEB-SDG57117.2023.10124559
- [12] Gandhi T, Navlakha M. Enhanced RSA cryptosystem. In: *5th International Conference on Advances in Science and Technology (ICAST)*, Mumbai, India. Piscataway, NJ, USA: IEEE; 2022. Available from: <https://ieeexplore.ieee.org/document/10039627>

- [13] Boneh D, Shaw J. Collusion-secure fingerprinting for digital data. *IEEE Transactions on Information Theory*. 1998;**44**:1897-1905. Available from: <https://ieeexplore.ieee.org/abstract/document/705568>
- [14] Kocher PC. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In: *Advances in Cryptology — CRYPTO'96*. Berlin/Heidelberg, Germany: Springer; 2001. pp. 104-113. Available from: https://link.springer.com/chapter/10.1007/3-540-68697-5_9
- [15] Shor PW. Algorithms for quantum computation: Discrete logarithms and factoring. In: *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*. Los Alamitos, CA, USA: IEEE Computer Society Press; 1994. pp. 124-134. [Online]. Available from: <https://ieeexplore.ieee.org/document/365700>
- [16] Gupta P, Saini S, Lata K. Securing QR codes by RSA on FPGA. In: *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. Piscataway, NJ, USA: IEEE; 2017. DOI: 10.1109/ICACCI.2017.8126188
- [17] Yang Y, Chao W, Li Z, Yang J. Efficient FPGA implementation of modular multiplication based on Montgomery algorithm. *Microprocessors and Microsystems*. 2016;**47**:209-215. Available from: <https://www.sciencedirect.com/science/article/abs/pii/S0141933116300916>
- [18] Sharma S, Zodpe H. Implementation of cryptography algorithm for E-passport security, inventive computation technologies (ICICT). *International Conference*. 2016;**3**:1-3. Available from: <https://www.sciencedirect.com/science/article/abs/pii/S0141933116300916>
- [19] Ghoreishi SS, Pourmina MA, Bozorgi H, Dousti M. High-speed RSA implementation based on modified Booth's technique and Montgomery's multiplication for FPGA platform. In: *ICICT*. Vol. 3. Piscataway, NJ, USA: IEEE; 2009. pp. 86-93. Available from: <https://www.sciencedirect.com/science/article/abs/pii/S0141933116300916>
- [20] Abubakar A, Muhammad SJ, Idrith Tijjani B, Zeki A, Chiroma H, Raji S, et al. Cryptanalytic attacks on Rivest, Shamir, and Adleman (RSA) cryptosystem. *Journal of Theoretical and Applied Information Technology*. 2014; **66**(2):123-134. [Online]. Available from: <https://www.sciencedirect.com/science/article/abs/pii/S0141933116300916>
- [21] Thobi A, Dhage S, Jadhav P, Chandrachood A. Implementation of RSA encryption algorithm on FPGA. *American Journal of Engineering Research*. 2015;**4**:144-151. Available from: <https://www.sciencedirect.com/science/article/abs/pii/S0141933116300916>
- [22] Pavani K, Sriramya P. Enhancing public key cryptography using RSA, RSA-CRT and N-prime RSA with multiple keys. In: *Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*. Piscataway, NJ, USA: IEEE; 2021. DOI: 10.1109/ICICV50876.2021.9388621
- [23] Aggarwal D, Maurer U. Breaking RSA generically is equivalent to factoring. *IEEE Transactions on Information Theory*. 2016;**62**:6251-6259. DOI: 10.1109/TIT.2016.2594197
- [24] Guleria V, Sabir S, Mishra DC. Security of multiple RGB images by RSA cryptosystem combined with FrDCT and Arnold transform. *Journal of Information Security and Applications*.

2020;**54**:102524. DOI: 10.1016/j.jisa.2020.102524

[25] Liu Y, Shen X, Liu J, Peng K. Optical asymmetric JTC cryptosystem based on multiplication-division operation and RSA algorithm. *Optics & Laser Technology*. 2023;**160**:109042. DOI: 10.1016/j.optlastec.2022.109042

[26] Kuppuswamy P, Al-Maliki SQYAK, John R, Haseebuddin M, Meeran AAS. A hybrid encryption system for communication and financial transactions using RSA and a novel symmetric key algorithm. *Bulletin of Electrical Engineering and Informatics*. 2023;**12**:1148-1158. DOI: 10.11591/eei.v12i2.4967

[27] Sharma S, Zodpe H. FPGA implementation of RSA cryptosystem. In: *International Conference on Inventive Computation Technologies (ICICT)*. Vol. 3. Piscataway, NJ, USA: IEEE; 2016. pp. 1-3

[28] Shehata K, Hussien H, Yehia S. FPGA implementation of RSA encryption algorithm for E-passport. In: *International Conference on Electrical Engineering (ICEENG)*. Piscataway, NJ, USA: IEEE; 2014. pp. 1-5

[29] Bo S, Kawakami K, Nakano K, Ito Y. An RSA encryption hardware algorithm using a single DSP block and a single block RAM on the FPGA. *International Journal of Networking and Computing*. Jul 2011;**1**(2):277-289. Article en ligne

[30] Nakano K, Kawakami K, Shigemoto K. RSA encryption and decryption using the redundant 390 number system on the FPGA. In: *IEEE International Symposium on Parallel Distributed Processing*. Piscataway, NJ, USA: IEEE; 2009. pp. 1-8

[31] Liu B. Research and implementation of RSA IP Core based on FPGA. In: *Data Processing Techniques and Applications for Cyber-Physical Systems (DPTA 2019)*. Piscataway, NJ, USA: IEEE; 2020. pp. 1311-1319

[32] Öztürk E. Design and implementation of a low-latency modular multiplication algorithm. *IEEE Transactions on Circuits and Systems I: Regular Papers*. 2020;**67**:1902-1911. DOI: 10.1109/TCSI.2020.2966755

[33] Ghayoula R, Fattahi J, Smida A, Gmati I, Pricop E, Ziadia M. FPGA implementation of SIMON-128 cryptographic algorithm using Artix-7. In: *2022 14th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, Ploiesti, Romania. Piscataway, NJ, USA: IEEE; 2022

[34] Ghayoula R, Smida A, Fattahi J, Ghayoula E. FPGA implementation of the RC4 stream Cipher using Artix-7. *International Journal of Embedded Systems and Emerging Technologies*. 2018;**4**(2)

[35] Huffmire T, Irvine C, Nguyen TD, Levin T, Kastner R, Sherwood T. *FPGA Updates and Programmability*. Berlin/Heidelberg, Germany: Springer Science + Business Media; 2010. pp. 87-96. ISBN 978-90-481-9157-4

[36] Rodriguez-Henriquez F, Saqib NA, Diaz Perez A, Koc CK. *Cryptographic Algorithms on Reconfigurable Hardware*. New York, NY, USA: Springer Science + Business Media, LLC; 2006. ISBN 0-387-33883-7

[37] Angheliescu P, Ionita S, Sofron E. Encryption technique with programmable cellular automata (ETPCA). *Journal of Cellular Automata*. 2010;**5**:1557-5969

- [38] Nibouche O, Nibouche M, Bouridane A. High-speed FPGA implementation of RSA encryption algorithm. In: Electronics Circuits and Systems ICECS 2003. Vol. 1. Piscataway, NJ, USA: IEEE; 2003. pp. 204-207. ISBN 0-7803-8163-7
- [39] Iana G, Anghelescu P, Traian N, Șerban G. Optimal implementation of cryptographic algorithm on FPGA. Scientific Bulletin Series Electronics and Computers Science. 2010;2: 17-23
- [40] Prasu G, Malabika B, Biswas M. A compact FPGA implementation of triple-DES encryption system with IP Core generation on on-Chip verification. In: Proceedings of the 2010 International Conference on Industrial Engineering and Operations Management. Piscataway, NJ, USA: IEEE; 2010
- [41] Saggese GP, Romano L, Mazzocca N, Mazzeo A. A tamper-resistant hardware accelerator for RSA cryptographic applications. Systems Architecture Journal. 2004;50(12):711-727
- [42] Zhang Y, Li Z, Yang L, Zhang S. An efficient CSA architecture for Montgomery modular multiplication. Microprocessors and Microsystems Journal. 2007;31(7):456-459
- [43] Kitsos P, Sklavos N, Galanis MD, Koufopavlou O. 64-bit block ciphers: Hardware implementations and comparison analysis. Computers and Electrical Engineering. 2004;30(8):593-604. Available from: https://www.sciencedirect.com/science/article/abs/pii/S0045790605000108?utm_source=chatgpt.com
- [44] Ghayoula R, Hajlaoui EA, Korkobi T, Traii M. FPGA implementation of RSA cryptosystem, international. Journal of Electrical and Computer Engineering. 2008;2:1667-1671. DOI: 10.1109/ISITIA.2015.7219961
- [45] Abdmouleh MK, Khalfallah A, Bouhlel MS. A novel selective encryption scheme for medical images transmission based on JPEG compression algorithm. Procedia Computer Science. 2017;112: 369-376
- [46] Xu G, Hoe JC, Franchetti F. Flexible hardware accelerator design generation with SPIRAL. In: 2022 IEEE High-Performance Extreme Computing Conference (HPEC). Piscataway, NJ, USA: IEEE; 2022. pp. 1-7
- [47] Bukhari K, Kuzmanov G, Vassiliadis S. DCT and IDCT implementations on different FPGA technologies. In: Proceedings of ProRISC. Veldhoven, Netherlands: ProRISC/IEEE; 2002. pp. 232-235
- [48] Huang J, Parris M, Lee J, Demara RF. Scalable FPGA-based architecture for DCT computation using dynamic partial reconfiguration. ACM Transactions on Embedded Computing Systems (TECS). 2009;9(1):1-18. DOI: insert-doi-here
- [49] Mazzeo A, Romano L, Saggese GP, Mazzocca N. FPGA-based implementation of a serial RSA processor. In: 2003 Design Automation and Test in Europe Conference and Exhibition. Piscataway, NJ, USA: IEEE; 2003. pp. 582-587
- [50] Oksuzoglu E, Savas E. Parametric secure and compact implementation of RSA on FPGA. In: 2008 International Conference on Reconfigurable Computing and FPGAs. Piscataway, NJ, USA: IEEE; 2008. pp. 391-396
- [51] Tahir AS. Design and implementation of the RSA algorithm using FPGA. International Journal of Computers & Technology. 2015;14(12): 6361-6367. DOI: insert-doi-here

- [52] Chandu Y, Kumar KR, Prabhukhanolkar NV, Anish A, Rawal S. Design and implementation of hybrid encryption for security of IoT data. In: 2017 International Conference on Smart Technologies for Smart Nation (SmartTechCon). Piscataway, NJ, USA: IEEE; 2017. pp. 1228-1231
- [53] Crockett LH, Elliot RA, Enderwitz MA, Stewart RW. The Zynq Book: Embedded Processing with the ARM Cortex-A9 on the Xilinx Zynq-7000 all Programmable SoC. Glasgow, Scotland, UK: Strathclyde Academic Media; 2014
- [54] Modular Exponentiation, [online] Available from: <https://www.geeksforgeeks.org/modular-exponentiation-power-in-modular-arithmetic/>.
- [55] Butler JT, Sasao T. Fast hardware computation of $x \bmod z$. In: 2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and PhD Forum. Piscataway, NJ, USA: IEEE; 2011. pp. 294-297
- [56] Chung J, Hasan MA. Montgomery reduction algorithm for modular multiplication using low-weight polynomial form integers. In: 18th IEEE Symposium on Computer Arithmetic (ARITH'07). Piscataway, NJ, USA: IEEE; 2007. pp. 230-239. Available from: <https://ieeexplore.ieee.org/document/4272870>
- [57] Shan R, Di S, Calhoun JC, Cappello F. Towards combining error-bounded lossy compression and cryptography for scientific data. In: 2021 IEEE High-Performance Extreme Computing Conference (HPEC). Piscataway, NJ, USA: IEEE; 2021. pp. 1-7. Available from: <https://ieeexplore.ieee.org/document/9622874>
- [58] Liu W, Zeng W, Dong L, Yao Q. Efficient compression of encrypted grayscale images. IEEE Transactions on Image Processing. 2009;19(4):1097-1102
- [59] Shan R, Di S, Calhoun JC, Cappello F. Exploring lightweight cryptography for efficient and secure lossy data compression. In: 2022 IEEE International Conference on Cluster Computing (CLUSTER). Piscataway, NJ, USA: IEEE; 2022. pp. 23-34. Available from: <https://ieeexplore.ieee.org/document/9912696>
- [60] Su Z, Di S, Gok AM, Cheng Y, Cappello F. Understanding impact of lossy compression on derivative-related metrics in scientific datasets. In: 2022 IEEE/IACM 8th International Workshop on Data Analysis and Reduction for Big Scientific Data (DRBSD). Piscataway, NJ, USA: IEEE; 2022. pp. 44-53
- [61] Khalil M, Ahmad I. A Customizable Cryptographic Service Provider for an Embedded Cryptohardware System [M. Sc. Thesis]. Johor, Malaysia: Universiti Teknologi Malaysia (UTM); 2007. Available from: [http://portal.utm.my/client/en_AU/main/search/detailnonmodal/ent:\\$002f\\$002fSD_ILS\\$002f0\\$002fSD_ILS:142096/one?qu=Illiasaak+Ahmad%2C&qf=ITYPE%09Material+Type%091%3ACLS_THESIS%09Closed+Access+Thesis](http://portal.utm.my/client/en_AU/main/search/detailnonmodal/ent:$002f$002fSD_ILS$002f0$002fSD_ILS:142096/one?qu=Illiasaak+Ahmad%2C&qf=ITYPE%09Material+Type%091%3ACLS_THESIS%09Closed+Access+Thesis)
- [62] Recommendation For Key Management – Part 1 (Revised), NIST, 2007
- [63] Subhayan S, Iqbal HSK, Kabirul I, Dipanwita RC, Parimal Pal C. Cryptosystem designed for embedded system security. In: Proceedings of the 16th International Conference on VLSI Design (VLSI'03); New Delhi, India. Piscataway, NJ, USA: IEEE; 4-8 Jan 2003. pp. 271-276. Available from: https://dl.acm.org/doi/10.5555/832285.835526?utm_source=chatgpt.com

Section 3

Future Security and Emerging Paradigms

Perspective Chapter: Postquantum Security

Satyanarayana Degala

Abstract

The advent of quantum computing technology carries a new evolution of computer technology called quantum computers, which are quite different from the currently available computers on the market. These computers have extensive computing capabilities; for instance, the quantum computers have one million times higher computing capacity than the currently available computers in the market. On the other hand, these quantum computers give different challenges to the online digital currency systems, precisely on banking financial transactions. The cryptographic algorithms used in financial transactions always assume that the current computing capabilities cannot decipher secret information with the presence of hard mathematical problems, such as integer factorization, discrete logarithm problem, and periodic finding problem. However, quantum computers can exceed the computing capabilities to decrypt secret codes and information. Hence, it is very important to discuss and review the postquantum security issues and developments. This book chapter presents information on the postquantum security mechanisms.

Keywords: quantum computing, quantum computers, network security, cryptography, postquantum security

1. Introduction

A new dimension of computer architecture in the evolution of computer technology is the quantum computers, which are quite different from the currently available computer systems in different aspects. These aspects include computing capabilities, system architecture, operating management, system interfaces, communication systems, and security systems. For example, the computing capabilities of quantum computers are 1 million times higher than the current super computers [1]. These computers are still under development, and it may take several years to come to the commercial market for public use. Even though these computers provide high computing capabilities, they pose many challenges to the present public service systems, significantly, online digital currency transactions, banking security systems, government service systems, and data centers. The present cryptographic algorithms are safe because the currently available computers cannot decipher secret information while doing the transactions. The mathematical problems associated with the current cryptosystems are intractable. Examples of such mathematical problems include

integer prime factorization, elliptic curve problem, and discrete logarithmic problem [2, 3]. Shor’s algorithm proved that the integer factorization problem can be solved with the time $O\left((\log n)^2(\log \log n)(\log \log \log n)\right)$ on quantum computers with $O(\ln \log n)$ time for the postprocessing task on nonquantum computers [2], which give a threat to Rivest-Shamir-Adleman (RSA) algorithm and its dependent subsystems. The quantum computers, in the near future, can easily break cryptographic keys and decipher secret information, which is dangerous for many online transactions and business applications. Hence, it is very important to review the postquantum security mechanism. This chapter presents information on quantum computers and their computing capabilities, various security issues, and postquantum cryptographic methods.

The remaining chapter is organized as follows: Section 2 describes the quantum computers and the technical specifications of quantum computers. Section 3 describes information security issues and quantum security concerns. Section 4 describes hard mathematical problems associated with cryptographic algorithms. Section 5 presents different types of postquantum cryptographic algorithms. Finally, Section 6 concludes the book chapter.

2. Quantum computers

There are different models of computer systems in the evolution of computer technologies starting from the first generation in 1940 to the present fifth generation computer system till date. **Table 1** shows details of various generations of computer systems [4].

The latest generation of computer technology is the fifth generation, which focuses on using artificial intelligence and associated technologies to add intelligence to computer systems. The sixth generation of computers is expected sooner by adopting the high processing capacity to the future computer systems. The sixth-generation computers are going to use quantum computing technology and nanotechnology for future-generation computing needs.

For the last few decades, digital computers have used binary bits for tasks in the computer system. However, quantum computers use qubits or quantum bits for computing tasks. The binary bits are either “1” or “0” at any given time. However, the qubits are considered to exist at same time by “1” and “0”. **Figure 1** shows the quantum bit (qubit). The basic properties of quantum computing are superposition and quantum entanglement. The superposition depicts that there will be multiple states between bit “0” and “1”, rather than two states either “0” or “1” as in the

Generation	Technology evolution	Time period
First generation	Vacuum tube based	1940–1950
Second generation	Transistor-based technology	1950–1960
Third generation	Integrated circuit–based technology	1960–1970
Fourth generation	Microprocessor technology	1970s–present
Fifth generation	AI-based technology	Present to future

Table 1.
Computer generations.

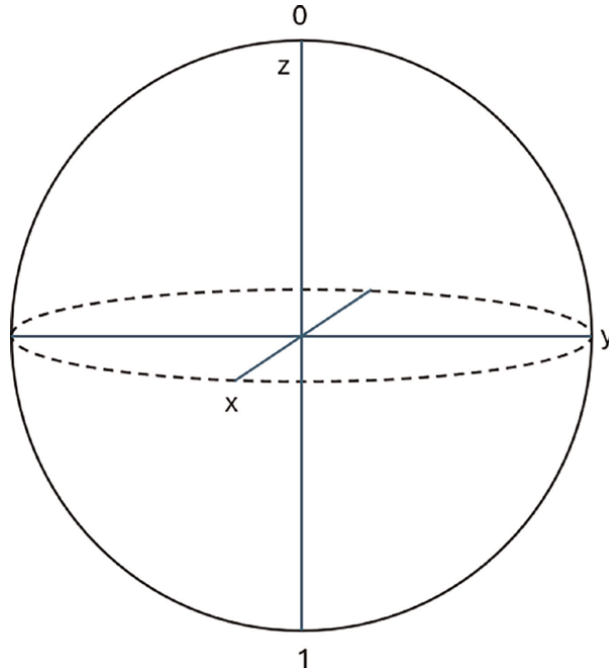


Figure 1.
 Quantum bit.

classical computers. Hence, it increases the parallelism of computations with qubits. The other basic property of quantum computing is the information flow between qubits, which is called quantum entanglement. Some of the operations that relate to quantum entanglement are hypodense coding, teleportation, and quantum cryptography.

The quantum computers, with the sufficient number of qubits, solve many complex problems, which cannot be solved by classical computers. For example, quantum computers can solve the integer factorization problem with a reasonable amount of time.

There are three types of quantum computers: quantum annealer, analog quantum computers, and universal quantum computers [5]. Among the three, universal quantum computers (UAC) are the most powerful, whereas the quantum annealer is the lowest powerful than the other two. The quantum annealer is useful to solve many optimization problems, such as machine learning, and optimization algorithms. These computers can work on specific functions. From the scientific community point of view, the quantum annealer does not bring many benefits than high power classical computers, such as supercomputer [6]. Analog quantum computers can simulate complex quantum problems, such as material science simulations, quantum chemistry, and quantum dynamics. The number of qubits used for the analog quantum computers is between 50 and 100 bits [6]. Universal quantum computers have higher power capability with applications in science and business areas. These quantum computers can be used for secure computing, cryptography, machine learning, and search engines. The estimated physical qubits for universal quantum computing can be 100,000 at present [6]. **Table 2** shows the information on types of quantum computers. **Table 3** shows some of the popular manufacturers of the quantum processor.

Types of quantum computers	Number of qubits	Computation power	Applications
Quantum annealer	512 (currently)	Low powerful	• Optimization problems
Analog quantum computer	256 (currently)	High powerful	• Sampling • Material science • Quantum chemistry
Universal quantum computers	Current estimation 100,000	Most powerful	• Cryptography • Secure computing • Machine learning

Table 2.
Types of quantum computers.

Manufacturer	Name of QC	Number of qubits	Release date	Type of QC
Google	Sycamore	54	2019	UAC
IBM	IBM Condor	1121	2023	UAC
D-Wave	D-Wave Advantage	5760	2020	Quantum annealer
Xanadu	X24	24	2020	UAC
SaxonQ	Princess	4	2024	UAC
Rigetti	Ankaa-2	84	2024	UAC
QunatWare	Tenor	64	2023	UAC
QuEra	Aquila	256	2022	Analog quantum

Table 3.
Quantum processor manufacturers.

3. Information and quantum security

The advancements in quantum computing technologies have led to many challenges to the currently available data storages and network security systems. For example, with the introduction of qubits in quantum computing, many traditional security algorithms [7], and cybersecurity attacks [8], got a threat that they can be solved easily with a reasonable amount of time. In 1986, Shor’s algorithm proved that it could break the integer factorization problems with sufficient number of qubits, which brings a threat to many cryptographic algorithms, especially the asynchronous or public key cryptographic algorithms. The public key cryptographic algorithms are mainly attacked by man in the middle (MITM) [9]. Cryptography plays an important role in network security. There are several application areas for cryptography, such as e-commerce, where a person can order products from remote places and the data is transferred over the network. The banking transactions are another application area, where the cryptography is abundantly used. The cryptography algorithms are of two types: symmetric key cryptography and asymmetric key cryptography. The symmetric key cryptography uses a shared key, whereas the asymmetric key cryptosystems use two keys: the public key and the private key. The symmetric key cryptography uses the same key for the encryption and decryption processes. The cryptography that uses the shared key has the advantage of defending the various attacks such as known plain text (KPT) [10], chosen plain text attack (CPA) [11], and linear cryptanalysis [12, 13]. The hash function-based encryption and decryption processes are also

Cryptography	
Public key cryptography	Symmetric key cryptography
• Encryption-based cryptography ◦ Example: RSA • Key exchange-based cryptography ◦ Example: Diffie-Hellman • Digital signature-based cryptography ◦ Example: RSA digital signature	• Secret key-based cryptography ◦ Example: Block cipher • Hash function-based cryptography ◦ Example: SHA1

Table 4.
Classification of cryptographic algorithms.

Mathematical hard problems	Public key cryptography methods
Integer factorization problem	• RSA encryption • RSA digital signature
Discrete logarithm problem	• ElGamal encryption • Diffie-Hellman key exchange • EiGamal digital signature
Elliptic curve problem	• Elliptic curve encryption • Diffie-Hellman key exchange • EiGamal digital signature
Lattice-based problems	• NTRU encryption • NTRU digital signature • Ring LWE

Table 5.
Mathematical problems used in public key cryptography.

considered symmetric cryptography [7]. The public key cryptography algorithms are broadly classified into three categories: encryption-based, digital signature-based, and key exchange-based algorithms. This public key cryptography depends on difficulty of solving mathematical problems, such as integer prime factorization, which is extensively used in RSA-based cryptosystems. **Table 4** shows classification of cryptographic algorithm types and their respective problems. **Table 5** shows various mathematical problems, and their associated algorithms used for the public key cryptography.

4. Shor’s algorithm

Many cryptosystems depend on the fact that certain problems, such as discrete logarithm, integer factorization, and periodic finding problem, cannot be solved by classical computers. However, in quantum computers, the computational speedup is increased, and these complex mathematical problems can be solved in a reasonable amount of time. In 1994, Peter Shor proposed a quantum computing algorithm that can be utilized to break many asynchronous cryptography algorithm schemes, such as Rivest-Shamir-Adleman, the elliptic curve Diffie-Hellman key exchange, and the finite field Diffie-Hellman key exchange [2]. The RSA algorithm relied on circumstance that the integer factorization problem does not have polynomial time solution for larger values of inputs, which makes the problem computationally intractable.

Shor's algorithm presents a quantum algorithm that can solve a prime factorization problem, which takes asymptotically $O((\log n)^2(\log \log n)(\log \log \log n))$ time on quantum computers with $O(\ln \log n)$ time for the postprocessing task on nonquantum computers to transform the output of quantum computers to factors of n .

The quantum computer model, which is considered by Shor's algorithm, contains two important components: quantum gate arrays and quantum acyclic circuits. In order to describe the quantum state, it needs $2^n - 1$ complex numbers to represent, where "n" is the number of bits required. The superposition of a state is represented as

$$\sum_{i=0}^{2^n-1} a_i |S_i\rangle \quad (1)$$

where a_i is the complex number and $|S_i\rangle$ is a base vector of Hilbert space. Here, the notation $|x\rangle$ represents the pure quantum state. It is also needed to note that $\sum_i |a_i|^2 = 1$ for Eq. (1).

