mathematics

IMPACT FACTOR 2.2

CITESCORE 4.6

Article

Quantum–Classical Optimization for Efficient Genomic Data Transmission

Ismael Soto, Verónica García and Pablo Palacios Játiva

Special Issue

Advanced Computational Modeling and Optimization in Operations Research and Supply Chain Management

Edited by

Prof. Dr. Pedro Ivan Palominos

MDPI

https://doi.org/10.3390/math13172792

mathematics

MDPI

Article

Quantum–Classical Optimization for Efficient Genomic Data Transmission

Ismael Soto [1,*], Verónica García [2] and Pablo Palacios Játiva [3]

[1] Multidisciplinary Research Center for Telecommunications Technologies (CIMTT), Departamento de Ingeniería Eléctrica, Universidad de Santiago de Chile, Santiago 9170124, Chile
[2] Departamento en Ciencia y Tecnología de los Alimentos, Universidad de Santiago de Chile, Santiago 9170022, Chile; veronica.garcia@usach.cl
[3] Escuela de Informática y Telecomunicaciones, Universidad Diego Portales, Santiago 8370190, Chile; pablo.palacios@mail.udp.cl
* Correspondence: ismael.soto@usach.cl; Tel.: +56-2-2718-0000

Abstract

This paper presents a hybrid computational architecture for efficient and robust digital transmission inspired by helical genetic structures. The proposed system integrates advanced modulation schemes, such as multi-pulse-position modulation (MPPM), high-order quadrature amplitude modulation (QAM), and chirp spread spectrum (CSS), along with Reed–Solomon error correction and quantum-assisted search, to optimize performance in noisy and non-line-of-sight (NLOS) optical environments, including VLC channels modeled with log-normal fading. Through mathematical modeling and simulation, we demonstrate that the number of helical transmissions required for genome-scale data can be drastically reduced—up to 95% when using parallel strands and high-order modulation. The trade-off between redundancy, spectral efficiency, and error resilience is quantified across several configurations. Furthermore, we compare classical genetic algorithms and Grover's quantum search algorithm, highlighting the potential of quantum computing in accelerating decision-making and data encoding. These results contribute to the field of operations research and supply chain communication by offering a scalable, energy-efficient framework for data transmission in distributed systems, such as logistics networks, smart sensing platforms, and industrial monitoring systems. The proposed architecture aligns with the goals of advanced computational modeling and optimization in engineering and operations management.

Keywords: Grover's algorithm; hybrid modulation schemes (MPPM-QAM-CSS); quantum Reed–Solomon codes; synthetic DNA encoding; visible light communication (VLC)

MSC: 92F05

Academic Editor: Eva H. Dulf

Received: 26 July 2025
Revised: 24 August 2025
Accepted: 28 August 2025
Published: 30 August 2025

Citation: Soto, I.; García, V.; Palacios Játiva, P. Quantum–Classical Optimization for Efficient Genomic Data Transmission. *Mathematics* **2025**, *13*, 2792. https://doi.org/10.3390/math13172792

1. Introduction

In industrial fermentation processes, the appearance of bacteriophages (phages) represents a significant threat, as they can disrupt continuous production, cause considerable economic losses due to the need for cleaning and sterilization of the equipment, and compromise the genetic stability of the bacterial strains used [1]. These viruses exhibit remarkable resistance to conventional disinfection methods, requiring the implementation of rigorous biosafety protocols for their detection and containment. However, phages also offer valuable opportunities in industrial biotechnology. Their specificity allows them to

eliminate contaminating bacteria without affecting productive strains and can be used as genetic vectors or in techniques such as phage display. Moreover, their ability to modulate the immune system in contexts involving animal or human cells has been explored with promising results [2]. Recently, certain virulent phages have been shown to be able to control pathogenic strains of *Escherichia coli* without inducing toxin production, positioning them as an effective and safe alternative to traditional antibiotics [3].

The application of artificial intelligence (AI) to biotechnological processes has revolutionized the modeling and optimization of solid-state fermentation (SSF), allowing more precise, efficient, and adaptable control of complex systems. Historically, SSF was characterized by its reliance on empirical methods with variable results, but the development of algorithms such as artificial neural networks (ANN), genetic algorithms (GA), and particle swarm optimization (PSO) has allowed a highly accurate representation of the dynamics of these processes, even in the absence of exact mechanistic models [4].

In recent years, various studies have confirmed the usefulness of machine learning (ML) techniques to model the non-linear and time-dependent interactions that characterize process bioengineering. These tools allow prediction of critical quality attributes (CQA) and adjustments in real time, which is especially useful in environments where experimental conditions are costly or difficult to replicate [5,6]. Since early efforts to describe the structure of deoxyribonucleic acid (DNA), computational modeling has played a central role in understanding its molecular dynamics. In 1995, Lavery introduced a pioneering vision on the need to replace analytical models with numerical simulations to represent the complexity of DNA at the atomic scale, thus addressing the challenges of conformational space and the computational limits of time [7]. This vision laid the groundwork for the subsequent development of computational biology. In the late 1990s, the field of DNA computing emerged, proposing to use the helical structure and self-assembly properties of DNA as a means of performing large-scale parallel calculations [8,9]. This approach was solidified with proposals to represent digital information in DNA sequences and perform algorithmic operations through strand displacement, establishing a bridge between molecular biology and computer science [10].

With the expansion of systems and synthetic biology in the 21st century, computational modeling became essential to design and optimize artificial genetic circuits, simulating the dynamics of regulatory networks and cellular processes [11,12]. More recently, research has shown how molecular computation models can manipulate digital information directly stored in DNA, integrating storage and processing processes within a single molecule [13,14].

Moreover, concepts from communication theory have been applied to biology, considering gene expression as a signal transmission process subject to error control and routing, suggesting a direct parallel with digital networks and information theory [15].

Thus, DNA modeling has evolved from physical and structural descriptions to becoming a computational platform in itself, enabling disruptive advances in gene synthesis, personalized therapies, and nanotechnology.

This work proposes a novel hybrid classical–quantum architecture for the digital transmission of synthetic DNA, integrating advanced modulation schemes such as multi-pulse-position modulation (MPPM), high-order quadrature amplitude modulation (QAM), and chirp spread spectrum (CSS) along with Reed–Solomon error correction coding, and multithreaded parallelism. The system design significantly reduces the number of required helical transmissions by leveraging high-order modulations and physical parallelism, while maintaining robustness against errors through redundancy. Furthermore, it incorporates quantum modules, such as Grover-assisted list decoding and S^2-CNN-based classification, to accelerate inference tasks, achieving a theoretical complexity improvement from $O(N)$

to $O(\sqrt{N})$ in candidate search. The simulation results validate the system performance in non-line-of-sight (NLOS) optical channels, showing bit error rates (BER) below 10^{-6} at moderate signal-to-noise ratios ($E_b/N_0 \approx 11$ dB). The proposed solution balances spectral efficiency, error resilience, and scalability, positioning it as a promising framework for future applications in distributed biotechnology, molecular communication, and optical genetic data storage.

Organization of the Paper. This article is organized as follows. The Section 1 presents the motivation, background, and general objectives of the study. The Section 2 presents related works. The Section 3 describes the transmission model, encoding parameters, and simulation setup. The Section 4 establishes the mathematical basis for Reed–Solomon correction and quantum-assisted decoding. Next, the Section 5 provides detailed descriptions of the encoding, modulation, transmission, and decoding procedures in both classical and quantum-enhanced configurations. The Section 6 presents the performance analysis, followed by the Section 7, which interprets the findings and explores the trade-offs between efficiency and robustness. Finally, the Section 8 summarizes the key contributions and outline future directions.

2. Related Work

2.1. Data Storage in DNA and Quantum Alignment

Since the emergence of the Needleman–Wunsch algorithm for sequence alignment in 1970, bioinformatics has evolved to tackle massive amounts of genomic data, a trend intensified by the COVID-19 pandemic and advances in sequencing technologies. Quantum computing has emerged as a promising alternative to overcome the limitations of classical systems. Cudby and Strelchuk proposed quantum algorithms for the alignment and central chain problem, showing specific advantages depending on the computational context [16]. For their part, Varsamis et al. developed gate-based quantum algorithms for reference-guided alignment, capable of integrating into existing classical systems [17]. Another line of research focuses on models like QiBAM, which adapts Grover to approximate alignments in genomic reads [18]. Beyond computational acceleration, works such as those by Rivelino and D'Acunto propose the possibility of using the intrinsic quantum properties of DNA or enzymes as quantum processing units [19,20]. Although still in the theoretical phase, these proposals open the door to hybrid architectures that could transform genetic storage and processing from the ground up.

2.2. Channel Coding in Synthetic DNA

The storage of information in synthetic DNA has emerged as a revolutionary solution to the growing demand for high-density and long-lasting storage media. Since the first channel models introduced by Hunt et al. [21], fundamental errors such as insertions, deletions, and substitutions have been identified, reflecting the noisy nature of synthesis and sequencing processes. These observations motivated the development of classical error correction codes, such as BCH and LDPC, to preserve data integrity.

Subsequently, studies such as that of Lenz et al. [22] formalized storage models based on unordered sets of fixed-length sequences, introducing efficient codes to correct errors at the full sequence level and at individual points. Other advances focused on asymmetric techniques, such as those proposed by Kiah et al. [23], which incorporated profile vectors to better model the DNA channel.

More recently, the use of modulation coding [24] and innovative schemes such as RRNS + MSA [25] has been explored, which leverage the structural redundancy of DNA to correct complex errors with high efficiency. Even algorithms based on probabilistic syn-

chronization and consensus have been proposed to correct large-scale desynchronization errors, as demonstrated by the work of Hamoum and Dupraz [26].

This body of research reflects a transition from classical models to specialized architectures that integrate probabilistic models, ensemble structures, and modulated encoding. The modern approach considers not only error correction but also read cost, encoding efficiency, and compatibility with molecular biology, laying the groundwork for an era of biocompatible mass storage.

2.3. Optical Transmission Optimization: From Bias Modulation to Quantum Transforms

In visible light communication (VLC) systems, the constraint of transmitting only non-negative optical signals has motivated the development of multiple coding and modulation strategies. One of the most traditional techniques involves applying a direct-current (DC) bias to ensure the unipolarity of the modulated signal, commonly employed in schemes such as DCO-OFDM [27]. However, this approach suffers from energy inefficiency, since the bias does not contribute any useful information [28]. To counteract this drawback, bias optimization methods and informative designs have been proposed [29,30]. Alternatively, techniques based on the Fourier transform, both classical and quantum (QFT), have gained prominence, especially in hybrid quantum–classical schemes. The latter allow for more efficient distribution of information in the complex plane and offer theoretical improvements in scalability and spectral precision [31,32].

2.4. Code Search in Non-Hermitian Channels

The evolution from reversible computing to contemporary techniques for quantum coding and simulation of open systems reflects a sustained advance at the intersection of physics, information theory, and communications. Early works, such as De Vos's [33], explored reversible combinatorial computation and its analogy to endoreversible thermodynamics, laying the physical foundations for efficient logic design. Subsequently, Nolan and Stark [34] developed a recursive method to estimate error probabilities in Reed–Solomon (RS) codes under various interference levels, providing key tools for analysis in non-ideal channels. In the quantum realm, Wei and Long [35] introduced an efficient algorithm, within the framework of duality-based quantum computing, to simulate the evolution of open systems using Kraus operators. Complementarily, Zheng [36] proposed an innovative approach to represent irreversible evolutions as linear combinations of unitary operators, leveraging the inherently reversible architecture of quantum computers. In this context, the selection of RS channel codes can be approached from two perspectives: open algorithms—also known as a priori methods, which make probabilistic assumptions about distances—and the closed model or a posteriori approach—which determines the code based on the final BER value, estimated, for example, using a digital twin. This methodology enables the emulation of physical irreversibility within a unified framework, expanding the scope of quantum simulation and laying the groundwork for representing open systems in a controlled and efficient manner. In recent years, digital simulations of Lindblad equations on quantum hardware have been implemented using quantum imaginary-time evolution [37]. In parallel, the decoding of RS codes has undergone significant improvements, such as the FFT-based approach and the modular method for solving the Welch–Berlekamp equation [38], as well as the integration of RS kernels into polar codes through more efficient construction and rate-adjustment techniques [39], with promising applications in next-generation communication systems.

2.5. Quantum Reed–Solomon Codes

The evolution of quantum Reed–Solomon codes (QRS) has followed a trajectory that combines classical algebraic foundations with advances in physical efficiency for their

implementation in real quantum systems. Initially, Grassl et al. introduced the quantum adaptation of Reed–Solomon codes over finite fields of characteristic two, using Fourier transforms to design encoding and decoding circuits [40]. Subsequently, Howard demonstrated how these codes could also be used in the construction of Hermitian operators and discrete Wigner functions, connecting the theory of classical codes with quantum state space [41]. More recently, significant advances have been achieved with the use of quantum multiplexing techniques, such as the one proposed by Nishio et al., which allows a considerable reduction in the number of controlled X (CX) gates needed to implement QRS in high-dimensional photonic systems [42] and Li in 2019 [43]. This optimization in circuit complexity positions QRS as a viable alternative for quantum communication tasks resilient to errors due to qudit loss, and extends its applicability to algorithms such as Grover and quantum walks. Complementarily, entanglement-assisted Reed–Solomon codes, first developed by Li in 2019 [43] and later extended by Nadkarni and Garani [44], offer optimal error correction schemes over qudits, with explicit encoding circuits and quadratic complexity. These advances reflect a convergence between algebraic theory and quantum resource engineering, which is key to the future scalability of quantum communications and computing.

2.6. Grover's Algorithm in Quantum Decoding

Grover's algorithm has evolved significantly since its original proposal as a probabilistic method for unstructured searches with a complexity of $O(\sqrt{N})$ [45]. Subsequent studies have demonstrated its sensitivity to errors in oracle markings, but also its resilience in mixed scenarios of defective and nondefective marked elements, where the quantum advantage is still preserved [46]. To improve the certainty of the result, deterministic variants such as the D2p protocol have been proposed, which guarantees a 100% success rate without requiring control over the oracle [47]. Other relevant extensions integrate estimation and discovery to find multiple solutions without prior knowledge of their number [48], or allow accelerating the search through partial oracles and dynamic correlations in the search space [49]. Moreover, recent research has generalized the algorithm to incorporate priorities in the marks through phase encoding, increasing its applicability in contexts with hierarchical elements [50]. These developments, along with their empirical validation against classical searches [51] and their scalability in quantum implementations of up to six qubits [52], consolidate Grover as a central tool for efficient search tasks in realistic quantum environments.

