 ISSN: **3107-6513**

Error Control Coding and Emerging Technologies for Reliable IoT Communication

Dr. S.V. Viraktamath [*1], *Sneha Mohan Kalkoti*[*2c], *Shrusti Kotyal* [*3], *Uddavva Atagal* [*4]

[1234]*Department of Electronics and Communication Engineering*

S D M College of Engineering and Technology

Dharwad, Karnataka State, India

[2c]snehakalkoti136@gmail.com

Abstract:

Error-Control Coding (ECC) plays a fundamental role in ensuring reliable, secure, and energy-efficient communication in large-scale IoT systems, where billions of low-power devices operate under noisy and rapidly changing channel conditions. Recent research highlights major progress in several areas, including advanced Turbo and partially coupled codes that deliver ultra-high reliability close to the Shannon capacity limit, and adaptive channel coding techniques that dynamically balance latency, throughput, and energy efficiency in 5G and emerging 6G networks.

In parallel, ECC is increasingly combined with security and intelligence at the system level. Hybrid schemes such as *Bose–Chaudhuri–Hocquenghem* BCH–Turbo codes enhance data robustness for mission-critical IoT applications, while integration with cryptographic authentication, blockchain-based access control, and TinyML-driven inference improves data integrity, resilience, and autonomous decision-making. Additionally, fault-tolerant and low-power hardware ECC architectures are being developed to meet the strict resource constraints of IoT edge devices. Together, these developments underscore the importance of cross-layer ECC integration as a key enabler for secure, scalable, and intelligent future 6G and edge-AI IoT ecosystems.

Keywords— *Internet of Things (IoT); Error Control Coding (L); Turbo Codes; Bose–Chaudhuri–Hocquenghem (BCH) Codes; Low-Density Parity-Check (LDPC) Codes; Adaptive Channel Coding; Bit Error Rate (BER); Fault-Tolerant Architectures; Blockchain-Based Access Control; Tiny Machine Learning (TinyML); Energy-Efficient Communication; 5G and 6G Networks Internet of Things (IoT), Bose-Chaudhuri-Hocquenghem (BCH), Artificial Intelligence (AI).*

I. INTRODUCTION

The rapid expansion of Internet of Things (IoT) deployments across domains such as healthcare, industrial automation, transportation, and smart-city infrastructure has intensified the demand for highly reliable data transmission under stringent energy and environmental constraints. Many IoT devices operate over noisy and time-varying wireless channels while relying on sub-milliwatt power budgets, rendering traditional retransmission-based reliability mechanisms inefficient or infeasible. Consequently, ECC has become a fundamental enabler of robust communication in resource-constrained IoT environments.

Although classical coding schemes—such as Hamming, Reed–Solomon, and Convolutional codes—provide essential error-correction capabilities, their fixed redundancy and limited adaptability are increasingly inadequate for the diverse and dynamic requirements associated with emerging 5G and 6G communication paradigms. Modern traffic categories, including enhanced Mobile Broadband (eMBB), ultra-reliable low-latency communication (URLLC), and massive Machine-Type Communication (mMTC), impose heterogeneous constraints on latency, throughput, and reliability, thereby necessitating adaptive and context-aware coding strategies.

Security considerations further compound these challenges, as IoT networks routinely handle sensitive and mission-critical data. Recent advances integrating ECC with lightweight cryptographic primitives and blockchain-based decentralized access-control mechanisms offer enhanced guarantees of data integrity, confidentiality, and traceability. Simultaneously, the emergence of Tiny Machine Learning (TinyML) enables microcontrollers to perform localized inference, facilitating context-aware error mitigation and anomaly detection directly at the network edge. This paper provides a comprehensive survey of contemporary ECC developments situated at the intersection of communication theory, embedded hardware design, and emerging decentralized intelligence.

II. FUNDAMENTALS OF ERROR CONTROL CODING

ECC is a key technique used in digital communication systems to improve transmission reliability by introducing structured redundancy into the transmitted data. During wireless transmission, signals are affected by channel impairments such as noise, interference, and multipath fading, which can lead to random and burst errors at the receiver. ECC enables the receiver to detect and correct these errors without requiring

retransmission, thereby reducing latency and conserving energy. This property is particularly important in Internet of Things (IoT) applications, where devices often operate with limited power resources and under strict delay constraints. The overall performance of an ECC scheme depends on several factors, including the code rate, decoding complexity, latency, and energy efficiency, all of which must be carefully optimized to suit the target application [1].