According to Shor's algorithm, a quantum gate can be expressed in terms of truth tables. For example, one gate is as follows [2]:

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle \\ |00\rangle &\rightarrow |00\rangle \\ |10\rangle &\rightarrow \frac{1}{\sqrt{2}}(|10\rangle + |11\rangle) \\ |11\rangle &\rightarrow \frac{1}{\sqrt{2}}(|10\rangle - |11\rangle) \end{aligned} \quad (2)$$

The same gate can also be represented in the matrix form. The matrix form for the above-mentioned gate is as follows:

$$\begin{array}{cccc} & |00\rangle & |00\rangle & |00\rangle & |00\rangle \\ |00\rangle & |1 & 0 & 0 & 0| \\ |00\rangle & |1 & 1 & 0 & 0| \\ |00\rangle & |1 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}| \\ |00\rangle & |1 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}| \end{array} \quad (3)$$

In the above matrix, the rows represent the input base vector, where the columns represent the output base vector. The entry (x, y) in the matrix describes that x base vector is input to the gate and then the corresponding coefficient y base vector is the output gate. The quantum gate assumes that the matrix is unitary; that is, matrix A is equal to the unitary if and only if

$$A^* = A^{-1} \quad (4)$$

where A^* is the transpose of matrix A and the A^{-1} is the inverse matrix of A.

Another important point is the number of qubits required to solve the integer factorization problem using Shor's algorithm. There are two types of qubits in quantum computing: logical qubits and physical qubits [14]. Most of the current manufacturing companies of quantum computers use physical qubits. The physical qubits are noisy, and they are prone to errors. A group of physical qubits are connected to each other to make sure that it can provide error corrections so that the system is *fault tolerant*. These kinds of qubits are called *logical qubits* or *perfect qubits*. There is an estimate that around 1000 physical qubits represent one logical qubit [15]. A researcher Stephane Beauregard estimated that $2n + 3$ logical qubits are sufficient to solve a k -bit integer factorization using Shor's algorithm [16]. In addition, it is also mentioned that a 2048-bit number needs 4099 logical qubits, which are approximately 4-million physical qubits.

Quantum Fourier transformation (QFT) is used for the implementation of Shor's algorithm in finding the integer prime factorization with the polynomial time [2]. This transformation of quantum computing is typically as discrete Fourier transformation (DFT). It can be represented as a matrix with a certain number of rows and columns indexed by quantum states. QFT is crucial part of Shor's algorithm.

If you consider a number "a" for some value "q" such that $0 \leq a < q$, then the transformation of a state $|a\rangle$ to another state is represented as

$$\frac{1}{q^{1/2}} \sum_{c=0}^{q-1} |c\rangle \exp\left(\frac{2\pi iac}{q}\right) \quad (5)$$

The entry of (a, c) in unitary matrix A_q is $\frac{1}{q^{1/2}} \exp(2\pi iac/q)$.

Shor's algorithm considers a simple construction of A_q with q as the power of 2. If $q = 2^l$, then the binary form of an integer "a" is $|a_{l-1}, a_{l-2}, \dots, a_0\rangle$.

The quantum Fourier transformation needs two types of gates: R_j and S_{jk} [2].

$$R_j = \begin{matrix} & |0\rangle & |1\rangle \\ |0\rangle & \begin{vmatrix} 1 & 1 \\ \sqrt{2} & \sqrt{2} \end{vmatrix} \\ |1\rangle & \begin{vmatrix} 1 & -1 \\ \sqrt{2} & \sqrt{2} \end{vmatrix} \end{matrix} \quad (6)$$

$S_{j,k}$ operates on the bit location at j and k.

$$S_{j,k} = \begin{matrix} & |00\rangle & |00\rangle & |00\rangle & |00\rangle \\ |00\rangle & |1 & 0 & 0 & 0| \\ |00\rangle & |0 & 1 & 0 & 0| \\ |00\rangle & |0 & 0 & 1 & 0| \\ |00\rangle & |0 & 0 & 0 & e^{i\theta_{k-j}}| \end{matrix} \quad (7)$$

Here, $\theta_{k-j} = \pi/2^{-j}$.

QFT is performed with the matrices from left to right order.

$$R_{l-1}S_{l-2,l-1}R_{l-2}S_{l-3,l-1}S_{l-3,l-2}R_{l-3} \dots R_1S_{0,l-1}S_{0,l-2} \dots S_{0,2}S_{0,1}R_0 \quad (8)$$

Note that the gates are applied for R_j in the reverse order R_{l-1} to R_0 . For $S_{j,k}$ gate, all the gates of $S_{j,k}$ are applied, which are between R_{j+1} and R_j and $k > j$.

After the sequence of transformations, the obtained quantum state is $\frac{1}{q^{1/2}} \sum_b \exp(2\pi i a c / q) |b\rangle$. Here, b represents the bit reversal of c .

In R matrix, the factors $\frac{1}{\sqrt{2}}$ multiply to bring finally to $\frac{1}{q^{1/2}} \cdot S_{j,k}$ does not change the bit value; instead, it changes their phase. The phase on the path $|a\rangle$ to $|b\rangle$ is [2]

$$\sum_{0 \leq j < l} \pi a_j b_j + \sum_{0 \leq j < k < l} \frac{\pi}{2^{k-j}} a_j b_k \quad (9)$$

5. Postquantum cryptography

The National Institute of Standards and Technology (NIST) initiated research, and they called for many rounds of research papers on postquantum cryptography methods. **Table 6** shows the information on the rounds and the papers. These research methods can be broadly classified as four groups from mathematical perspectives: the lattice-based cryptography algorithms, code-based cryptography algorithms, multivariate-based cryptography algorithms, and the hash-based cryptography algorithms [17].

5.1 Lattice-based cryptography

The lattice-based cryptography plays an important role in the postquantum security as the lattice problems are hard enough to resist the quantum computing methods. The lattice-based cryptography can achieve desired security levels for postquantum cryptography in the form of public and private key encryptions, digital signatures, key exchanges, homomorphic encryptions, and attribute encryptions [18]. Hence, the postquantum security considers lattice-based cryptography as a promising quantum-resistant security methodology.

The lattice is familiar in mathematics, and it can be modeled for various tasks of complex problems. The lattice can be used to provide the fault tolerance to the cryptosystems from the quantum computers. The lattice can be defined as [19].

Definition (lattice): the lattice L is defined as that L is generated from n linearly independent vectors $b_1, b_2, b_3, \dots, b_n$

$$L(B) = L(b_1, b_2, b_3, \dots, b_n) = \left\{ \sum_{i=1}^n a_i b_i \mid a_i \in \mathbb{Z} \right\} \quad (10)$$

Round number	Number of drafts/filtered algorithms	Year of commencement
Round 1	Received 82 draft algorithms.	2017
Round 2	26 algorithms are shortlisted.	2019
Round 3	15 drafts included.	2020
Round 4	4 algorithms are included.	2022

Table 6.
NIST postquantum algorithm screenings.

here, $b_1, b_2, b_3, \dots, b_n$ is called the base vector of the lattice $L(B)$. The B represents $m \times n$ matrix, where the columns are $b_1, b_2, b_3, \dots, b_n$ and then the lattice can be

$$L(B) = L(b_1, b_2, b_3, \dots, b_n) = \{Bx | x \in \mathbb{Z}^n\} \quad (11)$$

here, “ m ” is called dimension of the lattice and “ n ” is called the rank of the lattice. The lattice is called full rank lattice if $m = n$. The shortest distance function λ is defined as the length of shortest nonzero vector S

$$\lambda = \min(\|S\| | S \in L, S \neq 0) \quad (12)$$

Figure 2 shows a two dimensional lattice generated by a basis $\{(1,1), (1,-1)\}$.

The cryptosystems are more secure provided that the lattice problems need to be harder. Some of the well-known computational problems on lattice are shortest vector problem (SVP) [20], closest vector problem (CVP) [21], shortest independent vector problem (SIVP) [22], and bounded distance decoding problem (BDDP) [23]. The cryptosystems Ajtai-Dwork cryptosystem [24], learning with error (LWE) cryptosystems [25, 26], and N-th degree Truncated polynomial Ring Units (NTRU) [27] are some of the promising lattice-based cryptosystems for quantum computer attacks [28].

5.2 Code-based cryptography

The postquantum cryptography should be able to secure data in the cryptosystem from the quantum computer attacks. The code-based algebraic structures can resist quantum computer attacks as the subsystems of the code-based algebraic structures can increase the hardness as much as NP problems. The McEliece cryptosystem is the first code-based cryptography system that can sustain quantum computer attacks. The decoding of a linear code is an NP problem without the knowledge of algebraic structure [29, 30]. In addition, the distinguishability of scrambled parity check matrix and random matrix is also NP problem [30–32].

The first version of the cryptosystem was introduced in 1978 by McEliece [33]; later, the dual version of cryptosystem was introduced in 1986 and named as

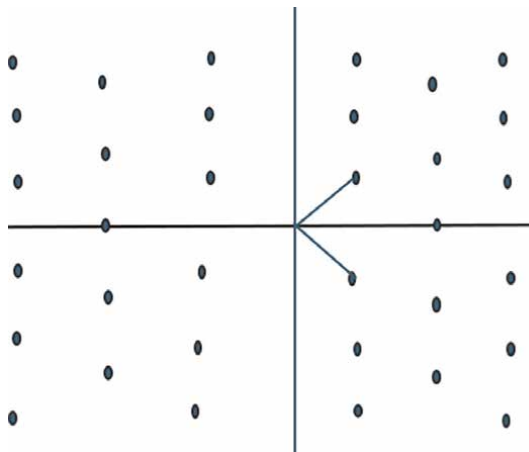


Figure 2.
 A two-dimensional lattice space.

Niederreiter cryptosystem [34]. The later version is faster, and it is used for digital signatures [30]. The encryption and decryption of McEliece cryptosystem are as follows [30]:

For a given message “ m ”, the encoded message c is:

$$c = mSGP \quad (13)$$

where S is scrambled matrix, G is the generator matrix, and P is the permutation matrix.

Using random vector ‘ e ’ of length ‘ n ’, it flips some of the bits of ‘ c ’. Here, $w(e) \leq t$, where $w(.)$ represents Hamming weight, and t is the error-correcting capability of the code. The encrypted message is

$$c^1 = c + e = mSGP + e \quad (14)$$

The decryption of the algorithm for McEliece cryptosystem is as follows:

$$c^1 P^{-1} = (mSGP + e) P^{-1} = mSG + eP^{-1} \quad (15)$$

where P^{-1} is the secret key.

Later, the decoding of $c^1 P^{-1}$ is done to obtain mS . Finally, the original message is obtained as follows:

$$m = mS * S^{-1} \quad (16)$$

In the Niederreiter cryptosystem, the encryption process is as follows:

$$c^1 = SHPm^T \quad (17)$$

Here, H is the parity check matrix. The decryption process of the Niederreiter cryptosystem is as follows [34]:

- $S^{-1}c^1 = HPm^T$
- Use syndrome decoding to obtain Pm^T
- $m = P^{-1} * Pm^T$

where m is the plain text.

5.3 Multivariate polynomial: Based cryptography

Another category of quantum-safe cryptographic algorithms are multivariate polynomial public key (MPPK) algorithms. The criticalness of the security for these algorithms does not depend on the discrete logarithm problem, or integer prime factorization problem; instead, it relates NP completeness of the Diophantine equation problem [35]. A Diophantine equation is a polynomial equation, which has two or more unknown integer coefficients, and the solution of these coefficients are interested in only integers. The private key of MPPK contains multivariate polynomials.

Cryptosystem mechanisms, such as encryptions, digital signatures, and key exchanges, use a set of multivariate polynomials over the finite fields, and it can increase the difficulty for classical computers as well as quantum computers by utilizing a high degree of nonlinear polynomial equations.

Kuang et al. used the multivariate polynomial for quantum-resistant digital signature schemes [36]. The digital signature contains three important tasks: key generation process, signing process, and the signature-verifying process. The key generation process uses the following multivariate-based polynomial [35, 36]:

$$\beta(y_0, y_1, y_2, \dots, y_n) = \sum_{i=0}^n \beta(y_0, y_1, y_2, \dots, y_n) y_0^i = \sum_{i=0}^n \sum_{j=1}^m c_{ij} y_j^i \quad (18)$$

where “n” and “m” are positive integers.

The private key contains two univariate polynomials, which can be represented as

$$\begin{aligned} f(y_0) &= f_0 + f_1 y_0 + \dots + f_\lambda y_0^\lambda \\ h(y_0) &= h_0 + h_1 y_0 + \dots + h_\lambda y_0^\lambda \end{aligned} \quad (19)$$

where λ is a positive integer and $f(y_0)$ and $h(y_0)$ are the two univariate polynomials.

The public key contains the two polynomials $P(y_0, y_1, y_2, \dots, y_n)$ and $Q(y_0, y_1, y_2, \dots, y_n)$ as shown below:

$$P(y_0, y_1, y_2, \dots, y_n) = R_0(\phi(y_0, y_1, y_2, \dots, y_n) - E_\phi(y_0)) \quad (20)$$

$$Q(y_0, y_1, y_2, \dots, y_n) = R_n(\psi(y_0, y_1, y_2, \dots, y_n) - E_\psi(y_0)) \quad (21)$$

Here, ϕ and ψ are the product polynomials of univariate polynomials $f(y_0)$ and $h(y_0)$, respectively, with the base polynomial $\beta(y_0, y_1, y_2, \dots, y_n)$. R_0 and R_n are random integers.

In digital signature, two parties exist: a signer and verifier. The signer uses the private key “S” and an algorithm that produces the message digital signature $(\mu, t) = S_s(\mu)$. After the digital signature process is finished by signer, it sends the signed document over the network. The verifier receives digital signature message (μ^1, t^1) followed by verifying with public key “v” and verifying algorithm $V_v()$ to identify whether t^1 is matched to the digital signature. If there is a match, then it returns (μ^1, VALID) ; otherwise, it returns $(\mu^1, \text{INVALID})$.

5.4 Hash-based cryptography

In order to resist quantum attacks, hash-based signature algorithms are used as an alternative to the classical digital signatures [37]. Since not much work is done toward identifying the collisions in hash functions, hash-based constructions with reasonable number of bits, the cryptosystems are safe from quantum attacks. The widely used one-way hash functions, such as secure hash algorithm 1 [38] and message digest 5 [39], are no longer secure with the advancements in computing technologies. For example, in 2017, the same hash values were obtained from the experiments of secure hash algorithm 1 with two different input messages [40].

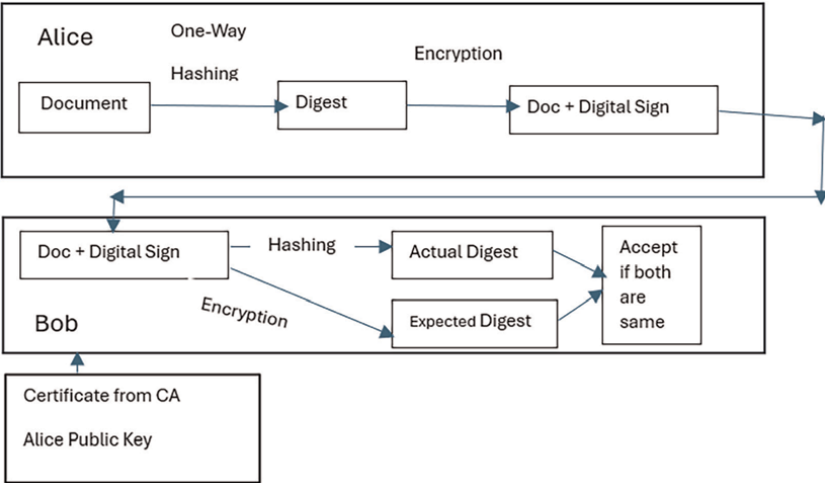


Figure 3.
Classical digital signatures.

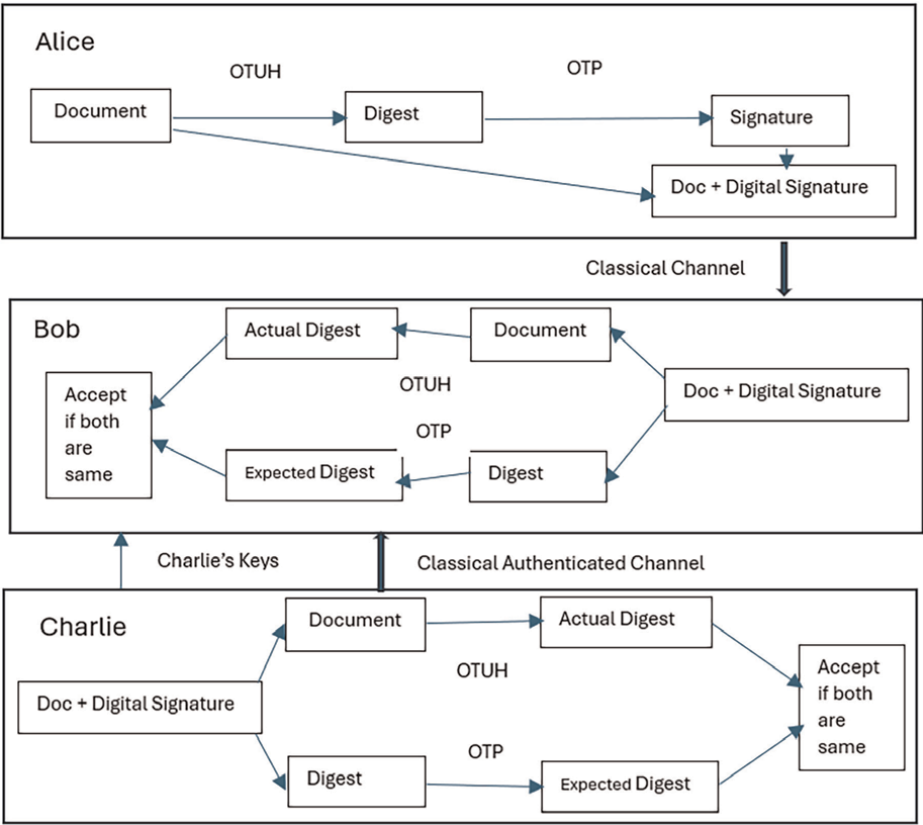


Figure 4.
QDS digital signatures.

The quantum secure communications [41] and the quantum key distributions [42–44] can only ensure confidentiality, which is one of the information security principles. However, the other information security principles that are authenticity, integrity, and nonrepudiation are not assured. On the other hand, classical digital signatures using one-way hash functions can satisfy all the four information security principles [13]; see **Figure 3** for the classical digital signature process. These one-way hash functions and public key encryption algorithms are breakable by quantum computers [45].

The quantum digital signature (QDS) uses quantum laws during the document-signing process, which maintains the following information theoretic principles: integrity, authenticity, and nonrepudiation [37]. **Figure 4** shows the schematic diagram of QDS. Due to the impractical requirements of the quantum system, such as quantum memory, and high-dimensional superposition of qubit states, QDS is not completely implemented, and it is still under development. Hence, the high efficiency of QDS still remains an unresolved open problem [37].

The one-time universal hashing (OTUH)-QDS method proposed can sign long documents by maintaining information theoretical properties [37]. The method not only used OTUH and one-time pad (OTP) as the underlying cryptosystem layer but also utilized secret sharing, which connects the quantum secret sharing, quantum key distribution, and/or future quantum internet.

6. Conclusions

The advancements in quantum computing and technology brought information or network security systems under threat. For example, many applications such as digital currency systems and driverless vehicles always depend on secure networks. Many cryptographic systems in information security rely on the hardness of mathematical problems, such as integer factorization, discrete logarithmic problems, and elliptic curve problem. However, with enough number of qubits, the quantum computers are able to break these mathematical problems. This book chapter presented the importance of postquantum security by portraying the information on the postquantum security issues. In addition, the chapter provided the required information on the quantum computers before starting the postquantum cryptographic methods. The basics on cryptography, the classifications of cryptography, and the mathematical problems relied on cryptography are presented. The first quantum computing algorithm proposed by Peter Shor has been briefly described in the chapter. The four postquantum categories of cryptographic methods, lattice-based cryptographic methods, code-based cryptographic methods, multivariate polynomial-based cryptographic methods, and hash-based cryptographic methods, are presented in the book chapter.

For future directions with the beginning of quantum computers era, it is a clear sign to strengthen our information security mechanisms and develop new encryption processes that are quantum resistant. Quantum cryptography scientists need to find proactive quantum-resistant solutions rather than reactive on when security vulnerabilities arise. NIST already started in 2016 to standardize and strengthen the postquantum security mechanisms at the early stage. The new approach of quantum computing emerges with the presence of quantum computing, for instance quantum key distribution uses superposition, and it is difficult to copy the unknown state of the key. In addition, the perspective of the encryption must be changed with the arrival of

quantum computers. The postquantum security mechanisms need to consider beyond the encryption processes, zero-trust philosophy, and verify at every point or level of the system, which is possible with high-speed quantum computers, whereas the classical computers were trying to balance the computing time and security levels.

Author details

Satyanarayana Degala
College of Engineering, University of Buraimi, Al Buraimi, Oman

*Address all correspondence to: degala.s@uob.edu.om

IntechOpen

© 2024 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Neild D. Google's Quantum Computer is 100 Million Times Faster than Your Laptop [Internet], ScienceAlert. 2015
- [2] Peter W. Shor, polynomial time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Scientific Computing*. 1997;**26**:1484. DOI: 10.1137/S0097539795293172
- [3] Hankerson D, Menezes A. Elliptic curve cryptography. In: van Tilborg HCA, Jajodia S, editors. *Encyclopedia of Cryptography and Security*. 2011. Springer, Boston, MA. DOI: 10.1007/978-1-4419-5906-5_245
- [4] Null L. *Essentials of Computer Organization and Architecture with Navigate Advantage Access*. 2024. p. 750
- [5] Thakare R, Phutane P. Unlocking the potential: A journey into quantum computing. In: 2024 International Conference on Innovations and Challenges in Emerging Technologies (ICICET). 2024. pp. 1-5. DOI: 10.1109/ICICET59348.2024.10616268
- [6] Desjardins J. The 3 Types of Quantum Computers and their Applications [Internet], Visualcapitalist. 2016. Available from: <https://www.visualcapitalist.com/three-types-quantum-computers/>
- [7] Debnath S, Chattopadhyay A, Dutta S. Brief review on journey of secured hash algorithms. In: 2017 4th International Conference on Opto-Electronics and Applied Optics (Optronix). Kolkata, India: IEEE; 2017. pp. 1-5. DOI: 10.1109/OPTRONIX.2017.8349971
- [8] Satyanarayana D, Alasmi AS. Detection and mitigation of DDOS based attacks using machine learning algorithm. In: 2022 International Conference on Cyber Resilience (ICCR). United Arab Emirates: Dubai; 2022. pp. 1-5. DOI: 10.1109/ICCR56254.2022.9995773
- [9] Althobaiti OS, Dohler M. Cybersecurity challenges associated with the internet of things in a post-quantum world. *IEEE Access*. 2020;**8**: 157356-157381. DOI: 10.1109/ACCESS.2020.3019345
- [10] Yuan Y, Mo Y. Security for cyber-physical systems: Secure control against known-plaintext attack. *Science China Technological Sciences*. 2020;**63**: 1637-1646. DOI: 10.1007/s11431-020-1621-y
- [11] Kiltz E, O'Neill A, Smith A. Instantiability of RSA-OAEP under chosen-plaintext attack. *Journal of Cryptology*. 2017;**30**:889-919. DOI: 10.1007/s00145-016-9238-4
- [12] Biryukov A, De Cannière C. Linear cryptanalysis for block ciphers. In: van Tilborg HCA, Jajodia S, editors. *Encyclopedia of Cryptography and Security*. 2011. Springer, Boston, MA. doi:10.1007/978-1-4419-5906-5_589
- [13] Menezes A. *Handbook of Applied Cryptography*. Boca Raton, FL, USA: CRC Press; 1997. DOI: 10.1201/9780429466335
- [14] Gambetta JM, Chow JM, Steffen M. Building logical qubits in a superconducting quantum computing system. *npj Quantum Information*. 2017; **3**(1):2
- [15] Classiq's Library, Quantum Cryptography - Shor's Algorithm

Explained (internet). Available from: <https://www.classiq.io/insights/shors-algorithm-explained>

[16] Beauregard S. Circuit for Shor's algorithm using $2n+3$ qubits. *Quantum Information Computer*. 2003;3(2): 175-185

[17] Li S, Chen Y, Chen L, Liao J, Kuang C, Li K, et al. Post-quantum security: Opportunities and challenges. *Sensors*. 2023;23(21):8744. DOI: 10.3390/s23218744

[18] Nejatollahi H, Dutt N, Ray S, Regazzoni F, Banerjee I, Cammarota R. Post-quantum lattice-based cryptography implementations: A survey. *ACM Computing Surveys*. 2019; 51(6) Article 129:41. DOI: 10.1145/3292548

[19] Khorasgani HA, Asaad S, Eghlidos T, Aref M. A lattice-based threshold secret sharing scheme. In: 2014 11th International ISC Conference on Information Security and Cryptology. Iran: Tehran; 2014. pp. 173-179. DOI: 10.1109/ISCISC.2014.6994043