2.7. Optimization of Quantum Coding Through Multiplexing

The development of quantum Reed–Solomon codes (QRS) has undergone significant technical evolution from their initial adaptation to their optimization for high-dimensional quantum architectures. In its early days, QRS codes were proposed as a natural extension of classical Reed–Solomon codes, using their algebraic structure to correct errors in qudit-based quantum systems [40]. However, their practical implementation has been limited by the high cost in physical resources, particularly the large number of controlled two-qubit (CX) gates required for encoding.

A recent line of research has shown that the use of quantum multiplexing—a technique that encodes information in multiple degrees of freedom of a photon—significantly reduces the number of CX gates required in QRS encoding circuits [42]. These advances have been key to the practical implementation of QRS schemes on high-dimensional photonic platforms. The most influential works have presented techniques to decompose multi-controlled gates into more efficient configurations through multiplexing [53], which

drastically improves encoding efficiency without sacrificing robustness against errors due to qudit loss.

Moreover, this approach is not limited to QRS codes: it has also been shown to be applicable to other relevant quantum algorithms such as Grover and quantum walks [42]. Although the most recent works do not always quantify a specific reduction (such as a factor of 24×), they do highlight a substantial improvement in the physical feasibility of these systems. In parallel, previous studies had already explored the reduction in the number of photons required for error correction schemes, consolidating the role of multiplexing as a critical enabler for the scalability of quantum communications.

Together, this technological trajectory positions quantum multiplexing as a strategic enabler to bring quantum Reed–Solomon codes from algebraic theory to efficient implementation in quantum hardware. This progress is particularly relevant for hybrid architectures that integrate classical redundancy with quantum inference, advancing towards practical and scalable solutions for future quantum communications.

2.8. Deep Learning in Optical Communication

In recent years, end-to-end deep learning has revolutionized the design of optical communication systems by replacing the traditional block approach (modulation, channel, equalization) with fully differentiable integrated architectures. This paradigm allows for the joint optimization of the transmitter and receiver as a single neural network, substantially improving metrics such as the BER and robustness against channel distortions.

An initial milestone was the proposal by Karanov et al. (2018), who implemented a complete optical fiber communication system as a differentiable neural network that includes the transmitter, the optical channel, and the receiver [54]. This approach, experimentally validated, surpassed conventional techniques based on PAM2/PAM4 and equalization, achieving rates of up to 42 Gb/s over 40 km with BER below the rigid FEC threshold [54].

Subsequently, Roumpos et al. (2023) introduced neural networks with activation functions inspired by non-linear optics, such as Photonic Sigmoid, demonstrating notable improvements in IM/DD links with up to 48 Gb/s over 42 km [55].

In parallel, Li and Wang (2022) proposed end-to-end trained autoencoders to compensate for chromatic dispersion, achieving lower BER compared to classical electrical methods and without the need for coherent detection [56].

Xu et al. (2024) designed a low-cost framework for 100 Gb/s passive optical networks (100G-PON), where the transmitter and receiver are modeled as single convolutional layers, reducing the necessary operations by more than 75% compared to traditional designs [57].

Furthermore, Roumpos et al. demonstrated that the use of neural networks informed by optical models improves generalization and efficiency by incorporating physical channel knowledge into network training [55].

In general, these advances reflect how end-to-end deep learning has moved from theory to experimental implementation, surpassing traditional architectures. Although the current focus is on IM/DD systems, the methodology is applicable to other domains such as VLC or digital biocommunications [58].

3. Materials and Methods

3.1. Formulation of the Optimization Problem

The problem of minimizing the total number of helical transmissions required to transfer a coded genomic data sequence is presented, using M-QAM modulation schemes and Reed–Solomon block coding over finite fields $GF(2^6)$. The transmission architecture is based on a multichannel helical structure, where the information is distributed in par-

allel through multiple strands. This formulation corresponds to a discrete combinatorial optimization problem with mixed constraints, typical of optical communication systems in biotechnological or molecular storage environments.

3.1.1. Objective Function

The objective is to minimize the total number of helical transmissions required:

$$\min_{M,\mu,h,k,n} \quad T' = \left\lceil \frac{B \cdot b \cdot \frac{n}{k}}{\mu \cdot h \cdot \log_2 M} \right\rceil \tag{1}$$

where T' represents the total number of transmission units required, considering coding redundancy and modulation efficiency.

3.1.2. Subject to Restrictions

$$\begin{aligned}
M &\in \{4, 16, 64, 256\} && \text{(M-QAM modulation available)} \\
\mu &\in \{4, 8, 16, 32, 64\} && \text{(active threads count)} \\
1 \leq h \leq h_{\max}, &\quad h \in \mathbb{Z}^+ && \text{(allowed helical levels)} \\
1 \leq k < n \leq n_{\max}, &\quad k, n \in \mathbb{Z}^+ && \text{(Reed–Solomon code parameters)} \\
\mathrm{BER}(M) &\leq \epsilon && \text{(maximum tolerated error rate)} \\
\mathrm{SNR}_{\mathrm{req}}(M) &\leq \mathrm{SNR}_{\mathrm{disp}} && \text{(viability condition in the VLC channel)} \\
\frac{n}{k} &\leq \rho_{\max} && \text{(maximum allowed redundancy ratio)}
\end{aligned}$$

3.1.3. Model Parameters

$$\begin{aligned}
B &: \text{Total number of information blocks to be transmitted} \\
b &: \text{Bits per block (for example, 6 if triplets are encoded in } GF(2^6)) \\
\epsilon &: \text{Maximum allowed bit error rate (BER)} \\
\mathrm{SNR}_{\mathrm{disp}} &: \text{Available signal-to-noise ratio in the optical channel} \\
\rho_{\max} &: \text{Maximum redundancy ratio: } \frac{n}{k}
\end{aligned}$$

The proposed model describes an optimization scheme for the efficient transmission of encoded genomic data, based on a multichannel helical architecture. The objective function minimizes the total number of required transmissions, simultaneously considering M-QAM modulation, Reed–Solomon block coding over $GF(2^6)$, the number of strands available for parallel transmission, and the helical levels used. The model integrates constraints inherent to the physical and logical system, such as the available signal-to-noise ratio (SNR), the allowed BER, and the maximum admissible redundancy in the coding. Since it is a discrete problem with multiple possible combinations, its resolution requires efficient search techniques that balance transmission robustness against errors and spectral efficiency, making it a representative case of combinatorial optimization with mixed constraints.

Within the framework of the optimization function defined in Equation (1), a constraint linked to the bit error rate (BER) is incorporated, which must be kept below a maximum allowable threshold. This condition is formulated as $\mathrm{BER}(M) \leq \epsilon$ (maximum tolerated error rate), where M represents the modulation scheme under consideration and ϵ corresponds to the admissible upper bound on the error rate. Compliance with this constraint ensures that the system meets the reliability requirements necessary for efficient and robust information transmission.

In practice, classical decoding of short RS codes, such as RS(15, 11), is extremely efficient due to the availability of algebraic algorithms (e.g., Berlekamp–Massey or Euclidean-

based methods) whose complexity scales polynomially in n and can be heavily optimized in software and hardware implementations. Thus, even though Grover's search provides a quadratic speedup over exhaustive candidate inspection, for small block lengths, the constant factors and initialization overheads dominate, leaving classical methods decisively superior. The quantum advantage only becomes meaningful when operating with larger codes or under higher error regimes where the expected list size L grows significantly.

Example 1 (Small code)**.** *The scaling plot in Figure 1 shows that Grover's quadratic speedup only becomes advantageous once the search space size N exceeds the crossover point* $N^\star = (C_q/C_c)^2$*. For small block lengths, N remains far below this threshold, so classical decoding is still faster. To make this concrete, consider the code* $RS(15, 11)$ *over* $\mathbb{F}_{16}$ *(*$m = 4$*,* $t = 2$*). The number of candidate error patterns up to weight t is*

$$N_{\text{cand}} = \binom{15}{0}15^0 + \binom{15}{1}15 + \binom{15}{2}15^2 = 23851.$$

With an expected list size $L \approx 1$*, Grover's algorithm would require*

$$\frac{\pi}{4}\sqrt{\frac{N_{\text{cand}}}{L}} \simeq 1.21 \times 10^2$$

iterations.

While this is a modest number in absolute terms, in practice, the setup and oracle costs of a quantum decoder are much higher than those of a classical algebraic RS decoder (e.g., Berlekamp–Massey or Euclidean algorithms). As a result, for short codes such as $RS(15, 11)$, the overhead dominates and classical decoding remains decisively superior. This aligns with the intuition from Figure 1, which explicitly shows that only when the problem size grows beyond $N^\star$, or when higher error rates inflate the expected list size L, does the quantum advantage become relevant.

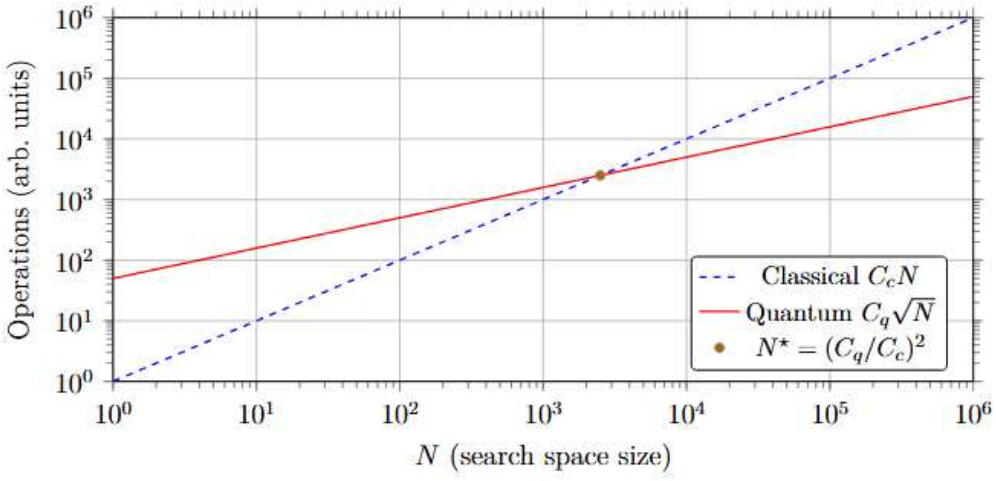

Figure 1. Scaling of classical linear search vs. Grover search. With $C_c = 1$ and $C_q = 50$ (illustrative), the crossover is at $N^\star = 2500$. For $N > N^\star$, the quantum method is asymptotically more efficient.

The classical complexity grows linearly with the search space size N, while the quantum method scales with $\sqrt{N}$. The crossover point $N^\star = (C_q/C_c)^2$ indicates the regime where the quantum algorithm begins to outperform the classical one, after overhead costs are taken into account. For problem sizes below $N^\star$, classical search is more efficient, whereas for $N > N^\star$, the quadratic speedup of Grover's algorithm becomes advanta-

geous. This visual representation clarifies the conditions under which a genuine quantum advantage emerges.

To address this combinatorial optimization problem with mixed constraints, it is crucial to consider approaches that allow efficient exploration of the discrete solution space defined by the modulation, coding, and helical architecture parameters. In this context, the comparison of three computational approaches is proposed: a classical method, the Genetic Algorithm (GA), and two quantum methods, Grover's search algorithms. The GA allows for a stochastic search based on selection, crossover, and mutation over discrete populations, while Grover offers a quadratic advantage in the search for optimal solutions in unstructured spaces. These methods are evaluated as potential strategies to minimize the total number of helical transmissions under the conditions imposed by the digital transmission system based on synthetic DNA and VLC.

3.2. System Diagram

Figure 2 presents a hybrid classical–quantum architecture for the transmission of genetic information through an NLOS optical channel, employing advanced coding, modulation, detection, and processing techniques. Each block of the diagram represents a key operation in the flow of information, from genomic input to the final reconstruction of the message. These processes were mathematically defined in the previous section, providing a precise formulation of the transformations applied at each stage. Complementarily, Table 1 summarizes these transformations, detailing the nature of the operation and its corresponding mathematical expression. In this way, a direct correspondence is established between the visual representation of the system, its formal modeling, and the tabular organization, facilitating a comprehensive understanding of the proposed scheme's operation.

Table 1. Summary of block operations in the classical–quantum hybrid genomic transmission architecture.

Block	**Domain**	**Mathematical Description**
	Classical Processing	
Genomic Input Block	Classical	$G = \{g_1, \ldots, g_N\},\ g_i \to \mathbf{u}_i \in GF(2^6)^k$
RS Encoder	Classical	$\mathbf{c}_i = \mathrm{RS}_k^n(\mathbf{u}_i) \in GF(2^6)^n$
MPPM Modulator	Classical	$\mathbf{s}_i \in \{0,1\}^m,\ \sum s_i = p$
16-QAM Modulator	Classical	$\log_2 M$ bits $\mapsto x_i \in \mathbb{C}$
CSS Chirp Modulator	Classical	$x(t) = \cos\left(2\pi\left(f_0 t + \frac{k}{2}t^2\right)\right)$
Matched Filter	Classical	$y(t) = (x * h)(t)$
LED (IM Modulation)	Classical	$I(t) \propto x(t)$
NLOS Optical Channel	Classical	$r(t) = \sum_i \eta_i x(t - \tau_i) + n(t)$
APD Photodetector	Classical	$i(t) = R \cdot r(t) + n_q(t)$
Digitization & Sync	Classical	$\mathbf{y}_i \in \mathbb{R}^m,\ y[n] \leftarrow y(t_n)$
Genomic Reconstruction	Classical	$\hat{G} = \{\hat{g}_1, \ldots, \hat{g}_N\}$
	Quantum Processing	
S^2-CNN Feature Extraction	Quantum	$\mathcal{Q}_{S^2}(\mathbf{y}_i) \to \mathbf{z}_i$
Softmax + Top-L Classifier	Quantum	$\hat{c}_i = \arg\max_j \mathrm{softmax}(\mathbf{z}_i)_j$
Quantum RS Decoder	Quantum	$\hat{\mathbf{u}}_i = \arg\min_{\mathbf{u} \in \mathcal{C}} \mathrm{dist}(\mathbf{c}_i, \mathcal{Q}_{\mathrm{RS}}(\mathbf{u}))$

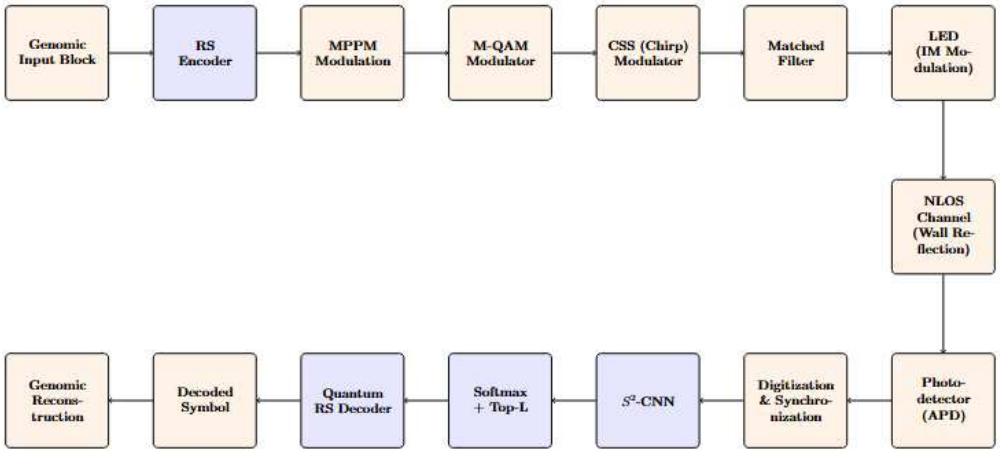

Figure 2. Hybrid classical–quantum architecture for the transmission and decoding of genomic blocks. The system implements Reed–Solomon coding, MPPM modulation, M-QAM, and CSS, followed by optical detection in the NLOS channel and quantum processing for accurate classification and reconstruction of the transmitted genetic material.