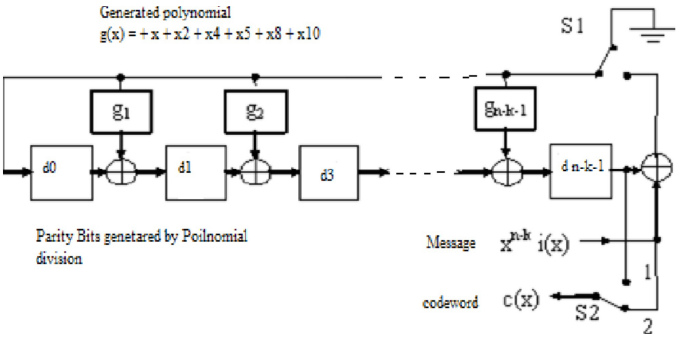

Fig. 1. BCH Encoder Block Diagram [2]

The BCH encoder introduces algebraic redundancy to enable multiple-bit error correction, making it suitable for short-packet IoT transmissions with strict reliability requirements as shown in fig1 [2]. Block codes form one of the fundamental classes of ECC techniques and operate by encoding fixed-length blocks of information bits into longer codewords using algebraic methods. The added redundancy allows the receiver to identify and correct errors based on the mathematical structure of the code. Among block codes, Bose–Chaudhuri–Hocquenghem (BCH) codes are widely adopted in IoT and embedded systems due to their ability to correct multiple random bit errors with relatively moderate computational complexity. Their deterministic correction capability and effectiveness for short data packets make BCH codes well suited for sensor data transmission, control signalling, and memory protection in IoT devices [2]. Convolutional codes differ from block codes in that they process continuous streams of data rather than fixed-size blocks. The encoding process uses shift registers and generator polynomials to produce parity bits that depend on both current and previous input bits, providing robust protection for streaming data. Decoding is typically performed using the Viterbi algorithm, which searches the trellis representation of the code to determine the most likely transmitted sequence. This optimal maximum-likelihood decoding approach, combined with low decoding latency, makes convolutional codes particularly suitable for real-time telemetry and control applications in IoT networks [3].

Turbo codes represent a major breakthrough in coding theory by demonstrating error-correction performance close to the theoretical Shannon capacity limit. This is achieved through the parallel concatenation of convolutional encoders and iterative decoding, where soft information is exchanged between component decoders to progressively improve bit reliability estimates. Building on these advances, LDPC codes further enhance performance using sparse parity-check matrices and iterative belief-propagation decoding, enabling highly parallel and energy-efficient implementations. Polar codes, based on the principle of channel polarization, are capacity-achieving codes and have been adopted in 5G New Radio control channels due to their strong theoretical foundation and reliable performance in modern wireless systems [4].

LDPC codes further enhance error-correction performance by employing sparse parity-check matrices combined with iterative belief-propagation decoding. The defining characteristic of LDPC codes is the sparsity of their parity-check matrices, which significantly reduces decoding complexity and enables highly parallel processing. This parallelism makes LDPC codes particularly attractive for hardware implementations that require high throughput and low energy consumption, such as base stations, gateways, and advanced IoT edge devices. Moreover, LDPC codes exhibit strong error-correction capability across a wide range of block lengths and code rates, allowing them to be flexibly adapted to diverse channel conditions and application requirements. As a result, LDPC codes have been widely adopted in modern communication standards, including Wi-Fi, satellite communications, and 4G/5G wireless systems [5].

Polar codes represent another major advancement in coding theory through the concept of channel polarization, which transforms a set of identical communication channels into a combination of highly reliable and highly unreliable subchannels. By transmitting information bits only over the reliable subchannels and assigning fixed values to the unreliable ones, Polar codes can theoretically achieve channel capacity with provable optimality. Their structured construction enables efficient encoding and decoding using successive cancellation and its enhanced variants, making them suitable for practical implementation. Owing to their strong theoretical foundation, low encoding complexity, and reliable performance for short and moderate block lengths, Polar codes have been adopted in 5G New Radio control channels. This adoption underscores their growing importance in next-generation wireless and IoT communication systems that demand high reliability, low latency, and efficient use of spectral resources [6].

III. TURBO CODES FOR RELIABLE IOT COMMUNICATION

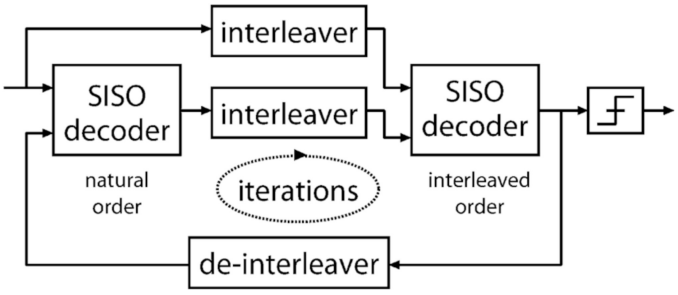

Fig. 2. Turbo Encoder and Iterative Decoder Architecture [7]

