entropy

MDPI

Article

LERMS: A Low-Latency and Reliable Downlink Packet-Level Encoding Transmission Method in Untrusted 5GA Edge Network

Zhongfu Guo [1,*], Xinsheng Ji [2,3,*], Wei You [1], Mingyan Xu [1], Yu Zhao [1], Zhimo Cheng [1], Deqiang Zhou [1] and Lingwei Wang [1]

[1] Department of Next-Generation Mobile Communication and Cyber Space Security, Information Engineering University, Zhengzhou 450001, China
[2] National Digital Switching System Engineering and Technological Research and Development Center, Zhengzhou 450000, China
[3] Purple Mountain Laboratories: Networking, Communications and Security, Nanjing 211111, China
* Correspondence: ndscgzf@163.com (Z.G.); ndscjxs@126.com (X.J.)

Abstract: The increasing demand for end-to-end low-latency and high-reliability transmissions between edge computing nodes and user elements in 5G Advance edge networks has brought new challenges to the transmission of data. In response, this paper proposes LERMS, a packet-level encoding transmission scheme designed for untrusted 5GA edge networks that may encounter malicious transmission situations such as data tampering, discarding, and eavesdropping. LERMS achieves resiliency against such attacks by using 5GA Protocol data unit (PDU) coded Concurrent Multipath Transfer (CMT) based on Lagrangian interpolation and Raptor's two-layer coding, which provides redundancy to eliminate the impact of an attacker's malicious behavior. To mitigate the increased queuing delay resulting from encoding in data blocks, LERMS is queue-aware with variable block length. Its strategy is modeled as a Markov chain and optimized using a matrix method. Numerical results demonstrate that LERMS achieves the optimal trade-off between delay and reliability while providing resiliency against untrusted edge networks.

Keywords: 5G-A core network; robust concurrent multipath transfer; interface diversity edge network; raptor codes; security

Citation: Guo, Z.; Ji, X.; You, W.; Xu, M.; Zhao, Y.; Cheng, Z.; Zhou, D.; Wang, L. LERMS: A Low-Latency and Reliable Downlink Packet-Level Encoding Transmission Method in Untrusted 5GA Edge Network. *Entropy* **2023**, *25*, 966. https://doi.org/10.3390/e25070966

Academic Editor: T. Aaron Gulliver

Received: 4 May 2023
Revised: 11 June 2023
Accepted: 20 June 2023
Published: 21 June 2023

1. Introduction

In the era of 5G [1], edge computing provides customized computing services for data-intensive [2] and time-sensitive [3] applications such as healthcare [4] and traffic management [5]; edge networks (ENs) [6] are designed to address the challenges of centralized cloud computing and unreliable communications [7,8]. 5G-Advanced networks [9] are expected to offer more powerful capabilities, including wide coverage [10], low latency [11], and highly reliable transmission [12].

The EN transmits important and sensitive data related to safety [13], but it is closer to users and vulnerable to wireless eavesdropping [14], network-layer attacks [15], and tampering [16]. The edge network often finds itself in a predicament of weak protection capabilities and urgent security needs.

Previous studies have focused on addressing these pressing demands [17], and researchers have conducted extensive studies on point-to-point (UE-gNB) security at both the physical layer [18] and MAC-layer [19]. Providing secure end-to-end (UE-DN) transmission in an untrusted network incurs additional performance and resource overheads [20]. Commonly used confidentiality protection schemes include 5G voice communication using IPSec encryption [21], control plane TLS authentication [22], and secret-sharing proposed

by Shamir et al. [23]. In the secret-sharing scheme, each share is delivered by a different courier to the recipient, providing an information-theoretic provable confidentiality transmission scheme.