Figure 2 presents a hybrid classical–quantum architecture for the transmission of genetic information through an NLOS optical channel, employing advanced coding, modulation, detection, and processing techniques. It is a quantum–classical (hybrid) simulation. Each block of the diagram represents a key operation in the flow of information, from genomic input to the final reconstruction of the message.

Below, the processes implemented by each of the blocks of the classical–quantum hybrid system are mathematically described. The genomic input block divides the complete sequence into blocks $G = \{g_1, g_2, \ldots, g_N\}$, where each g_i represents a set of nitrogenous bases (A, C, G, T). Each block is transformed into a binary representation in the finite field $GF(2^6)$, obtaining encoded symbols $\mathbf{u}_i \in GF(2^6)^k$.

In the proposed architecture, Reed–Solomon (RS) coding is applied before the assignment of blocks to the helical threads. This is because the $RS(4,2)$ code operates on blocks of symbols defined in $GF(2^6)$, generating two parity symbols from two data symbols. Once encoded, the four resulting symbols are distributed among the available threads: threads 1 and 2 transmit the data symbols, while threads 3 and 4 carry the parities, as shown in Table 2.

Table 2. Distribution of blocks encoded by thread in an $RS(4,2)$ scheme over $GF(2^6)$. Threads 1 and 2 transmit data, while threads 3 and 4 carry parity symbols.

Thread	Content
1	Data Symbol 1
2	Data Symbol 2
3	Parity Symbol 1 (RS)
4	Parity Symbol 2 (RS)

In the Multi-Pulse Position Modulation (MPPM) scheme, the parameter L represents the number of active pulses transmitted within a symbol composed of m time slots. For example, if $m = 4$ and $L = 2$, then in each MPPM symbol, two pulses are activated in specific positions, selected from among the possible combinations. This allows information to be encoded efficiently and with high immunity to noise, as the position of the pulses is what carries the information through QAM modulation (if used) and chirp spread spectrum coding (CSS). Thus, RS coding guarantees error protection before the information

is transformed by the physical modulation stages, ensuring both the feasibility of the scheme and its efficiency against noise in unguided optical environments.

3.2.1. Reed–Solomon (RS) Coding

$$\mathbf{c}(x) = \mathbf{m}(x) \cdot g(x), \tag{2}$$

where $\mathbf{m}(x)$ is the message polynomial in $GF(2^m)$, $g(x)$ is the degree generator polynomial $n-k$, and $\mathbf{c}(x)$ is the length-encoded RS codeword n.

3.2.2. Multi-Pulse Position Modulation (MPPM)

Transmit a sequence with L pulses in m time slots:

$$x(t) = \sum_{i=1}^{L} p(t - t_i). \tag{3}$$

Bits per symbol:

$$N_{\text{bits}} = \left\lfloor \log_2 \binom{m}{L} \right\rfloor. \tag{4}$$

3.2.3. 16-QAM with Gray Coding

Symbol generation for the 16-QAM square constellation:

$$s = (2a + 1 - \sqrt{M}) + j(2b + 1 - \sqrt{M}), \quad a, b \in \{0, \ldots, \sqrt{M} - 1\}. \tag{5}$$

Normalization of the average power:

$$x = \frac{s}{\sqrt{\frac{2(M-1)}{3}}}. \tag{6}$$

3.2.4. CSS (Chirp Spread Spectrum)

Linear frequency chirp signal:

$$x(t) = \cos\left(2\pi\left(f_0 t + \frac{k}{2}t^2\right)\right), \quad k = \frac{B}{T} \tag{7}$$

Time-shifted symbol based on data:

$$x_d(t) = x\left(t - \frac{dT}{2^n}\right), \quad d \in \{0, \ldots, 2^n - 1\} \tag{8}$$

3.2.5. Hybrid RS + MPPM + QAM + CSS Scheme

Overall processing pipeline:

$$\mathbf{c} = \text{RS\_Encode}(\mathbf{m}), \tag{9}$$

$$x_{\text{MPPM}}(t) = \sum_{i=1}^{L} p(t - t_i), \tag{10}$$

$$x_{\text{QAM}} = \text{GrayMap}(\mathbf{c}), \tag{11}$$

$$x_{\text{CSS}}(t) = x_{\text{MPPM}}(t - \tau_d). \tag{12}$$

where τ_d is the chirp delay corresponding to the bit-mapped data value.

3.2.6. VLC Channel Model Without Direct Component

The total optical power received at the receiver is modeled considering only the reflected paths:

$$p_r = \sum_{i=1}^{Z} p_t \cdot \left(\int_{\text{wall}} H_{\text{ref},i} \, dA_{\text{wall}} \right), \quad 0 \leq H_{\text{ref},i} \leq 1. \tag{13}$$

In this equation, p_r represents the total optical power received in the detector and p_t is the optical power transmitted by the emitter (LED). The term $H_{\text{ref},i}$ corresponds to the optical gain by reflection associated with the i-th indirect path, while Z denotes the total number of reflection paths considered. The integration is performed over the wall area where the reflection occurs.

3.3. Bits Required for Digital Transmission

Table 3 presents a comparison between different types of phages based on the size of their genomes, expressed in kilobases (kb), the total number of bases, and the estimated number of bits required for their digital transmission.

Table 3. Size of phage genomes and bits required for their binary transmission (2 bits/base).

Phage	Size (kb)	Bases	Required Bits
ΦX174	5.4	5400	10,800 bits
M13	6.4	6400	12,800 bits
λ	48.5	48,500	97,000 bits
T7	40	40,000	80,000 bits
T4	169	169,000	338,000 bits

Assuming a simple binary encoding in which each DNA base (A, T, C, G) is represented by two bits, it is observed that the size of the data to be transmitted grows linearly with the length of the genome. This relationship can be expressed as:

$$\text{Required bits} = N_{\text{bases}} \times 2,$$

For example, phage ΦX174, with a genome of 5.4 kb, requires approximately 10,800 bits for its transmission, while phage T4, with a much larger genome of 169 kb, requires 338,000 bits. This estimation provides a quantitative basis for designing efficient modulation and coding schemes, especially in low-power communication systems such as those based on VLC.

Assuming a transmission rate of 100 kbps in a VLC system, it would take approximately 0.108 s to transmit the genome of phage ΦX174 and 3.38 s for that of phage T4. This establishes latency and energy consumption thresholds for biotechnological applications in low-power environments.

Although brief, these transmission times become relevant in applications where precise control of genetic flow is required, such as in automated DNA synthesis, targeted gene editing, or molecular data storage. In controlled environments, such as biosafety cabinets or closed microfluidic systems, low transmission latency through VLC allows for the synchronization of encoding, replication, and analysis processes in real time. Moreover, the directed nature and freedom from electromagnetic interference of the visible optical channel offer a significant advantage over traditional wireless technologies, especially in environments where the use of radio frequency (RF) may be restricted due to electromagnetic contamination, interference with medical equipment, or biosecurity requirements.

Although the ultraviolet (UV) spectrum is adjacent to the visible range used in VLC,

its application in optical communication systems for confined environments presents significant limitations. Wavelengths shorter than 400 nm, corresponding to UV-A, UV-B, and UV-C, can induce mutagenic effects on genetic material, degrade sensitive biological structures, and pose a risk to the integrity of cell or microbial cultures. Although some research has explored the use of UV for non-line-of-sight (NLOS UV) optical communications outdoors, its implementation in biosafety laboratories or enclosed environments is not recommended without specific shielding. In contrast, the visible light used in VLC provides a safe, addressable, and compatible alternative with materials and biological organisms, making its integration into growth chambers, microbioreactors, or bioassembly systems feasible where optical monitoring or control is required without affecting the viability of the environment.

In comparison, technologies such as infrared (IR) or Bluetooth present higher levels of dispersion, continuous energy consumption, and collision risks in densely instrumented environments. Bluetooth, although efficient for personal devices, operates in the 2.4 GHz band, which can interfere with other wireless systems or may not be permitted in sensitive laboratories. For its part, VLC allows the establishment of dedicated optical channels, with minimal interference and an adaptable energy footprint, ideal for integrating distributed sensors or genetic encoders in confined biotechnological spaces.

Although the phage genome ΦX174 is composed of single-stranded DNA (ssDNA), the proposed coding model uses artificially synthesised double-stranded DNA (dsDNA) fragments. This synthetic representation does not replicate the original biological structure, but rather provides a stable and differentiable way to encode blocks of genetic information using specific lengths, facilitating their detection by techniques such as gel electrophoresis. The representation in synthetic double-stranded DNA fragments does not reduce the number of bits needed to encode the genome. Each base is still represented with two bits, so the total number of bits required depends exclusively on the length of the original genetic sequence.

3.4. Multithreaded Transmission of Synthetic DNA in Extended Finite Fields

In the digital transmission of synthetic DNA, it is proposed to represent each base as a symbol in a finite field $GF(p)$, with p being the prime. Using a field extension $GF(p^n)$, each symbol can represent the symbols n of the base field. Moreover, by having a parallel transmission system with μ threads, the total number of required transmissions can be significantly reduced Figure 3.

Ref.	Position of the coefficients of $\alpha^k \in GF(p^n)$					
	$n-1$	$n-2$	$\cdots$	2	1	0
S_{p-1}		✕				
⋮						
S_2	✕					
S_1					✕	
S_0						✕

Figure 3. Matrix illustrating the position of the coefficients in the polynomial representation of an element of $GF(p^n)$. The columns indicate decreasing powers and the rows, different symbols S_i.

For example, according to Table 3, the phage ΦX174, with a genome of 5.4 kb, requires approximately 10,800 bits for its transmission, while the phage T4, with a much larger genome of 169 kb, demands 338,000 bits. This estimation provides a quantitative

basis for designing efficient modulation and coding schemes, especially in low-power communication systems such as those based on VLC.

Assuming a transmission rate of 100 kbps in a VLC system, it would take approximately 0.108 s to transmit the genome of the ΦX174 phage and 3.38 s for the T4 phage. This establishes latency and energy consumption thresholds for biotechnological applications in low-power environments.

Although the phage genome ΦX174 is composed of single-stranded DNA (ssDNA), the proposed coding model uses artificially synthesized double-stranded DNA (dsDNA) fragments. This synthetic representation does not replicate the original biological structure but rather provides a stable and differentiable way to encode blocks of genetic information using specific lengths, facilitating their detection by techniques such as gel electrophoresis. The representation in synthetic double-stranded DNA fragments does not reduce the number of bits needed to encode the genome. Each base is still represented with two bits, so the total number of bits required depends exclusively on the length of the original genetic sequence.

3.5. Genome Encoding of Phage

Below is a practical example of how to represent the genome of the bacteriophage ΦX174 using blocks of three bases, encoded as elements of the finite field $GF(2^6)$, that is, with 64 possible values. Each value is associated with a single-length double-stranded DNA fragment, according to the experimental model proposed by Jirón et al. (2019), based on the differential migration of fragments in electrophoresis [59].

According to Table 4, each DNA base can be represented by two bits, since the genetic alphabet consists of four symbols (A, C, G, T). The binary assignment used is as follows:

Table 4. Binary encoding of DNA nitrogenous bases. Each base is represented by two bits, which allows triplets to be encoded as 6-bit blocks in $GF(2^6)$.

Base	Binary Encoding
A	00
C	01
G	10
T	11

Grouping every three consecutive bases, 6-bit blocks are obtained, which correspond to values between 0 and 63. Therefore, each block can be assigned to an element of $GF(2^6)$. Some examples are shown below (Table 5):

Table 5. Examples of encoding triplets of nitrogenous bases in 6-bit blocks. Each group of three bases (A, C, G, T) is transformed into a 6-bit binary sequence, which corresponds to a decimal value between 0 and 63, representable as an element of $GF(2^6)$.

Base 1	Base 2	Base 3	Binary	Decimal Value
A	C	G	00 01 10	6
T	T	A	11 11 00	60
G	A	C	10 00 01	33

Each value in $GF(2^6)$ is assigned a double-stranded DNA fragment with a specific and unique size, previously generated by PCR. The following table (Table 6) illustrates this correspondence:

Table 6. Mapping between values of the finite field $GF(2^6)$ and unique lengths of double-stranded DNA fragments. Each value $r \in [0, 63]$ represents a block of three nitrogenous bases and is encoded as a specific DNA fragment generated by PCR, with length S_r expressed in base pairs (bp).

Value $r \in GF(2^6)$	**Size of the DNA Fragment (S_r)**
0	100 bp
1	110 bp
2	120 bp
⋮	⋮
63	730 bp

The genome of phage ΦX174 has approximately 5400 bases. When grouped into blocks of three bases, the following are obtained:

$$\frac{5400 \text{ bases}}{3 \text{ bases/block}} = 1800 \text{ blocks}$$

Each of these 1800 blocks is converted into a value between 0 and 63, which is physically represented by a fragment of DNA of a specific size. This allows for a complete molecular representation of the genome.