Turbo codes achieve significant coding gains at low signal-to-noise ratios, which is critical for energy-constrained IoT devices operating in noisy wireless environments as shown in fig.2. [7]. Turbo codes are widely recognized for their exceptional error-correction capability, particularly in low signal-to-noise ratio (SNR) conditions where conventional coding schemes tend to fail. Their ability to operate reliably

near the Shannon capacity limit makes them especially attractive for IoT environments, which are often characterized by noisy channels, limited transmission power, and intermittent connectivity. IoT devices deployed in remote or harsh environments benefit significantly from coding schemes that can maintain reliable communication without frequent retransmissions. The robustness of Turbo codes under such adverse channel conditions has made them a preferred choice for reliability-critical wireless communication systems [7].

A typical turbo encoder is constructed using two recursive systematic convolutional encoders connected in parallel and separated by an interleaver. The interleaver plays a crucial role by rearranging the input bit sequence in a pseudo-random manner, thereby spreading burst errors over multiple code blocks.

$$x(t) = [u(t), p_1(t), p_2(t)], \qquad (1)$$

where $p_1(t)$and $p_2(t)$arise from the direct and interleaved branches. At the receiver, two soft-input soft-output decoders iteratively exchange extrinsic information using BCJR, Log-MAP, or Max-Log-MAP algorithms until convergence or a set iteration limit.

This randomization improves the effectiveness of iterative decoding at the receiver and enhances overall error-correction performance. The parallel concatenation structure enables Turbo codes to exploit both time diversity and coding gain, which is particularly beneficial for short-packet transmissions common in IoT applications.

At the receiver, Turbo decoding is performed using an iterative process based on soft-input soft-output (SISO) algorithms. Instead of making hard decisions on received bits, the decoder processes probabilistic information in the form of likelihood values. Algorithms such as BCJR, Log-MAP, and Max-Log-MAP iteratively exchange extrinsic information between component decoders, refining bit reliability estimates with each iteration. This iterative exchange gradually reduces uncertainty in the decoded sequence and leads to significant improvements in bit error rate (BER) performance, especially in low-SNR regimes [8].

Recent research efforts have focused on enhancing the practical performance of Turbo codes by addressing inherent limitations such as error floors, decoding complexity, and reduced efficiency for short block lengths. In conventional Turbo codes, the presence of low-weight codewords and suboptimal interleaver designs can lead to performance saturation at high signal-to-noise ratio (SNR) values, resulting in an error floor that limits achievable reliability. This issue is particularly critical in Internet of Things (IoT) applications, where high reliability is required even for short packets transmitted under stringent power and latency constraints.

To overcome these challenges, turbo-like and partially coupled turbo codes have been proposed as advanced coding structures that improve the minimum distance properties of Turbo codes. By introducing spatial or partial coupling between component encoders, these schemes enhance the exchange of extrinsic information during iterative decoding and promote faster convergence toward correct decisions. As a result, the probability of residual errors at high SNR values is significantly reduced, and decoding robustness is improved for short block lengths commonly used in IoT communication. These enhancements make advanced Turbo coding schemes well suited for reliable, low-latency, and energy-efficient IoT systems [9].

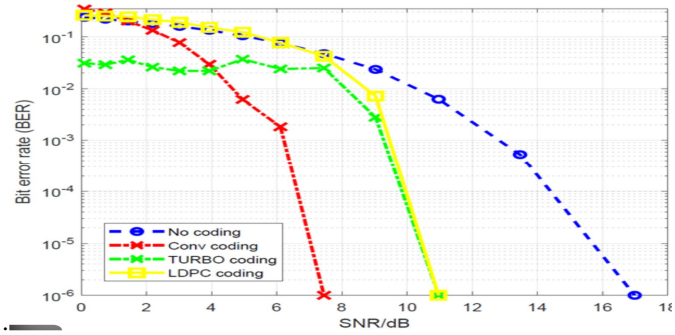

Fig. 3. BER versus SNR Performance Comparison of BCH, Turbo, and LDPC CODES [9].

Turbo and LDPC codes significantly outperform classical BCH codes in the low-SNR regime, demonstrating near-capacity performance through iterative decoding as shown in fig. 3 [9]. In addition to structural improvements, hybrid coding schemes that combine Turbo codes with algebraic codes such as BCH have gained considerable attention. Hybrid BCH–Turbo coding schemes leverage the strong burst-error correction capability of Turbo codes along with the deterministic and guaranteed correction capability of BCH codes. This combination significantly reduces the number of retransmissions, improves fault tolerance, and enhances overall system reliability. Such hybrid approaches are particularly beneficial for mission-critical IoT applications, including industrial automation, healthcare monitoring, and smart infrastructure, where data integrity and reliability are of paramount importance [10].