The enhanced core network (ECN) proposed by 3GPP in ([1], clause 5.33), as shown in Figure 1, could establish multiple PDU sessions (MP), which can provide end-to-end reliable communication by redundancy transmission. The multiplex transport protocol stack is deployed on both the data network (DN) and user elements (UEs). Additionally, the high-layer (above the IP layer [1]) splits and aggregates the service flows. However, implementing concurrent multipath transmission (CMT) in an untrusted MPEN faces many challenges, including increased attack surface due to MPEN [17], additional delays due to asynchronous delay and bandwidth [24,25], making it not feasible to interact with the PDU bearer from the perspective of the 5G session mechanism [26].

In this paper, we aim to design a low-latency, high-reliability transmission scheme suitable for untrusted edge networks. Randomly arriving source data packets are buffered and queued at the sending end before transmission. This article continues previous work [27] and uses Raptor coding for efficient CMT. We have introduced a Lagrangian interpolation coding [28] scheme to realize CMT based on secret sharing. Different coding parameters can be adjusted for different resilience levels, which is suitable for an untrusted network environment, as the concatenated encoder provides reliability and robustness against malicious behavior. Encoding is performed at the granularity of data packets. Longer encoding block length can ensure reliability, but the corresponding queuing delay will increase [29]. The length of encoding blocks must be carefully considered to balance delay and reliability. We use a Markov decision process to formulate the optimization problem and construct it as a matrix-based linear programming to obtain the optimal variable-length coding strategy. The contributions of this paper are as follows:

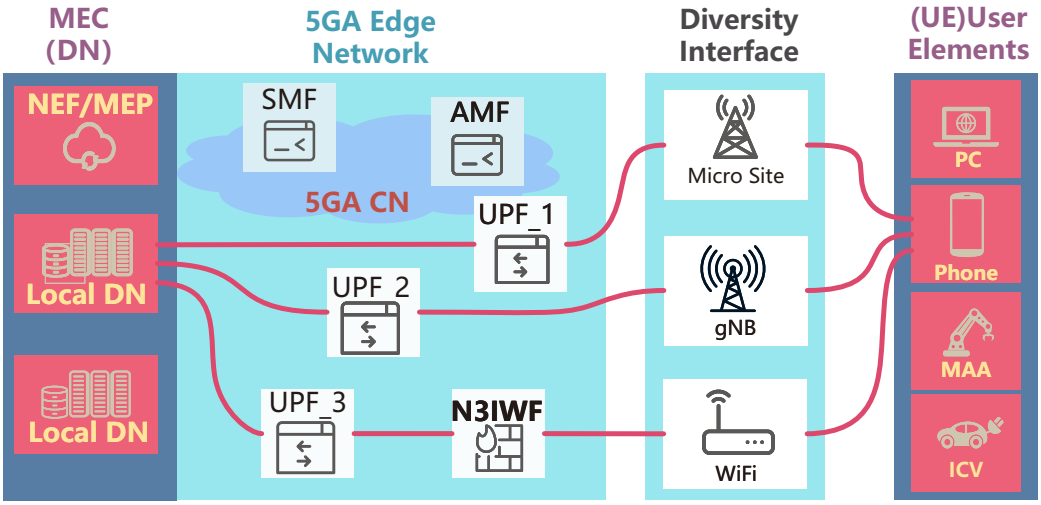


Figure 1. End-to-end redundant transmission in edge networks: A schematic illustration of MPEN utilizing redundant PDU sessions in an edge network with diversity interface for concurrent multipath transmission.

- Proposes LERMS (LERMS is short for (Lagrangian–polynomial and Raptor Encoder concurrent Multiple-pdu-Sessions transmission)), which is the first concatenated encoder scheme that is suitable for untrusted edge network environments. LERMS provides secure, reliable downlink transmission capabilities in the face of an untrusted network, such as transmission failures, data theft, and malicious tampering.
- A queue-aware variable block length encoding scheme is designed and optimized using a matrix-based approach to minimize queuing delay while ensuring reliability.
- Proposes a multi-service flow aggregation transmission scheme that reduces the probability of data packet random idle filling, ensuring security when the flow is small, and improves the transmission efficiency of the edge network.