Agarose gel electrophoresis allows the visualization of DNA fragments according to their size. The resulting band pattern can be interpreted using a decoding key that links each band to a $GF(2^6)$ value, and therefore, to the original block of three bases. This process is reversible and allows the digital reconstruction of the genetic sequence.

Figure 4 illustrates a partial example of how to digitally represent fragments of the viral phage genome ΦX174 using molecular computing techniques. Using a finite number of DNA fragments of different lengths and finite field-based encoding, the implementation of biological storage, genetic encryption, and experimental information processing models in biotechnological environments is facilitated.

Thread	Coefficients of α^k in $GF(2^6)$					
	α^5	α^4	α^3	α^2	α^1	α^0
S_3	✕		✕			
S_2				✕		
S_1	✕					
S_0		✕				✕

Figure 4. Polynomial representation matrix of symbols in $GF(2^6)$. Each row represents a strand S_i and each column indicates the power of α corresponding to the binary coefficient of the encoded block. The encoded values for each strand are as follows: $S_3 = \texttt{101000} = 40$, $S_2 = \texttt{000100} = 4$, $S_1 = \texttt{100000} = 32$, and $S_0 = \texttt{010001} = 17$. Each row represents a thread S_i and each column indicates the power of α corresponding to the binary coefficient of the encoded block. The encoded values for each thread are as follows: $S_3 = \texttt{101000} = 40$, $S_2 = \texttt{000100} = 4$, $S_1 = \texttt{100000} = 32$, and $S_0 = \texttt{010001} = 17$.

3.6. Relationship Between Genetic Coding and M-QAM Modulation

According to Figure 3, the digital representation of phage genomes using blocks of three bases encoded in $GF(2^6)$ allows efficient compression and structuring of genetic information, as illustrated in Table 5 of binary encoding and in Figure 4. Each triplet of bases (6 bits) can be transmitted as a single symbol, allowing the use of M-QAM constellations to efficiently modulate the block sequence. As shown in Table 7 comparing

constellations, schemes such as 16-QAM and 64-QAM offer a good trade-off between spectral efficiency and noise robustness, making them suitable for different genome sizes. For example, for small genomes like that of the ΦX174 phage, a 16-QAM modulation is sufficient and reliable; while for larger genomes, such as the T4 phage, the use of 256-QAM might be more convenient if the channel allows it. A comparative overview of the most suitable M-QAM modulations for different types of phages is presented in Table 8. The proposed helical visualization models the parallel transmission of these encoded units, assigned to differentiated strands, simulating a distributed and highly scalable molecular communication architecture.

Table 7. Comparison of M-QAM constellations.

M-QAM Modulation	Bits per Symbol	Comment
4-QAM	2	Robust but slow
16-QAM	4	Good balance
64-QAM	6	More efficient, requires a good channel
256-QAM	8	High efficiency, very sensitive to noise

Table 8. Choice of constellation by type of phage.

Phage	Total Bits	Suggested Option	Justification
ΦX174	10,800	16-QAM	Small, secure, and fast
M13	12,800	16-QAM	Same as ΦX174
λ (lambda)	97,000	64-QAM	More bits, without demanding as much as 256-QAM
T7	80,000	64-QAM	Good balance between efficiency/robustness
T4	338,000	256-QAM	Needs efficiency; requires a reliable channel

The first level of Figure 5 shows four blocks of six bits, corresponding to combinations of three DNA bases (each encoded in two bits). This gives a total of 24 available bits, which can be efficiently represented using 4-QAM modulation (2 bits per symbol). However, since 4-QAM transmits only two bits per symbol, 12 symbols are required to transmit the 24 bits. To improve the robustness of the system against transmission errors, it is possible to use a Reed–Solomon code $RS(4,2)$ on $GF(2^6)$. This code takes two information symbols (12 bits) and adds two parity symbols, allowing for the correction of up to one erroneous symbol (equivalent to six bits) per block. Since there are four differentiated blocks at the first level, this coding scheme can be applied to each set of four symbols, ensuring a more reliable transmission without exceeding the channel capacity represented at that helical level.

If a Reed–Solomon code $RS(4,2)$ over $GF(2^6)$ is used, each 12 bits of information (two blocks of 6 bits) is encoded as 24 bits, including two parity symbols. Thus, the 10,800 bits corresponding to the genome of the phage ΦX174 require:

$$\frac{10{,}800 \text{ bits}}{12 \text{ bits per data block}} = 900 \text{ RS blocks.}$$

Each of the 900 RS blocks encodes 24 bits, so the total amount of information transmitted amounts to:

$$900 \times 24 = 21{,}600 \text{ bits.}$$

Since each helical figure transmits 720 bits, it is necessary to perform:

$$\frac{21{,}600}{720} = 30 \text{ transmissions.}$$

Therefore, with error correction coding included, the total number of necessary helical transmissions doubles compared to the case without coding.

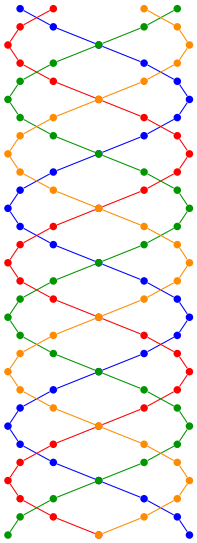

Figure 5. Helical visualization of the digital transmission of blocks encoded in $GF(2^6)$, where each sphere represents a block of three genome bases (6 bits) transmitted by a differentiated strand. This representation illustrates a model of parallel communication through multiple strands, facilitating the encoding and reconstruction of synthetic DNA.

3.7. Parity Storage

Since each strand transmits a block of six bits (corresponding to three DNA bases encoded in $GF(2^6)$), the first level of Figure 5 contains a total of 24 bits, that is, four blocks of six bits. Although visually, two overlapping strands (red and yellow) are observed, the system considers four differentiated helical paths.

If a Reed–Solomon code $RS(4, 2)$ is used on $GF(2^6)$, it takes two information symbols (12 bits) and generates two additional parity symbols (another 12 bits), resulting in a total of four encoded blocks. This configuration fits perfectly with the first helical level, which has four differentiated strands, as we can see in Figure 6.

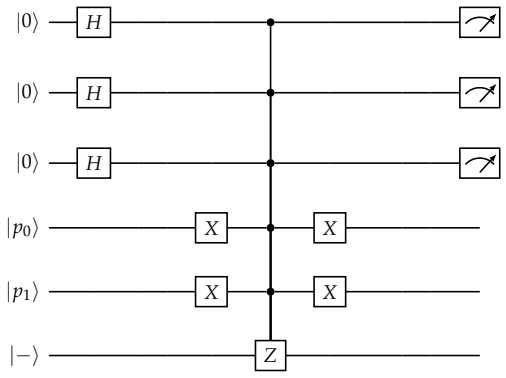

Figure 6. Quantum circuit for Top-L classification via amplitude amplification. Each state represents a candidate symbol; interference selects the most probable.

Therefore, it is possible to efficiently distribute the blocks among the threads, as

detailed in Table 2, where threads 1 and 2 transmit the data symbols, and threads 3 and 4 carry the parity symbols generated by the code $RS(4,2)$.

This assignment allows for maintaining a parallel structure consistent with the helical transmission model. Moreover, the use of the $RS(4,2)$ code allows the correction of up to one erroneous symbol per block of four, that is, up to six corrupted bits, which increases the robustness of the system against transmission or reading errors in electrophoresis.

3.8. AI-Enhanced Receiver for Symbol Recovery in Hybrid VLC Systems

3.8.1. Description of Convolutional and Attentional Operations in S^2-CNN

It can be assumed that a photodetector has a single sensitive element arranged in a line. However, this does not exclude the possibility of reorganizing it into a matrix matrix by software, a configuration in which the S^2-CNN algorithm could achieve its full efficiency. The following equations describe the fundamental processing stages in the classical S^2-CNN architecture, detailing the mechanisms of convolution, normalization, pooling, projection and attention applied to the input tensor through an MLP. These steps are essential for the effective extraction of spatial and spectral features before classification.

$$\begin{aligned} \mathbf{Y}_1^i &= \mathbf{X} * \mathbf{W}_1^i + b_1^i \\ \mathbf{B}_1^i &= \gamma \cdot \frac{\mathbf{Y}_1^i - \mu_{\text{batch}}}{\sqrt{\sigma_{\text{batch}}^2 + \varepsilon}} + \beta \\ \mathbf{M}_1^i &= \text{MaxPool}_{k\times k,s}(\mathbf{B}_1^i) \\ [\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_3] &= \text{Split}(\text{MLP}(\mathbf{M}_1^i)) \\ \mathbf{Z} &= \text{SA}(\text{concat}(\text{SL}_1(\mathbf{L}_1), \text{SL}_2(\mathbf{L}_2), \mathbf{L}_3)) \end{aligned} \tag{14}$$

where $\mathbf{X}$ denotes the input tensor, $\mathbf{W}_1^i$ the convolutional filters, and b_1^i the biases. The parameters γ and β are liabilistic coefficients of batch normalization. The operators SL_j correspond to spectral projections. SA denotes spatial self-attention.

3.8.2. Quantum Implementation of List Decoding: From Guruswami–Sudan to Oracle-Based Circuits

This circuit abstracts the Guruswami–Sudan algorithm into a quantum list decoder. Steps include: (1) parallel superposition of candidates, (2) oracle marking of valid solutions, and (3) selection by measurement or Grover amplification.

Grover-based search should be understood as a hybrid backup strategy: inactive in the high-SNR, small-L regime, where classical decoding dominates, and activated only under near-failure conditions or with long RS codes, where the list size K makes the quantum speedup significant. This prevents overestimating the benefit for small codes and preserves the conceptual contribution of integrating quantum search into the error-correction chain Figure 7.

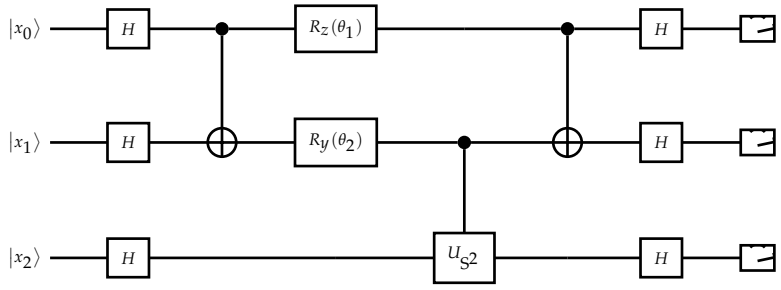

Figure 7. Quantum list decoding: S^2-CNN-inspired circuit for oracle-based verification of symbol candidates.

4. Theoretical Foundations of Error Correction in Classical and Quantum Genetic Encoding

4.1. Theoretical Analysis of Classical Correction

In this section, the theoretical foundations supporting the error correction capability of the classic Reed–Solomon coding scheme applied to digital genetic sequences are presented.

Theorem 1 (Correction Capacity of RS(n,k))**.** *Let a digital genetic sequence be encoded using a Reed–Solomon code $RS(n,k)$ over the finite field $GF(2^m)$. This code can correct up to $t = \left\lfloor \frac{n-k}{2} \right\rfloor$ errors per block of length n, as long as the number of erroneous symbols does not exceed t.*

Proof. By definition, a Reed–Solomon code $RS(n,k)$ has a minimum distance $d = n-k+1$. According to error correction code theory, any code with a minimum distance d can correct up to

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor$$

errors.

Substituting d into the formula, we obtain:

$$t = \left\lfloor \frac{(n-k+1)-1}{2} \right\rfloor = \left\lfloor \frac{n-k}{2} \right\rfloor.$$

Thus, as long as the channel introduces at most t errors per block, the RS decoder can identify and correct them deterministically. □

Theorem 2 (Complete Reconstruction of the Original Sequence)**.** *Let there be a sequence of B data blocks encoded using $RS(n,k)$ and transmitted through a channel that introduces at most $t = \left\lfloor \frac{n-k}{2} \right\rfloor$ errors per block. Then, RS decoding allows for the complete recovery of the original sequence without any loss of information.*

Proof. By Theorem 1, each block of length n can correct up to t errors. If the number of errors introduced by the channel in each block does not exceed this limit, then the decoder can exactly recover the original symbols k from each block.

Since the encoding is applied to independent blocks and there are a total of B blocks, if each is correctly decoded, the complete reconstruction of the original sequence is straightforward by concatenating the $k \cdot B$ recovered symbols. □

Remark 1. *The error correction capability of the classical scheme depends on the selected (n,k) configuration. Greater redundancy (lower k/n) increases error tolerance but also increases the total number of required transmissions, as reflected in the objective function of the proposed optimization model.*

4.2. Theoretical Analysis of Quantum Correction

Theorem 3 (Efficient Recovery with Grover on List Decoding)**.** *Let there be a digital genetic transmission system based on blocks encoded with Reed–Solomon $RS(n,k)$ over $GF(2^m)$, modulated with M-QAM, and transmitted through a helical architecture with μ strands and h levels. Assume that each received block produces a list $\mathcal{L}$ of L candidates through list decoding. If there exists a quantum oracle O_f such that $f(c) = 1$ if c is the correct message, then Grover allows recovery of the correct message with arbitrarily high probability in $O(\sqrt{L})$ steps per block. Moreover, this quantum search does not modify the target function T', but reduces the total computational time for recovery.*

Proof. Given an $RS(n,k)$ code, a list $\mathcal{L}$ of possible candidate blocks can be generated from

symbols altered during transmission. Let us assume that for each block of length n, list decoding returns a set $\mathcal{L} = \{c_1, \ldots, c_L\}$ that contains at least one original valid code.

Let $f : \mathcal{L} \to \{0, 1\}$ be a validation function such that $f(c) = 1$ if the candidate c meets additional criteria (for example, structural redundancy, genetic markers or expected length of the physical fragment). This function can be implemented as a quantum oracle O_f.

The Grover algorithm allows one to find the unique element $c^\star \in \mathcal{L}$ such that $f(c^\star) = 1$, with a probability close to 1 in $O(\sqrt{L})$ steps. This represents a quadratic improvement over a classical search on $\mathcal{L}$.