IV. ADAPTIVE CHANNEL CODING FOR 5G AND IOT NETWORKS

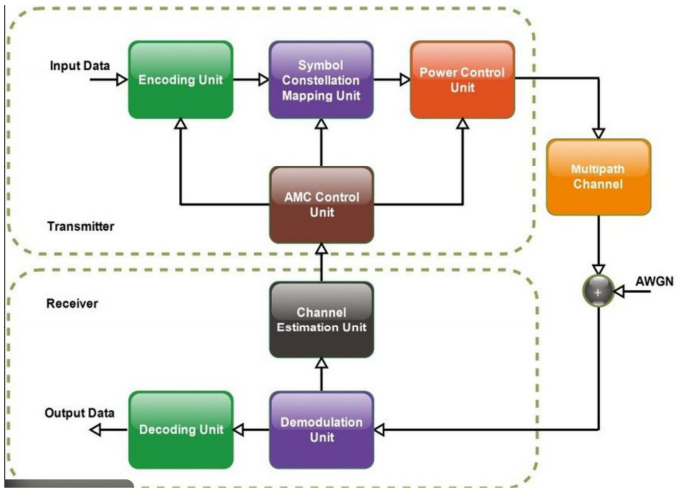

Fig. 4. Adaptive Channel Coding Loop Using CQI Feedback [11]

Adaptive Channel Coding dynamically adjusts the coding rate based on channel quality indicators to balance reliability and energy efficiency as shown in fig. 4. [11]. ACC enables communication systems to dynamically adjust coding

parameters in response to variations in wireless channel conditions. In IoT environments, channel quality can fluctuate significantly due to factors such as node mobility, interference from neighbouring devices, and multipath fading. These variations directly affect link reliability and can lead to increased packet errors if fixed-rate coding schemes are used. ACC enhances system robustness by adapting the level of redundancy based on real-time channel assessments, thereby maintaining reliable communication while reducing unnecessary retransmissions. This adaptive behaviour is particularly beneficial for energy-constrained IoT devices, as it balances reliability with power efficiency and helps sustain acceptable quality of service over time. Channel quality metrics such as signal-to-noise ratio (SNR) estimates or Channel Quality Indicator (CQI) feedback from the receiver are commonly used to guide these adaptive decisions [11].

Early ACC schemes primarily relied on punctured convolutional codes to provide variable code rates by selectively removing parity bits from a low-rate mother code. Although this method enabled basic rate adaptation, its coarse granularity and limited flexibility made it less effective in rapidly changing channel conditions. To overcome these shortcomings, modern communication systems have adopted rate-compatible Turbo and LDPC codes, which allow fine-grained adjustment of redundancy while preserving a common encoder–decoder structure. This seamless rate adaptation simplifies implementation, reduces signalling overhead, and enables efficient operation across a wide range of channel scenarios encountered in contemporary wireless and IoT networks [12]. Early ACC strategies employed punctured convolutional codes to realize multiple effective rates (e.g., 1/2, 2/3, 3/4). Modern 5G systems utilize rate-compatible Turbo and LDPC code families (RCPT and RC-LDPC), allowing smooth adjustment of redundancy without modifying decoder structure. The achievable throughput is commonly expressed as

$$T = R_c \log_2(1 + \text{SNR}), \qquad (2)$$

where R_c denotes the coding rate. Maximizing T while satisfying BER and latency requirements defines the adaptation objective. Hybrid coding designs have been proposed to address the short-packet and burst-mode characteristics of many IoT applications. Polar–Convolutional concatenated codes, for example, have demonstrated up to 10 dB coding gain at BER 10^{-5} in UFMC-based 5G systems, offering substantial robustness compared with standalone Polar code.

ACM extends conventional channel adaptation techniques by jointly optimizing both the modulation order and the channel coding rate according to instantaneous channel conditions. Unlike schemes that adapt only the coding rate, ACM exploits the combined degrees of freedom offered by modulation and coding to improve spectral efficiency and link reliability. In favourable channel conditions, the system selects higher-order modulation schemes together with weaker error-correction coding to maximize throughput and efficiently utilize available bandwidth. Conversely, under poor channel conditions caused by fading, interference, or mobility, lower-order modulation combined with stronger coding is employed to ensure reliable data transmission and maintain acceptable error performance. This joint optimization enables the system to satisfy predefined bit error rate and latency constraints while adapting smoothly to rapidly changing wireless environments, making ACM particularly effective in time-varying and frequency-selective channels commonly encountered in IoT and mobile communication systems [13].