[20] Khot S. Hardness of approximating the shortest vector problem in lattices. *Journal of the ACM*. 2005;52: 789-808

[21] Goldreich O, Micciancio D, Safra S, Seifert JP. Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. *Information Processing Letters*. 1999;71:55-61

[22] Peikert C. Public-key cryptosystems from the worst-case shortest vector problem. In: *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, Bethesda, Maryland. Bethesda: ACM; 2009. pp. 333-342

[23] Lyubashevsky V, Micciancio D. On bounded distance decoding, unique shortest vectors, and the minimum distance problem. In: *Proceedings of the Annual International Cryptology Conference*. Santa Barbara, CA, USA; 2009. pp. 577-594

[24] Goldreich O, Goldwasser S, Halevi S.. Eliminating decryption errors in the Ajtai-Dwork cryptosystem. In: *Proceedings of the Annual International Cryptology Conference*. Santa Barbara, CA, USA; 1997. pp. 105-111

[25] Brakerski Z, Gentry C, Halevi S. Packed ciphertexts in LWE-based homomorphic encryption. In: *International Workshop on Public Key Cryptography*. Berlin/Heidelberg, Germany: Springer; 2013. pp. 1-13

[26] Regev O. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*. 2009; 56:34

[27] Pöppelmann T, Güneysu T. Towards efficient arithmetic for lattice-based cryptography on reconfigurable hardware. In: *Proceedings of the International Conference on Cryptology and Information Security in Latin America*. Santiago, Chile; 2012. pp. 139-158

[28] Asif R. Post-quantum cryptosystems for internet-of-things: A survey on lattice-based algorithms. *IoT*. 2021;2(1): 71-91. DOI: 10.3390/iot2010005

[29] Berlekamp ER, McEliece RJ, Van Tilborg HCA. On the inherent intractability of certain coding problems. *IEEE Transactions on Information Theory*. 1978;24(3):384-386

[30] Haidary Makoui F, Gulliver TA, Dakhilalian M. A new code-based digital signature based on the McEliece

cryptosystem. IET Communications. 2023;**17**:1199-1207. DOI: 10.1049/cmu2.12607

[31] Nojima R, Imai H, Kobara K, Morozov K. Semantic security for the McEliece cryptosystem without random oracles. Designs, Codes and Cryptography. 2008;**49**(1-3):289-305

[32] Cayrel PL, Gaborit P, Girault M. Identity based identification and signature schemes using correcting codes. In: Proceedings of International Workshop on Coding and Cryptograph. Amsterdam: IOS Press; 2007. pp. 69-78

[33] McEliece R. A public-key cryptosystem based on algebraic coding theory. Deep Space Network Program Report. 1978;**44**:114-116

[34] Niederreiter H. Knapsack-type cryptosystems and algebraic coding theory. Problem Control Information Theory. 1986;**15**(2):159-166

[35] Kuang R, Perepechaenko M, Toth R, et al. Performance comparison of quantum-safe multivariate polynomial public key encapsulation algorithm. EURASIP Journal on Information Security. 2024;**2024**:23. DOI: 10.1186/s13635-024-00170-7

[36] Kuang R, Perepechaenko M, Barbeau M. A new quantum-safe multivariate polynomial public key digital signature algorithm. Scientific Reports. 2022;**12**:13168. DOI: 10.1038/s41598-022-15843-x

[37] Yin HL, Fu Y, Li CL, Weng CX, Li BH, Gu J, et al. Experimental quantum secure network with digital signatures and encryption. National Science Review. 2023;**10**:nwac228

[38] Wang X, Yin YL, Yu H. Finding collisions in the full SHA-1. In: Shoup V, editor. Advances in Cryptology-

CRYPTO 2005. Berlin: Springer; 2005. pp. 17-36

[39] Wang X, Yu H. How to break MD5 and other hash functions. In: Cramer R, editor. Advances in Cryptology- Eurocrypt 2005. Berlin: Springer; 2005. pp. 19-35

[40] Stevens M, Bursztein E, Karpman P, et al. The first collision for full SHA-1. In: Katz J, Shacham H, editors. Advances in Cryptology – CRYPTO 2017. Cham: Springer; 2017. pp. 570-596

[41] Sheng YB, Zhou L, Long GL. One-step quantum secure direct communication. Scientific Bulletin. 2022;**67**:367-374

[42] Bennett CH, Brassard G. Quantum cryptography: Public key distribution and coin tossing. Theoretical Computer Science. 2014;**560**:7-11

[43] Chen YA, Zhang Q, Chen TY, et al. An integrated space-toground quantum communication network over 4,600 kilometres. Nature. 2021;**589**:214-219

[44] Zhang Y, Ni Q. Design and analysis of random multiple access quantum key distribution. Quantum Engineering. 2020;**2**:e31

[45] Lyubashevsky V. Lattice-based digital signatures. National Science Review. 2021;**8**:nwab077

Information Coding and the Search for Extraterrestrial Intelligence

Jonathan Blackledge

Abstract

Communication systems that operate under low-power constraints and in noisy transmission environments require specialised, application-specific coding strategies. In such contexts, the ‘Chirplet Transform’ emerges as a highly effective method for information transfer under noisy conditions. A bit can be encoded as a chirplet—a frequency-modulated signal of finite duration—using a phase shift, for example, to distinguish between binary values (0 and 1). A sequence of these chirplets forms a ‘chirplet stream’, which is then frequency modulated and transmitted. Upon reception, the signal is frequency demodulated by correlating it with a matching chirplet replica, enabling recovery of the original bit stream—albeit with some potential for bit errors. This technique demonstrates a notably lower bit error rate than conventional modulation methods, especially in high-noise scenarios. However, the central question remains: ‘Is this method truly unique?’ Are there alternative coding schemes that could match or even exceed the performance of chirplet modulation in such environments? These questions are key drivers of the investigation presented here. We propose that if chirplet modulation possesses truly unique properties, it may represent the only viable technique for transmitting intelligible, bit-based information across vast interstellar distances—even through the so-called ‘waterhole’, the quietest region of the interstellar radio spectrum. In this context, we examine the theoretical foundations of chirp-based modulation and demodulation for both ‘unbounded’ and ‘bounded’ cases (i.e., chirps of a finite duration or chirplets). This includes a comparative analysis of chirplets relative to other potential modulation basis functions, aiming to determine whether their properties confer uniqueness in information encoding. Subsequently, we explore how ‘discrete chirplet modulation’ can be employed to encode bit streams into transmission signals. The corresponding demodulation process is analysed using a ‘relative information entropy metric’ to assess whether the reconstructed bit stream consists of random noise or meaningful, intelligible content. This is further complemented by a ‘blind demodulation method’ based on ‘prime number factorisation’, offering an additional layer of signal analysis and security. To support this investigation, “prototype Python source code” is provided, enabling readers to explore, test, and extend the proposed approach.

Keywords: Chirplet transform, Chirplet coding, binary strings, information entropy, intelligibility test, optimality and uniqueness, intelligibility source conjecture

1. Introduction

On 17 February 1600, in the Campo de' Fiori—a central market square in Rome—Giordano Bruno (1548–1600) stood naked, his tongue having been cut out 'because of his wicked words'. He was then burned alive, and his ashes thrown into the Tiber River. Bruno postulated the multiplicity of inhabited worlds; that not only did what we now call 'exoplanets' exist, but that many will have intelligent life [1]. This was over 400 years before the first confirmation of such planets in 1992 [2].

As of 7 November 2024, there are ~6000 confirmed exoplanets in over 4000 planetary systems, with over 1000 systems having more than one planet. It is because of Bruno's sacrifice to a scientific idea that he could not validate that Giordano Bruno is referred to as 'Martyr of Science'. Because of this, a 'Giordano Bruno Memorial Award' is given annually for 'significant contributions to the art and science associated with the Search for Extraterrestrial Intelligence (SETI)' [3]. At the time, Bruno was one of many whose ideas were radical in the extreme; ideas that challenged the underlying philosophical and theological basis of the governing authorities, predicated on monotheism.

Giordano Bruno is among many historical and contemporary thinkers who have embraced the idea of Cosmic Pluralism—the philosophical belief that the universe contains numerous planets or natural satellites beyond Earth that may harbour extraterrestrial life. In some interpretations, this includes the possibility of intelligent life forms possessing technological civilisations equal to or even more advanced than our own.

This belief has been shared by a diverse range of philosophers, astronomers, and scientists across cultures and eras, including Anaximander (610–546 BCE), Fakhruddin Razi (1150–1210), Johannes Kepler (1571–1630), Benjamin Franklin (1706–1790), Friedrich Herschel (1738–1822), Carl Sagan (1934–1996), and Frank Drake (1930 to present). Cosmic Pluralism is a philosophy that transcends cultural and historical boundaries. It addresses a question both profound and persistent: Are we alone in the universe? This is a question that is as natural to ask, as it is difficult to answer.

Since the early 1960s, SETI has been 'listening' to cosmic radio noise and attempting to interpret it to ascertain whether there is intelligible information embedded in it. To date, nothing has been found. For this reason, we focus on the use of chirplet-based modulation as a robust method for encoding and decoding extraterrestrial signals, making use of its unique Fourier eigenfunction properties [4]. It is argued that, if an advanced civilisation wanted to communicate *via* radio signals, it would most likely use chirplet modulation [5]. This is due to the superior noise resistance and information recovery capabilities that chirplet modulation can offer, and further, such capabilities are unique.

The chapter explores some practical implementations of this approach and includes some example prototype Python-based chirplet modulation/demodulation code to help illustrate the basic idea and allow interested readers to undertake further developments. A prime factorisation-based key exchange mechanism also proposed, coupled with some open questions concerning alternative encoding methods.

In order to put the problem of detecting intelligible signals embedded in cosmic noise into a perspective, we can consider it in terms of a decryption problem. This is discussed in the following section.

2. The ultimate decryption problem

In cryptography, the cryptographer has control over the noise (the cypher), while the cryptanalyst knows what to aim for—the known unknowns. Typically, cryptanalysis assumes that the encrypted data (the ciphertext) represents an encrypted version of a known natural language (the plaintext), for example. Furthermore, it may be possible to assume knowledge of the algorithm used to encrypt the plaintext; in other words, the algorithm is accessible. This assumption aligns with the Kerckhoff-Shannon principle, which states that ‘a crypto-system should be secure even if everything about the system, except the key, is public knowledge’. Or, as Claude Shannon succinctly put it, ‘The enemy knows the system’ (i.e. the encryption algorithm) [6].

Knowledge of the plaintext language allows cribs to be created, providing the cryptanalyst with a target for attacking the ciphertext and recovering part or all of the plaintext. However, this approach can be easily thwarted by first translating the plaintext into another natural language before encryption. Regarding relatively recent developments in cryptography, the Kerckhoffs-Shannon principle no longer strictly applies. Modern cyphers—whether key-dependent or otherwise—can be routinely generated using evolutionary computing and symbolic regression to simulate real noise. Nevertheless, the challenge of exchanging such cyphers remains, which can be addressed using a no-keys-based protocol (e.g. the Shamir three-pass method).

Regardless of the encryption method used, one fundamental principle of cryptography remains constant: One Message, One Key, One Cypher. This principle is the foundation of the One-Time Pad (OTP)—a cypher system that offers unconditional security. Even in the era of quantum computing, the OTP retains its theoretical security, making it a prime example of post-quantum cryptography. By strictly adhering to this principle, it is possible to ensure that message transmission remains secure, regardless of advancements in computational power.

Recent advancements, enabled by the intersection of computing power, data coding, and the Internet, have facilitated the development of information-hiding techniques. These methods include hiding plaintext in plaintext (coverttext) through steganography and hiding ciphertext in coverttext through steganocryptography. In this context, a fundamental steganocryptographic law has emerged: One Message, One Key, One Cypher, One Coverttext. Steganocryptography has the unique advantage of embedding encrypted data within other data types, thereby eliminating an obvious indicator to potential attackers that the data has been encrypted and may contain sensitive information. Furthermore, steganocryptography has become increasingly viable in today’s data-rich, or ‘stega-rich’, environment.

When comparing the world of cryptography, as outlined above, with the Search for Extraterrestrial Intelligence (SETI), a fundamental difference arises. No matter how sophisticated an encryption method may be, or how rigorously its implementation adheres to the fundamental laws discussed, encryption and its analysis are ultimately products of the human mind and technology. In this sense, cryptanalysis operates within the realm of known unknowns, where identifying and understanding such unknowns requires academic research and/or effective intelligence services.

In the case of SETI, however, if measurable signals containing intelligent information exist, the challenge is far greater. Here, there is a long list of unknown unknowns

associated with their “decryption.” Known unknowns refer to things we are aware we do not know—uncertainties or gaps in knowledge that we can clearly identify. This concept was famously highlighted by former US Secretary of Defence Donald Rumsfeld in a 2002 speech, where he categorised knowledge into three types:

- Known knowns: Things we know we know.
- Known unknowns: Things we know we do not know.
- Unknown unknowns: Things we do not know that we do not know.

Known unknowns are identifiable questions or uncertainties. For instance, if a scientist knows that a certain experiment could produce several possible outcomes, the specific outcome is a known unknown—they are aware of the uncertainty but do not yet have the answer. In fields like decision theory, recognising known unknowns is important because it allows for planning and strategies to address those uncertainties. These can be managed through research, investigation, or gathering more data, while unknown unknowns present much greater challenges due to their unpredictability.

In this context, and that of SETI, the list of questions and assumptions is unending. For example:

- i. Assuming that such signals are part of the electromagnetic spectrum, what would be their frequency spectrum—radio frequencies?
- ii. Assuming that these radio signals have intelligent information embedded in them, how would this information be coded, and what ‘language’ would such codes represent?
- iii. How would such coded information be transformed to compensate for the noise that will accumulate over many light years of electromagnetic wave propagation, leading to the information becoming significantly distorted and possibly unintelligible as a result of this distortion?

3. Fundamental assumptions

In order to do something, we must consider a framework based on a set of assumptions that allow us to adopt specific approaches to signal processing and analysis. In this work, we consider the following assumptions. In the context of SETI, some of these are standard, and some are non-standard, but all are debatable.

1. *Frequency band.* The specific range of radio frequencies is between 1.42 and 1.72 GHz. Known as the ‘waterhole’, this frequency range is considered an ideal band for interstellar communication and is so named because it lies between two natural spectral lines of hydrogen and hydroxyl—the components of water, a molecule essential for life as we know it. The significance of the waterhole lies in

its quiet nature in terms of cosmic radio noise. Hydrogen is the most abundant element in the universe, and its natural emission at 1.42 GHz (the 21 cm line) makes this range easily detectable by radio telescopes. Since this part of the radio spectrum is relatively free from interference from natural cosmic sources, it can be hypothesised that an intelligent civilisation might choose to transmit signals within this ‘quiet’ band. The idea is that if advanced civilisations are trying to communicate, they might logically choose this region because it is both detectable and symbolic, potentially recognising the importance of water for life. Thus, SETI researchers often focus their efforts on scanning this frequency range when searching for extraterrestrial signals.

2. *Bit streams coding.* Sending signals as ‘bit streams’ would be a practical and efficient way for an intelligent civilisation to encode and transmit information. Some of the key reasons why life—specifically intelligent or technologically advanced civilisations—might choose to send signals as bit streams are as follows:

- *Simplicity and universality:* A bit stream consists of two fundamental values, typically represented as 0 s and 1 s, corresponding to ‘binary’ data. Binary encoding is the simplest and most universal way to represent information, making it likely to be understood by any technologically advanced civilisation.
- *Universality of mathematics:* Since mathematics is considered a universal language, bit streams, which rely on basic principles of numbers, might be easily recognised and decoded by any civilisation familiar with logic and computation. It cannot, of course, be assumed that an extraterrestrial civilisation would use mathematical symbols and analysis similar to humans. Their primary form of encoding could be non-linear or much more complex.
- *Information entropy:* The use of bits to convey intelligible information becomes a reasonable approach when paired with an understanding of information entropy. This concept is key to estimating the minimum number of bits required to encode a string of symbols, thereby establishing a fundamental limit on how much information can be communicated through a sequence of bits—a ‘bit stream’. The concept of ‘information entropy’ originated from a physical thought experiment undertaken by Leo Szilard (a research student and colleague of Albert Einstein) in 1929, which demonstrated that the mere possession of information by ‘intelligent beings’ could, in principle, have thermodynamic consequences. This work refined the famous Maxwell’s demon scenario and reversed the classic Joule expansion thought experiment [7].
- *Efficiency in data transmission:* Information can be efficiently compressed and transmitted in binary form. Large amounts of data—whether mathematical or linguistic—can be encoded into a bit stream, enabling the efficient use of bandwidth and energy during transmission.

- *Technological convergence:* If life elsewhere follows a path similar to Earth's technological development, they would likely have evolved technologies similar to digital computing and binary data processing. Since many modern communication systems (like the Internet, radio, and satellite communication) rely on bit streams, this method might represent a form of communication that other civilisations would also develop, making it a common 'language' across different technological species.
- *Potential for structured information (encoding patterns and structures):* Bit streams allow the sending of structured information, such as prime numbers, mathematical sequences, or even detailed scientific data, which could demonstrate intelligence and provide context for communication. For example, a stream of prime numbers (only divisible by 1 and themselves) could be recognised as a non-random signal from an intelligent source.
- *Communication protocols:* Bit streams can follow certain protocols that involve headers, checksums, and message payloads, ensuring that the receiver knows how to interpret the data properly. This organisation can help in deciphering and understanding the message.
- *Potential for multi-purpose communication:* Bit streams can encode any kind of data—text, images, sound, or other formats—by simply converting them into binary sequences. This allows a civilisation to send diverse forms of information in a unified format, which can be more easily processed by machines or computational systems.
- *Noise resistance:* Bit streams are particularly suited for transmission over noisy channels. Error-correcting codes and redundancy can be incorporated to help the receiving civilisation recover the message, even if parts of the signal are corrupted or degraded during transmission.
- *Robustness to interference:* Digital signals (bit streams) tend to be more robust against noise and interference compared to analogue signals, making them ideal for long-distance communication through space, where signals may weaken or face interference from cosmic phenomena.
- *Modulation and bandwidth:* Bit streams can be modulated onto various carrier frequencies for transmission, allowing for the transmission of complex and rich data, such as images or even multimedia, using simple binary encoding techniques.

3. Sending signals as bit streams offers a straightforward, efficient, and universally understandable means of interstellar communication. It minimises ambiguity, allows for error correction, and can encode complex information in a simple form. Since binary systems are a foundation of both human and, potentially, extraterrestrial technology, it stands to reason that advanced civilisations would recognise and employ bit streams for communication over vast distances. However, there is a major issue as to which modulation method will provide optimality, given that the information must be made highly robust to noise, where it is assumed that the noise is additive.

4. *Chirplet modulation*. Chirplet modulation refers to a signal modulation technique that uses chirplets—short-duration, frequency-modulated signals that vary in frequency over time. Chirplets are a generalisation of chirps, which are signals in which frequency increases (up-chirp) or decreases (down-chirp) linearly over time. Chirplet modulation is used in applications where robust, flexible, and efficient communication is necessary, especially in environments with noise or interference.

In chirplet modulation, data (like binary information) is encoded by varying the parameters of the chirplet. The frequency, phase, or amplitude of the chirplet can be modulated to encode different values. Unlike basic chirp signals, chirplets allow for non-linear frequency modulations and can be adapted for complex signal processing tasks. This flexibility is key to why chirplet modulation is useful in noisy or dynamic environments, as the chirplet shape can be optimised for specific conditions. Thus, in communications, chirplet modulation is employed in environments with high interference or multi-path effects. The flexibility of chirplet signals allows for more robust transmission of data even in challenging conditions, and their overall performance far outweighs amplitude, frequency, or phase modulation, for example. Coupled with this realisation is a further and important assumption, namely, that the ability for chirplet modulation to convey binary information in the presence of additive noise with very low signal-to-noise ratios (SNRs) is in fact unique. This says that no other basis function can provide the same performance as that provided by a chirp. In this context, it is argued that any form of communication that other technologically advanced civilisations may develop would be based on the chirplet modulation of a bit stream, an issue that is explored further in the following section.

Given the expansion of the universe, a question arises as to whether the frequency characteristics of chirplets make them susceptible to Doppler shifts. In general, this is indeed the case, but primarily for signals emitted many billions of years ago, which would have undergone frequency shifts comparable to the optical redshifts observed in distant galaxies. As with optical redshifts, the farther away the source of the emission, the greater the resulting frequency shift. There is thus a centre frequency shift and consequently, the entire frequency content of the chirplet shifts downward (redshift) if the source is receding, or upward (blueshift) if approaching. Time stretching will occur in which the duration of the chirplet increases (it becomes more spread out in time) for a redshift (and compresses for a blueshift). This affects the slope of the chirp—the rate of frequency change (chirp rate) becomes slower under redshift. Since the signal is stretched in time, its bandwidth contracts proportionally, and thus, a redshifted chirplet has a narrower spectral width. Crucially, however, there is time-frequency signature preservation in that the overall time-frequency shape of the chirplet is preserved, just scaled. This means a Doppler-shifted chirplet remains recognisable in the time-frequency plane—essential for matched filtering and demodulation.

Moreover, if a chirplet experiences significant Doppler red-shifting, it would suggest that the signal originated so far in the past and has travelled such an immense distance—spanning billions of light-years—that cosmic noise would likely have obliterated any intelligible information it once carried, regardless of the information modulation technique applied. Therefore, in a practical or universal context, it is generally assumed that the analysis of extraterrestrial signals occurs within the “near field”—typically much less than a billion light-years away.

4. Modulation methods used by SETI

The Search for Extraterrestrial Intelligence (SETI) represents one of the most profound scientific undertakings of modern times. At its core lies the challenge of detecting communications from civilisations elsewhere in the universe. Central to this effort is the concept of ‘modulation’—the method by which information is encoded onto electromagnetic waves for transmission. SETI scientists must hypothesise what forms of modulation Extraterrestrial Intelligences (ETIs) might use, guided by physics, communication theory, and the assumptions of universality and intelligibility.

4.1 The SETI signal dilemma

SETI does not operate under the assumption that extraterrestrials are attempting to contact us specifically. Instead, it presumes that a civilisation may broadcast detectable signals for various purposes: beacons, inadvertent leakage, or directed communication. Given the unknown nature of alien technologies, SETI adopts a conservative and pragmatic approach: it searches for signals that are distinguishable from natural astrophysical processes and likely to be energy-efficient and robust over interstellar distances [8].

4.2 Example modulation method

A likely signal from an ETI would satisfy the following:

- *Spectral distinctiveness*: Easily separated from natural radio emissions.
- *Energy efficiency*: To maximise reach and minimise cost.
- *Noise resilience*: Capable of surviving cosmic background noise and interstellar medium effects.
- *Universality*: Based on simple mathematical or physical principles, assuming shared physics.
- *Detectability*: Favouring signals with low information entropy for ease of recognition [9].

These criteria inform the likely modulation methods in the search for ETI communications, leading to some of the more common techniques which are now briefly considered.

1. *Narrowband continuous wave (CW) signals*: Narrowband signals, typically less than 1 Hz wide, are the primary target of traditional SETI programmes. A narrowband continuous wave—essentially a pure sine wave—is extremely rare in nature, making it a strong candidate for artificial origin. These signals do not carry complex information but could act as beacons, announcing presence [10].
2. *On-off keying (OOK)*: Also known as ‘Amplitude-Shift Keying’, OOK involves toggling the signal on and off to represent binary 1 s and 0 s. This is simple to generate and detect and may be used to encode a message in a beacon-type signal.

3. *Frequency-shift keying (FSK)*: FSK involves switching the carrier between discrete frequencies. It offers better noise immunity than OOK and is compatible with low-power transmission. FSK can be spread over narrow or wide frequency bands, and its spectral features remain distinct, even with interstellar dispersion (Messerschmitt, 2012).
4. *Phase shift keying (PSK)*: In PSK, the phase of a carrier wave is modulated to represent data. PSK is widely used in Earth-based digital communication (e.g. Wi-Fi and satellite links) due to its spectral efficiency and robustness. However, PSK's detection is more complex, potentially making it less favourable for an initial beacon signal intended to be easily recognised.
5. *Pulse modulation and time-based encoding*: SETI has also considered 'pulse modulation'—transmitting signals in timed bursts with precise intervals. Timing sequences based on prime numbers or mathematical constants are hypothesised as universal [11]. Pulsed signals are resistant to Doppler shifts and dispersion and might encode information in the time between bursts.
6. *Chirp signals*: Chirped signals combat dispersion caused by the interstellar medium, as they can be de-chirped to recover the original message. They are also easier to detect using matched filtering, a technique used in radar and communications. It is this approach that is the focus in this work, which revises and expands further on [12].

4.3 Technological implementations in modern SETI

There are a range of technological implementations that have been pioneered by SETI and continue to be so.

The 'Breakthrough Listen' initiative is one of the most comprehensive SETI programmes to date. It applies advanced algorithms, including machine learning, to search vast radio bandwidths for a variety of modulated signals—narrowband, pulsed, chirped, or modulated in complex ways [13]. It expands the modulation search space beyond traditional methods.

The now-retired 'SETIhome' project harnessed volunteer computing to search for narrowband and pulsed signals in Arecibo data. It considers Doppler drift and frequency modulation effects due to relative motion.