The total number of required helical transmissions is given by the objective function:

$$T' = \left\lceil \frac{B \cdot b \cdot \frac{n}{k}}{\mu \cdot h \cdot \log_2 M} \right\rceil,$$

which depends solely on the physical parameters of the system (modulation, coding, architecture). Since quantum search acts after transmission, it does not modify T', but it does improve the overall efficiency of the system by reducing the decoding time. □

Remark 2. *The quantum advantage manifests itself exclusively in the decoding stage. The physical architecture of the system, determined by M, μ, h, n, and k, defines the spectral efficiency and the required number of transmissions. However, in high error rate scenarios, the combination of list decoding and quantum search allows maintaining the robustness of the system without increasing T', offering an efficient and hybrid solution.*

Theorem 4 (BER with Spreading Factor in RS + MPPM + QAM + CSS Systems)**.** *The BER of a hybrid optical communication system that integrates RS encoding, MPPM, QAM, and CSS depends on the spreading factor SF, which corresponds to the number of shifts 2^{SF} in the chirp. The overall BER is expressed as:*

$$\begin{aligned} BER = \frac{1}{2^{SF}} \cdot \Bigg\{ & \frac{\log_2 \binom{m}{L}}{\log_2 \binom{m}{L} + L \log_2 M} \cdot BER_{MPPM} \\ & + \frac{L \log_2 M}{\log_2 \binom{m}{L} + L \log_2 M} \cdot \Bigg[(1 - SER_{MPPM}) \\ & \cdot BER_{QAM} \\ & + \frac{SER_{MPPM}}{2} \Bigg] \Bigg\}. \end{aligned} \tag{15}$$

Proof. Consider a transmission system where the RS encoding over $GF(2^m)$ is applied first, followed by cascaded modulation stages. MPPM with parameters m (time slots) and L (active pulses), 16-QAM ($M = 16$) with Gray mapping, and finally, CSS with 2^{SF} spectral spreading shifts.

Each hybrid symbol transmits a total of

$$R_{\text{symbol}} = \log_2 \binom{m}{L} + L \log_2 M$$

bits, where:

- $\log_2 \binom{m}{L}$: bits conveyed by MPPM via the selection of L active slots among m,
- $L \log_2 M$: bits carried by the QAM symbols embedded in each of the L active positions.

The total BER is then the weighted sum of the bit errors introduced by MPPM and QAM, where the weights correspond to the proportion of bits to which each technique contributes R_{symbol}.

- $\mathrm{BER}_{\mathrm{MPPM}}$: errors in the detection of active positions.
- $\mathrm{BER}_{\mathrm{QAM}}$: errors in symbol detection when MPPM is correct.
- $\mathrm{SER}_{\mathrm{MPPM}}$: probability of incorrect detection of MPPM symbol, making QAM decoding unreliable.

When MPPM is incorrectly decoded, associated QAM bits are assumed to be random, contributing to the probability of error 1/2. The final weighted average BER becomes:

$$\begin{aligned}\mathrm{BER}_{\mathrm{hybrid}} = {} & \frac{\log_2 \binom{m}{L}}{R_{\mathrm{symbol}}} \cdot \mathrm{BER}_{\mathrm{MPPM}} \\ & + \frac{L \log_2 M}{R_{\mathrm{symbol}}} \cdot \left[(1 - \mathrm{SER}_{\mathrm{MPPM}}) \cdot \mathrm{BER}_{\mathrm{QAM}} + \frac{\mathrm{SER}_{\mathrm{MPPM}}}{2} \right].\end{aligned} \tag{16}$$

Finally, CSS introduces a spreading effect in which each bit is distributed over a longer duration of symbols by a factor 2^{SF}, reducing the effective data rate and scaling the BER by the same factor. Hence,

$$\mathrm{BER} = \frac{1}{2^{\mathrm{SF}}} \cdot \mathrm{BER}_{\mathrm{hybrid}}.$$

This completes the proof. □

5. Algorithmic Framework for Classical and Quantum Genetic Encoding

The following describes the algorithms used in the process of encoding, modulation, transmission, and decoding of digital genetic sequences, both in their classical version and in their Grover-assisted quantum version.

5.1. Algorithm A1: Classic Genetic Encoding with Reed–Solomon

The Algorithm A1 transforms a sequence of genetic bases (A, C, G, T) into blocks encoded on a finite field $GF(2^m)$. First, a binary conversion of each base is performed using a fixed two-bit encoding. The bits are then grouped into symbols of the m bits, which are interpreted as elements of a finite field. Finally, these symbols are grouped into blocks of k data, on which a Reed–Solomon code $RS(n, k)$ is applied to add parity and allow error correction.

5.2. Algorithm A2: Helical Genetic Modulation and Decoding

The Algorithm A2 takes the encoded blocks as input and assigns them to physical DNA fragments of unique length. Then, it groups the symbols for helical transmission using M-QAM modulation, considering an architecture of μ strands and h levels. Transmission is carried out in parallel, and fragments are recovered using physical techniques (for example, electrophoresis). Finally, Reed–Solomon decoding is performed to recover the original data and reconstruct the genetic sequence through reverse translation.

5.3. Algorithm A3: Grover-Assisted Genetic Decoding

The Algorithm A3 extends classical decoding using quantum techniques. From the physical fragments observed, the symbols in $GF(2^m)$ are identified. Then, list decoding (for example, Guruswami–Sudan) is applied to each block, generating sets of possible solutions. For each list of candidates, Grover's algorithm is used with a quantum oracle $f(c)$ that distinguishes the correct message. This allows reducing the search time for the correct solution to $O(\sqrt{L})$. Finally, the genetic sequence is reconstructed through reverse binary conversion.

The complete details of the encoding, modulation, and decoding procedures (classical and quantum) can be found in Appendix B.

6. Results

6.1. Performance Comparison

In this section, we will perform a general performance comparison between classical and quantum algorithms, whose evaluation criteria are shown in Table 9.

Table 9. Comparison between classical and quantum algorithms.

Criterion	Genetic Algorithm (Classical)	Grover (Quantum)
Type of search	Stochastic over population	Quantum superposition
Theoretical complexity	$O(N)$	$O(\sqrt{N})$ (Grover)
Requires fitness evaluation	Yes, explicit	Yes, via quantum oracle
Scalability	Limited by CPU/RAM	Potentially exponential
Precision	Depends on parameters	Depends on the number of layers/depth
Requires special hardware	No	Yes, simulator or quantum processor

6.2. Evaluation of Transmission Scenarios

This section presents a quantitative analysis of the number of helical transmissions required to encode and transmit the phage genome ΦX174 in different M-QAM modulation configurations, the number of parallel strands, and the use of Reed–Solomon $RS(4,2)$ coding. A total of 5400 bases are considered, grouped into blocks of three bases encoded as elements in $GF(2^6)$, as we can see in Figure 8.

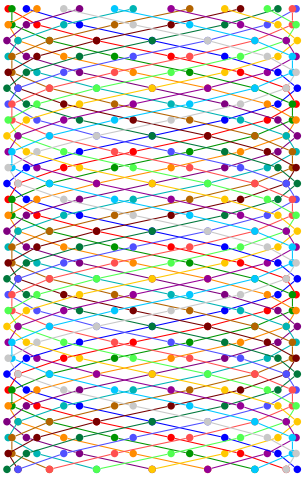

Figure 8. Helical visualization of digital transmission using 16 differentiated strands. Each sphere represents a symbol (for example, four bits in 16-QAM) in a parallel channel. This model allows for a massive and parallel architecture for sending data encoded in $GF(2^6)$ through multiple synchronized streams.

6.3. Analysis of Transmission Efficiency with Varying Modulation and Coding

To evaluate the impact of modulation schemes and error correction on transmission efficiency, we consider the complete encoding of the bacteriophage genome ΦX174, composed of 5400 DNA bases. Each block of three bases is mapped to a symbol in the finite field $GF(2^6)$, yielding 1800 original data blocks. Using different combinations of modulation (4-QAM, 16-QAM, 64-QAM, 256-QAM), coding (with or without RS(4, 2)), and helical strands (16 or 64), we compute the number of required transmissions to deliver the full genome. The RS-coded cases introduce redundancy to protect data, increasing the total number of transmitted bits. Higher-order QAM schemes, however, allow more efficient packing of information per symbol, significantly reducing the required number of transmissions. The most efficient configuration (256-QAM without RS and using 64 threads) requires just a single helical transmission, while the least efficient (4-QAM with RS over 16 threads)

requires 23 transmissions. This illustrates a clear trade-off between redundancy for error correction and transmission efficiency (Table 10).

Table 10. Number of helical transmissions required with different configurations.

Modulation	Bits/Symbol	Threads	RS Coding	Transmissions
4-QAM	2	16	Yes	23
16-QAM	4	16	Yes	12
64-QAM	6	16	Yes	8
256-QAM	8	64	No	1
64-QAM	6	16	No	4

6.4. General Comparison of Configurations

Table 11 presents an extended summary of the number of helical transmissions required in various combinations of modulation, number of strands, and presence of Reed–Solomon $RS(4,2)$ coding. This analysis allows for evaluating the balance between spectral efficiency, parallelism, and error robustness, as well as considering configurations without coding.

Table 11. Number of helical transmissions required according to system parameters.

Modulation	Bits/Symbol	Threads	RS	Necessary Transmissions
4-QAM	2	4	Yes	30
16-QAM	4	4	Yes	12
64-QAM	6	4	Yes	5
256-QAM	8	4	Yes	3
16-QAM	4	16	Yes	3
64-QAM	6	16	Yes	2
256-QAM	8	16	Yes	2
64-QAM	6	64	Yes	2
256-QAM	8	64	Yes	2
4-QAM	2	4	No	15
16-QAM	4	4	No	6
64-QAM	6	4	No	3
256-QAM	8	4	No	2

6.4.1. Evaluation of Transmission Scenarios

This section presents a quantitative analysis of the number of helical transmissions required to encode and transmit the phage genome ΦX174 in different M-QAM modulation configurations, the number of parallel strands, and the use of Reed–Solomon coding $RS(4,2)$.

6.4.2. Comparative Visualization and Interpretative Analysis

To illustrate the visual impact of the previous configurations, Figure 9 shows a comparative representation of the helical transmissions required according to modulation and number of strands. It is evident how greater spectral efficiency and parallelism significantly reduce the number of required transmissions.

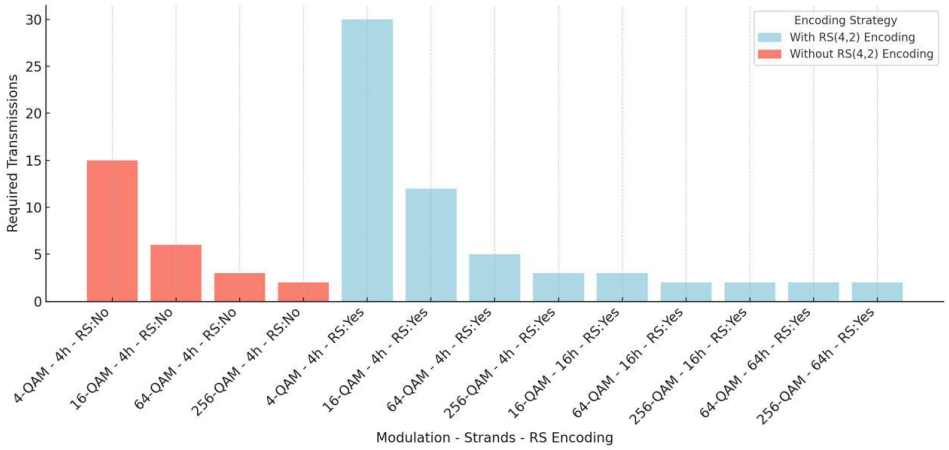

Figure 9. Representation of the helical transmission model for different modulation and parallelism schemes.

In terms of compensation, it is observed that:

- The use of RS coding increases the number of bits transmitted, but provides robustness against errors.
- Higher-order modulations (such as 64-QAM or 256-QAM) significantly reduce transmissions, but can be more sensitive to noise.
- Increasing the number of threads increases the physical complexity of the system, but drastically improves parallelism and efficiency.

Therefore, optimal selection depends on the balance between genetic integrity, transmission speed, and the physical implementation capacity of the helical channel.

6.5. BER Curves

As mentioned earlier, the assignment precedes the stages of MPPM shown in Figure 10, QAM modulation (if used), and CSS coding. Thus, RS coding ensures error protection before the information is transformed by the physical modulation stages, ensuring both the feasibility of the scheme and its efficiency against noise in unguided optical environments.

However, the parameter w does not appear explicitly in the previous equations but is often used as an alternative notation to represent the number of bits per symbol or the number of active positions (equivalent to L) in some modulation contexts. In this document, the standard notation with L is maintained for the number of active pulses, avoiding ambiguity with other variables, such as bandwidth (B) or bit rate (R_b).

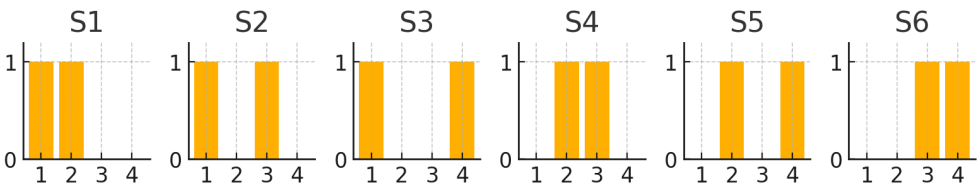

Figure 10. Symbol constellation of MPPM(4, 2): each symbol activates two pulses in a frame of four time slots. There are $\binom{4}{2} = 6$ possible combinations.

Figure 11 presents the performance of BER as a function of the signal-to-noise ratio E_b/N_0 for the multiple transmission schemes characterized in Table 12.

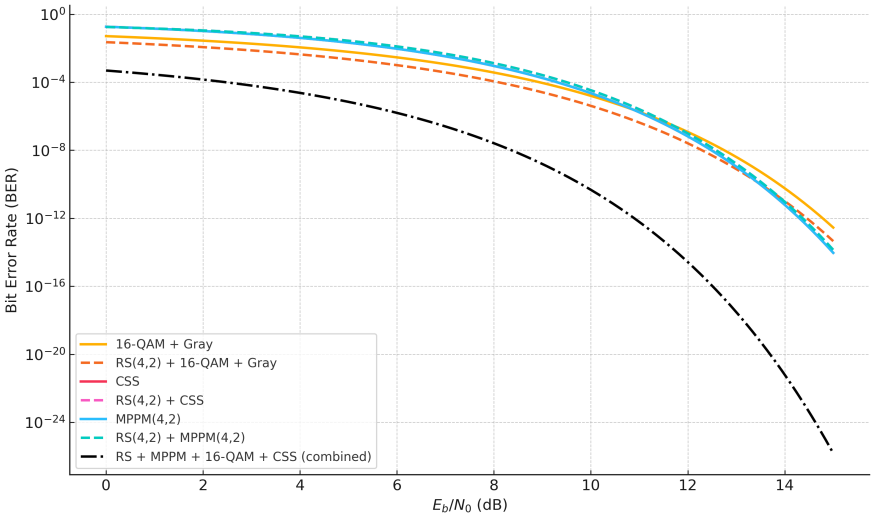

Figure 11. BER curves for different modulation and coding techniques over an AWGN channel. The combination RS(4, 2) + MPPM(4, 2) + 16-QAM with Gray coding and CSS achieves the lowest BER across all E_b/N_0 levels.