In recent years, machine-learning-based techniques have emerged as a powerful alternative to traditional rule-based adaptation strategies. Conventional ACC schemes typically rely on predefined thresholds and analytical models that may not accurately capture the complexity and uncertainty of real-world wireless channels. Machine-learning approaches, particularly reinforcement learning, allow the communication system to autonomously learn optimal adaptation policies by continuously interacting with the environment. By observing historical channel conditions, decoding success rates, traffic characteristics, and energy consumption patterns, learning agents can make informed coding decisions that optimize long-term system performance. These data-driven techniques are capable of handling non-linear channel behaviour and unforeseen operating conditions more effectively than static adaptation rules. As a result, machine-learning-based ACC methods improve adaptation accuracy, enhance reliability, and reduce energy consumption, making them well suited for large-scale, autonomous, and energy-constrained IoT networks [14].

V. SECURITY, ADAPTIVITY, AND SCALABILITY I N IOT COMMUNICATION

Although ECC significantly improves communication reliability by mitigating channel-induced errors, it does not inherently address security threats that arise at the protocol or application layers. IoT systems typically operate over open and shared wireless media, where transmitted data can be easily intercepted or manipulated by malicious entities. As IoT devices are often deployed in unattended or hostile environments, they are especially vulnerable to attacks such as spoofing, replay attacks, impersonation, and unauthorized access. These security challenges are further aggravated by the limited computational capability, memory, and battery life of IoT nodes, which restrict the use of conventional heavyweight cryptographic algorithms. Consequently, relying solely on encryption-based security mechanisms may not be feasible for many IoT applications. To address this gap, researchers have emphasized the integration of ECC with lightweight authentication and integrity verification techniques that can offer basic security assurances while maintaining low complexity and energy consumption. Such integrated approaches allow IoT systems to achieve simultaneous protection against both transmission errors and malicious data manipulation, thereby enhancing the overall trustworthiness and resilience of IoT communication frameworks [15].

Blockchain-based access control frameworks have emerged as a promising approach to further strengthen security and trust in large-scale IoT deployments. Traditional centralized access control systems suffer from single points of failure, limited scalability, and vulnerability to targeted attacks. In contrast, blockchain employs a decentralized ledger architecture that distributes trust across multiple participating nodes, eliminating dependence on a central authority. The immutability of

blockchain records ensures that access transactions cannot be altered or repudiated, providing transparent and verifiable audit trails. Additionally, distributed identity management and smart contracts enable automated enforcement of access policies, allowing only authenticated and authorized entities to interact with IoT devices and services. When combined with ECC-protected data transmission, blockchain-based access control enhances both communication reliability and security, resulting in a robust end-to-end framework suitable for heterogeneous, distributed, and large-scale IoT environments [16].

Beyond physical and link layers, secure and scalable communication increasingly relies on decentralized trust mechanisms. Blockchain-based access control, discussed further in Section VII, complements ECC by providing immutable auditability and distributed identity management across edge, fog, and cloud layers. Together, these approaches form a unified security–reliability framework suitable for next-generation IoT ecosystems. In addition to security challenges, scalability remains a critical concern in dense IoT deployments that involve hundreds or even thousands of devices sharing limited wireless resources. As the number of connected nodes increases, the wireless medium becomes highly congested, leading to increased contention for channel access, frequent packet collisions, and elevated levels of interference. These effects can significantly degrade network throughput, increase latency, and reduce overall reliability, particularly in large-scale IoT applications such as smart cities, industrial automation, and environmental monitoring. Without appropriate adaptation mechanisms, fixed-rate coding and static access strategies are often unable to cope with such dynamic and crowded network conditions.

To address these scalability challenges, cross-layer optimization approaches have gained considerable attention in recent research. These approaches coordinate ECC at the physical layer with Medium Access Control (MAC) layer protocols, enabling joint adaptation to both channel conditions and network load. By dynamically adjusting coding redundancy in response to factors such as traffic intensity, collision probability, and contention levels, cross-layer designs can significantly improve packet delivery success rates and mitigate throughput degradation in congested scenarios. Such coordinated strategies highlight the importance of jointly considering reliability, security, and scalability to ensure the sustainable and efficient operation of large-scale IoT networks [17].