This paper is organized as follows: Section 2 presents the design of the secure transmission method and flow aggregation transmission scheme as preliminary work for 5GA LERMS. Section 3 describes the system model. In Section 4, the trade-off between delay and reliability for the LERMS strategy is demonstrated. Section 5 presents numerical results. This paper concludes in Section 6, with the future research directions proposed. Table 1 summarizes the important variables used throughout the paper.

Table 1. Basic notations.

Symbol	Definition
Ps, Pc	The source packets, coded packets.
$\mathbb{P}$,$\mathbb{E}$	The probability, expectation value.
$\mathbb{F}$,$\mathbb{N}$	The finite field, the set of natural numbers.
$\vec{\sum}$	The data collection operator for receiver.
T	The time span of timeslot.
$\Omega(x)$	The degree distribution.
g	The block-length of codes.
$g[t]$,$\gamma_v^j[t]$	The encoding action within t-th timeslot.
d^{th}	The constraints of delay.
B	The upper bounds of $g[t]$.
Z	The transmitting side queue buffer size.
ϕ	The size of each data packet.
$\varepsilon_v^j = [\varepsilon_1, \ldots, \varepsilon_j]$	The erasure probability of j sessions.
$\Lambda_l[t]$, $\boldsymbol{\lambda} = [\lambda_l^0, \lambda_l^1, \ldots, \lambda_l^N]$,	The number Ps arrivals within t-th timeslot for user l. The probability distribution of $\Lambda_l[t] = n$.
N_L	The number of disjoint PDU sessions established in MPEN.

2. 5GA LERMS Preliminary

We consider the business scenario in the edge network, as illustrated in Figure 1. The UE subscribes to the edge computing service, and the edge network that transmits is responsible for transmitting the downlink service flow to the UE. The downlink service flow requires low delay and high reliability in transmission. With abundant transmission resources, the 5GA network allows for the establishment of multiple PDU sessions to carry the downlink traffic. To ensure the secure transmission of the downlink traffic in an untrusted network environment, this section begins by analyzing the key performance indicators in the edge network environment. We then propose a multiplex transmission scheme based on the concept of secret sharing, which provides feasibility for secure transmission. Finally, we discuss a joint coding scheme for downlink traffic in the core network.

2.1. 5GA Edge Network Requirements Analysis and Challenges

As an emerging network architecture that provides high-performance communication between terminals (UE) and edge computing nodes (DN), the 5G Advance (5GA) edge network [6] aims to meet key requirements such as low latency, high reliability, and security in the 5GA mobile network. In this section, we will analyze these requirements in detail and discuss the challenges of implementing them in an untrusted network environment.

Latency (L). In classical transmission systems, the overall transmission latency comprises propagation latency (t_p), signifying the physical delay of signals within the transmission medium, and transmission latency (t_c), determined by $t_c = W/\phi$, where W represents the bandwidth and ϕ is the transmitted data volume. However, factors such as congestion and bit errors prevent the total latency from being a simple sum of t_p and t_c, causing trans-

mission latency fluctuations within a specific range [30,31]. In coded transmission systems, end-to-end transmission delay must be reconsidered. Data are coded and transmitted in block units. Current application layer data enter the cache and queue up, waiting for the previous generation of coded data blocks to finish transmission [32]. Consequently, the end-to-end transmission delay is defined as $\overleftrightarrow{T} = T + t_{en} + t_{de} + t_q$, encompassing codec delay and queuing delay. This paper's low latency discussion primarily centers on t_q and t_c.