The Additive White Gaussian Noise (AWGN) channel represents an essential tool for the theoretical analysis of coding and modulation schemes, thanks to its simplicity and its ability to establish ideal performance limits. In particular, it is suitable as a first approximation to optical fiber modeling, especially in the initial stages of evaluating digital schemes [60]. However, its applicability in fiber optic communication systems is limited, as it does not consider fundamental physical phenomena that influence optical propagation, such as chromatic dispersion, polarization mode dispersion (PMD), attenuation, and non-linear effects induced by signal intensity, such as the Kerr effect.

For short-distance optical links or in early stages of digital system design, the AWGN model can be useful as an initial approximation. Nevertheless, for a more realistic and accurate modeling of optical fibers, it is necessary to use more complex models based on the Non-linear Schrödinger Equation (NLSE) and numerical techniques such as the Split-Step Fourier Method (SSFM), which allow a proper simulation of the interaction between dispersion and non-linear effects.

In summary, while the AWGN channel provides a useful starting point, a comprehensive analysis of optical communication systems requires models that accurately reflect the physical characteristics of the fiber.

The blue curve corresponds to a baseline system that uses 16-QAM modulation ($M = 16$) without any channel coding, which relies only on gray mapping to minimize symbol errors. The orange curve illustrates the improvement gained by incorporating a classical Reed–Solomon code RS(4, 2) over $GF(2^8)$, showing enhanced resilience to noise due to the error-correction capability ($t = 8$ symbols). The solid green curve depicts a hybrid modulation scheme based on MPPM with parameters $m = 4$, $L = 2$, combined with QAM and Chirp Spread Spectrum (CSS) with spreading factor $n = 4$, but without redundancy. A dashed green variant represents the same hybrid structure with alternative physical parameters such as the chirp bandwidth B or the pulse timing. Finally, the dashed red curve shows the full hybrid configuration that integrates RS (4, 2) coding over $GF(2^6)$, MPPM(4, 2), 16-QAM with Grey coding, and CSS spreading. This integrated approach achieves the most robust performance, attaining BER values below 10^{-6} at moderate E_b/N_0 levels (around 11 dB), demonstrating its effectiveness for transmission over noisy optical channels, especially in NLOS scenarios.

Table 12. Simulation parameters for transmission methods.

Parameter	Value	Description
n, k	4, 2	RS(n,k) code parameters: n is the codeword length and k is the message length.
$\mathbf{m}(x)$	–	Message polynomial over $GF(2^m)$ used in RS encoding.
$g(x)$	–	Generator polynomial of degree $n-k$ used in RS encoding.
$\mathbf{c}(x)$	–	Codeword polynomial after RS encoding.
M	16	Modulation order for QAM (16-QAM: 4 bits/symbol).
Gray Mapping	Enabled	Ensures adjacent QAM symbols differ by only one bit.
a, b	0 to $\sqrt{M}-1$	Indices used to generate QAM symbols.
s	–	Unnormalized complex QAM symbol.
x	–	Normalized transmitted QAM symbol.
m	4	Number of time slots in an MPPM symbol frame.
L	2	Number of active pulses in MPPM per symbol.
N_{bits}	2	Number of bits encoded per MPPM symbol: $\lfloor \log_2 \binom{m}{L} \rfloor$.
f_0	variable	Initial frequency of CSS chirp.
B	variable	Bandwidth of the chirp signal used in CSS.
T	1	Duration of each CSS symbol (normalized).
k	B/T	Chirp rate used in the frequency sweep of CSS.
$x(t)$	–	CSS waveform as a function of time.
d	0 to 2^n-1	Index for data-dependent CSS delay.
τ_d	$dT/2^n$	Time delay applied to MPPM waveform in CSS.
n (CSS)	4	Number of bits encoded per CSS symbol (defines 2^n shifts).
E_b/N_0	0 to 15 dB	Energy-per-bit to noise ratio used in simulation.
Channel	AWGN	Additive White Gaussian Noise channel model.

Table 13 presents the effective throughput (in bits per symbol) achieved by hybrid configurations combining Reed–Solomon coding, MPPM, high-order QAM, and CSS. The results highlight how the choice of parameters affects overall efficiency. For example, using RS(255, 239) with 64-QAM and CSS(3) yields 10.310 bits/symbol, while extending to 256-QAM with CSS(4) increases it to 12.248 bits/symbol. Even with more redundant codes such as RS(255, 191), throughputs above 11 bits/symbol are achievable when combined with MPPM(6, 3) and 256-QAM.

Table 13. Effective throughput for hybrid RS + MPPM + QAM + CSS configurations.

RS(n, k)	MPPM (m, L)	QAM (M)	CSS Bits (n)	RS Rate	MPPM Bits	QAM Bits	Throughput
RS(255, 239)	(4, 2)	16	2	0.937	2	4	7.498
RS(255, 239)	(4, 2)	64	3	0.937	2	6	10.310
RS(255, 239)	(4, 2)	256	4	0.937	2	8	12.248
RS(255, 223)	(5, 2)	64	3	0.875	3	6	10.500
RS(255, 223)	(6, 3)	64	4	0.875	3	6	11.375
RS(255, 191)	(6, 3)	256	4	0.749	3	8	11.242

These configurations reflect trade-offs between redundancy and spectral efficiency. Higher-order modulations and broader CSS spreading enhance throughput but increase sensitivity to noise, whereas stronger RS coding improves robustness at the cost of added

redundancy. MPPM contributes to temporal resolution and energy efficiency, especially in optical systems.

To evaluate robustness, Figure 12 shows the BER performance of a 64-QAM system over an NLOS VLC channel, modeled with log-normal fading to reflect multipath dispersion and intensity fluctuations typical in indoor optical environments.

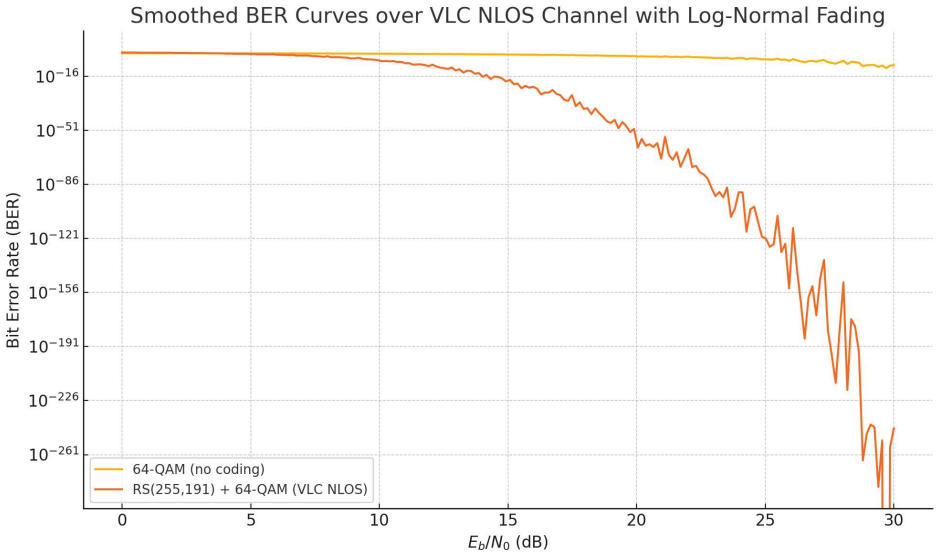

Figure 12. Smoothed BER curves for 64-QAM over a VLC NLOS channel modeled with log-normal fading. The use of RS(255, 191) significantly improves noise resilience by reducing the BER at all SNR levels.

The 64-QAM (no coding) curve reveals a steep BER floor under fading conditions, while the RS(255, 191) + 64-QAM configuration achieves BER below 10^{-5} for $E_b/N_0 > 20$ dB, confirming the critical role of forward error correction in maintaining link reliability in challenging VLC NLOS scenarios.

Together, the table and figure demonstrate that these hybrid schemes can be tuned to balance transmission efficiency and resilience, according to the operational constraints of the system and the channel conditions (Table 14).

Table 14. Simulation parameters for BER over VLC NLOS channel with MPPM, QAM, and CSS.

Parameter	Value	Description
		Channel and Fading Model
Channel Model	VLC NLOS	Non-line-of-sight visible light communication with optical diffusion.
Fading Model	Log-normal	Models multipath scattering and reflection in intensity-based optical channels.
Log-normal Mean (μ)	0	Mean of log-amplitude distribution (linear domain).
Log-normal Std. Dev. (σ)	0.5	Standard deviation of log-amplitude fading (typical for indoor optical NLOS).
SNR Range (E_b/N_0)	0 to 30 dB	Signal-to-noise ratio in dB.
Monte Carlo Realizations	100	Number of fading realizations per SNR point for averaging.

Table 14. *Cont.*

Parameter	Value	Description
	Modulation and Coding	
MPPM Parameters	$m = 4$, $L = 2$	Multi-Pulse Position Modulation: 2 active pulses in 4 time slots.
MPPM Bits per Symbol	$\left\lfloor \log_2 \binom{4}{2} \right\rfloor = 2$	Number of encoded bits per MPPM symbol.
QAM Order	64-QAM	Quadrature Amplitude Modulation with $M = 64$ (6 bits per symbol).
Gray Mapping	Enabled	Mapping symbols so adjacent constellation points differ by 1 bit.
CSS Spreading Factor	$n = 4$	Number of bits encoded by CSS delay shifts ($2^n = 16$ positions).
CSS Bandwidth (B)	Variable	Bandwidth of chirp waveform (application-dependent).
CSS Symbol Duration (T)	1 (normalized)	Normalized duration of one chirp symbol.
	Error Correction	
Coding Scheme	RS(255, 191)	Reed–Solomon code over $GF(2^8)$ with $n = 255$, $k = 191$.
RS Rate	$R = 191/255 \approx 0.749$	Proportion of useful data in each RS block.
Error Correction Capability	$t = \frac{n-k}{2} = 32$	Maximum number of correctable symbol errors per block.
BER Estimation Method	Analytical + Block Error Model	Combines theoretical BER of QAM with RS block failure probability.

7. Discussion and Analysis of Results

The comparison between classical and quantum search algorithms, as presented in Table 9, highlights key advantages and limitations of both approaches. The Grover algorithm exhibits a quadratic speed-up, reducing the search complexity from $O(N)$ to $O(\sqrt{N})$, which is particularly valuable in unstructured search problems. While genetic algorithms rely on stochastic evaluation over populations and are widely deployable on classical hardware, Grover exploits quantum superposition and amplitude amplification to converge faster. However, its implementation requires quantum hardware or simulators, limiting its current scalability in real-world deployments.

The evaluation of helical architecture-based transmission scenarios demonstrates the importance of combining modulation order, parallelism, and error correction. For the transmission of the ΦX174 phage genome encoded on $GF(2^6)$, the results show that increasing the modulation order (e.g., from 4-QAM to 256-QAM) dramatically reduces the number of helical transmissions. For example, using 256-QAM with 64 strands achieves the transmission in a single step, while lower-order modulations require up to 23 transmissions when combined with Reed–Solomon (RS) coding. This confirms that higher spectral efficiency and parallelization significantly enhance system performance.

However, the use of RS codes introduces redundancy, effectively doubling the transmitted bits. This increases the number of required transmissions, but provides strong protection against errors in optical wireless environments. As shown in Table 11, the trade-off is evident: Uncoded systems require fewer transmissions but become unreliable in noisy or fading conditions.

In terms of BER performance, Figure 11 shows that the hybrid configuration using RS(4, 2), MPPM(4, 2), 16-QAM with Gray mapping, and Chirp Spread Spectrum (CSS) achieves BER values below 10^{-6} at $E_b/N_0 \approx 11$ dB. This configuration combines the temporal precision of MPPM, the spectral compactness of QAM, and the noise resistance of CSS, providing a powerful scheme for optical transmission.

Performance under NLOS VLC conditions, modeled with log-normal fading, further underscores the importance of forward error correction. As shown in Figure 12, a 64-QAM

system without coding experiences a floor BER, while RS(255, 191) coding reduces the BER below 10^{-5} for $E_b/N_0 > 20$ dB. This validates the critical role of RS codes in maintaining reliable links under multipath dispersion and intensity fluctuations.

Lastly, the throughput analysis in Table 13 shows that hybrid schemes can exceed 12 bits per symbol using high-order QAM and CSS with low-redundancy RS codes. However, stronger RS coding still achieves more than 11 bits/symbol when combined with efficient MPPM parameters. These results emphasize the need to carefully balance redundancy, modulation, and coding according to the constraints of the system and channel conditions.

In summary, the combination of advanced modulation techniques, forward error correction, and parallel transmission architectures provides a scalable and robust framework for high-efficiency optical wireless communication, especially in bioinspired or helical transmission scenarios. The potential integration of quantum search mechanisms could further enhance encoding and optimization in such systems.

8. Conclusions

The results obtained in this study focus on advanced computational modeling and optimization in operations research and supply chain management. The proposed helical transmission system, along with hybrid modulation and quantum–classical coding techniques, constitutes a concrete example of how advanced computational models can be applied to complex problems of logistics and distributed information processing.

In particular, the integration of multithread modulation, Reed–Solomon coding, and hybrid schemes such as RS-MPPM-QAM-CSS demonstrates the ability of optimization approaches to significantly reduce the number of transmissions required, improving spectral and energy efficiency. This is especially relevant for communication systems in distributed logistics networks, industrial sensors, or real-time monitoring platforms.

Furthermore, the comparative analysis between classical and quantum algorithms, such as Grover's algorithm, provides a prospective view on how emerging artificial intelligence and quantum computing techniques can improve decision-making processes in complex industrial environments, aligning with the trends promoted by the Special Issue: the convergence between artificial intelligence, mathematical modeling, and computational optimization.