VI. FAULT-TOLERANT AND LOW-POWER ECC ARCHITECTURES

Fault-tolerant ECC hardware is essential for maintaining reliability in IoT deployments, where microcontrollers and memories are frequently exposed to radiation-induced upsets and transient faults. Hardware-level multi-bit error correction has proven effective: Mahadevaswamy et al. Demonstrated a BCH-coded 32-bit ALU on a Spartan-3 FPGA using a shortened (63, 36) BCH code capable of correcting up to five-bit errors. The design incurred roughly 70% hardware overhead—substantially lower than the >200% overhead associated with Triple Modular Redundancy—while achieving complete recovery from injected bit-flip faults, confirming BCH as a lightweight protection mechanism for embedded systems. In advanced memory technologies, where CMOS–nanodevice hybrids exhibit high defect rates, ECC also plays a key role in defect management. Kavitha et al. Introduced a group-based BCH protection scheme in which multiple memory regions share Galois-field hardware and apply codes of differing strength. Coupled with adaptive logical-to-physical remapping, this approach mitigates permanent and transient faults while preserving storage capacity, illustrating how ECC can function as an active fault-management tool rather than solely as a data-integrity safeguard.

IoT devices are increasingly deployed in harsh and unpredictable environments such as industrial plants, space systems, outdoor sensing fields, and high-radiation zones. In such conditions, hardware components are exposed to factors including radiation, temperature fluctuations, and aging effects, which can induce transient faults such as single-event upsets and permanent hardware failures. These faults often manifest as bit flips in memory elements, registers, or processing units, leading to corrupted data and unreliable system behaviour. Ensuring fault tolerance under these constraints is critical, especially for safety-critical and mission-critical IoT applications where data integrity and continuous operation are essential.

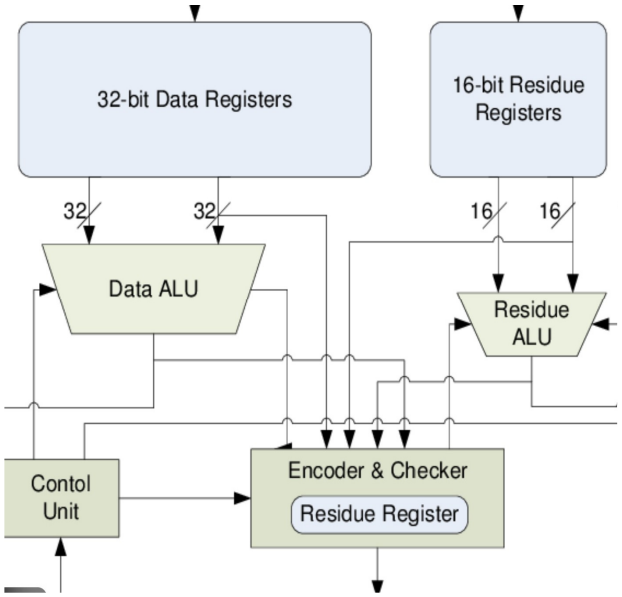

Fig.5. BCH-Based Fault-Tolerant ALU Architecture [18]

BCH codes provide effective protection against transient and permanent hardware faults with significantly lower overhead than redundancy-based techniques as shown in fig.5 [18].

To address these challenges, BCH-based fault-tolerant architectures have been widely adopted due to their ability to correct multiple random bit errors with relatively low hardware complexity. Unlike redundancy-based approaches such as Triple Modular Redundancy (TMR), which replicate hardware modules and rely on majority voting, BCH-based solutions provide error correction through coding techniques that introduce minimal additional logic and power overhead. This makes them particularly suitable for resource-constrained IoT devices, where silicon area, power consumption, and cost are critical design considerations. By offering strong error-correction capability without excessive duplication of hardware

 ISSN: 3107-6513

resources, BCH-based architectures provide an efficient and scalable approach to fault tolerance in harsh operating environments [18]. In parallel with fault tolerance, energy efficiency remains a primary design objective for ECC implementations in IoT systems. Decoding operations can be computationally intensive, especially for advanced coding schemes, and may significantly impact battery life if not carefully optimized. As a result, energy-efficient ECC architectures focus on reducing decoding complexity through techniques such as clock gating, which disables inactive circuit blocks, approximate arithmetic that lowers computational precision without significantly affecting decoding accuracy, and early termination strategies that halt decoding iterations once convergence is achieved.

Low-power Turbo decoder designs exemplify these optimization strategies by combining algorithmic simplifications with hardware-level power-saving techniques. Such designs significantly reduce energy consumption while preserving high decoding throughput and acceptable error-rate performance. These characteristics make low-power Turbo decoders well suited for Narrowband IoT (NB-IoT) and wireless sensor network applications, where long device lifetime and reliable communication are paramount. The demonstrated efficiency of these architectures highlights the importance of co-designing ECC algorithms and hardware to meet the stringent power and performance requirements of modern IoT systems [19].