4.4 Challenges and philosophical considerations

The main challenge in predicting modulation schemes is the lack of shared context between humans and ETIs. While narrowband signals are an engineering 'default', more advanced civilisations might use spread spectrum, quantum modulation, or even gravitational waves, for example, in their endeavour to undertake cosmic communications. Thus, SETI must balance 'search specificity with search breadth'—casting a wide net while prioritising signals humans can realistically detect [14].

Additionally, researchers debate whether SETI should expect intentional beacons (deliberately designed for detection) or 'techno-signatures'—leakage or unintentional emissions. Modulation methods differ greatly between these scenarios.

Modulation is central to SETI, not only as a technical consideration but also as a philosophical one. The modulation method chosen by an alien civilisation may reveal

much about their technology, intent, and assumptions about communication. Current SETI efforts focus on simple, robust, and energetically efficient modulation schemes. These include narrowband CW, OOK, and FSK, while increasingly exploring complex and dynamic forms like chirps and bursts. As technology improves and data volumes grow, the modulation parameter space will continue to expand, increasing the chances of detecting an intelligent signal. It is in this context that this material has been composed.

5. Overview of the material

Chirplets (windowed chirp functions) and the chirplet transform [15] are important concepts in the physical sciences. In signals and systems engineering, chirplets find broad application, notably in chirp spread spectrum techniques, which form a core component of the IEEE wireless telecommunications standards [16]. Other significant uses include pulse compression, which enhances radar sensitivity and resolution by shaping transmitted pulses to improve their autocorrelation characteristics, as demonstrated in chirp radar systems [17].

In optical fibre communications, the propagation of light pulses can achieve substantial spectral broadening and pulse compression at higher pulse energies through the use of large-mode-area photonic crystal fibres combined with chirped input pulses [18]. In non-relativistic quantum mechanics, transient phenomena such as the quantum shutter problem [19] are modelled using Fresnel integrals, whose kernels involve quadratic phase functions. This approach can be extended to the semi-relativistic regime through the application of fractional calculus [20].

Chirped mirrors, which incorporate *chirped spaces* in their dielectric design, are engineered to reflect specific wavelengths of light, thereby compensating for dispersion effects introduced by certain optical components [21]. More recently, chirp functions have found applications in cryptology, including key exchange protocols [22], self-authentication of digital signals [6], and robust watermarking techniques [23]. Intriguingly, chirp functions also play a critical role in cosmology, where gravitational wave chirps serve as key signatures for detecting gravitational waves from binary systems [24].

The examples and references above underscore the foundational significance of chirp functions. From characterising physical phenomena to driving innovations in communication engineering (e.g. [25, 26]), the chirp function remains a cornerstone across multiple scientific fields. In this context, we revisit the unique properties of the chirp function, demonstrating that it possesses a conjugate Fourier eigenfunction. This observation leads us to conjecture that the chirp function is singular in this regard, as no other phase-only function exhibits such a conjugate eigenfunction.

This uniqueness, coupled with the autocorrelation properties of chirp functions, underpins their exceptional utility. Building on this foundation, we reevaluate the principles of chirp coding and the transmission of bitstreams through noisy communication channels characterised by low signal-to-noise ratios (SNRs). We propose a chirplet modulation algorithm, complemented by an intelligibility test for the recovered bitstream. This work builds upon and extends material presented in Refs. [12, 27].

In this work, we propose a communication protocol based on chirplet modulation, wherein secure transmission is achieved through shared prior knowledge between the sender and receiver regarding the signal's operational bandwidth. Modulation and

demodulation are enabled by factoring a semi-prime number—derived from this known bandwidth—into its two prime components. When the semi-prime is sufficiently large, the protocol offers a uniquely robust form of communication security.

We further examine the implications of this method, particularly its potential resilience to future breakthroughs in quantum computing. More broadly, this framework is presented as a conceptual ‘toolbox’ for analysing cosmic signals in the search for intelligent, technologically advanced life. It offers a speculative yet intriguing pathway for transitioning from ‘unknown unknowns’ to ‘known unknowns’ in the context of interstellar communication.

6. Communications theory: In search of the optimal solution

When a signal is distorted by additive noise, it can be modelled as:

$$\text{Signal} = \text{Transform}[\text{Information}] + \text{Noise}.$$

This raises the fundamental question: what transform provides a uniquely optimal method for transmitting recoverable information through noisy environments?

Let us consider a transform denoted by $\mathcal{T}$. In terms of time t , the signal model can be expressed as:

$$s(t) = \mathcal{T}[f(t)] + r(t),$$

where $s(t)$ represents the transmitted signal, $f(t)$ represents the information, and $r(t)$ is the noise. The ideal solution to the inverse problem—extracting information from noise—is given by:

$$f(t) = \mathcal{T}^{-1}[s(t)],$$

where $\mathcal{T}^{-1}$ is the inverse transform, and it is assumed that $\mathcal{T}^{-1}[r(t)] = 0$. The key challenge, then, is to identify a practical transform that satisfies this condition.

6.1 Chirp convolution and correlation

Consider the scenario where the transform is based on a convolution integral with a (unit-amplitude) frequency-modulated chirp as the kernel. For a chirp parameter α , the signal can be written as:

$$s(t) = p(t) \otimes f(t) + r(t), \quad (1)$$

where

$$p(t) = \exp(i\alpha t^2),$$

and the convolution is defined as:

$$p(t) \otimes f(t) \equiv \int_{-\infty}^{\infty} p(t - \tau) f(\tau) d\tau.$$

By correlating $s(t)$ with $p^*(t)$ (the complex conjugate of $p(t)$), we obtain:

$$p^*(t) \odot s(t) = [p^*(t) \odot p(t)] \otimes f(t) + p^*(t) \odot r(t), \quad (2)$$

where $\odot$ represents the correlation integral:

$$p^*(t) \odot p(t) \equiv \int_{-\infty}^{\infty} p^*(t + \tau) p(\tau) d\tau.$$

Using the definition and properties of the delta function:

$$\delta(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \exp(it\tau) d\tau, \quad \delta(\alpha t) = \frac{1}{|\alpha|} \delta(t), \quad |\alpha| > 0,$$

we evaluate $p^*(t) \odot p(t)$ as follows:

$$p^*(t) \odot p(t) = \int_{-\infty}^{\infty} \exp[-i\alpha(t + \tau)^2] \exp(i\alpha\tau^2) d\tau.$$

Expanding and simplifying, this becomes:

$$= \exp(-i\alpha t^2) \int_{-\infty}^{\infty} \exp(-2i\alpha t\tau) d\tau = \exp(-i\alpha t^2) 2\pi \delta(-2\alpha t).$$

Using $\delta(-2\alpha t) = \frac{1}{|2\alpha|} \delta(t)$, we find:

$$p^*(t) \odot p(t) = \frac{\pi}{|\alpha|} \delta(t).$$

Substituting this result into Eq. (2), we obtain:

$$p^*(t) \odot s(t) = \frac{\pi}{|\alpha|} f(t) + p^*(t) \odot r(t).$$

Assuming that $r(t)$ is a zero-mean, delta-uncorrelated stochastic function, it follows that $p^*(t) \odot r(t) = 0$. The reasons for this crucial point are discussed in the following section.

6.2 Correlation with a stochastic function

Delta-uncorrelated noise is typically modelled as a random process with the following properties:

- The noise is uncorrelated: At any two distinct times t_1 and t_2

$$E[n(t_1)n(t_2)] = 0 \quad \text{for } t_1 \neq t_2.$$

where

$$E[X] = \int_{-\infty}^{\infty} xp(x)dx$$

denotes the expectation operator for the probability density function $p(x)$ of the random variable X and x represents the possible values that X can take. By the assumption of delta-uncorrelated noise, the noise function $r(t)$ is statistically independent of the chirp function. This means there is no systematic relationship between the chirp and the noise values at any given time.

- Zero mean: Since the noise has a zero mean, $E[n(t)] = 0$.

Given that the noise has zero mean and is uncorrelated, the expected value of the product of the chirp and noise at any time is zero, that is,

$$E[p(t)r(t)] = E[p(t)]E[n(t)] = 0$$

This indicates that, on average, the product of the chirp signal and the noise is zero. Therefore, the correlation integral

$$\int_{-\infty}^{\infty} p^*(t)n(t+\tau)dt$$

also evaluates to zero because the expectation of the product of a deterministic chirp and a zero mean, uncorrelated noise process is zero at every point in time.

In conclusion, the correlation of the chirp function with delta-uncorrelated noise is zero because: (i) the noise has zero mean; (ii) the noise and the chirp are uncorrelated (statistically independent); and (iii) the expected value of the product of a deterministic signal (chirp) and a zero mean, uncorrelated random noise process is zero. Thus, mathematically, the correlation integral evaluates to zero, which means that the chirp function is orthogonal to the delta-uncorrelated noise in the correlation sense.

In respect of this result (i.e. $p^*(t) \odot r(t) = 0$), the information function is given by

$$f(t) = \frac{|\alpha|}{\pi} p^*(t) \odot s(t).$$

This solution demonstrates that the amplitude of the reconstruction is proportional to $|\alpha|$, meaning a larger chirping parameter α results in a higher signal-to-noise ratio (SNR)—or more specifically, a higher information-to-noise ratio, $\|f(t)\|/\|r(t)\|$.

This derivation assumes the ideal case where $t \in (-\infty, \infty)$. The practical application of this method using chirplets (finite-duration chirps) will be considered later. Nonetheless, this analysis illustrates how a transform $\mathcal{T}$, based on convolution with a linear frequency-modulated chirp, can achieve robust information recovery in noisy environments. Theoretically, we can achieve an exact inverse solution for additive zero-mean noise that is typically Gaussian distributed—taking into account the Central Limit Theorem.

6.3 Spectral analysis

To reevaluate the analysis presented in the previous section, it is helpful to examine the problem from the perspective of the spectral properties of $p(t)$. In the context

of Eq. (2), it becomes evident that we require a function $p(t)$ with a Fourier transform $P(\omega)$ such that (using the correlation theorem):

$$P^*(\omega)P(\omega) \equiv |P(\omega)|^2 = C, \quad \text{and} \quad P^*(\omega)N(\omega) = 0$$

where C is a constant, and $N(\omega)$ represents the Fourier transform of $r(t)$. Using the convolution and correlation theorems, Eq. (2) transforms into:

$$P^*(\omega)S(\omega) = P^*(\omega)P(\omega)F(\omega) + P^*(\omega)N(\omega)$$

where $S(\omega)$ and $F(\omega)$ denote the Fourier transforms of $s(t)$ and $f(t)$, respectively. This yields the desired result:

$$F(\omega) = \frac{1}{C} P^*(\omega)S(\omega)$$

This expression provides a theoretically perfect reconstruction of the information spectrum $F(\omega)$, regardless of the signal-to-noise ratio (SNR), provided $P^*(\omega)N(\omega) = 0$.

The above analysis highlights the principal conditions that any chosen function $p(t)$ must satisfy:

- i. A constant power spectrum $|P(\omega)|^2$ across all frequencies.
- ii. (ii) Zero correlation with additive noise.

Condition (i) is always satisfied if $p(t)$ has a phase-only spectrum. Using the notation $\leftrightarrow$ to denote the Fourier transform, we can write:

$$p(t) \leftrightarrow C \exp[i\theta(\omega)]$$

where $\theta(\omega)$ is the spectral phase. This condition is further explored in Section 7.

6.4 Autocorrelation analysis

The analysis of reconstructing $f(t)$ from $s(t)$, as expressed in Eq. (1), leads to the question: *What convolution functions yield a broad spectrum when autocorrelated?* While many functions exhibit a constant power spectrum (Condition (i)), they may not necessarily satisfy Condition (ii). Convolution functions that generate broad spectra when autocorrelated are typically noise-like or rapidly varying functions. Examples include:

6.4.1 Gaussian function

A Gaussian function, when convolved or autocorrelated with itself, produces another Gaussian. In the frequency domain, this results in a Gaussian spectrum, which can be broad depending on the width of the original Gaussian.

6.4.2 Rectangular (boxcar) function

The Fourier transform of a rectangular window function is a sinc function. When auto-correlated, it produces a spectrum with broad frequency components due to the sinc function's characteristics.

6.4.3 Exponential decay

Exponential decay functions can produce broad spectra, particularly when the decay rate is fast. Autocorrelation of such functions generates power spectra that extend across a wide range of frequencies.

6.4.4 Brownian motion ($1/\omega$ noise)

While Brownian motion has a power spectrum proportional to $1/\omega^2$, its derivatives or variations can result in broader spectra. When autocorrelated, these functions often exhibit frequency components spread over a wide range.

6.4.5 Impulse (Dirac Delta) function

The convolution of a signal with an impulse function preserves the original signal. Autocorrelation of an impulse produces a flat power spectrum since:

$$p(t) = \delta(t) \leftrightarrow 1(\omega) \quad \forall \omega$$

In the time domain, this translates to:

$$s(t) = \delta(t) \otimes f(t) + r(t)$$

Following the correlation theorem:

$$\delta(t) \otimes s(t) = \delta(t) \otimes \delta(t) \otimes f(t) + \delta(t) \otimes r(t)$$

This simplifies to:

$$f(t) = s(t) - r(t)$$

However, without prior knowledge of $r(t)$, $f(t)$ cannot be recovered. This principle is analogous to additive encryption methods like the Vernam Cypher, which relies on the addition (encryption) and subtraction (decryption) of a pseudo-random cypher $r(t)$.

6.4.6 Random noise functions

Random noise, such as white noise, generates a flat power spectrum when autocorrelated. Enhancements like Power Spectral Equalisation can further refine this property. For instance, let $p(t) \leftrightarrow M^*(\omega)/|M(\omega)|^2$, where $M(\omega)$ represents white noise with uniform power spectral density. In Fourier space, Eq. (1) becomes:

$$S(\omega) = \frac{M(\omega)}{|M(\omega)|^2} F(\omega) + N(\omega)$$

Correlating with $m(t)$, we have:

$$M^*(\omega)S(\omega) = F(\omega)$$

assuming that $m(t) \otimes r(t) = 0$, as two independent random noise functions do not correlate.

This approach aligns with the Wiener-Khinchin theorem (e.g. [28, 29]) states that the power spectral density of a stationary random process is the Fourier transform of its autocorrelation function. Time domain functions with rapid variations (high frequencies) correspond to broad spectra in the frequency domain.

At first glance, random noise functions may appear to provide a recoverable coded version of the information function. However, they require prior knowledge of $m(t)$. This approach can be used as the basis for an encrypted information-hiding scheme, for example. In this case, the spectrum $N(\omega)$ is that of a covertext in which the encrypted spectrum $M^*(\omega)F(\omega)$ is embedded, subject to a given SNR, for example [30, 31]. However, such a scheme is unsuitable for an optimal communication systems where the recipient lacks knowledge on the spectrum $M(\omega)$. In contrast, chirp functions, defined by a single parameter α , offer a simpler and more effective approach for reconstructing information embedded in noise.

7. Phase-only functions

In considering the solution of Eq. (1) for $f(t)$, given $p(t)$, and noting the relationship in Eq. (2), an important question arises: How many impulse response functions $p(t)$ possess a phase-only spectrum? Considering the Fourier transform of a unit Gaussian function,

$$\exp(-\alpha t^2) \leftrightarrow \sqrt{\frac{\pi}{\alpha}} \exp\left(-\frac{\omega^2}{4\alpha}\right),$$

substituting $\alpha \rightarrow -i\alpha$ gives

$$\exp(i\alpha t^2) \leftrightarrow \sqrt{\frac{i\pi}{\alpha}} \exp\left(-\frac{i\omega^2}{4\alpha}\right).$$

In general, this can be expressed as

$$\exp\left(\pm \frac{i\alpha t^2}{2}\right) \leftrightarrow \lambda_{\pm} \exp\left(\mp \frac{i\omega^2}{2\alpha}\right), \quad (3)$$

where

$$\lambda_{\pm} = (1 \pm i) \sqrt{\frac{\pi}{\alpha}}.$$

This implies that under the Fourier transform operator, the chirp function has a conjugate eigenfunction. Specifically, for $\alpha = 1$,

$$\mathcal{F}[\exp(\pm it^2/2)] = \lambda_{\pm} \exp(\mp i\omega^2/2),$$

where $\mathcal{F}[f(t)] \equiv \int_{-\infty}^{\infty} f(t) \exp(-i\omega t) dt$, and $\lambda_{\pm}$ are the eigenvalues.

7.1 Orthogonality of Chirplets

Chirplet-modulated communication schemes rely on the *orthogonality* of chirplets within a time-frequency tiling framework. This orthogonality allows for efficient signal separation and robust demodulation.

Two chirplets $p_i(t)$ and $p_j(t)$ are said to be orthogonal if:

$$\langle p_i, p_j \rangle = \int p_i(t) \cdot p_j^*(t) dt = 0 \quad \text{for } i \neq j$$

Orthogonality can be approximately maintained if chirplets differ in time centre, frequency centre, or chirp rate. In practical implementations, they are quasi-orthogonal due to finite duration and bandwidth constraints. Orthogonality in the time-frequency plane ensures:

- minimal interference between symbols (low cross-correlation);
- greater robustness in noisy environments;
- effective matched filtering at the receiver.

7.2 Minimum SNR for intelligibility

Consider a signal model of the form:

$$s(t) = f(t) + r(t)$$

where $f(t)$ is the transmitted chirplet signal and $r(t)$ is Additive White Gaussian Noise (AWGN). Demodulation uses matched filtering:

$$\langle s(t), p(t) \rangle = \langle f(t), p(t) \rangle + \langle r(t), p(t) \rangle$$

Due to orthogonality, only the intended chirplet returns a strong projection; others yield near-zero results in ideal conditions. For a matched filter receiver, the output signal-to-noise ratio is:

$$\text{SNR}_{\text{out}} = \frac{E}{N}$$

where $E = \|f(t)\|^2$ is the energy of the signal, and N is the noise power spectral density. The bit error rate (BER) for binary detection is:

$$\text{BER} = Q\left(\sqrt{2 \cdot \text{SNR}_{\text{out}}}\right)$$

where Q denotes the Q -function, which describes the tail probability of the standard normal distribution—essentially, the probability that a Gaussian random variable exceeds a certain value. To achieve a BER less than 10^{-2} , a practical minimum SNR is:

$$\text{SNR}_{\text{min}} \approx 5 \text{ to } 7 \text{ dB}$$

The implications of this result for practical considerations are as follows:

- High orthogonality improves resistance to noise and interference.
- Time-bandwidth product (TBP) affects chirplet energy concentration; larger TBP enhances robustness.

Property	Description
Orthogonality	Ensures minimal cross-talk between chirplets, enabling clear symbol separation
Matched Filtering	Maximises SNR at the receiver, allowing for effective demodulation
Minimum SNR	Approximately 5–7 dB under AWGN for intelligible signal recovery
Practical Threshold	Depends on chirplet duration, bandwidth, and chirp rate separation

Table 1.

Summary of key properties influencing chirplet-based modulation under noisy conditions.

- Coherent detection (with known phase/frequency) outperforms non-coherent methods.

These properties are tabulated in **Table 1**.

7.3 Beyond chirps

There are phase-only functions beyond chirps that exhibit special properties under Laplace, wavelet, and even other transform domains (e.g. the Gabor and Mellin transforms). Each transform emphasises different signal characteristics (time, scale, frequency, and poles), and different phase functions can be engineered to exploit those characteristics for purposes such as secure communication, feature extraction, signal modulation, time-scale or time-frequency localisation. For example

$$\mathcal{L}[\exp(i\alpha t)] = \frac{1}{s - i\lambda}$$

where $\mathcal{L}$ denotes the Laplace transform operator, revealing that there is a pole at $s = i\lambda$, indicating resonance or system modes, widely used in control theory and signal synthesis.

Logarithmic phase-only functions such as $\exp(i\log(t))$ or $\exp(i\log^k(t))$ have logarithmic phase evolution and show interesting behaviour under the wavelet transform, which is naturally scale-aware. Wavelets operate well on logarithmic or fractal-like signals, making such phase functions suitable for scale-invariant detection or analysis.

Wavelets offer time-scale localisation rather than time-frequency localisation. Certain phase-only signals can exhibit orthogonality or compact support in scale, depending on the mother wavelet chosen. For example, using a Morlet or complex Mexican hat wavelet (which has a complex-valued envelope), one could analyse a signal characterise by $\exp[i\theta(t)]$ where $\theta(t)$ is non-linear (but smooth). If $\theta(t)$ varies slowly compared to the scale of the wavelet, the transform behaves like a matched filter, emphasising localised oscillatory behaviour even in phase-only data.

Depending on the transform domain and functional form, such a function can satisfy:

- *Orthogonality*: Many phase-only functions are orthogonal over time intervals (especially chirps).
- *Energy-spreading*: Similar to chirps, some phase-only functions “spread out” energy across scales or frequencies.
- Invariance in certain domains (e.g. Laplace transform gives rational expressions).

However, it is arguable that the chirp function is unique, as discussed in the following section.

7.4 Uniqueness of the chirp function

A critical question follows: Is the chirp function the only phase-only function whose spectrum is also phase-only? Although many amplitude-only functions, such as $\text{sech}(x)$, $1/\sqrt{|x|}$, the Dirac comb $\text{comb}(x)$, and $\exp(-x^2/2)H_n(x)$ (where $H_n(x)$ are Hermite polynomials), exhibit self-characterisation under the Fourier transform, they do not satisfy the condition that the power spectrum remains constant.

This uniqueness property leads to the *chirp conjecture*: The chirp function $\exp(\pm it^2/2)$ is the only phase-only function whose Fourier spectrum is also phase-only. Furthermore, functions like $\exp(i\alpha t^n)$ for $n > 2$ do not have closed-form Fourier transforms, and their solutions involve asymptotic or numerical evaluation. Using the method of stationary phase, the Fourier transform of such functions reveals increasingly higher-order phases as n increases, making these functions unsuitable for maintaining a constant power spectrum.

7.5 A formal statement and proof

This uniqueness conjecture can be formally expressed in the following theorem:

Theorem 1.1 If $F(\omega) \leftrightarrow f(t)$, that is

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \exp(-i\omega t) dt, \quad f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) \exp(i\omega t) d\omega,$$

where $F(\omega)$ exists if $f(t)$ is square-integrable ($\|f(t)\|_2^2 < \infty$), then the phase-only function $f(t) = \exp(\pm it^2/2)$ has a conjugate phase-only eigenfunction that is unique.

Proof: Using Eq. (3), for $\alpha = 2$,

$$\exp\left(\pm \frac{it^2}{2}\right) \leftrightarrow (1 \pm i)\sqrt{\pi} \exp\left(\mp \frac{i\omega^2}{2}\right),$$

where $(1 \pm i)\sqrt{\pi}$ are the complex eigenvalues. Consider the ‘Bluestein decomposition’ [32]:

$$\exp\left(i \frac{\omega^2}{2}\right) F(\omega) = \int_{-\infty}^{\infty} f(t) \exp\left[i \frac{(\omega - t)^2}{2}\right] \exp\left(-i \frac{t^2}{2}\right) dt.$$

Then, if $f(t) = \exp(it^2/2)$, $f(t) \exp(-it^2/2) = 1$, and the integral reduces to

$$\exp\left(i \frac{\omega^2}{2}\right) F(\omega) = \int_{-\infty}^{\infty} \exp\left[i \frac{(\omega - t)^2}{2}\right] dt.$$

Since

$$\int_{-\infty}^{\infty} \exp \left[i \frac{(\omega - t)^2}{2} \right] dt = (1 + i) \sqrt{\pi},$$

it therefore follows that

$$F(\omega) = (1 + i) \sqrt{\pi} \exp \left(-i \frac{\omega^2}{2} \right),$$

thereby recovering the eigenfunction relationship in Eq. (3). The uniqueness of this result supports the conjecture.

7.6 Implications and extensions

The conjecture has significant implications. For example, in the search for extraterrestrial intelligence, if an alien civilisation intended to encode information in a robust manner that survives distortion and noise, they might use a chirp-based function due to its unique Fourier properties. Practical implementation would involve compact-support chirps, or *chirplets*, which introduce additional parameters such as the oscillation period T . These aspects are explored in subsequent sections.

7.7 Discussion

The chirp function, characterised by $\phi(t) = \pm t^2/2$ or $\pm(c + t^2/2)$ for a constant c , appears to be the only phase-only function with a conjugate eigenfunction property. While the proof here focuses on quadratic phases, extending it to establish that no other phase function shares this property would convert the conjecture into a full theorem. This unique relationship underscores the fundamental nature of the chirp function in signal processing and beyond.