These results contribute both to academic advancement and to practical impact in industrial engineering and smart logistics, proposing a robust and scalable framework that can be extended to various applications within the future supply chain.

Author Contributions: Conceptualization, I.S.; methodology, I.S. and V.G.; software, I.S.; validation, I.S., V.G. and P.P.J.; formal analysis, I.S. and P.P.J.; investigation, I.S. and V.G.; resources, I.S.; data curation, I.S. and V.G.; writing—original draft preparation, I.S.; writing—review and editing, I.S. and V.G.; visualization, I.S.; supervision, I.S.; project administration, I.S.; funding acquisition, I.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Projects FOVI240009, Project FONDECYT Iniciación 11240799 and 563.B.XVI.25 UDLA/Ecuador; and Project Dicyt/USACH/code 062413SG.

Institutional Review Board Statement: Not applicable for studies not involving humans or animals.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
ANN	Artificial Neural Networks
APD	Avalanche Photodetector
AWGN	Additive White Gaussian Noise
BCH	Bose–Chaudhuri–Hocquenghem
BER	Bit Error Rate
CNN	Convolutional Neural Network
CQA	Critical Quality Attributes
CSS	Chirp Spread Spectrum
CX	Controlled X (gate)
DOAJ	Directory of Open Access Journals
dsDNA	Double-Stranded DNA
FEC	Forward Error Correction
GA	Genetic Algorithm
GF	Galois Field (Finite Field)
IM/DD	Intensity Modulation/Direct Detection
IR	Infrared
LD	Linear Dichroism
LDPC	Low-Density Parity-Check
ML	Machine Learning
MLP	Multilayer Perceptron
MPPM	Multi-Pulse Position Modulation
NLOS	Non-Line-of-Sight
PAM	Pulse Amplitude Modulation
PCR	Polymerase Chain Reaction
PSO	Particle Swarm Optimization
QAM	Quadrature Amplitude Modulation
QRS	Quantum Reed–Solomon
RF	Radio Frequency
RRNS	Residue Number System
RS	Reed–Solomon
SER	Symbol Error Rate
SF	Spreading Factor
SNR	Signal-to-Noise Ratio
ssDNA	Single-Stranded DNA
SSF	Solid-State Fermentation
UV	Ultraviolet
VLC	Visible Light Communication

Appendix A. Colateral Proofs

Theorem A1 (BER of M-QAM with CSS (Spreading Factor))**.** *Let it be a digital communication system that uses M-QAM modulation in an AWGN channel with coherent detection. If a CSS scheme with spreading factor SF is applied, then the BER is approximated by:*

$$BER_{M\text{-}QAM,\,SF} \approx \frac{4}{\log_2 M}\left(1-\frac{1}{\sqrt{M}}\right) \cdot Q\left(\sqrt{\frac{3\cdot \log_2 M}{M-1}\cdot 10^{\frac{E_b/N_0}{10}}\cdot 2^{SF}}\right) \tag{A1}$$

Proof. The standard BER expression for M-QAM modulation in an AWGN channel is:

$$\mathrm{BER}_{M\text{-QAM}} \approx \frac{4}{\log_2 M}\left(1-\frac{1}{\sqrt{M}}\right)\cdot Q\left(\sqrt{\frac{3\cdot\log_2 M}{M-1}\cdot\frac{E_b}{N_0}}\right) \tag{A2}$$

When CSS is applied with spreading factor SF, a processing gain is introduced:

$$G_p = 10\cdot\log_{10}(2^{SF}) = SF\cdot 3.01 \text{ dB},$$

which effectively increases the E_b/N_0 ratio:

$$\left(\frac{E_b}{N_0}\right)_{\text{effective}} = \left(\frac{E_b}{N_0}\right) + SF\cdot 3.01.$$

Substituting this into (A2) yields equation (A1). □

Appendix A.1. BER of M-QAM with CSS and RS Code $RS(n,k)$

Theorem A2 (BER of M-QAM with CSS and RS Code $RS(n,k)$)**.** *Let it be a digital system employing M-QAM modulation over an AWGN channel, with Reed–Solomon coding $RS(n,k)$ and CSS with spreading factor SF. Then, the approximate bit error rate is:*

$$BER_{final} \approx \frac{t+1}{n\cdot\log_2 M}\cdot\binom{n}{t+1}\cdot\left(BER_{M\text{-}QAM,\,SF}\right)^{t+1}, \tag{A3}$$

where:

$$BER_{M\text{-}QAM,\,SF} \approx \frac{4}{\log_2 M}\left(1-\frac{1}{\sqrt{M}}\right)\cdot Q\left(\sqrt{\frac{3\log_2 M}{M-1}\cdot 10^{\frac{E_b/N_0+SF\cdot 3.01}{10}}}\right) \tag{A4}$$

Proof. For RS(n,k) coding, the code corrects up to $t = \lfloor\frac{n-k}{2}\rfloor$ errors per block. The probability of block error is dominated by the occurrence of $t+1$ or more symbol errors:

$$P_{\text{block}} \approx \binom{n}{t+1}\cdot\left(\mathrm{BER}_{M\text{-QAM, SF}}\right)^{t+1}.$$

By averaging the block error over $n\cdot\log_2 M$ bits, the final BER becomes:

$$\mathrm{BER}_{\text{final}} \approx \frac{t+1}{n\cdot\log_2 M}\cdot P_{\text{block}}.$$

Substituting (A4) yields the result. □

Appendix A.2. Lower Bound for Spreading Factor in RS–MPPM–QAM–CSS Systems

Theorem A3 (Lower Bound for Spreading Factor in RS–MPPM–QAM–CSS Systems)**.** *Let a hybrid optical communication system integrate Reed–Solomon coding $RS(n,k)$, Multi-Pulse Position Modulation (MPPM) with N slots and w pulses, and M_q-QAM. Assume CSS modulation is applied with spreading factor SF over a non-line-of-sight VLC channel.*

If the target bit error rate is BER_{target}, then the spreading factor must satisfy:

$$2^{SF} \geq \frac{1}{BER_{target}} \cdot \left(\frac{t+1}{nw \cdot \log_2 M_q \cdot \binom{N}{w}} \cdot \binom{n}{t+1} \cdot BER_{phys}^{t+1} \right), \tag{A5}$$

Proof. The total bit content per hybrid symbol is:

$$R_{\text{symbol}} = \log_2 \binom{N}{w} + w \log_2 M_q.$$

The RS block error probability for $t+1$ or more symbol errors is approximated by:

$$P_e^{\text{RS}} \approx \binom{n}{t+1} \cdot (\text{BER}_{\text{phys}})^{t+1}.$$

This is averaged over $n \cdot w \cdot \log_2 M_q$ bits, yielding:

$$\text{BER}_{\text{uncorrected}} \approx \frac{t+1}{nw \cdot \log_2 M_q} \cdot \binom{n}{t+1} \cdot (\text{BER}_{\text{phys}})^{t+1}.$$

After applying CSS, the BER is reduced by a spreading gain factor $1/2^{\text{SF}}$:

$$\text{BER}_{\text{final}} = \frac{1}{2^{\text{SF}}} \cdot \text{BER}_{\text{uncorrected}}.$$

To ensure $\text{BER}_{\text{final}} \leq \text{BER}_{\text{target}}$, we impose:

$$2^{\text{SF}} \geq \frac{\text{BER}_{\text{uncorrected}}}{\text{BER}_{\text{target}}}.$$

□

Theorem A4 (Practical list sizes from our SNR/RS/CSS model)**.** *Using Equations* (A1) *and* (A3) *and the target* BER*, we obtain the expected symbol error rate and the error-weight distribution per RS block under our channel and modulation parameters. For an $RS(n,k)$ code over $\mathbb{F}_{2^m}$ with unique decoding radius $t = \lfloor (n-k)/2 \rfloor$, a conservative upper bound for the candidate space explored by the list/Grover stage is*

$$N_{\text{cand}}(t) \;=\; \sum_{i=0}^{t} \binom{n}{i} (2^m - 1)^i,$$

while the expected list size L at the operating SNR follows from the tail of the (binomial/Poisson) error-weight distribution implied by our BER *model. (In practice, L is very close to 1 in the operating regime with low post-CSS* BER*; when pushing the rate or lowering the SNR, L grows, and the quantum module becomes relevant.)*

Proof. Let $\text{RS}(n,k)$ be defined over $\mathbb{F}_{2^m}$ with a unique decoding radius $t = \lfloor (n-k)/2 \rfloor$, and denote $q = 2^m$.

(i) Bound for $N_{\text{cand}}(t)$. An error pattern of weight i is determined by: (a) the choice of i positions out of n, contributing $\binom{n}{i}$ possibilities, and (b) the selection, for each of these positions, of a non-zero erroneous symbol in $\mathbb{F}_q$, contributing $(q-1)^i$ possibilities. By the multiplicative principle, the total number of error patterns of weight i is

$$\binom{n}{i} (q-1)^i.$$

If the search procedure (list/Grover) only explores patterns of weight at most t, the size of the search space is bounded by

$$N_{\text{cand}}(t) \;=\; \sum_{i=0}^{t} \binom{n}{i} (q-1)^i,$$

which sums the counts for all admissible weights. By construction, this is a conservative upper bound on the number of candidates the method needs to examine, since any further restriction on the exploration (e.g., syndrome screening) can only reduce this total.

(ii) Expected list size L. Under the channel/modulation model, let p_s be the per-symbol error probability induced by Equations (A1) and (A3). Assuming effective independence between symbols (a standard approximation in the operating regime), the error weight E per block satisfies

$$E \sim \text{Binomial}(n, p_s).$$

For MDS codes like RS, if $E \le t$, unique decoding is valid, and the set of marked solutions in the search space (the marked items for Grover) is exactly one: the true error pattern (equivalently, the transmitted codeword). If $E > t$, there may be zero or multiple candidates compatible with the decoder/list's internal checks (depending on the list radius and additional tests). By linearity of expectation,

$$\mathbb{E}[L] \;=\; \mathbb{E}[\,\#\text{marked solutions}\,] \;=\; \Pr[E \le t] \cdot 1 \;+\; \mathbb{E}[\,\#\text{solutions} \mid E > t\,] \Pr[E > t].$$

In the operating regime of interest (post-CSS with low BER), $p_s \ll 1$, and the tail $\Pr[E > t] = \sum_{i=t+1}^{n} \binom{n}{i} p_s^i (1-p_s)^{n-i}$ is exponentially small in $t+1$. Thus,

$$\mathbb{E}[L] \;=\; \Pr[E \le t] \;+\; O\big(\Pr[E > t]\big) \;=\; 1 - \Pr[E > t] \;+\; O\big(\Pr[E > t]\big) \;=\; 1 - o(1),$$

i.e., $L \simeq 1$ when the system operates at high SNR (or moderate rate). For large n and small p_s with $\lambda = np_s$ fixed, the Poisson approximation $E \approx \text{Poisson}(\lambda)$ and the tail

$$\Pr[E > t] \;\approx\; 1 - e^{-\lambda} \sum_{i=0}^{t} \frac{\lambda^i}{i!}. \tag{A6}$$

can be used to explicitly set $L \approx 1 - \Pr[E > t]$ and quantify its growth as SNR decreases or the rate increases.

(iii) Grover's complexity. If the search space has size $N_{\text{cand}}(t)$ and there are L marked solutions on average, the expected number of Grover iterations is

$$T_G \;\approx\; \frac{\pi}{4} \sqrt{\frac{N_{\text{cand}}(t)}{L}}. \tag{A7}$$

In the regime where $L \simeq 1$, this reduces to $T_G \approx \frac{\pi}{4} \sqrt{N_{\text{cand}}(t)}$, consistent with the numerical example shown.

With (i)–(iii), the bound for $N_{\text{cand}}(t)$ and the asymptotic characterization of L based on the tail of the error weight distribution induced by the BER model are justified. □

Appendix B. Procedures

Algorithm A1 Genetic encoding in $GF(2^m)$ with Reed–Solomon (n, k)—Part I

Input: Genetic sequence of N bases (A, C, G, T), parameters m, k, n
Output: Encoded blocks with symbols of length n
Step 1: Binary Encoding
for each base b in the sequence **do**
    Encode b as: A $\rightarrow$ 00, C $\rightarrow$ 01, G $\rightarrow$ 10, T $\rightarrow$ 11
end for
Step 2: Grouping into blocks of b bits
Group every b bits ($b = m$) and convert to decimal value $r \in [0, 2^m - 1]$
Interpret r as a symbol in $GF(2^m)$
Step 3: Reed–Solomon Encoding
Group the symbols into blocks of k data
for each block $(d_1, \ldots, d_k)$ **do**
    Calculate $n - k$ parity symbols $(p_1, \ldots, p_{n-k})$
    Form the encoded block: $[d_1, \ldots, d_k, p_1, \ldots, p_{n-k}]$
end for

Algorithm A2 Helical genetic modulation and decoding—Part II

Input: Encoded blocks, parameters M, μ, h, b
Output: Reconstructed genetic sequence
Step 4: Assignment of Physical Fragments
for each symbol r in the encoded blocks **do**
    Assign a DNA fragment S_r with a unique length
end for
Step 5: M-QAM Modulation
Group bits by helical level: μ strands, h levels $\Rightarrow \mu \cdot h$ symbols
for each group of $\log_2 M$ bits **do**
    Map to a point in the M-QAM constellation
end for
Step 6: Transmission and Recovery
Transmit in parallel by μ threads
Read fragments using physical technique (e.g., electrophoresis)
for each observed band **do**
    Estimate length $\rightarrow$ identify symbol r
end for
Step 7: Decoding
Group into blocks of length n
Apply Reed–Solomon decoding (n, k)
for each corrected symbol **do**
    Convert to b bits
    Translate to base triplets using inverse table
end for

Algorithm A3 Genetic decoding assisted by Grover—quantum version

Input: Recovered physical fragments, parameters n, k, m, list $\mathcal{L}$ of candidates
Output: Corrected data blocks translated to genetic bases
Step 1: Physical Measurement of Fragments
for each observed fragment **do**
    Measure length $\Rightarrow$ identify symbol $r \in GF(2^m)$
end for
Step 2: List Decoding
for each block of n symbols **do**
    Apply list decoding (e.g., Guruswami–Sudan)
    Generate list $\mathcal{L}_i = \{c_1, \ldots, c_L\}$
end for
Step 3: Quantum Search for the Correct Message
for each list $\mathcal{L}_i$ **do**
    Define oracle function $f(c)$ that returns 1 if c is correct
    Apply Grover's algorithm on $\mathcal{L}_i$
    Identify $c^\star$ such that $f(c^\star) = 1$
end for
Step 4: Sequence Reconstruction
for each block $c^\star$ **do**
    Convert symbols to bits
    Translate to genetic bases using the inverse binary table
end for