Ultimately, IoT system design must negotiate the trade-off between correction capability and energy consumption. Techniques such as adaptive voltage scaling and early-termination decoding can reduce energy usage by up to 25% under favorable channel conditions. Furthermore, emerging error-resilient architectures tolerate residual BER on the order of $10^{-6}$by relying on application-level masking and graceful degradation, thereby extending device lifetime without compromising functional reliability.

VII. EMERGING TECHNOLOGIES FOR INTELLIGENT IOT RELIABILITY

TinyML represents an important advancement in embedded intelligence by enabling neural network inference directly on ultra-low-power microcontrollers with strict constraints on memory, computation, and energy consumption. Unlike conventional machine-learning approaches that rely on cloud or edge servers, TinyML allows intelligent decision-making to occur locally at the device level. This on-device processing significantly reduces communication latency, minimizes bandwidth usage, and enhances data privacy—factors that are critical for large-scale IoT systems operating in remote, bandwidth-limited, or delay-sensitive environments. In wireless communication scenarios, TinyML models can be trained to analyse locally observed signal characteristics such as received signal strength, noise variance, packet error rates, and historical decoding outcomes. By learning temporal and spatial patterns in these parameters, TinyML-enabled devices can accurately predict channel quality variations and proactively adapt ECC parameters, such as coding rate, decoding iterations, or redundancy level. This real-time, autonomous adaptation improves communication reliability and energy efficiency without continuous reliance on cloud connectivity, making TinyML particularly suitable for self-sustaining and scalable IoT deployments [20].

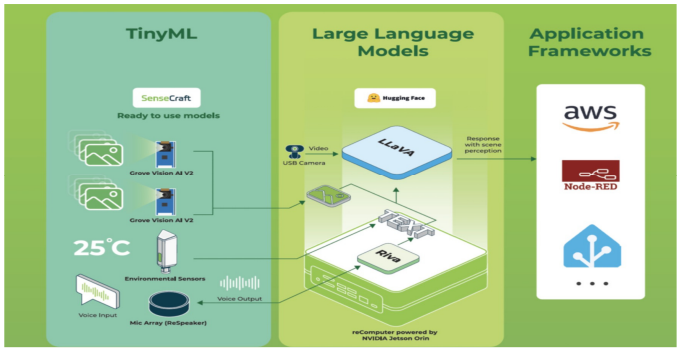

Fig. 6. TinyML-Based On-Device Intelligence for Adaptive ECC [20]

TinyML enables real-time channel prediction and ECC adaptation on ultra-low-power microcontrollers, reducing latency and energy consumption as shown in fig.6 [20]. Soft-decision decoding techniques further enhance ECC performance by exploiting the probabilistic information inherently present in received signals. In practical wireless channels, received symbols carry not only binary decisions but also reliability information that reflects confidence levels associated with each bit. Hard-decision decoding discards this valuable information by converting received signals into binary values before decoding, which can lead to suboptimal performance. In contrast, soft-decision decoding incorporates likelihood or log-likelihood values into the decoding process, enabling more accurate error correction. Algorithms such as soft-bit Viterbi decoding and Maximum a Posteriori (MAP) decoding evaluate multiple candidate paths or codewords based on probabilistic metrics, allowing the decoder to make informed decisions even under severe noise and fading conditions. Although soft-decision decoding introduces higher computational complexity, advances in algorithmic optimization and low-power hardware design have made these techniques feasible for resource-constrained IoT devices. The resulting signal-to-noise ratio gains translate into improved reliability or reduced transmission power requirements, making soft-decision decoding an effective and practical solution for modern IoT communication systems seamlessly orchestrating these diverse layers without excessive overhead. Future IoT chipsets will likely feature built-in ECC engines, hardware neural accelerators, and lightweight cryptographic cores integrated into a unified SoC.

CONCLUSION

ECC has evolved well beyond its traditional role as a physical-layer error mitigation technique and has emerged as a critical cross-layer component in modern Internet of Things (IoT) communication systems. As IoT networks continue to scale in size, heterogeneity, and application diversity, the limitations of fixed-rate and isolated reliability mechanisms have become increasingly evident. Advanced coding schemes such as Turbo, LDPC, and Polar codes provide near-capacity performance and robust error correction under challenging channel conditions,

forming the foundation for reliable low-power wireless communication.

The integration of adaptive channel coding and adaptive coded modulation enables IoT systems to dynamically respond to time-varying channel conditions and network congestion, achieving an optimal balance between reliability, spectral efficiency, and energy consumption. Fault-tolerant ECC architectures, particularly those based on algebraic codes such as BCH, further enhance system robustness by protecting IoT devices against hardware-induced errors and environmental disturbances without incurring excessive overhead. At the same time, cross-layer optimization approaches that coordinate ECC with medium access control and higher-layer protocols address scalability challenges in dense IoT deployments.