Reliability (R). In this paper, we consider each PDU session as a packet erasure channel, denoted by $\varepsilon_v^j = [\varepsilon_1, \ldots, \varepsilon_j]$ to represent the erasure probability of each PDU session [27]. High reliability refers to the transmission robustness against higher erasure probabilities. In other words, when a generation of source data packets departs from the sender within a specified time, and after applying redundancy/retransmission/coding and other reliable transmission techniques, we call it reliable transmission [17]. However, such losses may still be manipulated by malicious nodes, such as attackers, or intercepted by eavesdroppers. We define the total erasure probability as $\mathrm{E}_{N_L} = \sum_{j=1}^{N_L} \gamma_v^j \varepsilon_j$, where $\gamma_v^j = [\gamma_1, \ldots, \gamma_j]^T$ represents the traffic load weight of each path, with $\sum_{j=1}^{N_L} \gamma_v^j = 1$, and $0 \leq \mathrm{E}N_L < 1$. We refer to the current MPEN as E_{N_L}-Level reliability.

Security. Security performance refers to the robustness of maintaining correct data delivery in the face of attackers [33]. Unlike traditional trust management [34] to ensure system security, our focus is on enhancing communication security through the lens of information theory. It takes into account active attackers who may tamper with or pollute the data received by the destination, and passive attackers who attempt to eavesdrop on and steal the data. We consider rational attackers who target only a subset of PDU sessions in the edge network rather than all of them. By treating all malicious actions as packet errors, we can employ redundant coding techniques to protect and repair the corrupted or stolen data, thereby enhancing the security performance in the edge network environment. This approach ensures the integrity and confidentiality of data transmission, even in the presence of adversaries targeting edge networks. Therefore, we can regard the redundancy of multi-path coded transmission as the security level, and we can simply refer to the edge network where N_L PDU sessions are transmitted as N_L-level secure transmission.

To sum up, service flows carried by edge networks require low-latency, high-reliability transmission with elastic capabilities in an untrusted network environment. However, achieving these requirements in practice is not easy. Due to the untrustworthy characteristics of edge networks, such as congestion, bit errors, and malicious attacks, more advanced transmission schemes and technologies need to be researched and designed to meet these challenges and ensure safe and efficient data transmission in edge networks.

2.2. A Secure Transmission Method

In this section, we analyze the transfer security of secret sharing. Secret sharing is a method of splitting data and generating multiple pieces of data through Lagrangian interpolation to transmit data over untrusted multiple paths. This approach is designed to achieve a level of security that protects data from potential attackers while providing protection against partial data-tampering attacks.

The core idea of secret sharing technology is to decompose the original data into multiple derivative pieces of data using the Lagrangian interpolation method. Let us suppose we need to transmit the original data D and decompose its encoding into n pieces of derivative data, among which any k shares of data $(k \leq n)$ are sufficient to reconstruct the original data (In the secret-sharing technique [23,35], the original data is split into multiple parts, referred to as "shares" (derived data). These shares in LERMS are then transmitted through multiple paths, with each path carrying one share of the data). First, a polynomial of degree $k-1$ is generated using Lagrange interpolation:

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{(k-1)} x^{(k-1)} \tag{1}$$

where $a_0 = D, a_1, a_2, \ldots, a_{(k-1)}$ are randomly selected coefficients. Then, we calculate the value of $f(x)$ at n different points to generate n derivative data:

$$D_i = f(x_i), i = 1, 2, \ldots, n \tag{2}$$

To reconstruct the original data, they can be calculated by Lagrangian interpolation:

$$D = L(0) = \sum_{i=1}^{k} l_i(0) D_i \tag{3}$$

$l_i(x)$ is a Lagrangian basis function that satisfies $l_i(x_j) = \delta_{ij}$ that is, when $i = j$, $l_i(x_j) = 1$; when $i \neq j$, $l_i(x_j) = 0$, defined as:

$$l_i(x) = \prod_{j \neq i}^{k} \frac{(x - x_j)}{(x_i - x_j)}, j \neq i, j = 1, 2, \ldots, k \tag{4}$$

In the edge network, each PDU session is considered an independent link provider, with a probability of being infiltrated by malicious behavior. The secret sharing transmission scheme based on Lagrangian interpolation in this scenario offers the following properties: **Information security**: In order to restore the original data through Lagrangian interpolation, the attacker must obtain at least k shares of derived data simultaneously, which increases the difficulty of their job and enhances information security. **Integrity**: Since the original data D is only dependent on the linear combination of the derived data D_i, an attacker cannot change the value of the original data by tampering with part of the derived data unless they control at least k shares of the derived data at the same time. **Reliability**: As any k shares of derived data can be used to restore the original data, the reliability of the data can be guaranteed even if part of the derived data is lost or damaged during transmission.