References

1. Urban-Chmiel, R.; Pyzik, E. Selected mechanisms of action of bacteriophages in bacterial infections in animals. *Viruses* **2025**, *17*, 101. [CrossRef]
2. Yuan, K. The application value assessment of CRISPR in screening and optimizing bacteria for industrial production. *Theor. Nat. Sci.* **2025**, *82*, 27–32. [CrossRef]
3. Cho, E.; Kim, J.; Won, T.-B.; Ryu, S.; Jeon, B. Virulent bacteriophages for controlling Shiga toxin-producing *Escherichia coli* (STEC) without inducing toxin production. *J. Infect. Dis.* **2025**, jiaf035. [CrossRef]
4. Cuellar, H.B.; Trujillo, M.A.; de la, L.; Membrillo Venegas, I. Modelado y optimización de la fermentación en estado sólido: El poder de la inteligencia artificial en procesos biotecnológicos. *Terys* **2024**, *3*, 170–174. [CrossRef]
5. Mishra, S.; Tiwari, A.M. Current Trends of Artificial Intelligence in Biotechnology. *Open Access J. Data Sci. Artif. Intell.* **2024**, *2*, 000112. [CrossRef]
6. Garza-Ulloa, J. Artificial Intelligence Models Applied to Biomedical Engineering. In *Applications of AI in Biomedical Engineering*; Academic Press: Cambridge, MA, USA, 2021. [CrossRef]
7. Lavery, R. Modelling the DNA double helix: Techniques and results. In *Computational Approaches to Biomolecular Modeling*; Springer: Berlin/Heidelberg, Germany, 1995. [CrossRef]
8. Gibbons, A.; Amos, M.; Hodgson, D.A. Models of DNA Computation. In *DNA Based Computers*; Springer: Berlin/Heidelberg, Germany, 1997. [CrossRef]
9. Lewin, D.I. DNA computing. *IEEE Comput.* **2001**, *4*, 5–8. [CrossRef]
10. Wang, B.; Wang, S.S.; Chalk, C.; Ellington, A.D.; Soloveichik, D. Parallel molecular computation on digital data stored in DNA. *Proc. Natl. Acad. Sci. USA* **2022**, *120*, e2217330120. [CrossRef]
11. Mol, M.; Singh, S. Computational Design of Biological Systems. In *Synthetic Biology*; Elsevier: Amsterdam, The Netherlands, 2015. [CrossRef]
12. Gayathiri, E.; Prakash, P.; Kumaravel, P.; Jayaprakash, J.; Ragunathan, M.G.; Sankar, S.; Piaraj, S.; Thirumalaivasan, N.; Thiruvengadam, M.; Govindasamy, R. Computational approaches for modeling and structural design of biological systems: A comprehensive review. *Prog. Biophys. Mol. Biol.* **2023**, *185*, 17–32. [CrossRef]
13. Wang, B.; Wang, S.S.; Chalk, C.; Ellington, A.D.; Soloveichik, D. Parallel molecular computation on digital data stored in DNA. *bioRxiv* **2022**. [CrossRef]
14. Lin, Y.S.; Liang, Y.P.; Chang, Y.H.; Shih, W.K.; Lim, W.S. Bridging DNA Storage and Computation. In Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing, Avila, Spain, 8–12 April 2024. [CrossRef]
15. Cevallos, Y.; Tello-Oquendo, L.; Inca, D.; Palacios, C.; Rentería, L. Genetic Expression in Biological Systems: A Digital Communication Perspective. *Open Bioinform. J.* **2019**, *12*, TOBIOIJ-12-45. [CrossRef]

16. Cudby, J.; Strelchuk, S. Quantum Algorithms for Genome Sequencing and Analysis. In Proceedings of the 2024 IEEE International Conference on Quantum Computing and Engineering (QCE), Montreal, QC, Canada, 15–20 September 2024. [CrossRef]
17. Varsamis, G.D.; Karafyllidis, I.G.; Gilkes, K.M.; Arranz, U.; Martin-Cuevas, R.; Calleja, G.; Dimitrakis, P.; Kolovos, P.; Sandaltzopoulos, R.; Jessen, H.C.; et al. Quantum Gate Algorithm for Reference-Guided DNA Sequence Alignment. *Comput. Biol. Chem.* **2023**, *107*, 107959. [CrossRef]
18. Sarkar, A.; Al-Ars, Z.; Almudever, C.G.; Bertels, K. QiBAM: Approximate Sub-String Index Search on Quantum Accelerators Applied to DNA Read Alignment. *Electronics* **2023**, *10*, 2433. [CrossRef]
19. Rivelino, R. Deterministic Storage of Quantum Information in the Genetic Code. *arXiv* **2024**. [CrossRef]
20. D'Acunto, M. Quantum Computation by Biological Systems. *IEEE Trans. Mol. Biol.-Multi-Scale Commun.* **2023**, *9*, 257–262. [CrossRef]
21. Hunt, F.; Perkins, S.; Smith, D.H. Channel models and error correction codes for DNA information storage. *Int. J. Inf. Coding Theory (IJICOT)* **2015**, *3*, 120–136. [CrossRef]
22. Lenz, A.; Siegel, P.H.; Wachter-Zeh, A.; Yaakobi, E. Coding over Sets for DNA Storage. *IEEE Trans. Inf. Theory* **2018**, *66*, 2331–2351. [CrossRef]
23. Kiah, H.M.; Puleo, G.J.; Milenkovic, O. Codes for DNA Sequence Profiles. *IEEE Trans. Inf. Theory* **2016**, *62*, 3493–3503. [CrossRef]
24. Zan, X.; Xie, R.; Yao, X.; Xu, P.; Liu, W. A Robust and Efficient DNA Storage Architecture Based on Modulation Encoding and Decoding. *J. Chem. Inf. Model.* **2023**, *63*, 3967–3976. [CrossRef]
25. Luo, J.; Mu, L.; Huang, Y.J.W.; Yan, Y.; Han, G.; Zhong, Y. Two RRNS-Based Error Correction Schemes for DNA Storage Channels. *IEEE Commun. Lett.* **2024**, *28*, 2729–2733. [CrossRef]
26. Hamoum, B.; Dupraz, E. Synchronization Algorithms from High-Rate LDPC Codes for DNA Data Storage. In Proceedings of the IEEE DSP Conference, Rhodes, Greece, 11–13 June 2023; pp. 1–5. [CrossRef]
27. Yu, Z.; Baxley, R.J.; Zhou, G.T. EVM and Achievable Data Rate Analysis of Clipped OFDM Signals in Visible Light Communication. *EURASIP J. Wirel. Commun. Netw.* **2012**, *2012*, 321. [CrossRef]
28. Gao, Q.; Gong, C.; Li, S.; Xu, Z. DD-informative Modulation for Visible Light Communications Under Lighting Constraints. *IEEE Wirel. Commun.* **2015**, *22*, 86–93. [CrossRef]
29. Yao, S.; Zhang, X. Joint Beamforming and DC Bias Optimization in VLC with Dimming Control. In Proceedings of the 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), Sydney, Australia, 4–7 June 2017; pp. 1–5. [CrossRef]
30. Obeed, M.; Dahrouj, H.; Salhab, A.M.; Zummo, S.A.; Alouini, M.-S. DC-Bias Allocation in Cooperative VLC Networks via Joint Information and Energy Transfer. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 9–13 December 2018; pp. 1–6. [CrossRef]
31. Camps, D.; Beeumen, R.V.; Yang, C. Quantum Fourier Transform Revisited. *Numer. Linear Algebra Appl.* **2018**, *25*, e2331. [CrossRef]
32. Troyer, M. Review: A Comparison of Quantum and Traditional Fourier Transform Computations. *Authorea Prepr.* **2020**. [CrossRef]
33. De Vos, A. Reversible and endoreversible computing. *Int. J. Theor. Phys.* **1986**, *29*, 1251–1264. [CrossRef]
34. Nolan, T.C.; Stark, W.E. A recursive method for calculating error probabilities for a Reed–Solomon codeword. In Proceedings of the VTC '98. 48th IEEE Vehicular Technology Conference. Pathway to Global Wireless Revolution (Cat. No.98CH36151), Ottawa, ON, Canada, 21–21 May 1998; pp. 686–690. [CrossRef]
35. Wei, S.; Long, G.-L. Efficient simulation of open quantum system in duality quantum computing. In Proceedings of the SPIE 9996, Quantum Information and Computation XIV, Beijing, China, 12–14 October 2016. [CrossRef]
36. Zheng, C. Universal quantum simulation of single-qubit nonunitary operators using duality quantum algorithm. *Sci. Rep.* **2021**, *11*, 3960. [CrossRef]
37. Kamakari, H.; Sun, S.N.; Motta, M.; Minnich, A.J. Digital quantum simulation of open quantum systems using quantum imaginary–time evolution. *PRX Quantum* **2022**, *3*, 010320. [CrossRef]
38. Tang, N.; Han, Y.S. A new decoding method for Reed–Solomon codes based on FFT and modular approach. *IEEE TRansactions Commun.* **2022**, *70*, 7790–7801. [CrossRef]
39. Zhao, J.; Zhang, W.; Liu, Y. An efficient construction method based on partial distance of polar codes with Reed–Solomon kernel. *arXiv*, **2023**. [CrossRef]
40. Grassl, M.; Geiselmann, W.; Beth, T. Quantum Reed–Solomon Codes. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 231–244. [CrossRef]
41. Howard, M. Classical codes in quantum state space. *J. Phys. A Math. Theor.* **2015**, *48*, 495303. [CrossRef]
42. Nishio, S.; Piparo, N.L.; Hanks, M.; Munro, W.J.; Nemoto, K. Resource Reduction in Multiplexed High-Dimensional Quantum Reed–Solomon Codes. *Phys. Rev. A* **2023**, *107*, 032620. [CrossRef]
43. Li, L.; Zhu, S.; Liu, L.; Kai, X. Entanglement-assisted quantum MDS codes from generalized Reed–Solomon codes. *Quantum Inf. Process.* **2019**, *18*, 153. [CrossRef]
44. Nadkarni, P.J.; Garani, S.S. Entanglement-Assisted Reed–Solomon Codes over Qudits: Theory and Architecture. *Quantum Inf. Process.* **2021**, *20*, 129. [CrossRef]

45. Arikan, E. An information-theoretic analysis of Grover's algorithm. In *Quantum Communication and Information Technologies*; NATO Science Series; Springer: Dordrecht, The Netherlands, 2002; Volume 113, pp. 339–347. [CrossRef]
46. Kravchenko, D.; Nahimovs, N.; Rivosh, A. Grover's Search with Faults on Some Marked Elements. In *SOFSEM 2016: Theory and Practice of Computer Science*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2015. [CrossRef]
47. Roy, T.; Jiang, L.; Schuster, D. Deterministic Grover search with a restricted oracle. *Phys. Rev. Res.* **2022**, *4*, L022013. [CrossRef]
48. Lee, S.; Nam, S.Y. Finding All Solutions with Grover's Algorithm by Integrating Estimation and Discovery. *Electronics* **2023**, *13*, 4830. [CrossRef]
49. Bolton, F.M. Accelerated quantum search using partial oracles and Grover's algorithm. *arXiv* **2024**. [CrossRef]
50. Tani, K.; Tsuchiya, S.; Tani, S.; Takeuchi, Y. Quantum algorithm for unstructured search of ranked targets. *arXiv* **2024**. [CrossRef]
51. Khurana, S.; Nene, M.J. Implementation of Database Search with Quantum Computing: Grover's Algorithm vs Linear Search. In Proceedings of the 2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE), Ballari, India, 2–3 November 2023. [CrossRef]
52. Vemula, D.R.; Konar, D.; Satheesan, S.; Kalidasu, S.M.; Cangi, A. A Scalable 5, 6-Qubit Grover's Quantum Search Algorithm. *arXiv* **2022**. [CrossRef]
53. Nishio, S.; Piparo, N.L.; Hanks, M.; Munro, W.J.; Nemoto, K. Reducing the Resources Needed to Implement Quantum Error Correction Codes Using Quantum Multiplexing. In Proceedings of the Conference on Lasers and Electro-Optics (CLEO), Sapporo, Japan, 31 August–5 September 2022. [CrossRef]
54. Karanov, B.; Chagnon, M.; Thouin, F.; Eriksson, T.A.; Bülow, H.; Lavery, D.; Bayvel, P.; Schmalen, L. End-to-End Deep Learning of Optical Fiber Communications. *J. Light. Technol.* **2018**, *36*, 4843–4855. [CrossRef]
55. Roumpos, I.; Marinis, L.D.; Kirtas, M.; Passalis, N.; Tefas, A.; Contestabile, G.; Pleros, N.; Moralis-Pegios, M.; Vyrsokinos, K. High-Performance End-to-End Deep Learning IM/DD Link Using Optics-Informed Neural Networks. *Opt. Express* **2023**, *31*, 43489–43507. [CrossRef]
56. Li, M.; Wang, S. End-to-End Learning for Chromatic Dispersion Compensation in Optical Fiber Communication. *IEEE Commun. Lett.* **2022**, *26*, 1852–1856. [CrossRef]
57. Xu, Y.; Guan, X.; Jiang, W.; Wang, X.; Hu, W.; Yi, L. Low-Complexity End-to-End Deep Learning Framework for 100G-PON. *IEEE/OSA J. Opt. Commun. Netw.* **2024**, *16*, 50–58. [CrossRef]
58. Candia, D.A.; Játiva, P.P.; Meza, C.A.; Sánchez, I.; Ijaz, M. Performance analysis of the particle swarm optimization algorithm in a VLC system for localization in hospital environments. *Appl. Sci.* **2024**, *14*, 2514. [CrossRef]
59. Jirón, I.; Soto, S.; Marín, S.; Acosta, M.; Soto, I. A new DNA-based model for finite field arithmetic. *Heliyon* **2019**, *5*, e02901. [CrossRef]
60. Zweck, J.; Menyuk, C.R. Validity of the Additive White Gaussian Noise Model for Quasi-Linear Long-Haul Return-to-Zero Optical Fiber Communications Systems. *J. Light. Technol.* **2009**, *27*, 3324–3335. [CrossRef]