Emerging technologies are further transforming the role of ECC in next-generation IoT networks. Blockchain-based access control frameworks complement ECC by introducing decentralized trust, secure identity management, and immutable audit trails, thereby strengthening end-to-end security. Similarly, on-device intelligence that supports real-time prediction of channel conditions and autonomous adaptation of ECC parameters, reducing latency and dependence on cloud resources. Together, these advancements highlight a paradigm shift toward intelligent, adaptive, and secure communication architectures. Looking ahead, continued research in AI-assisted ECC design, hardware–software co-optimization, and energy-aware implementations will be essential to meet the stringent requirements of future IoT systems. As IoT applications expand into mission-critical domains such as smart healthcare, industrial automation, and autonomous infrastructure, the convergence of advanced coding theory, machine intelligence, and system-level optimization will play a pivotal role in realizing scalable, sustainable, and resilient next-generation IoT networks.

REFERENCES

[1] R. S. Ahirwal and A. Marmat, “A Partially Coupled Turbo Code Design for Error Detection and Correction in IoT Networks,” *Int. J. Adv. Eng. Res. Sci.*, 2024.

[2] M. J. Adamu *et al.*, “An Efficient Turbo Decoding and Frequency-Domain Equalization for LTE-Based NB-IoT Systems,” *Sensors*, 2021.

[3] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes,” *Proc. IEEE ICC*, 1993.

[4] F. Qureshi and S. Bhalla, “A Review on Turbo Codes for Error Detection and Correction in Wireless Networks,” *IJAEM*, 2023.

[5] R. Udaypurey *et al.*, “IoT Reliability Enhancement using Turbo Code Error Control,” *IJSART*, 2023.

[6] S. Zhao *et al.*, “Turbo-Like Codes for Ultra-Reliable Short Packet Communications,” *IEEE Trans. Ind. Inform.*, 2022.

[7] L. Srividya and P. Sudha, “Adaptive Channel Coding to Enhance Performance in Rayleigh Channel,” *IJACSA*, 2024.

[8] S. Prajapati *et al.*, “Parametric Analysis of UFMC with 5G NR Polar and Convolutional Codes,” *J. Eng. Technol. Ind. Appl.*, 2025.

[9] A. J. Goldsmith and S.-G. Chua, “Adaptive Coded Modulation for Fading Channels,” *IEEE Trans. Commun.*, 1998.

[10] M. J. Adamu *et al.*, “Simplified Frequency-Domain Turbo Equalization for NB-IoT,” *Sensors*, 2021.

[11] A. Mohammad *et al.*, “Hybrid Token and Biometric-Based Authentication for IoT Security,” *Computers*, 2022.

[12] S. Tripathi and S. De, “Channel Adaptive Protocols for Low Power IoT Devices,” *IEEE Internet Things J.*, 2020.

[13] K. Chounos *et al.*, “Scalability Analysis of Wi-Fi HaLow in Dense IoT Networks,” *arXivB preprint*, 2025.

[14] M. V. P. Mahadevaswamy *et al.*, “BCH-Based Fault Tolerant ALU Design Using FPGA,” *IJSCE*, 201s2.

[15] D. Kavitha *et al.*, “Error Tolerant Group Address Mapping for Hybrid CMOS–Nano Memories,” *IJERA*, 2014.

[16] G. Divya Bharathi and S. Surendra, “Low Power Turbo Decoder Design for Wireless Sensor Networks,” *IJSDR*, 2016.

[17] C. Zhang *et al.*, “Design of Low-Power Turbo Encoder and Decoder for NB-IoT,” *Chinese J. Electron.*, 2024.

[18] A. Punia *et al.*, “A Systematic Review on Blockchain-Based Access Control Systems in Cloud Environment,” *J. Cloud Comput.*, 2024.

[19] T. Malche *et al.*, “A TinyML Approach to Real-Time Snoring Detection in Resource-Constrained Wearable Devices,” *Eng. Proc.*, 2024.

[20] I. H. Ali, “Implement Wireless Transceiver System Based on Convolutional Coding; Aided by Soft-Bit Decoding,” *IJCCCE*, 2018.

[21] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell Syst. Tech. J.*, 1948.

[22] D. J. C. MacKay, “Good Error-Correcting Codes Based on Very Sparse Matrices,” *IEEE Trans. Inf. Theory*, 1999.

[23] E. Arıkan, “Channel Polarization: A Method for Constructing Capacity-Achieving Codes,” *IEEE Trans. Inf. Theory*, 2009.

[24] F. Kschischang and B. Frey, “Factor Graphs and the Sum–Product Algorithm,” *IEEE Trans. Inf. Theory*, 2001.