8. The Chirplet transform applied to functions of compact support

Consider the chirplet transform applied to a function $f(t)$ that is of compact support where $f(t) \exists \forall t \in [-T/2, T/2]$, and defined as

$$\exp(i\alpha t^2) \otimes f(t) = \int_{-T/2}^{T/2} \exp[i\alpha(t - \tau)^2] f(\tau) d\tau$$

Note that $\otimes$ now denotes the finite convolution integral. The autocorrelation function $c(t)$ of the function $p(t) = \exp(i\alpha t^2)$, $t \in [-T/2, T/2]$ is given by

$$c(t) = \exp(-i\alpha t^2) \odot \exp(i\alpha t^2) = \exp(-i\alpha t^2) T \text{sinc}(\alpha T t)$$

where $\text{sinc}(x) \equiv \sin(x)/x$ and $\odot$ denotes the correlation integral. This result is easily derived as follows:

$$c(t) = \int_{-T/2}^{T/2} \exp[-i\alpha(t+\tau)^2] \exp(i\alpha\tau^2) d\tau = \exp(-i\alpha t^2) \\ \times \int_{-T/2}^{T/2} \exp(-2i\alpha t\tau) d\tau = \exp(-i\alpha t^2) T \text{sinc}(\alpha T t)$$

Noting that

$$T \text{sinc}(\alpha T t) \leftrightarrow \frac{\pi}{\alpha} H(\omega), \quad H(\omega) = \begin{cases} 1, & |\omega| \leq \alpha T; \\ 0, & |\omega| > \alpha T. \end{cases}$$

and that

$$\exp(i\alpha t^2) \leftrightarrow (1+i) \left(\frac{\pi}{2\alpha}\right)^{\frac{1}{2}} \exp(-i\omega^2/4\alpha)$$

then, from the product theorem,

$$c(t) \leftrightarrow \frac{1}{2\pi} (1+i) \left(\frac{\pi}{2\alpha}\right)^{\frac{1}{2}} \exp(-i\omega^2/4\alpha) \otimes \frac{\pi}{\alpha} H(\omega)$$

However, if we consider the case when $\alpha T \gg 1$, then $c(t) \simeq T \text{sinc}(\alpha T t)$ and thus, the spectrum of $c(t)$ is considerably simplified to the form

$$C(\omega) \simeq \frac{\pi}{\alpha} H(\omega), \quad \alpha T \gg 1$$

In this case (i.e. the case when $\alpha T \gg 1$), the bandwidth of the autocorrelation function is determined by αT alone. Thus, for a chirplet (a chirp function of a finite extent), increasing the value of the chirp parameter α and/or the time window of the chirplet T linearly increases the bandwidth of the autocorrelation function.

These results yield the following conclusion: By applying a chirplet transformation to an ‘information function’ $f(t)$ and transmitting the transform through a ‘channel’ characterised by additive noise, as denoted by a random variable of time $r(t)$, say, we can consider a model for the recorded signal that is given by

$$s(t) = p(t) \otimes f(t) + r(t)$$

Thus, by correlating this signal with the complex conjugate of $p(t)$ —denoted by $p^*(t)$ —we can derive an estimate $\hat{f}(t)$, say, for $f(t)$ given by

$$\hat{f}(t) = T \text{sinc}(\alpha T t) \otimes f(t) + p^*(t) \odot r(t), \quad \alpha T \gg 1$$

Since

$$T \text{sinc}(\alpha T t) \otimes f(t) \leftrightarrow \frac{\pi}{\alpha} H(\omega) F(\omega)$$

the function $f(t)$ is reconstructed, whose bandwidth is determined by αT , subject to a perturbation determined by the correlation function $p^*(t) \odot r(t)$. Thus, $\hat{f}(t)$ is both a

band-limited and a noise-limited reconstruction of the information function $f(t)$ whose bandwidth is determined by αT subject to ‘stochastic distortion’ by $p^*(t) \odot r(t)$. However, given that $p^*(t)$ and $r(t)$ can be taken to be uncorrelated, then as $T \rightarrow \infty$, $p^*(t) \odot r(t) \rightarrow 0$ and hence,

$$f(t) = p^*(t) \otimes s(t)$$

It is this result that is the fundamental and crucial ‘key’ to using chirplets in a range of applications, including the one that is considered in the section that follows. Coupled with what we should now call the ‘chirplet conjecture’, the results given above identify the chirplet transform as being unique. In the following section, we consider the generic implementation of the chirplet transform on a bit-for-bit basis. This is known as chirplet modulation.

8.1 Chirplet-based modulation: An overview and example

Chirplet-based modulation is a communication technique that uses chirplets that are short, time-localised waveforms that vary in frequency (chirp) and are shaped by parameters like amplitude, phase, duration, and bandwidth. The core idea is based on chirp signals, which are signals in which the frequency increases or decreases over time, as discussed in previous sections. Chirplet-based modulation builds on this by shaping the chirp signals more finely, leading to better adaptability in representing signals across both time and frequency domains.

The method leverages the fact that chirplets, due to their variable frequency, are well-suited for environments with high interference or noise, such as multi-path channels or wireless networks. They can provide a better balance between time localisation (which is good for resisting noise) and frequency localisation (which is useful for bandwidth efficiency). This makes chirplet-based modulation a flexible and robust method in specific scenarios.

8.2 Key characteristics of Chirplet modulation

The key characteristics associated with chirplet modulation include the following:

- i. Time-frequency flexibility: Chirplets have excellent flexibility in the time-frequency domain. One can modulate the frequency, amplitude, and phase of chirplets dynamically, making them adaptable for various channel conditions.
- ii. Resistance to noise: Due to their spread across a range of frequencies, chirplets are less prone to interference or noise compared to conventional signals like sinusoidal waveforms.
- iii. Efficient bandwidth usage: Chirplets can pack information efficiently across the time-frequency plane, making good use of bandwidth while resisting distortion and interference.
- iv. Non-stationary signals: Chirplet modulation is especially useful for non-stationary signals, where the frequency content of the signal changes over time. This is because chirplets are inherently capable of modelling these variations.

8.3 Chirplet signals

A chirplet signal can be described mathematically by controlling parameters like the starting frequency, ending frequency, duration, and time-varying behaviour (amplitude, phase, etc.). A common and more general form of a chirplet is

$$s(t) = A(t) \cos(\omega t + \alpha t^2 + \phi)$$

where $A(t)$ is the time-varying amplitude. ω_0 is the starting frequency, α defines the rate of frequency change (the chirp rate) and ϕ is the phase shift.

A typical example of chirplet-based modulation is where data is encoded into the time-frequency structure of chirplets. The step-by-step process is as follows:

1. Data Mapping: Binary data (e.g. '1' and '0') is mapped onto different chirp rates. A '1' might be represented by an upward frequency chirp (positive chirp rate), while a '0' might be represented by a downward frequency chirp (negative chirp rate). However, there is a range of different techniques that are applicable in this respect, including using a change in the polarity of the chirplet to delineate between a 1 and a 0, for example.
2. Chirplet generation: For each binary digit, a chirplet signal is generated. Thus, based on the encoding scheme discussed in point 1, for bit '1', the signal might start at a low frequency and sweep upwards, and for '0', the signal might start at a high frequency and sweep downwards.
3. Transmission: The modulated chirplet signal is transmitted over a communication channel. Its frequency sweep structure helps it resist interference and noise.
4. Demodulation: On the receiving end, the chirplet signal is analysed by detecting its frequency sweep direction, which allows the receiver to decode the binary information. The receiver detects whether the chirp is upward or downward to recover the original (binary) data stream, typically by correlating the 'chirplet stream' with the original chirplet. For example, if we consider the binary sequence '1010', the transmitted chirplet-modulated signal could be based on the following: '1' → Upward chirp; '0' → Downward chirp; '1' → Upward chirp; '0' → Downward chirp. However, this form of chirplet coding is one of a number that can be applied, and, in this chapter, a bit is encoded using a chirplet that is composed of both a downward and upward chirp—a two-sided chirp. Differentiation of each bit is then predicated on changing the polarity of the (two-sided) chirplet. Irrespective of the precise modulation method that is considered, chirplets, due to their spread in the time-frequency domain, are robust against multi-path interference, noise, or Doppler effects, which is especially useful in wireless or underwater communications, for example.

8.4 Applications of Chirplet-based modulation

The application of chirplet-based modulation includes:

Wireless communication: Chirplet modulation is advantageous in environments with multi-path propagation or high interference. Its time-frequency adaptability makes it ideal for robust communication.

Radar systems: Chirplets are used in radar to detect objects with high resolution and clarity, as they allow for the detection of signals across both time and frequency domains.

Underwater communication: Chirplet modulation is used in sonar systems and underwater communication systems due to its ability to handle noisy, multi-path environments.

Ultra-wideband (UWB) systems: Chirplets are useful in ultra-wideband systems, where the signal is spread over a wide frequency range, providing high data rates with low-power consumption.

In summary, chirplet-based modulation represents a powerful approach to communication in challenging environments. By using time-localised signals with varying frequencies, chirplets offer a robust way to encode and transmit data. The inherent adaptability of chirplets to different channel conditions, along with their noise resistance and efficient bandwidth usage, makes them suitable for a range of applications, from wireless communication to radar and sonar systems, for example. In the following section, a discrete chirplet modulation scheme is presented which is based on a two-sided complex chirplet using a change in polarity to differentiate between a 0 and a 1 in a bit stream.

9. Discrete Chirplet modulation and demodulation

Let $f_n \equiv \{0, 1\}^N$ represent a bitstream consisting of N bits. The function f_n , $n = 1, 2, \dots, N$, serves as a binary representation of information, where each 1 in the stream corresponds to a Kronecker delta function:

$$\delta_{n-m} = \begin{cases} 1, & m = n; \\ 0, & m \neq n. \end{cases}$$

9.1 Modulation

In two-sided chirplet modulation, the bitstream is replaced with a chirplet stream composed of a concatenation of the functions $p_n^\pm = \pm \exp[i\alpha(n - T/2)^2]$, $n \in [0, T]$, where p_n^+ encodes a '1' and p_n^- encodes a '0'. This process involves a bit-to-chirplet stream transformation, denoted by $\rightarrow$, which can be expressed as:

$$\{0, 1\}^N \rightarrow \{\text{cat}(p_n^\pm)\}^{N \times T}$$

where

$$\text{cat}(p_n^\pm) = \begin{cases} p_n^\pm, & n \in [0, T]; \\ p_n^\pm, & n \in [T, 2T]; \\ p_n^\pm, & n \in [2T, 3T]; \\ \vdots & \\ p_n^\pm, & n \in [(N-1)T, NT]. \end{cases}$$

Here, cat represents the concatenation of $p_n^\pm$. The j^{th} bit, f_n , $n = 1, 2, \dots, N$, in the bitstream $\{0, 1\}^N$ is thus transformed into the j^{th} chirplet in the chirplet stream. The output (the transmitted signal) is taken as the real component of $\text{cat}(p_n^\pm)$.

9.2 Demodulation

Chirp modulating a bitstream in this manner allows each bit to be recovered by correlating the stream with the complex conjugate of an identical chirp function, p_n^+ . The output of this correlation can be expressed as:

$$f_n = (p_n^+)^* \odot \text{Re}[\text{cat}(p_n^\pm)] \sim \text{cat}(\delta_{n-m}^\pm)$$

where

$$\delta_{n-m}^\pm = \begin{cases} \pm 1, & m = n; \\ 0, & m \neq n. \end{cases}$$

A subsequent process is required to convert the Kronecker δ -stream back into the original bitstream, that is,

$$\{\text{cat}(\delta_{n-m}^\pm)\}^{N \times T} \rightarrow \{0, 1\}^N.$$

Since each reconstructed Kronecker delta is either positive or negative at positions $(mT/2) - 1$, for $m = 1, 3, 5, \dots$, where T is an even integer, this conversion can be achieved through the following procedure:

$$\forall j \in [1, N] : \begin{cases} \text{if } \text{Re}[s_{n=[(2j-1)T/2]-1}] \geq 0, & f_j = 1; \\ \text{if } \text{Re}[s_{n=[(2j-1)T/2]-1}] < 0, & f_j = 0. \end{cases} \quad (4)$$

Using this process, the original bitstream $\{0, 1\}^N$ is successfully recovered.

9.3 Discussion

The value of T determines the array length used to compute the discrete chirplet. Given T , a value of α must be chosen to prevent aliasing. To derive a non-aliasing criterion for α given T , we observe that the autocorrelation of $\exp(i\alpha t)$, $t \in [-T/2, T/2]$ has a bandwidth approximately equal to αT , provided $\alpha T \gg 1$.

Under this condition, the bandwidth of $\exp(i\alpha t^2)$ is also on the order of αT . From the sampling theorem, it follows that if Δ denotes the sampling interval, then:

$$\Delta \leq \frac{1}{2\alpha T}.$$

Consequently, for $\Delta = 1$, we obtain $\alpha \leq \frac{1}{2T}$, where $\alpha = \frac{1}{2T}$ corresponds to a Nyquist sampled chirplet.

Chirplet modulation applied to each individual bit—rather than performing a chirplet transform on the entire bit stream—offers the potential for enhanced *reconstructive power* at the bit level. However, this benefit comes with a computational *cost*: the original bit stream of length N is expanded into a chirp-stream of length $N \times T$.

Consider the transmission of such a chirp-stream through an additive noise environment, where the received (real-valued) digital signal is recorded in the form (for $n = 1, 2, \dots, NT$):

$$s_n = \text{SNR} \times \text{Re}[\text{cat}(p_n^\pm)] + r_n, \quad \|r_n\|_\infty = 1 \quad (5)$$

then, in the context of the chirp conjecture discussed earlier, we consider the reconstruction of any element of f_n (with bandwidth αT) from s_n to be uniquely optimal.

In a communication system, the two-sided baseband function $\text{cat}(p_n^\pm)$ is assumed to undergo frequency modulation (following digital-to-analogue conversion) for the purpose of transmission. Consequently, the (digital) signal s_n given in Eq. (5) is assumed to result from frequency demodulation back to baseband, followed by analogue-to-digital conversion.

The noise function r_n is therefore influenced by noise characteristics associated with the sideband frequency range $\omega_0 - \alpha T/2$ to $\omega_0 + \alpha T/2$, where ω_0 is the carrier frequency used during frequency-modulated transmission. This bandwidth determines the noise type and signal-to-noise ratio (SNR) of the resulting baseband signal—specifically, it defines both the statistical distribution of r_n and the value of SNR in Eq. (5).

The algorithms presented in the following section are designed to test this model using two prototype Python functions: one for modulating a bit stream and another for demodulating the received signal to recover the bit stream, subject to a specified SNR.

9.4 Numerical implementation in Python

Two Python functions, namely CM (Chirplet Modulation) and CD (Chirplet Demodulation), are provided in Appendix A. In this implementation, the additive noise function is modelled as zero mean and normally distributed.

Both functions are extensively commented to narrate their structure, reflecting the processes and notation introduced earlier using Nyquist-sampled chirplets. They have been designed as a minimalist solution for the numerical implementation of the chirplet modulation method discussed previously. Readers are encouraged to extend or adapt the code to suit their own applications and areas of interest.

Possible extensions include the analysis of bit error rates (BER) resulting from the demodulation process, particularly under varying SNR conditions. One such metric is based on the root mean square error:

$$\text{BER} = \|f_j - \hat{f}_j\|_2$$

where f_j denotes the input bit stream to the CM function, and $\hat{f}_j$ represents the output bit stream recovered by the CD function after demodulation.

The reader should note that in function CD, Eq. (4) has been implemented in order to generate the reconstructed bit stream. This approach guarantees that a bit stream is always produced on output. This includes the case when there is no chirp-modulated signal embedded in the noise. In which case, the bit stream should be assumed to contain unintelligible information that is the result of noise alone, a scenario that becomes increasingly likely as the SNR decreases. Thus, the question arises as to how we can evaluate the intelligibility (or otherwise) of the bit streams that are generated upon demodulation. This is discussed in the following section.

9.5 Chirplet modulation and error correction codes

Chirplet modulation/demodulation and error correction techniques such as low-density parity-check (LDPC) or Reed-Solomon codes (RSC) serve complementary

purposes, but they operate at different conceptual layers in a communication system. LDPC refers to a class of linear error-correcting codes that are widely used in modern digital communication systems because they offer near-optimal performance close to the Shannon limit (the theoretical maximum efficiency of data transmission over a noisy channel).

Working in contexts like SETI, deep-space communication, or unknown signal decoding, then:

- Chirplet demodulation helps uncover potential structured signals embedded in noisy cosmic backgrounds.
- If a digital structure is found, error correction (like RSC or LDPC) can help reconstruct content, even if some parts are corrupted or missing.

Thus, chirplet demodulation is a front-end signal detection and feature extraction tool, whereas LDPC/RSC are back-end error correction tools. They are not mutually exclusive—they complement each other in a full communication pipeline.

Bit error correction algorithms typically assume that the recovered bit stream (based, in this case, on chirplet demodulation) contains intelligible information. However, in the context of bit streams obtained from a SETI signal, before applying such algorithms, tests must be considered to ascertain whether or not the stream contains intelligible information. This is considered for the following section.

10. A test in intelligibility: Generalisation of the Kullback-Leibler divergence

Intelligibility is a relative concept. Consequently, a relative metric must be employed to measure how a binary string compares to another string known to result from a genuinely random process. Moreover, this comparison should be conducted on a statistically significant basis, by assessing how the probability distribution associated with the binary string differs from a reference probability distribution in terms of its information content.

In this context, we consider a generalisation of the Kullback-Leibler divergence, which is a statistical measure used to quantify how one probability distribution differs from a second, reference probability distribution. It is often described as a measure of the ‘distance’ between two distributions, although technically, it is not a true distance metric (since it is not symmetric and does not satisfy the triangle inequality).

For a two continuous distribution $P(x)$ and $Q(x)$, the measure is defined as [33].

$$D_{KL}(P\|Q) = \int_{-\infty}^{\infty} P(x) \log \left[\frac{P(x)}{Q(x)} \right] dx$$

For application to a binary stream, we consider an analogous Relative (Binary) Information Entropy measure, defined as:

$$R = - \sum_{n=1}^2 p_n \log_2 \left(\frac{p_n}{q_n} \right)$$

where p_n represents the binary histogram of the binary string $f_n \equiv \{0, 1\}^L$, say, and q_n represents the binary histogram of a reference binary string $g_n \equiv \{0, 1\}^L$. Both strings are taken to be of a finite length L .

10.1 The relative information entropy signal

Suppose the string f_n represents a non-random, *intelligible* binary sequence (e.g. encoded text from a natural language), while g_n is a genuinely random binary string. We aim to define a metric R that yields a significantly different value when comparing an intelligible string f_n with a random string g_n , as opposed to the case where both f_n and g_n are random.

Ideally, such a metric would allow for a threshold-based classification: if R falls below a certain threshold, f_n could be classified as intelligible; if it exceeds the threshold, f_n would be classified as random. However, this binary decision framework relies on clear statistical significance, which may not always be present, and fails to capture gradual or ambiguous transitions between randomness and intelligibility.

To address this limitation, we instead examine the statistical distinction between the two scenarios:

1. f_n is intelligible and g_n is random.
2. Both f_n and g_n are random binary strings.

We quantify this distinction using a generalisation of the Kullback-Leibler divergence, referred to as the *Relative Entropy Signal* (RES), defined as:

$$R_m = - \sum_{n=1}^2 p_{nm} \log_2 \left(\frac{p_{nm}}{q_{nm}} \right), \quad m = 1, 2, \dots, M \quad (6)$$

Here, q_{nm} denotes the m^{th} binary histogram derived from the m^{th} random bit stream. The values of p_{nm} vary depending on the scenario:

- Case (i): p_{nm} is the m^{th} binary histogram of a non-random (intelligible) binary string.
- Case (ii): p_{nm} is the m^{th} binary histogram of a random binary string.

In each case, Eq. (6) is computed under the assumption that $p_{nm} = p_n$ for all m .

10.2 Basic statistical analysis

The application of this approach, using simulated data based on bit streams derived from natural language, yields the following results:

- *Case (i)*: When one of the strings is intelligible, $R_m > 0$ for all m , and the resulting distribution of R_m is approximately Gaussian with a well-defined non-zero mean, denoted $M_{(i)}$.

- *Case (ii)*: When both strings are random, $R_m > 0$ also produces an approximately Gaussian distribution, but with a higher mean value $M_{(ii)} > M_{(i)}$. The same inequality holds for other statistical measures, including the standard deviation, mode, and median.

However, in both cases, application of the Jarque-Bera (JB) test for normality reveals that the JB statistic is significantly larger than the corresponding χ^2 threshold. This indicates that the null hypothesis of normality must be rejected: the sequence R_m does not conform strictly to a normal distribution [34].

To obtain the statistically significant results, it is necessary to use relatively long binary strings ($L \gg 1$) and a large number of segments for the RES computation ($M \gg 1$).

It is important to emphasise that this test for the “intelligibility” of a binary string is based on a narrow definition of intelligibility; namely, that the string encodes data from a natural language. In a broader context, however, “intelligibility” should be understood as any structure or signal that suggests the string arises from a non-random process.

Since all data can ultimately be represented as binary strings—regardless of the encoding format (e.g. ASCII or otherwise)—this approach offers a general framework for distinguishing between random and non-random binary data. The method involves computing the Relative Entropy Signal (RES) from Eq. (6) and analysing its statistical distribution.

The resulting differences in basic statistics for R_m allow for the differentiation between random and intelligible binary streams, at least across a range of natural languages. Standard statistical metrics can support first-order classification, which can be further refined using additional parameters, such as spectral properties and other signal transformations.

Ultimately, the aim is to analyse a binary stream to assess whether it is truly random, intelligible (i.e. carrying structured or communicable information), or partially random in nature. A further extension to the use of Eq. (6) is to construct feature vectors whose elements are based on computing a broader range of statistical, spectral, and other transformation metrics (from the signal R_m) for application in the development of an expert system, for example. Such an approach will then be expected to be replaced with a deep learning ‘solution’, based on large quantities of simulated (random and intelligible) data.

11. Blind demodulation using prime number factorisation

Suppose an electromagnetic signal is received which, after demodulation to baseband, is known to have resulted from chirplet modulation. However, the values of the chirp rate α and the chirp duration T are unknown. The question then arises: what values of α and T should be used to reconstruct the chirplet of the form

$$\exp(i\alpha t^2), \quad t \in [-T/2, T/2]$$

given that the signal’s bandwidth is known and given by $\Omega = \alpha T / (2\pi)$ Hz?

This is a classic example of a *blind deconvolution* problem. In principle, one could attempt to recover the pair $[\alpha, T]$ by performing an exhaustive search—testing different combinations of α and T and evaluating whether the demodulated output yields an

intelligible bit stream, as defined in the previous section. However, such brute-force methods are computationally infeasible, especially when the search space is continuous or large.

This problem bears some analogy to prime factorisation: given a semi-prime number pq , recovering its prime factors p and q is computationally hard, a fact that underpins the security of the RSA encryption algorithm [35]. In our case, however, the unknowns α and T are not primes but instead a floating-point number and an integer, respectively. Nonetheless, the underlying computational challenge—recovering unknown parameters from a known output—is conceptually similar.

Prime number factorisation has attracted increasing attention due to the rise of quantum computing and the advent of Shor’s algorithm for efficiently factoring large integers [36], along with subsequent developments [37]. In this context, consider a scenario where Alice wishes to communicate securely with Bob using chirplet modulation of a bit stream.

Rather than relying on a separate key exchange protocol—where Alice transmits the chirplet modulation parameters $[\alpha, T]$ to Bob beforehand—Alice instead employs a shared deterministic rule, known to Bob, based on a maximum prime number factorisation scheme. In addition to this rule, the two parties are assumed to share knowledge of the transmission bandwidth.

The rule operates as follows: Values for α and T are derived from the prime factors of a semi-prime number whose value is as close as possible to the maximum representable length of a number defined by the floating-point precision used to express the shared bandwidth. This floating-point precision is assumed to be the highest available on the processors used by both Alice and Bob, ensuring that calculations match exactly on both ends.

Within this scheme, α can be any floating-point number, whereas T —being an integer—must be constrained by a threshold to ensure that the array size used for digital chirplet modulation and demodulation remains computationally manageable.

11.1 Basic protocol

Let B denotes the bandwidth in hertz available for communication between Alice and Bob, expressed as:

$$B = b_1.b_2b_3 \dots b_n = b_1b_2b_3 \dots b_n \times 10^{-(n-1)},$$

where each b_j is a base-10 digit ($b_j \in \{0, 1, 2, \dots, 9\}$), and the sequence $b_1b_2b_3 \dots b_n$ represents the n -digit significand. The value of n corresponds to the decimal precision available to both Alice and Bob.

The goal is to identify the largest pair of prime numbers p and q such that their product, a semi-prime pq , satisfies:

$$pq = b_1b_2b_3 \dots b_m, \quad \text{with } m \leq n,$$

and such that $n - m$ is minimised.

This is achieved by iteratively truncating the least significant digits of the significand—removing one digit at a time from right to left—and checking whether the resulting integer can be factored into exactly two prime numbers. The process stops upon finding the first such instance, at which point p and q define the modulation parameters used for secure communication.

11.2 Numerical example

To illustrate this scheme, suppose Alice and Bob are communicating over a portion of the electromagnetic spectrum with wavelengths ranging from 18 to 21 cm. This corresponds to a frequency range of 1.42 to 1.67 GHz, yielding an available bandwidth of:

$$\Omega = 0.25 \text{ GHz.}$$

Assume further that both Alice and Bob are capable of processing data with a maximum floating-point precision of 16 decimal digits. Using the relation $\Omega = \frac{\alpha T}{2\pi}$, we can compute:

$$\alpha T = 2\pi \times 0.25 = 1.570796326794897.$$

and Alice now needs to choose values of α and T , based on the largest prime numbers whose ‘re-floated product’ yields the closest match to the floating point value of αT as given. In the example given below, this is done by systematically decreasing the length of the string one number at a time until a re-floated product of the prime numbers is found within the floating point precision of αT as specified:

In this case, prime number decomposition yields the following:

$$\begin{aligned} 1570796326794897 &= 2 \times 3 \times 751 \times 348601049; \\ 157079632679489 &= 13 \times 12083048667653. \\ \Rightarrow 1.3 \times 1.2083048667653 &= 1.570796326794890 \end{aligned}$$

Thus, Alice chooses $\alpha = 1.3$ and (rounding up) $T = 121, 1208, 12083, \dots$. However, to produce a Nyquist sampled chirplet, we only need to specify the period of the chirplet T given that $\alpha = 1/2T$. In this way, Alice and Bob are only required to agree on the value of T , or, in the absence of this knowledge, to search through the list of values of T . Aside from a T -search, Bob only needs to know the operational bandwidth of the communications channel to implement chirp demodulation, coupled with the protocol considered.