To summarize, the CMT of secret sharing is robust and provides security guarantees for data transmission. We aim to further explain the resilient delivery methods that our proposed scheme LERMS can provide by identifying the types of security challenges it addresses. As shown in Figure 2, an untrusted edge network may contain invalid links that result in complete data loss, malicious links that tamper with or forge data, or colluding attackers that steal or modify data to launch a Byzantine attack. These malicious behaviors can be viewed as code errors that need to be corrected. Drawing from the concept of redundancy checks, we adjust the generation strategy of derived data to tolerate different types of malicious behaviors during transmission. In the next section, we will discuss more specific secure transmission schemes in the double-layer concatenated encoder.

2.3. Downlink Multi-Service Stream Joint Coding

In an untrusted edge network environment, the data transmission of edge computing nodes faces numerous challenges, including transmission efficiency, privacy, security, and reliability [36]. Due to the minimum limit of the length of the data flow resulting from the splitting of path transmission data and security coding, padding short packets is often necessary to meet the requirement. However, this can lead to a waste of resources.

To tackle these issues and demonstrate the demand and advantages of aggregating and transmitting multiple downlink service flows, we have developed an approach based on the current core network data transmission process. This approach enables the efficient transmission of aggregated data flows while preserving data privacy. As an example, we introduce an illustrative scenario called Single-Owner Multi-Device Data Transmission with Joint Encoding (SOMD-JE), which serves to further prove the value and practical effect of aggregated transmission in 5GA edge networks.

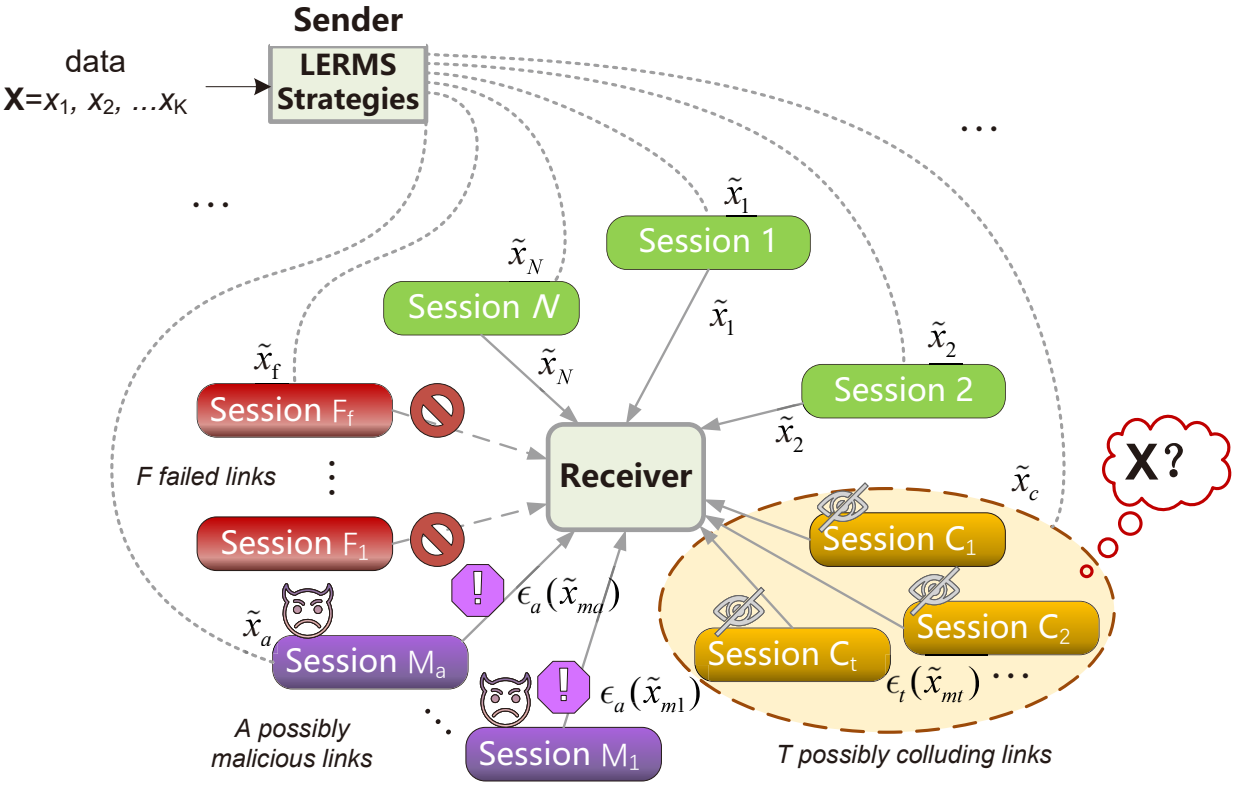


Figure 2. This diagram illustrates the transmission model (MPEN) with untrusted paths, which is an extended application of the problem model in [28]. The focus of this paper is to transmit **X** from the sender to the receiver through N_L PDU sessions with low-latency and high-reliability characteristics, despite facing multiple threats. By carefully designing the LERMS strategies, the receiver can collect whole data from a subset of PDU sessions' messages, even in the presence of failed links ($F_1, \ldots, F_{\mathrm{f}}$) and malicious links ($M_1, \ldots, M_{\mathrm{a}}$), while also ensuring data privacy from colluding links ($C_1, \ldots, C_{\mathrm{t}}$).

In the SOMD-JE scenario, we jointly encode multiple downlink service flows, which may belong to different devices but share the same owner. By analyzing the data characteristics of these flows, we found that aggregating and transmitting them does not lead to privacy leakage, and instead, the SOMD-JE improves transfer efficiency through server flows aggregation. For instance, in a smart home scenario, users' mobile phones, smart screens, VR devices, and elderly health monitoring devices all belong to the same owner (As shown in Figure 3, the receiver has multiple devices including terminals, medical monitoring devices, and smart home devices), so aggregated data flows can be transmitted without worrying about privacy leaks. Similar settings exist on hospital wards and factory floors.

This flows aggregation approach overcomes the waste of resources caused by traditional padding methods, makes full use of edge network transmission resources, and ensures safe, reliable, and efficient transmission while maintaining data privacy. The specific scheme for aggregating and transmitting downlink service flows needs to be adaptively adjusted within the current core network data transmission process. For further details, please refer to Appendix A.

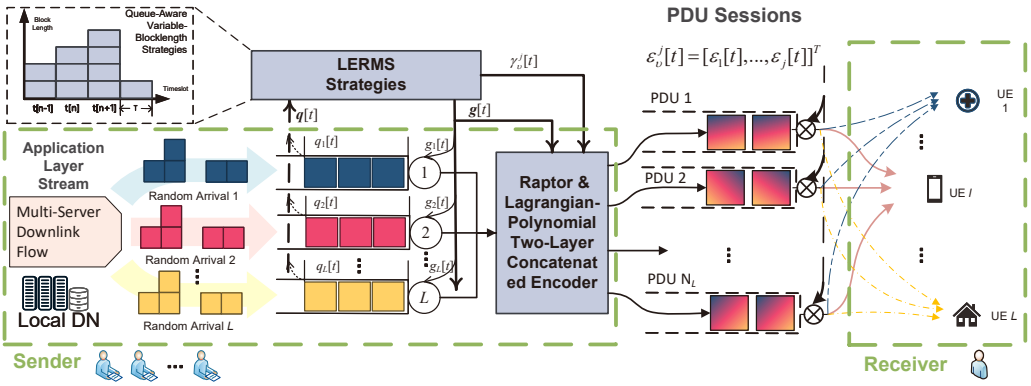


Figure 3. System model: downlink-server flow transmission with joint encoding.