The key point here is that the bandwidth Ω can be defined with high decimal precision. Provided that Ω is a rational number, the product $\alpha T = 2\pi\Omega$ becomes an irrational number, since π is irrational (i.e. a non-terminating, non-repeating decimal). As a result, αT can be represented to arbitrary decimal accuracy, increasing the complexity of its re-floated representation and, consequently, the size of the prime numbers required in the factorisation process.

Another important consideration is that the bandwidth of the transmitted signal—after chirplet modulation—increases linearly with the value of T . This leads to a demodulated signal with enhanced resolution. However, this improvement comes at the cost of increased computational effort due to the need to process larger arrays. Nonetheless, the trade-off is significant, particularly in scenarios involving low signal-to-noise ratio (SNR) environments.

This approach opens the possibility of using Shor’s algorithm on a quantum computer to factor semi-prime representations of αT , provided that αT is expressed as a decimal number with sufficient precision to make classical factorisation computationally infeasible.

For instance, as of now, no classical algorithm has successfully factored the RSA numbers RSA-240 (240 decimal digits, approximately 795 bits) through to RSA-2048 (617 decimal digits, 2048 bits). It is therefore conceivable that Alice and Bob could adopt

chirplet modulation based on the semi-prime representation of αT , but only if both parties have access to quantum computing resources capable of executing Shor's algorithm. The security of such a communication protocol would be inherently ensured by the computational hardness of the factorisation problem.

More significantly, this protocol may offer a practical strategy for selecting an appropriate value of T when demodulating signals in scenarios where T is a known unknown. This is particularly relevant in the context of analysing cosmic signals using chirplet demodulation, where the intelligibility of output bit streams is used as a diagnostic criterion.

The fundamental question then becomes whether an intelligent agent would choose chirplet modulation as a means of communication in the first place—an 'unknown unknown'. This critical uncertainty forms the conceptual core of the present thesis, in conjunction with the conjecture proposed in Section 7.

12. Summary and conclusions

In revisiting the chirp function, we have developed the conjecture that there exists one and only one phase-only function that possesses a conjugate eigenfunction under the Fourier transformation, specifically the function $\exp(\pm it^2/2)$. We have explored the problem of chirplet modulating a bit stream (using a two-sided chirplet) and briefly outlined a numerical scheme, along with some simplistic Python functions provided in the Appendix, which can be used to test the basic process. These functions can, of course, be extended to include studies of single-sided chirplet modulation and/or up-down frequency sweep chirplets to represent 0 and 1, respectively (or -1 and 1 , as utilised in the functions CM and CD; see Appendix A).

We have further considered a method of testing the reconstructed bit stream for intelligibility, or otherwise. Finally, a protocol has been established which is based on a factorisation of the bandwidth αT of the chirplet by treating the bandwidth in terms of a semi-prime, thereby illustrating a method of securing chirplet modulation which has an analogy with the RSA algorithm. Finally, given that chirplet modulation would appear to provide a uniquely optimal solution for communicating information subject to additive noise with low SNRs, we note that in the example given in Section 11, the bandwidth considered (i.e. 0.25 GHz) has not been chosen arbitrarily; it is the bandwidth of the "waterhole" [38].

The technical aspects of the coding theory presented in this work, including the minimalist functions provided in Appendices A and B, and the intelligibility test discussed in Section 10, are not intended as rigorous, ready-to-implement methods in the search for extraterrestrial intelligence. Instead, they represent a conceptual approach centred on a more fundamental issue: the conjecture that efforts in the search for extraterrestrial intelligence should prioritise the chirplet demodulation of signals, particularly those in the 'waterhole' frequency range. This conjecture is grounded in the apparent uniqueness of the inverse solution to Eq. (1), for the relatively simple impulse response function (IRF) $p(t) = \exp(iat^2)$. Moreover, this IRF is relatively simple to 'engineer' as part of a general-purpose communications system. In this context, we can consider chirp modulation as a biologically inspired solution to the problem, given that so many animals have evolved to use chirps for communication and/or echo location, for example. In the search for extraterrestrial intelligence, chirplet demodulation of radio signals might therefore be taken to be consistent with the evolution of life, at least life on Earth.

This issue still requires the ‘chirp conjecture’, introduced in Section 7.4, to be further developed and rigorously formulated. The argument presented in Section 7.5 is based solely on the apparent uniqueness of the chirp function to retain a conjugate eigenfunction upon Fourier transformation. While this property lends support to the conjecture, it does not constitute a definitive proof that chirp modulation is not merely a viable modulation scheme, but indeed the only one suitable for interstellar communication—at least using electromagnetic (radio) waves. Only if such a proof can be established, might the approach considered in this work provide a solid foundation for the routine analysis of SETI signals using chirplet demodulation alone.

Acknowledgements

The author acknowledges the support of the Science Foundation Ireland and the advice of Prof Roy Hoskins.

Conflict of interest

The author declares no conflict of interest.

A. Prototype Python functions for (two-sided) Chirplet modulation and demodulation

The following prototype (two-sided) chirplet modulation and demodulation functions are written in Python. They are designed to provide readers with a very simplistic and minimalist example of Python code in order to illustrate the basic processing algorithms described in Section 9. In this way, the functions can be easily extended and developed further, according to the interests and applications of the reader.

The Python code is a direct implementation of the algorithm presented in Section 9. This includes the use of a complex chirplet for the sake of consistency with Section 9. However, this complex chirplet can be replaced with just the real or imaginary component alone, that is, $\pm \cos[\alpha(n - T/2)^2]$ or $\pm \sin[\alpha(n - T/2)^2]$, respectively. Demodulation is then undertaken using the same real or imaginary component. In this context, and, the discussion provided in Section 7, we note that $\cos(iat^2)$ and $\sin(iat^2)$ are phase-only functions with a phase-only spectra, that is [39].

$$\cos(iat^2) \leftrightarrow \sqrt{\frac{\pi}{\alpha}} \cos\left(\frac{\omega^2}{4\alpha} - \frac{\pi}{4}\right)$$

and

$$\sin(iat^2) \leftrightarrow -\sqrt{\frac{\pi}{\alpha}} \cos\left(\frac{\omega^2}{4\alpha} - \frac{\pi}{4}\right)$$

In either case (i.e. use of a complex or real chirplet), a single-sided chirp can also be implemented using a chirp-up or chirp-down modulation protocol. These “variations on a theme” of the chirplet modulation process are important in the “engineering” of

the demodulation process and all such variations should be considered in the demodulation of the SETI signals.

On the other hand, the use of a complex chirplet facilitates the inclusion of a complex bit stream, when $f_n \equiv \{0, 1\}^N + i\{0, 1\}^N$. In order to reconstruct a complex bit stream from the real component of the chirp modulated signal (that is transmitted), the Hilbert transform [4] must be applied to recover the complex signal prior to demodulation.

This is achieved using the Python function “hilbert” from “scipy.signal import hilbert,” for example. In this way, the binary information content of the transmission can be increased twofold. The representative Python code required to undertake this is given in Appendix B:

```
import numpy as np
import matplotlib.pyplot as plt

def CM(T, SNR):

    """
    FUNCTION: Two-sided Chirp Modulation (CM) function.

    Parameters:
    T (int): Chirplet array size.
    SNR (float): Signal-to-Noise Ratio (linear scale).

    Behavior:
    - Reads a bitstream from 'bits.txt'.
    - Converts the bitstream into -1/+1 values.
    - Applies chirp modulation.
    - Adds Gaussian noise based on the specified SNR.
    - Plots the resulting signal.
    - Saves the modulated signal to 'cm.txt'.
    """

    # Read bitstream from file 'bits.txt'
    with open('bits.txt', 'r') as file:
        f = file.read()

    # Convert characters to ASCII and then to -1 and 1
    f = np.array([ord(char) for char in f])
    N = f.size
    K = np.where(f == 48, -1, 1) # 48 = '0', 49 = '1'

    # Ensure T is even
    if T % 2 == 1:
        T += 1

    # Generate chirplet
    t = np.arange(1, T + 1) - (T / 2)
    alpha = 1 / (2 * T)
    p = np.exp(1j * alpha * t**2)
```

```
# Modulate each bit with the chirplet
s = np.zeros((N, T), dtype=complex)
for k in range(N):
    s[k, :] = K[k] * p

# Flatten, take real part, and normalize
s = np.real(s.flatten())
s = s / np.max(np.abs(s))

# Plot the result.
plt.subplot(2,1,1); plt.plot(s)
plt.title("Chirp Modulated Signal of 010")
plt.xlabel("Array Value"); plt.ylabel("Amplitude")
plt.grid(True)

# Generate and normalize Gaussian noise
NT = s.size
r = np.random.randn(NT)
r = r / np.max(np.abs(r))

# Add noise and plot
s = SNR * s + r

# Plot the result.
plt.subplot(2,1,2); plt.plot(s)
plt.title("Chirp Modulated Signal with Noise")
plt.xlabel("Array Value"); plt.ylabel("Amplitude")
plt.grid(True);
plt.tight_layout(); plt.show()

# Save to file (with 6 dp accuracy)
np.savetxt('cm.txt', s, fmt='% .6f')

# Example usage:
# T = 1208 from numerical example in Section 11.2
# SNR = 0.1 (linear scale)
CM(1208, 0.1)
```

Example output: If file “bits.txt” contains a string of bits, such as 101,010..., for example, then the chirplet-modulated signal (subject to additive and normally distributed noise) is written to the file cm.txt. The signal depends on the parameters T and SNR. File paths and parameters require adjustment for a specific case. Plots of the chirplet modulation of the bit stream 010 with and without additive noise are provided, as shown in **Figure 1**.

The noisy signal given in **Figure 1** is illustrative of a baseband and digitised signal that is recorded in the search for extraterrestrial intelligence. It is based on a model for encoding binary data that determines the demodulation by correlation with an identical replica chirplet. The function for doing this is as follows:

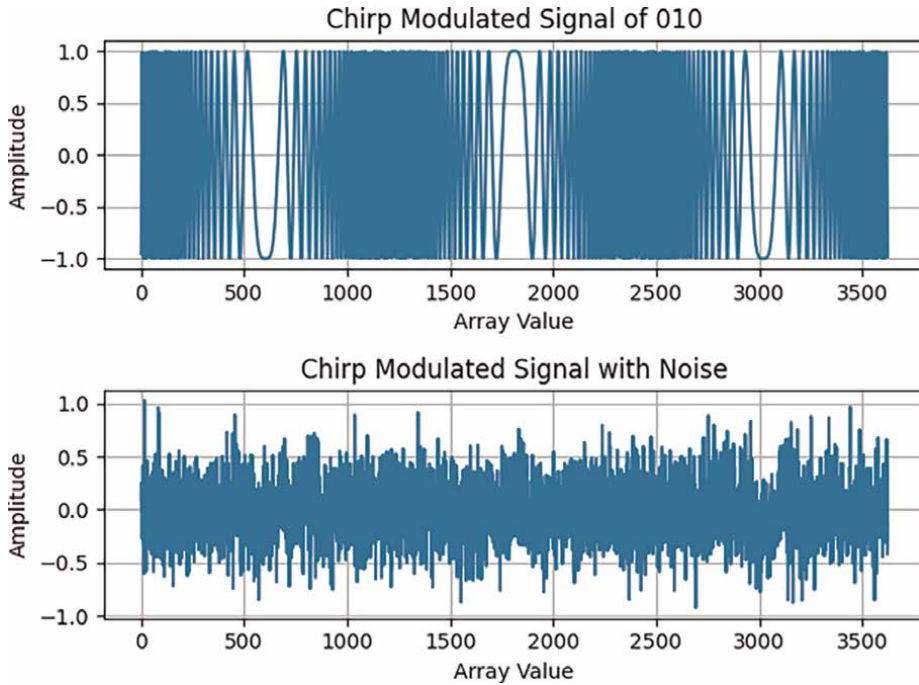


Figure 1. The real component of a (two-sided) chirplet-modulated signal of the bit stream “010” for $T = 1208$ (above) and the modulated signal for $\text{SNR} = 0.1$ (linear scale) $\equiv -10$ dB (below).

```
import numpy as np
import matplotlib.pyplot as plt
```

```
def CD(T):
    """
    Chirp Demodulation (CD) function.

    Parameters:
    T (int): Chirplet array size.

    Outputs:
    - Plots the demodulated signal.
    - Writes the recovered bit stream to 'cd.txt'.
    """
    # Load the chirplet-modulated signal
    s = np.loadtxt('cm.txt')
    # Ensure T is even for symmetric
    # chirplet generation.
    if T % 2 == 1:
        T += 1
    # Time array for chirplet
    t = np.arange(1, T + 1) - (T / 2)
    alpha = 1 / (2 * T)
```

```
# Create complex chirplet
p = np.exp(1j * alpha * t**2)

# Demodulate using convolution with time-reversed
# conjugated chirplet.
d = np.convolve(s, np.conj(np.flip(p)), mode='same')
# Normalize and take real part.
d = np.real(d / np.max(np.abs(d)))

# Plot the demodulated signal
plt.figure(figsize=(10, 4))
plt.plot(d); plt.title('Demodulated Signal')
plt.xlabel('Array Value'); plt.ylabel('Amplitude')
plt.grid(True); plt.tight_layout(); plt.show()

# Recover bitstream by sampling expected
# 'chirplet centres'.
NT = len(d)
num_bits = round(NT / T)
bits = []

for k in range(1, num_bits + 1):
    center_index = int((2 * k - 1) * (T / 2) - 1)
    bit_val = 1 if d[center_index] >= 0 else 0
    bits.append(bit_val)
```

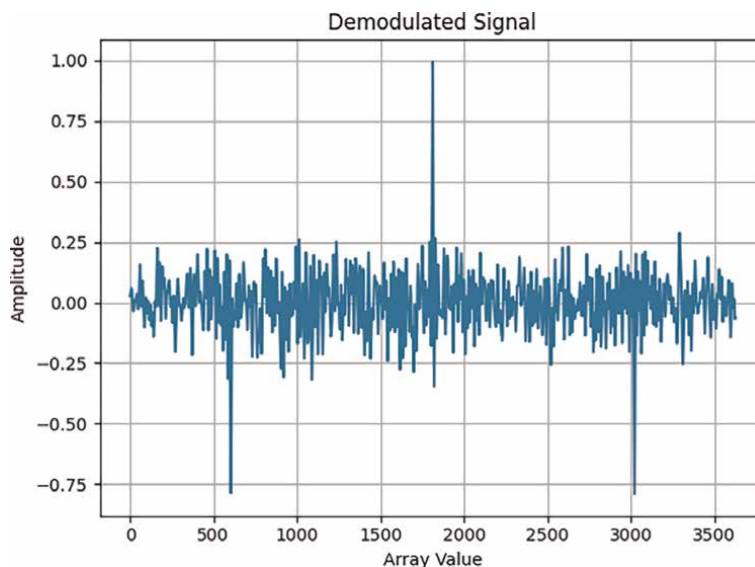


Figure 2.
 Demodulated signal: The result of correlating the noisy chirplet-modulated signal given in **Figure 1** with an identical replica of the (modulating) chirplet. The negative and positive outliers represent the bits associated with the 3-bit stream 010.

```

# Write recovered bits as a continuous
# string to 'cd.txt'.
with open('cd.txt', 'w') as fid:
    fid.write("".join(map(str, bits)))

# Example usage
CD(1208)

```

Example output: The programme inputs file cm.txt, applies chirplet demodulation to the data and output the resulting binary stream to the file cd.txt. The success of this depends on the accuracy of the chirp array size – parameter $T = 1208$. **Figure 2** shows the (accurate) demodulation of the (noisy) signal given in **Figure 1**, illustrating the positions where there is a near-perfect match (through application of the correlation process) to the chirplet used to modulate the binary stream 010.

B. Two-sided Chirplet modulation/demodulation of a complex bit array

The Python functions provided in this appendix are modified versions of the code presented in Appendix A. However, in this case, no provision is given for generating any plots of the data.

The function “CM” performs chirplet modulation on a complex bit stream, where the real and imaginary parts are each composed of an equal number of bits. The function outputs the real part of the modulated signal, incorporating additive noise.

The function “CD” reads this modulated data and applies the Hilbert transform to reconstruct the imaginary component. It then chirplet-demodulates the resulting complex signal, recovering both the real and imaginary bit streams, which are subsequently written to file.

This approach effectively doubles the amount of information encoded in the transmitted bit stream by leveraging the complex domain during chirplet modulation:

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import hilbert

def CM(T, SNR):
    """
    FUNCTION:
    Chirp modulation for a complex bit-stream code.

    Parameters:

    T (int): Chirplet array size.
    SNR (float): Signal-to-Noise Ratio (linear scale).
    Inputs:

    Realbits.txt - Default.txt file
    consisting of the real bit-stream code.

```

Imagbits.txt - Default.txt file
consisting of the imaginary bit-stream code.

Output:

Writes the chirplet modulated signal
with added noise to default file 'cm.txt'.

"""

```
# Read real and imaginary bitstream from files
# into array fr and fi, respectively
with open('Realbits.txt', 'r') as file:
    fr = file.read()
with open('Imagbits.txt', 'r') as file:
    fi = file.read()

# Convert to numeric arrays
fr = np.array([ord(char) for char in fr])
fi = np.array([ord(char) for char in fi])
N = fr.size # Compute size

# Convert arrays (composed of values 48 & 49)
# into array composed of values -1 & 1.
Kr = np.where(fr == 48, -1, 1)
Ki = np.where(fi == 48, -1, 1)
K=Kr+1j*Ki # Generate complex array.

# Compute Nyquist sampled chirplet of length T
# If T is odd, set T to T+1
if T % 2 == 1:
    T += 1

t = np.arange(1, T + 1) - (T / 2)
alpha = 1 / (2 * T)
# Generate complex chirplet array
p = np.exp(1j * alpha * t**2)

# Chirp modulate the bit array
s = np.zeros((N, T), dtype=complex)
for k in range(N):
    for j in range(T):
        s[k, j] = K[k] * p[j]

# Concatenate, take the real part
# and normalize the output.
s = np.real(s.flatten())
s = s / np.max(np.abs(s)) # Normalize
```

```
# Compute normally distributed
# random numbers and normalize.
NT = s.size
r = np.random.randn(NT)
r = r / np.max(np.abs(r)) # Normalize

# Add noise using linear scale SNR,
# and write the signal to file.
s = SNR * s + r
np.savetxt('cm.txt', s, fmt='%.6f')

# Example usage:
CM(1208, 0.1)

import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import hilbert

def CD(T):
    """
    FUNCTION:
    Chirplet demodulation for complex bit-stream code.

    Parameters:

    T (int): Chirplet array size.

    Inputs: cm.txt
    Default.txt file for chirplet modulated signal.
    Outputs:

    Writes the real and imaginary bit-stream to
    default.txt files, cdreal and cdimag, respectively.

    """

    # Read chirplet modulated data, i.e.
    # load data from default.txt file.
    s = np.loadtxt('cm.txt')

    # Compute the Hilbert transform to
    # generate complex array.
    s = hilbert(s)

    # Adjust T if it is odd
    if T % 2 == 1:
        T += 1
```

```
# Generate complex chirplet
t = np.arange(1, T + 1) - (T / 2)
alpha = 1 / (2 * T)
p = np.exp(1j * alpha * t**2)

# Demodulate data using flipped convolution
# with conjugated chirplet.
d = np.convolve(s, np.conj(np.flip(p)), mode='same')
d = d / np.max(np.abs(d)) # Normalize
dr = np.real(d) # Take real part.
di = np.imag(d)

# Recover bit array.
NT = len(dr) # Compute size of array.
step = round(NT / T) # Compute step size.
fr = [] # Initialize the real bit array.
fi = [] # Initialize the imaginary bit array.

for k in range(1, step + 1):
    K = 2 * k - 1
    # Center each chirplet.
    index = int(K * (T / 2) - 1)
    tempr = dr[index]
    tempi = di[index]

    if tempr < 0:
        fr.append(0)
    else:
        fr.append(1)

    if tempi < 0:
        fi.append(0)
    else:
        fi.append(1)

# Write result to file 'cdreal.txt'
with open('cdreal.txt', 'w') as fid:
    # Write bits as a continuous string
    fid.write("".join(map(str, fr)))

# Write result to file 'cdimag.txt'
with open('cdimag.txt', 'w') as fid:
    # Write bits as a continuous string
    fid.write("".join(map(str, fi)))

# Example usage:
CD(1208)
```

Author details

Jonathan Blackledge^{1,2,3,4,5}

1 School of Electrical and Electronic Engineering, Technological University Dublin, Ireland

2 Department of Computer Science, University of Western Cape, South Africa

3 Centre for Advanced Studies, Warsaw University of Technology, Poland

4 Chandigarh Engineering College, Chandigarh Group of Colleges, India

5 INTI International University, Malaysia

*Address all correspondence to: jonathan.blackledge59@gmail.com

IntechOpen

© 2025 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Rowland DI. Giordano Bruno: Philosopher/Heretic. Chicago, Illinois, U.S: University of Chicago Press; 2009. ISBN: 9780226730240
- [2] Perryman M. The Exoplanet Handbook. 2nd ed. Cambridge, UK: Cambridge University Press; 2018. ISBN-13978-1108419772
- [3] Giordano Bruno Memorial Award. New Jersey, US: The SETI League Inc; 2024. [Online]. Available from: <http://www.setileague.org/awards/brunolnk.htm> [Accessed: December 15, 2024]
- [4] Blackledge J. Digital Signal Processing. 2nd ed. Chichester, UK: Horwood Publishing; 2006. ISBN: 1-904275-26-5. [Online]. Available from: <https://arrow.tudublin.ie/engschelebk/4/>
- [5] Pachori RB. Time-Frequency Analysis Techniques and their Applications. Florida, US: CRC Press; 2023. ISBN 9781032435763
- [6] Blackledge J. Cryptography and steganography: New algorithms and applications. Lecture Notes. Warsaw: Centre for Advanced Studies, Warsaw University of Technology; 2011. DOI: 10.21427/D7ZS63. [Online]. Available from: <https://arrow.tudublin.ie/engscheleart2/40/>
- [7] Szilard L. On the decrease in entropy of a thermodynamic system by the intervention of intelligent beings. *Zeitschrift fur Physik*. 1929;65:840-866. [Online]. Available from: <https://fab.cba.mit.edu/classes/863.18/notes/computation/Szilard-1929.pdf> [Accessed: April 24, 2025]
- [8] Tarter J. The search for extraterrestrial intelligence (SETI). *Annual Review of Astronomy and Astrophysics*. 2001;39: 511–548. [Online]. Available from: https://www.researchgate.net/publication/228733522_The_Search_for_Extraterrestrial_Intelligence_SETI [Accessed: June 21, 2025]
- [9] Shostak S. Confessions of an Alien Hunter: A Scientist's Search for Extraterrestrial Intelligence. Washington, US: National Geographic; 2008. ISBN: 139781426203923. [Online]. Available from: https://openlibrary.org/books/OL22666102M/Confessions_of_an_alien_hunter [Accessed: June 21, 2025]
- [10] Cocconi G, Morrison P. Searching for interstellar communications. *Nature*. 1959;184(4690):844-846. [Online]. Available from: <https://aporcel.wordpress.com/wp-content/uploads/2023/11/seti-cocconi-morrison-nature-1959.pdf> [Accessed: June 22, 2025]
- [11] Drake F, Sobel D. Is Anyone out There? The Scientific Search for Extraterrestrial Intelligence. New York, US: Delacorte Press; 1992. ISBN: 9780385305327
- [12] Blackledge J. On the chirp function, the Chirplet transform and the optimal communication of information. *IAENG International Journal of Applied Mathematics*. 2020;50(2):285-319. [Online]. Available from: <https://arrow.tudublin.ie/engscheleart2/218/> [Accessed: December 11, 2024]
- [13] Enriquez JE et al. The breakthrough listen search for intelligent life: 1.1–1.9 GHz observations of 692 nearby stars. *The Astrophysical Journal*. 2017;849(2):104
- [14] Benford J, Benford G, Benford D. Cost-optimised interstellar beacons: METI. *Astrobiology*. 2010;10(5):491-498
- [15] Mann S, Haykin S. The Chirplet transform: Physical considerations. *IEEE*

Transactions on Signal Processing. 1995; **43**(11):2745-2761

[16] IEEE Computer Society. IEEE Standard 802.15.4a-2007. 2007. [Online]. Available from: <https://ieeexplore.ieee.org/document/4299496> [Accessed: December 12, 2024]

[17] Klauder JR, Price AC, Darlington S, Albersheim WJ. The theory and design of chirp radars. Bell System Technical Journal. 1960;**39**(4):745-808. [Online]. Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1960.tb03942.x> [Accessed: December 14, 2024]

[18] Dombi P, Racz P, Veisz S, Baum P. Conversion of chirp in fiber compression. Optics Letters. 2014;**39**(8). Available from: http://real.mtak.hu/23578/1/Dombi_FiChirpConv_OL.pdf [Accessed: December 15, 2024]

[19] Moshinsky M. Diffraction in time. Physical Review. 1952;**88**(625):1952. DOI: 10.1103/PhysRev.88.625

[20] Blackledge JM. Quantum mechanics and control using fractional calculus: A study of the shutter problem for fractional quantum fields. MDPI Applied Mechanics: Special Issue on Mechanics and Control using Fractional Calculus. 2022;**3**(2):413-463. [Online]. Available from: <https://www.mdpi.com/2673-3161/3/2/26> [Accessed: December 15, 2024]

[21] Cook CQ, Ariel A. Theory of chirped photonic crystals in biological broadband reflectors. Optica. 2016;**3**(12):1436-1439. [Online]. Available from: https://www.researchgate.net/publication/311289844_Theory_of_chirped_photonic_crystals_in_biological_broadband_reflectors [Accessed: December 15, 2024]

[22] Al-Ismaili A. Dynamic block encryption with self-authenticating key exchange [PhD thesis]. Loughborough University; 2006

[23] Blackledge JM, Iakovenko O. Resilient digital watermarking for document authentication. IAENG International Journal of Computer Science. 2014;**41**(1):1-17. [Online]. Available from: <https://arrow.tudublin.ie/engscheart2/197/> [Accessed: December 9, 2024]

[24] Tiwari V, Klimenko S, Necula V, Mitselmakher G. Reconstruction of chirp mass in the search of compact binaries. Classical and Quantum Gravity. 2016;**33**(1). [Online]. Available from: <https://arxiv.org/abs/1510.02426> [Accessed: December 11, 2024]

[25] Darlington S. Chirp Pulse Transmission. US Patent 2678997. 1954

[26] Darlington S. Chirp Pulse Equalisation. US Patent 3618095. 1969

[27] Blackledge J, Tobin P. The chirp function revisited: A uniqueness conjecture for Chirplet modulation. In: 30th Irish Signals and Systems Conference (ISSC 2019) Mon 17th - Tue 18th June 2019. Ireland: Hosted by Maynooth University; 2019

[28] Leibovich N, Barkai E. Aging Wiener-Khinchin theorem. Physical Review Letters. 2015;**115**(8):1-6. DOI: 10.1103/PhysRevLett.115.080602. . [Online] Available from: https://www.researchgate.net/publication/278733907_Aging_Wiener-Khinchin_Theorem [Accessed: 10 December 2024]

[29] Champeney DC. Power spectra and Wiener's theorems. In: A Handbook of Fourier Theorems. Cambridge, UK: Cambridge University Press; 1987. ISBN 9780521265034

- [30] Blackledge JM, Tobin P, Myeza J, Adolfo CM. Information hiding with data diffusion using convolutional encoding for super-encryption. *International Journal for Pure and Applied Mathematics (Mathematica Aeterna)*. 2017;7(4):319-356. [Online]. Available from: <https://arrow.tudublin.ie/engscheart2/127/> [Accessed: December 11, 2024]
- [31] Blackledge JM, Tobin P, Myeza J, Adolfo CM. Information hiding using convolutional encoding. In: *ISSC2018, IEEE UK and Ireland Signal Processing Chapter*; 21-22 June 2018; Queen's University Belfast. 2018. [Online]. Available from: <https://arrow.tudublin.ie/engscheart/267/> [Accessed: December 11, 2024]
- [32] Bluestein L. A linear filtering approach to the computation of discrete Fourier transform. *IEEE Transactions on Audio and Electroacoustics*. 1970;18(4):451-455. [Online]. Available from: <https://ieeexplore.ieee.org/document/1162132> [Accessed: December 5, 2024]
- [33] Kullback S, Leibler RA. On information and sufficiency. *Annals of Mathematical Statistics*. 1951;22(1):79-86. DOI: 10.1214/aoms/1177729694
- [34] Blackledge JM, Mosola N. A statistically significant test to evaluate the order or disorder for a binary string of a finite length *ISSC2020*. In: *IEEE UK and Ireland Signal Processing Chapter and IEEE Computational Intelligence Society (UK and Ireland)*, Letterkenny Institute of Technology; 11-12, June 2020. NewYork, US: IEEE; 2020. [Online]. Available from: <https://arrow.tudublin.ie/engscheart/311/>
- [35] Rivest R, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*. 1978;21(2):120-126. [Online]. Available from: <https://dl.acm.org/citation.cfm?doid=359340.359342> [Accessed: December 11, 2024]
- [36] Shor PW. Algorithms for quantum computation: Discrete logarithms and factoring. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science, IEEE Xplore*. New York, US: IEEE; 1994. [Online]. Available from: <https://ieeexplore.ieee.org/document/365700> [Accessed: December 5, 2024]
- [37] Saini K, Gupta BB, Raj P, editors. Post-quantum cryptography algorithms and approaches for IoT and blockchain security. In: *Advances in Computers Series*. Vol. 138. Amsterdam, Netherlands: Elsevier; 2025. ISBN: 9780323914789
- [38] The SETI League Inc. General Information, What is the Water-Hole? California, USA: The SETI League, Inc.; 2019. [Online]. Available from: <http://www.setileague.org/general> [Accessed: December 5, 2024]
- [39] Wikipedia. Fourier Transform. 2024. Available from: https://en.wikipedia.org/wiki/Fourier_transform [Accessed: December 5, 2024]

Quantum-Resilient Cryptographic Techniques for AI-Driven 6G Communication

*A.F.M. Shahen Shah, A.F.M. Suaib Akhter
and Muhammet Ali Karabulut*

Abstract

The rise of 6G communication networks, powered by artificial intelligence (AI), promises a new era of connectivity with ultra-fast data speeds, intelligent applications, and seamless integration across a wide range of devices. However, the immense volume of sensitive data handled by these ultra-dense networks brings heightened concerns about privacy and security. Quantum cryptography, which leverages the principles of quantum mechanics, offers groundbreaking potential for securing communications; yet, the advent of quantum computing poses significant threats to traditional cryptographic systems. This challenge underscores the need for quantum-resilient cryptographic solutions. This review delves into the intersection of quantum-resilient cryptographic techniques and AI-driven 6G communication networks. It provides a detailed examination of existing quantum-resistant algorithms, evaluates their suitability in AI-enabled 6G systems, and highlights critical challenges and potential future directions. By addressing the pressing demand for secure and resilient communication frameworks, this study contributes to the development of 6G systems that can withstand the security threats posed by the quantum computing era.

Keywords: artificial intelligence, quantum cryptography, quantum-resilient cryptographic techniques, security threats, 6G

1. Introduction

Today's fast-growing quantum computing capabilities risk the reliability of traditional encryption methods and constitute a huge security risk to future communication networks. Data security is especially important in new generation communication infrastructures like 6G, which are powered by artificial intelligence and feature enormous data rates. In this context, quantum-resistant cryptography (QRC) is a study field that seeks to assure secure communication by developing and implementing cryptographic algorithms that are resistant to the computational capability of quantum computers. Quantum-resistant cryptographic systems provide security mechanisms that can withstand future quantum attacks by depending on mathematical issues that are impervious to quantum algorithms like Shor's algorithm.

These techniques will play a vital role in protecting confidentiality, data integrity, and authentication in 6G communication networks.

6G communication is the next-generation wireless communication technology that is currently under active development and is expected to offer significant improvements over 5G technology. 6G aims to provide higher speeds, lower latency, and higher capacity. For example, 6G aims to reach data rates of up to 1 Tbps and reduce latency to less than 0.1 ms. These developments will pave the way for next-generation applications such as holographic telepresence, real-time digital twins, and seamless human-machine interaction. The core technologies of 6G include terahertz frequency bands, artificial intelligence (AI) integration, and advanced antenna technologies.

With features such as high speed, low latency, and extensive coverage, 6G communication networks provide an effective platform for the growth of AI-based applications. However, the integration of AI in these networks, particularly with the development of Internet of Things (IoT) devices, raises substantial data security and privacy concerns. At this juncture, QRC is important to ensure that the interaction between 6G and AI is secure. AI algorithms can improve system reliability and efficiency by optimizing QRC procedures, key management, and intrusion detection. On the other hand, QRC helps to protect sensitive data and provide dependable communication by ensuring the secure operation of AI-based 6G apps. Therefore, the co-evolution of 6G and AI will accelerate the development and proliferation of QRC and play an important role in constructing a secure digital future.

With the rapid growth of the information age, technological innovations such as AI, 6G, and QRC are vital to determining the future of our digital world. The interplay of these technologies presents unique prospects for developing a safe, intelligent, and efficient communication infrastructure. 6G provides real-time data analysis and decision-making by delivering the high bandwidth and low latency required for AI applications. On the other hand, AI is utilized to build and optimize QRC algorithms, resulting in stronger protection systems against complex cyber-attacks. QRC serves as the foundation for a dependable and secure digital environment by protecting critical data transported across 6G networks and AI algorithms from future quantum computing threats. The synergistic interaction of these technologies will accelerate innovation, especially in areas such as autonomous vehicles, smart cities, and personalized healthcare, and will provide significant transformations in every aspect of society and the economy.

1.1 Background and motivation

The rise of 6G communication networks, powered by AI, promises a new era of connectivity with ultra-fast data speeds, intelligent applications, and seamless integration across a wide range of devices [1]. However, the immense volume of sensitive data handled by these ultra-dense networks brings heightened concerns about privacy and security. Quantum cryptography, which leverages the principles of quantum mechanics, offers groundbreaking potential for securing communications; yet, the advent of quantum computing poses significant threats to traditional cryptographic systems. This challenge underscores the need for quantum-resilient cryptographic solutions. This chapter explores the intersection of quantum-resilient cryptographic techniques and AI-driven 6G systems, offering a comprehensive analysis of existing quantum-resistant algorithms, their feasibility in AI-enabled 6G environments, and the associated implementation challenges. Furthermore,

it explores future research directions to address vulnerabilities in securing these advanced networks. By tackling the critical need for robust and adaptive security frameworks, this study contributes to the development of future-proof security solutions for a hyperconnected world.

The 6G technology under development aims to reach data speeds of up to 1 Tbps, which represents a significant increase over the current 5G technology. These higher speeds will further increase security vulnerabilities and therefore further emphasize the importance of quantum-resistant cryptographic techniques.

2. Overview of 6G communication networks

This part examines the revolutionary shift from 5G to 6G communication networks, emphasizing the technological innovations that reshape connectivity, network features, and applications. Although 5G has proven to be a significant advancement in providing quicker speeds, reduced latency, and improved device connectivity, 6G imagines a more cohesive and smart communication network. By integrating AI and utilizing terahertz frequency bands, 6G aspires to provide unmatched speed, reliability, and coverage, enabling advanced applications such as holographic telepresence, real-time digital twins, and smooth human-machine collaboration. This segment explores the transition from 5G to 6G, highlighting its essential characteristics, foundational technologies, and the critical function AI serves in promoting flexibility, effectiveness, and safety in this advanced networking framework [2, 3]. A clear and concise view of how 6G surpasses its predecessors in performance, efficiency, and application scope is presented in **Table 1**.

2.1 Role of artificial intelligence in 6G

Artificial Intelligence (AI) is set to significantly influence the creation and functioning of 6G communication networks. As networks grow in complexity, AI guarantees that 6G systems remain efficient, adaptable, secure, and able to satisfy the requirements of new technologies. The role of AI in 6G covers various areas, including resource management and optimization, predictive analytics, and improved security measures. **Figure 1** shows the role of AI in 6G.

Dynamic resource management: AI facilitates instantaneous resource distribution in 6G networks, guaranteeing the efficient allocation of bandwidth, energy, and computing resources. Through the examination of network traffic trends, user activities, and surrounding conditions, AI guarantees optimal resource utilization. This ability is essential for sustaining performance in situations with significant data requirements, like holographic communication and immersive virtual environments [6].

Network optimization: AI algorithms play a crucial role in enhancing the performance and efficiency of 6G networks. By utilizing machine learning, AI detects bottlenecks, forecasts demand variations, and adaptively modifies network setups to enhance throughput and reduce latency. AI aids in energy efficiency by turning network components on or off according to usage, thereby enhancing the sustainability objectives of 6G [7].

Anomaly detection and security: In 6G networks, AI significantly enhances security by identifying and mitigating cyber threats in real-time. AI-based anomaly detection systems monitor network behavior to detect irregular patterns, such as unauthorized access or distributed denial-of-service attack (DoS) attacks. By swiftly addressing

Aspect	4G (LTE)	5G	6G
Data Rate	Up to 1 Gbps	Up to 10 Gbps	1 Tbps or more
Latency	~50 ms	~1 ms	<0.1 ms
Frequency Spectrum	Below 6 GHz	Sub-6 GHz and mmWave (24–100 GHz)	Terahertz (0.1–10 THz)
Network Density	Thousands of devices per square km	Millions of devices per square km	Billions of devices per square km
Connectivity	Mobile broadband	Enhanced broadband, IoT, URLLC	Ubiquitous connectivity, extreme IoT
Energy Efficiency	Moderate	Improved	Highly energy-efficient (green networks)
AI Integration	None	Limited	Fully integrated into network architecture
Use Cases	HD video streaming, web browsing	AR/VR, autonomous vehicles, smart cities	Holographic communication, real-time digital twins, telepresence
Sustainability	Not energy-efficient	Moderate	High focus on energy and environmental efficiency
Security Mechanisms	Basic encryption and authentication	Advanced encryption, limited AI security	AI-driven threat detection and self-healing systems
Coverage	Mostly terrestrial	Terrestrial with limited satellite support	Global, including terrestrial, satellite, and deep-sea connectivity
Technology Enablers	OFDMA, MIMO	Massive MIMO, beamforming, network slicing	Cell-free massive MIMO, reconfigurable surfaces, integrated sensing & communication

Table 1.
A clear and concise view of how 6G surpasses its predecessors in performance, efficiency, and application scope [4, 5].

threats, AI helps maintain the integrity and strength of 6G systems, which face increasing targeting due to their vast scale and vital applications [8].

Traffic management: With billions of devices linked to 6G, AI-driven traffic management guarantees seamless data flow and prioritizes essential applications. AI is capable of categorizing data traffic, forecasting congestion areas, and distributing resources properly to ensure quality of service. For instance, in a medical environment, AI can favor telehealth solutions instead of nonessential data flows to guarantee continuous service [9].

Predictive maintenance: Predictive maintenance is yet another field where AI excels in 6G networks. Through the examination of both real-time and historical performance data, AI can forecast possible system failures and arrange maintenance in advance. This decreases downtime, cuts costs, and guarantees continuous connectivity for users [7].

In summary, AI acts as the foundation of 6G networks, facilitating their agile, smart, and safe functioning. Incorporating it into the fundamental structure of 6G guarantees that these networks can address the demands of a hyperconnected environment, facilitating the emergence of next-generation applications and services [10].

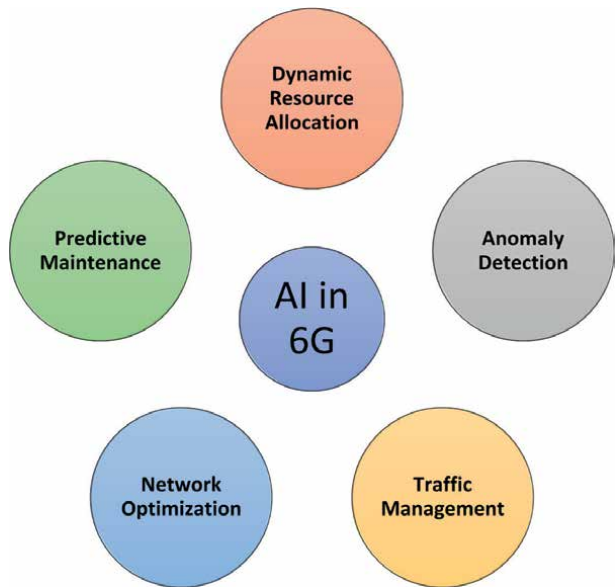


Figure 1.
Role of artificial intelligence in 6G.

3. Security challenges in 6G networks

As 6G technology evolves to meet the demands of a hyperconnected world, its advanced capabilities introduce a new set of security challenges. The integration of AI, edge computing, and terahertz communication expands the threat landscape, requiring novel security solutions to ensure reliability, privacy, and scalability. Emerging use cases, such as autonomous systems, smart grids, and extended reality, bring unique vulnerabilities that demand tailored approaches. This section explores these challenges, starting with the stringent requirements of 6G applications and their implications for security.

3.1 New 6G requirements

Future 6G applications will demand stricter requirements and extended network capabilities compared to 5G [11]. These requirements will enable diverse key use cases while shaping how security measures are implemented in 6G. For Enhanced Ultra-Reliable, Low-Latency Communication (eURLLC), latency-sensitive security workflows must ensure service quality. High reliability necessitates efficient security to protect service availability. In Further Enhanced Mobile Broadband (FeMBB), extreme data rates pose challenges for traffic processing, attack detection, and pervasive encryption. Distributed security solutions, such as local traffic processing using Distributed Ledger Technology (DLT), can alleviate these challenges by offering transparency and redundancy. Similarly, Ultra Massive Machine-Type Communication (umMTC) will drive stringent security needs for the Internet of Everything (IoE), requiring scalable AI/ML-based privacy solutions for resource-constrained and highly mobile devices operating across different administrative domains.

3.2 New applications

6G will drive numerous new technologies and applications, including UAV-based mobility, Connected Autonomous Vehicles (CAV), Smart Grid 2.0, Collaborative Robots, Hyper-Intelligent Healthcare, Industry 5.0, Digital Twins, and extended reality [12]. Each application demands tailored security measures to address its unique requirements and counter advanced attackers. For example, smart grids may need robust encryption, while autonomous vehicles require secure real-time communication protocols.

3.3 Privacy

Privacy protection will be a cornerstone of 6G networks but will face several challenges [13]. The enormous volume of small data exchanges and the integration of intelligence at the network edge increase the risk of data exposure and regulatory difficulties. Additionally, resource-constrained mobile devices running sophisticated applications will face challenges in incorporating privacy-protecting mechanisms. AI/ML introduces both opportunities and risks for privacy, as ML models can either enhance privacy or become targets for attacks such as data poisoning, reverse engineering, and adversarial interference.

3.4 AI and machine learning exploits

AI will be the backbone of 6G network management, optimizing traffic, detecting anomalies, and ensuring seamless connectivity. However, AI itself introduces new risks:

- *Adversarial attacks*: Malicious actors can manipulate AI models with poisoned data or adversarial inputs, compromising network performance and security.
- *Model theft*: Trained AI models can be stolen through side-channel attacks, leading to intellectual property theft and network disruptions.
- *Bias exploitation*: Inaccuracies in training datasets can be exploited to bypass security measures, resulting in misclassification of threats [14].

3.5 Post-quantum cryptographic threats

6G networks must be resilient to quantum computing, which threatens to break traditional cryptographic schemes such as rivest–shamir–adleman (RSA) and elliptic curve cryptography (ECC):

- *Need for quantum-resilient algorithms*: Transitioning to post-quantum cryptography (PQC) requires secure and efficient algorithms capable of handling the demands of high speed, high-volume 6G communication.
- *Integration complexity*: Updating legacy systems and ensuring seamless compatibility with PQC standards presents significant challenges [15].

3.6 Security in software-defined networks (SDN) and network slicing

6G networks rely heavily on SDN and network slicing for flexibility and efficiency, but these features also introduce vulnerabilities:

- *Slice-specific threats*: Each network slice, tailored to a specific use case, may have unique security requirements. A breach in one slice could potentially compromise others.
- *Centralized control risks*: The SDN controller, as a single point of management, becomes an attractive target for attackers [16].

3.7 Physical layer security

6G's reliance on THz communication and massive multiple-input, multiple-output (MIMO) systems brings new physical layer security concerns:

- *Eavesdropping risks*: High-frequency signals are susceptible to eavesdropping due to their directional propagation.
- *Jamming attacks*: The dense deployment of devices and reliance on shared spectrum make 6G networks more vulnerable to jamming [17].

3.8 Cyber-physical system (CPS) security

6G networks will be integral to critical infrastructure, including healthcare, transportation, and energy systems. Securing these cyber-physical systems is critical:

- *Life-critical applications*: A breach in CPS could endanger lives, such as in autonomous vehicles or remote surgeries.
- *Supply chain attacks*: Compromising hardware or software during manufacturing can introduce vulnerabilities at scale [18].

3.9 Regulatory and standardization challenges

6G development is progressing rapidly, but the absence of standardized global security protocols poses risks:

- *Fragmented standards*: Inconsistent security frameworks across countries could leave gaps exploitable by attackers.
- *Slow adoption*: Regulatory bodies may lag behind technological advancements, leading to unsecured deployments [19].

4. Quantum cryptography: An overview

4.1 Principles of quantum mechanics in cryptography

Quantum cryptography utilizes the distinctive concepts of quantum mechanics to enable secure communication. Essential principles comprise:

- *Superposition*: Quantum bits (qubits) can occupy various states at the same time, allowing for intricate encoding methods. This concept is essential for the improved security and computational capabilities of quantum systems [20].

- *Entanglement*: Linked quantum particles guarantee that any disturbance to one particle is observable in its entangled partner. This feature allows for secure communication pathways, where any interference affects the entangled state [21, 22].
- *No-cloning theorem*: An exact duplicate of an unknown quantum state cannot be made, which allows for the detection of eavesdropping. This theorem is essential for safeguarding the integrity of transmitted quantum information [23].

These principles form the foundation of quantum cryptographic methods, offering a level of security unattainable with classical techniques.

4.2 Key distribution protocols

Quantum cryptography is best known for its application in quantum key distribution (QKD), which enables two parties to share encryption keys securely. Prominent protocols include:

- *BB84 protocol*: The initial functional QKD protocol that utilizes polarized photons for the secure encoding and transmission of information [24]. This protocol remains the cornerstone of modern QKD systems.
- *E91 protocol*: Based on quantum entanglement, this protocol guarantees that any eavesdropping efforts disturb the entangled state, indicating the breach [25].
- *Decoy state QKD*: An improvement over BB84, this method incorporates decoy states to identify photon number splitting attacks, thus boosting security and effectiveness [26].

These protocols exemplify the potential of quantum cryptography in achieving unbreakable key exchanges.

4.3 Strengths and limitations of quantum cryptography

4.3.1 Strengths

- *Unconditional security*: Security is rooted in the laws of physics rather than computational complexity, providing resilience against current and future attacks [27].
- *Eavesdropping detection*: Any attempt to intercept communication alters the quantum states, alerting the communicating parties [28].
- *Future proofing*: Resistant to quantum computer-based attacks on classical encryption methods, ensuring long-term data security [29].

4.3.2 Limitations

- *Hardware challenges*: Quantum devices require highly sensitive and precise instruments, such as single-photon detectors and entangled photon sources, for operation [30].

- *Scalability issues:* Deploying quantum cryptography on a large-scale is complex and resource-intensive due to current technological limitations [31].
- *Limited communication distance:* Signal degradation in quantum channels restricts the range of QKD systems. Efforts like satellite-based QKD are being explored to overcome this limitation [32].

While quantum cryptography offers transformative security capabilities, its practical adoption requires overcoming significant technical and logistical hurdles.

5. Quantum-resilient cryptographic techniques

With the advancement of quantum computing technology, the cryptographic algorithms safeguarding existing communication systems are becoming more and more exposed. This chapter investigates quantum-resistant cryptographic methods crucial for safeguarding 6G communication systems, highlighting post-quantum cryptography (PQC) along with its application.

5.1 Post-quantum cryptography: A primer

Post-quantum cryptography (PQC) denotes algorithms intended to withstand assaults from quantum computers while still being compatible with traditional systems. In contrast to quantum cryptography, which employs quantum mechanics principles, PQC modifies conventional cryptographic systems to protect against quantum dangers. These algorithms seek to ensure a smooth shift into the quantum age without needing specific quantum equipment. Recent initiatives by groups such as the National Institute of Standards and Technology (NIST) have been crucial in standardizing PQC techniques to protect digital communication systems [33].

5.2 Prominent quantum-resistant algorithms

Various post-quantum cryptographic algorithms have been developed, each leveraging different mathematical foundations to counteract quantum threats. Here is an overview of the most prominent approaches:

5.2.1 Lattice-based cryptography

Overview: Lattice-based cryptography relies on the hardness of problems like the Shortest Vector Problem (SVP) and Learning with Errors (LWE). These problems are computationally challenging for both classical and quantum computers, making them a strong foundation for cryptographic systems.

Advantages: High computational efficiency and versatility, supporting encryption, key exchange, and digital signature schemes.

Applications: Widely used in NIST finalist schemes like Kyber (encryption) and Dilithium (digital signatures), which offer robust security and practical implementation [34].

5.2.2 Code-based cryptography

Overview: Code-based cryptography builds on the difficulty of decoding random linear codes, as demonstrated in the McEliece cryptosystem.

Advantages: Long-standing security track record and resistance to quantum attacks.

Challenges: Requires larger key sizes, which can hinder performance in resource-constrained environments.

Applications: Suitable for environments where security outweighs constraints on memory and bandwidth [35].

5.2.3 Multivariate quadratic cryptography

Overview: Multivariate quadratic cryptography uses systems of multivariate quadratic equations over finite fields for security.

Advantages: Efficient computation and compact signature sizes make this approach attractive for specific use cases.

Challenges: Vulnerable to specialized algebraic attacks, leading to limited adoption in large-scale systems.

5.2.4 Hash-based cryptography

Overview: Hash-based cryptography relies on the collision resistance of cryptographic hash functions, with Merkle trees and eXtended Merkle Signature Scheme (XMSS) being prominent examples.

Advantages: Simple design and minimal reliance on untested mathematical assumptions, ensuring long-term security.

Challenges: Inefficient for applications requiring frequent key exchanges due to constraints like one-time key usage [36].

5.3 Comparative analysis of post-quantum algorithms

Each quantum-resistant algorithm offers unique benefits and trade-offs, making them suitable for different applications in AI-driven 6G communication:

Lattice-based cryptography: Known for its high efficiency, it requires rigorous parameter optimization to prevent side-channel attacks. It is ideal for general-purpose encryption in 6G systems.

Code-based cryptography: Provides strong security guarantees but struggles with scalability due to its large key sizes. It is suitable for high-security, low-latency applications like secure control systems.

Hash-based cryptography: Simplistic and well-understood, but its limitations in dynamic key exchanges make it less suitable for high-frequency communication networks.

The choice of algorithm depends on the specific requirements of 6G systems, including latency, bandwidth, computational resources, and energy constraints. The adoption of hybrid systems combining multiple quantum-resilient techniques is also being explored for comprehensive security solutions [37].

6. Applicability in AI-driven 6G networks

Incorporating quantum-resilient cryptographic methods into AI-based 6G networks is essential for providing secure, reliable, and efficient communication

in future applications. By protecting data integrity and privacy, allowing real-time encryption, and facilitating AI-driven anomaly detection, these methods tackle the increased security challenges presented by quantum computing. Moreover, strong systems for authentication, key management, and end-to-end encryption provide consistent security throughout versatile and diverse 6G settings. Finding a balance between these security measures and performance needs is crucial for real-time AI applications such as autonomous systems, smart healthcare, and immersive technologies, establishing a secure and scalable 6G environment.

6.1 Integrating quantum-resilient cryptography with AI systems

The integration of quantum-resilient cryptographic techniques into AI-driven 6G networks is essential for ensuring secure and efficient communication. Key considerations include:

Data integrity and privacy: Post-quantum cryptography (PQC) ensures that sensitive AI-generated data, such as predictive models, user analytics, and decision-making outputs, remains secure against the computational threats posed by quantum computers. For instance, lattice-based schemes like Learning with Errors (LWE) can secure critical AI datasets used in smart city infrastructures or financial modeling systems. This is particularly crucial for protecting federated learning systems, where user devices collaborate on model training without compromising data privacy [38].

Real-time encryption: AI-enabled systems in 6G, such as autonomous drones and real-time analytics platforms, require rapid encryption and decryption processes. Techniques like isogeny-based cryptography, which offer smaller key sizes, are promising candidates for achieving both security and computational efficiency in scenarios demanding real-time data exchange [39].

AI-enhanced security: Machine learning models can complement cryptographic techniques by providing real-time anomaly detection and breach prediction. For example, an AI model could monitor patterns in encrypted traffic and detect deviations indicative of potential attacks, such as quantum-assisted brute-force decryption attempts. This layered security framework combines proactive AI-driven insights with robust cryptographic protocols [40].

6.2 Security frameworks for 6G communication

Quantum-resilient cryptographic techniques are instrumental in designing robust security frameworks for AI-driven 6G networks:

- *Authentication protocols:* Lattice-based cryptography, such as the NTRUEncrypt protocol, is well-suited for secure authentication in 6G. For instance, these protocols can secure communication between IoT devices in smart homes or industrial settings, preventing unauthorized access or impersonation attacks [41].
- *Key management systems:* Dynamic 6G environments, such as autonomous vehicular networks or drone swarms, require efficient and secure key distribution mechanisms. AI can optimize these processes by predicting network changes and preemptively generating quantum-resistant keys. For example, reinforcement learning algorithms can adapt key renewal frequencies based on network conditions, ensuring seamless integration of PQC in dynamic scenarios [41].

- *End-to-end encryption*: Secure communication across devices, from IoT end-points to centralized AI systems, can be achieved through post-quantum encryption techniques like hash-based signatures (e.g., XMSS). This is particularly important in critical applications such as military communication systems or national healthcare networks, where secure end-to-end data transmission is paramount [42].

6.3 Performance implications in real-time AI applications

Quantum-resilient cryptographic solutions must balance security with performance, especially in real-time AI applications where latency and resource efficiency are critical.

- *Autonomous systems*: Ultra-low latency is crucial for AI-driven autonomous vehicles, drones, and robotics. For example, a self-driving car in a 6G-connected smart city must exchange encrypted data on traffic conditions with nearby vehicles and infrastructure in milliseconds. Lightweight quantum-resistant algorithms like ring-LWE can provide the required balance between speed and security in these scenarios [43].
- *Smart healthcare*: 6G-enabled AI systems in telemedicine, such as real-time diagnostic platforms and robotic surgery, demand robust encryption to protect sensitive patient data while ensuring uninterrupted service. For instance, a surgical robot performing a remote operation must communicate securely with medical professionals in real-time, where any delay caused by encryption overhead could be life-threatening [44].
- *Immersive applications*: Augmented reality (AR) and virtual reality (VR) environments powered by 6G require lightweight cryptographic solutions to avoid disruptions in user experience. For instance, a VR-based remote education platform must ensure the secure streaming of interactive content to students worldwide without compromising on responsiveness. Cryptographic protocols optimized for low-power devices and edge computing can enhance these applications [45].

7. Challenges and open research directions

7.1 Technical and implementation barriers

The adoption of quantum-resilient cryptographic techniques in AI-driven 6G networks faces several technical and practical challenges:

1. *Computational overhead*: Post-quantum cryptographic algorithms, such as lattice-based and code-based cryptography, demand significantly higher computational resources compared to classical algorithms. This requirement can strain devices with limited processing power, including IoT nodes and edge devices, potentially affecting overall network efficiency and latency [46].
2. *Key size and bandwidth*: Many post-quantum cryptographic schemes require substantially larger key sizes and signature lengths. This increases storage re-

quirements and impacts transmission efficiency, particularly in bandwidth-constrained environments characteristic of 6G networks. These challenges become critical for applications involving massive IoT and ultra-reliable low-latency communication (URLLC) [47].

3. *Compatibility issues*: Incorporating quantum-resilient cryptographic methods into existing network infrastructures and communication protocols often requires significant modifications. This leads to compatibility concerns, higher implementation costs, and a complex transition phase, making backward compatibility with legacy systems a substantial hurdle [46].
4. *AI-driven optimizations*: AI-driven 6G networks rely on dynamic, adaptive configurations. Implementing AI-based optimization for managing cryptographic processes, such as key management and real-time authentication, is a complex task. The interplay between cryptographic demands and AI algorithm overheads further complicates system-level optimizations [48].

7.2 Standardization efforts for post-quantum cryptography

The successful implementation of post-quantum cryptography in 6G communication systems necessitates a standardized global framework to ensure interoperability, security, and efficiency.

1. *NIST post-quantum cryptography standardization*: The National Institute of Standards and Technology (NIST) is leading the global effort to identify, evaluate, and standardize post-quantum cryptographic algorithms. These efforts aim to balance the trade-offs between security, computational efficiency, and practical implementation challenges [49].
2. *Industry collaboration*: Collaboration among academia, industry, and government bodies is critical to fostering the development of practical and interoperable post-quantum cryptographic standards. Initiatives like the European Telecommunications Standards Institute (ETSI) [50] and the International Telecommunication Union (ITU) play a pivotal role in harmonizing efforts across stakeholders [51].
3. *Regulatory challenges*: Cross-border data encryption laws and regulations pose challenges to the deployment of standardized cryptographic solutions. Ensuring compliance with diverse legal frameworks is essential for the seamless adoption of post-quantum cryptography [10].

7.3 Future directions in AI and quantum-resilient security

The intersection of AI and quantum-resilient cryptographic techniques opens several promising avenues for future research and development:

1. *AI-assisted cryptography*: AI can be leveraged to enhance cryptographic algorithms by optimizing key generation, identifying vulnerabilities, and enabling adaptive defense mechanisms against evolving threats. This dynamic approach aligns well with the flexible architecture of 6G networks [52].

2. *Hybrid cryptographic models*: Combining classical, quantum, and post-quantum cryptographic techniques can create robust, multilayered security frameworks. These models can provide resilience against threats originating from both classical and quantum computing capabilities [53].
3. *Lightweight cryptography*: Developing energy-efficient post-quantum cryptographic solutions is crucial for IoT and edge devices with limited power and computational resources. Research in this domain focuses on minimizing key sizes, reducing algorithmic complexity, and improving energy efficiency without compromising security [54].
4. *Quantum cryptography integration*: Exploring the integration of quantum cryptography (e.g., quantum key distribution) with post-quantum techniques is an emerging area of research. This approach aims to create a unified security paradigm that addresses threats from both current and future quantum technologies [55].

These challenges and research opportunities emphasize the importance of interdisciplinary collaboration and innovation in safeguarding 6G networks against emerging security threats.

8. Case studies and applications

The integration of quantum-resilient cryptography into AI-driven 6G systems is being actively explored across various real-world scenarios:

8.1 Examples of AI-driven 6G use cases

- *Smart cities*: Secure communication protocols are essential for AI-driven infrastructure, such as intelligent traffic management systems and automated energy grids, to operate without the risk of data breaches. The implementation of quantum-resistant cryptographic algorithms ensures that the vast amounts of data exchanged within smart cities remain protected from potential quantum attacks [56].
- *Healthcare systems*: In smart healthcare, post-quantum cryptographic techniques safeguard sensitive medical records while ensuring the reliability of AI applications like remote surgery and diagnostics. The adoption of quantum-safe encryption methods is crucial to maintaining patient confidentiality and the integrity of medical data against future quantum threats [57].
- *Autonomous vehicles*: Self-driving cars depend on real-time communication with infrastructure and other vehicles. Quantum-resilient encryption ensures secure exchanges, preventing malicious interference that could compromise safety. The integration of post-quantum cryptography in vehicular communication systems is vital to protect against emerging cyber threats in the quantum era [58].

8.2 Practical deployment of quantum-resilient solutions

- *Quantum-secure IoT networks*: Pilot studies in IoT deployments have implemented lattice-based cryptography to secure data from sensors in smart factories,

demonstrating scalability and efficiency. These studies highlight the feasibility of deploying quantum-resistant algorithms in resource-constrained IoT devices, ensuring long-term security [59].

- *Quantum key distribution trials:* Telecommunications companies have initiated trials integrating quantum key distribution (QKD) systems with 6G infrastructure to enhance data security for AI applications. For instance, O₂ Telefónica has been exploring QKD to secure mobile networks, aiming to create quantum-safe communication channels resistant to future quantum attacks [60].
- *AI-augmented security frameworks:* AI algorithms are being utilized to monitor and adapt post-quantum cryptographic processes in dynamic 6G environments, ensuring optimized performance and security. Projects like CONFIDENTIAL6G focus on developing cryptographic quantum-resistant protocols and security-proof tools, leveraging AI to enhance confidentiality in 6G networks [61].

8.3 Lessons learned from pilot studies

8.3.1 Successes

- Post-quantum algorithms have demonstrated strong resistance to potential quantum attacks, providing a robust foundation for future-proof security in 6G networks [62].
- Hybrid cryptographic models, combining traditional and quantum-resistant techniques, have facilitated smoother transition strategies, allowing for the gradual integration of post-quantum solutions without disrupting existing systems [63].

8.3.2 Challenges

- Increased computational overhead has impacted real-time applications, particularly in resource-constrained environments, necessitating further optimization of post-quantum algorithms to balance security and performance [64].
- Lack of standardization and interoperability has posed barriers to large-scale deployment, highlighting the need for unified standards and collaborative efforts to ensure seamless integration of quantum-resilient cryptography across diverse platforms [65].

These case studies underscore the potential of quantum-resilient cryptography in securing AI-driven 6G networks, while also emphasizing the necessity for ongoing refinement and scalability to address emerging challenges in the quantum era.

9. Conclusion

With the spread of 6G communication networks and the acceleration of AI integration, the importance of quantum-resistant cryptographic techniques is increasing. In order to build a secure and reliable digital ecosystem, the development of QRC

algorithms and their integration into 6G infrastructure is of great importance. In this way, a communication network that is protected against the potential threats of quantum computers can be established, and the confidentiality of sensitive data can be secured. The combined use of AI and QRC will form the basis of smart and secure communication systems of the future and accelerate the digital transformation of society. By exploring the synergistic link between 6G, AI, and QRC, future research on how to improve their integration for safe and intelligent communication systems will be guided.

Author details

A.F.M. Shahan Shah^{1*}, A.F.M. Suaib Akhter² and Muhammet Ali Karabulut³

1 Department of Electronics and Communication Engineering, Yildiz Technical University, Istanbul, Turkey

2 Department of Computer Engineering, Sakarya University of Applied Science, Sakarya, Turkey

3 Department of Electronics Engineering, Turkish Air Force Academy, National Defense University, Istanbul, Turkey

*Address all correspondence to: shah@yildiz.edu.tr

IntechOpen

© 2025 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Shah AFMS. A survey from 1G to 5G including the advent of 6G: Architectures, multiple access techniques, and emerging technologies. In: Proc. of IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA: IEEE; 2022. pp. 1117-1123
- [2] Raihan A. An overview of the implications of artificial intelligence (AI) in sixth generation (6G) communication network. Research Briefs on Information and Communication Technology Evolution. 2023;**9**:120-146
- [3] Telecoming. 2025. Available from: <https://www.telecoming.com/blog/the-road-to-6g-evolution-of-5g-technology/>
- [4] Mihret E, Haile G. 4G, 5G, 6G, 7G and future mobile technologies. Journal of Computer Science and Information Technology. 2021;**9**(2):75
- [5] Kshirsagar PR, Reddy DH, Dhingra M, Dhabliya D, Gupta A. A review on comparative study of 4G, 5G and 6G networks. In: In 2022 5th International Conference on Contemporary Computing and Informatics (IC3I). Uttar Pradesh, India: IEEE; 2022. pp. 1830-1833
- [6] Ponnusamy V, Vasuki A. AI-enabled intelligent resource management in 6G. In: AI and Blockchain Technology in 6G Wireless Network. Springer Nature Singapore: Singapore; 2022. pp. 71-92
- [7] Suresh K et al. Sustainable resource allocation and base station optimization using hybrid deep learning models in 6G wireless networks. Sustainability. 2024;**16**(17):7253
- [8] Saeed MM et al. Anomaly detection in 6G networks using machine learning methods. Electronics. 2023;**12**(15):3300
- [9] Tariq M, Naeem F, Poor HV. Toward experience-driven traffic management and orchestration in digital-twin-enabled 6G networks. arXiv preprint arXiv:2201.04259. 2022
- [10] Chataut R, Nankya M, Akl R. 6G networks and the AI revolution—Exploring technologies, applications, and emerging challenges. Sensors. 2024;**24**(6):1888
- [11] Alwis CD. et al. Survey on 6G Frontiers: Trends, Applications, Requirements, Technologies and Future Research. IEEE Open Journal of the Communications Society. 2021;**2**:836-886
- [12] You X et al. Towards 6G wireless communication networks: Vision enabling technologies and new paradigm shifts. SCIENCE CHINA Information Sciences. 2021;**64**(1):1-74
- [13] Wang M et al. Security and privacy in 6G networks: New areas and new challenges. Digital Communications and Networks. 2020;**6**(3):281-291
- [14] Ylianttila M et al. 6G white paper: Research challenges for trust, security and privacy. arXiv preprint arXiv:2004.11665. 2020
- [15] Chamola V, Jolfaei A, Chanana V, Parashari P, Hassija V. Information security in the post quantum era for 5G and beyond networks: Threats to existing cryptography, and post-quantum cryptography. Computer Communications. 2021;**176**:99-118

- [16] Cunha J et al. Enhancing network slicing security: Machine learning, software-defined networking, and network functions virtualization-driven strategies. *Future Internet*. 2024;**16**(7):226
- [17] Ara I, Kelley B. Physical layer security for 6G: Toward achieving intelligent native security at layer-1. *IEEE Access*. 2024;**12**:82800-82824
- [18] RS, P. An intelligent dynamic cyber physical system threat detection system for ensuring secured communication in 6G autonomous vehicle networks. *Scientific Reports*. 2024;**14**(1):20795
- [19] Alsaedi WK, et al. Spectrum Options and Allocations for 6G: A Regulatory and Standardization Review. *IEEE Open Journal of the Communications Society*. 2023;**4**:1787-1812
- [20] Clarke J, Wilhelm FK. Superconducting quantum bits. *Nature*. 2008;**453**(7198):1031-1042
- [21] Einstein A, Podolsky B, Rosen N. Can quantum-mechanical description of physical reality be considered complete? *Physical Review*. 1935;**47**(10):777-780
- [22] Bell JS. On the Einstein-Podolsky-Rosen paradox. *Physics Physique Физика*. 1964;**1**(3):195-200
- [23] Wootters WK, Zurek WH. A single quantum cannot be cloned. *Nature*. 1982;**299**:802-803
- [24] Bennett CH, Brassard G. Quantum cryptography: Public key distribution and coin tossing. In: *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing*. Bangalore, India: IEEE; 1984
- [25] Ekert AK. Quantum cryptography based on Bell's theorem. *Physical Review Letters*. 1991;**67**(6):661-663
- [26] Hwang W-Y. Quantum key distribution with high loss: Toward global secure communication. *Physical Review Letters*. 2003;**91**(5):057901
- [27] Scarani V, Bechmann-Pasquinucci H, Cerf NJ, et al. The security of practical quantum key distribution. *Reviews of Modern Physics*. 2009;**81**(3):1301-1350
- [28] Gisin N, Ribordy G, Tittel W, Zbinden H. Quantum cryptography. *Reviews of Modern Physics*. 2002;**74**(1):145-195
- [29] Shor PW. Algorithms for quantum computation: Discrete logarithms and factoring. In: *Proceedings of 35th Annual Symposium on Foundations of Computer Science*. 1994. pp. 124-134
- [30] Diamanti E, Lo H-K, Qi B, Yuan Z. Practical challenges in quantum key distribution. *NPJ Quantum Information*. 2016;**2**:16025
- [31] Tannous R et al. Quantum cryptography: Future directions and challenges. *Journal of Quantum Information Science*. 2022;**12**(1):45-67
- [32] Liao S-K, Cai W-Q, Liu W-Y, et al. Satellite-to-ground quantum key distribution. *Nature*. 2017;**549**:43-47
- [33] Chen L et al. Report on post-quantum cryptography. 2016
- [34] Lyubashevsky V, Peikert C, Regev O. On Ideal Lattices and Learning with Errors over Rings. 2010
- [35] Patarin J. Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): Two New Families of Asymmetric Algorithms. 1996
- [36] Merkle RC. A certified digital signature. 1989

- [37] Campagna M, Chen L. Quantum Safe Cryptography and Security: An Introduction, Benefits, and Impact. 2014
- [38] Bernstein DJ, Lange T. Post-quantum cryptography. *Nature*. 2017;**549**(7671):188-194
- [39] Chen L et al. Report on post-quantum cryptography. Vol. 12. Gaithersburg, MD, USA: US Department of Commerce, National Institute of Standards and Technology; 2016
- [40] Alshaer NA, Ismail TI. AI-driven quantum technology for enhanced 6G networks: Opportunities, challenges, and future directions. *Journal of Laser Science and Applications*. 2024;**1**(1):21-30
- [41] Abdallah W. A physical layer security scheme for 6G wireless networks using post-quantum cryptography. *Computer Communications*. 2024;**218**:176-187
- [42] Alagic G, Alagic G, Apon D, Cooper D, Dang Q, Dang T, Kelsey J, Lichtinger J, Liu Y-K, Miller C, Moody D, Peralta R, Perlner R, Robinson A, Smith-Tone D. Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process. 2022
- [43] Partala J. Post-quantum cryptography in 6G. *Mobile Wireless Networks*. 2021;**6G**:431-448
- [44] Kumar A, Ottaviani C, Gill SS, Buyya R. Securing the future internet of things with post-quantum cryptography. *Security and Privacy*. 2022;**5**(2):e200
- [45] Bhattacharya P et al. Coalition of 6G and blockchain in AR/VR space: Challenges and future directions. *IEEE Access*. 2021;**9**:168455-168484
- [46] Hoang VT, Ergu YA, Nguyen VL, Chang RG. Security risks and countermeasures of adversarial attacks on AI-driven applications in 6G networks: A survey. *Journal of Network and Computer Applications*. 2024;**104031**:1-31
- [47] Shaik RB, Nagaradjane P, Ioannou I, Sittakul V, Vasiliou V, Pitsillides A. AI/ML-aided capacity maximization strategies for URLLC in 5G/6G wireless systems: A survey. *Computer Networks*. 2024;**249**:110506
- [48] Sharma D, Tilwari V, Pack S. An overview for designing 6G networks: Technologies, spectrum management, enhanced air Interface and AI/ML optimization. *IEEE Internet of Things Journal*. 2024
- [49] Chen L, Moody D, Liu Y. NIST post-quantum cryptography standardization. *Transition*. 2017;**800**(131A):164
- [50] ETSI. Quantum-Safe Cryptography White Paper. 2023. Available from: <https://etsi.org>
- [51] TU-T Study Group. Security Considerations for Next-Generation Networks. ITU White Paper. 2022
- [52] Ishtaiwi A et al. Artificial intelligence in cryptographic evolution: Bridging the future of security. In: *Innovations in Modern Cryptography*. IGI Global; 2024. pp. 31-54
- [53] Fathalla E, Azab M. Beyond Classical Cryptography: A Systematic Review of Post-Quantum Hash-Based Signature Schemes, Security, and Optimizations. *IEEE Access*; 2024
- [54] Khan AS et al. A survey on 6G enabled light weight authentication

protocol for UAVs, security, open research issues and future directions. *Applied Sciences*. 2022;**13**(1):277

[55] Zohaib M, Altuwaijri FS, Hyrynsalmi S. Integrating quantum computing and blockchain: Building the foundations of secure, efficient 6g technology. In: *Proceedings of the 1st ACM International Workshop on Quantum Software Engineering: The Next Evolution*. 2024. pp. 27-34

[56] [Internet]. 2025. Available from: <https://www.ericsson.com/en/reports-and-papers/further-insights/impact-of-quantum-computing-on-5g-6g-security>

[57] [Internet]. 2025. Available from: <https://www.azoai.com/news/20241105/Unlocking-6G-with-QuantumSecurity-and-AI-Integration.aspx>

[58] [Internet]. 2025. Available from: <https://hub.telefonica.com/en/research/confidential-6g-project>

[59] [Internet]. 2025. Available from: https://irr.singaporetech.edu.sg/articles/conference_contribution/A_Survey_on_PostQuantum_Cryptography_for_5G_6G_Communications/26129599

[60] [Internet]. 2025. Available from: <https://www.telefonica.de/news/pressreleases-telefonica-germany/2024/11/kick-off-into-the-quantum-ageo2-telefonica-enhances-mobilenetwork-efficiency-and-securitywith-quantum-technologies-on-aws.html>

[61] [Internet]. 2025. Available from: <https://hub.telefonica.com/en/research/confidential-6g-project>

[62] Available from: <https://www.6gacademy.com/what-it-is-post-quantum-security>

[63] [Internet]. 2025. Available from: https://en.wikipedia.org/wiki/Post-quantum_cryptography

[64] [Internet]. 2025. Available from: https://irr.singaporetech.edu.sg/articles/conference_contribution/A_Survey_on_PostQuantum_Cryptography_for_5G_6G_Communications/26129599

[65] [Internet]. 2025. Available from: <https://decentcybersecurity.eu/evaluating-the-security-of-6g-networkswith-post-quantum-cryptography-in-2024/>

*Edited by Jonathan Blackledge
and A.F.M. Shahen Shah*

Cryptography and coding theory share a common mathematical foundation in algebra, number theory, and information theory. While coding theory focuses on ensuring reliable transmission and storage of data in the presence of noise and errors, cryptography is concerned with safeguarding data from adversarial threats, ensuring confidentiality, integrity, and authenticity.

In the modern landscape of data-driven technologies, these two perspectives have become increasingly intertwined. Secure communications, distributed systems, and intelligent networks require both error resilience and cryptographic protection to operate effectively. This book presents a comprehensive exploration of recent advances in coding theory, highlighting its profound applications across informatics, data analysis, and cryptography. Bringing together contributions from leading experts, the volume provides both theoretical insights and practical perspectives, making it a valuable resource for researchers, professionals, and advanced students alike. Topics covered include error-correcting codes, modern coding strategies for large-scale data systems, emerging applications in network coding, machine learning, and secure communications. Specific emphasis is placed on the role of coding theory in ensuring data integrity and robustness in an age of unprecedented information exchange. From enhancing the reliability of cloud storage to strengthening cryptographic protocols against evolving cyber threats and utilizing novel methods of data encryption, such as those employing Machine Learning and post-quantum security, the chapters demonstrate how coding techniques continue to adapt to contemporary challenges in digital technology. Written with clarity and depth, the book also serves as a bridge between the mathematical basis for coding and cryptography and their real-world applications. Each chapter combines rigorous discussion with illustrative examples, offering readers both a conceptual understanding and the tools for practical implementation. The book is not only a reference for specialists but also an invitation to explore how abstract mathematical ideas translate into innovations that safeguard and empower today's digital society.

Published in London, UK

© 2025 IntechOpen

© Ihor Tsyvinskyi / iStock

IntechOpen

ISBN 978-1-83634-036-2