3. System Model

3.1. PDU Session Queuing and Encoding Transmission Model

For data packets arriving from the application layer, we assume that the arrival of downlink data packets of different users is completely independent and identically distributed (*i.i.d.*). Let $\Lambda_l[t]$ denote the number of *Ps* arriving in the t-th timeslot for user l ($l = 1, 2, \ldots, L$). Given that the maximum value of $\Lambda_l[t]$ is N, the probability distribution of $\Lambda_l[t]$ for user l is expressed as $\boldsymbol{\lambda} = [\lambda_l^0, \lambda_l^1, \ldots, \lambda_l^N]^T$, where $\lambda_l^n = \mathbb{P}\{\Lambda_l[t] = n\}$ denotes the probability of user l receiving n packets in the t-th timeslot. The average arrival rate is defined as $\bar{\Lambda}_l = \sum_{n=0}^{N_\Lambda} n\lambda_l^n$.

For each user, a buffer with a size of Z, randomly arriving packets are accumulated, and $g_l[t]$ packets are selected from the user l's cache once the encoding of the previous block of packets is completed. Therefore, $q_l[t] \in \mathcal{Z} = \{0, 1, \ldots, \mathrm{Z}\}$, in the $[t+1]$-th timeslot, $q_l[t]$ evolves as (we define $(x)^* = \max\{x, 0\}$)

$$q_l[t+1] = \min\{(q_l[t] - g_l[t])^* + \Lambda_l[t+1], \mathrm{Z}\}. \tag{5}$$

We assume that the size of the data block $g_l[t]$ selected for each user and each generation of encoding does not exceed B, i.e., $g_l[t] \in \mathcal{N} = \{0, 1, \ldots, B\}$. The feasible region of each user's cache queue length $q_l[t]$ is given by $q_l[t] \in \mathcal{Z}$. Under the block length selection strategy $\mathcal{N}$, $\mathcal{N}(q) = \{g \in \mathbb{N} | (0, q - B + N_\Lambda)^* \le g \le \min(q, B)\}$, where the feasible range $\mathcal{N}(q)$ guarantees that each user's sending buffer queue will not underflow or overflow, we also have $B \ge N_\Lambda$ which ensures that the system will not be congested.

Under the LERMS strategies, at time slot t we encode and transmit a generation of $\boldsymbol{g}_\sigma[t]$ data packets (*Ps*) (Assuming that all packets have the same length and carry ϕ bit information), where $\boldsymbol{g}_\sigma[t] = \Sigma_{l=1}^{L} g_l[t]$. The specific encoder process will be described in detail in the next subsection. LERMS choose N_L PDU sessions for transmission, and the transmission vector is denoted as $\gamma_v^j = [\gamma_1, \ldots, \gamma_j]^T$, where γ_v^j indicates which channels are used for transmission. The output share of Lagrangian coding is also determined based on the number of channels. In summary, we express the LERMS strategies action as (g_l, γ_v^j), which changes in units of time slots, i.e., $(g_l[t] = g_l, \gamma_v^j[n] = \gamma_v^j)$, indicating that the LERMS strategy action in the t-th time slot is g_l, γ_v^j. The LERMS action of each generation of data remains unchanged in its occupied time slot, and g_l, γ_v^j is set to 0 for the time slot not occupied. Assuming that the number of data packets in each generation does not exceed B, we have $\gamma_j \in \{0, 1\}$, and let $\gamma_v = \sum_{j=1}^{N_L} \gamma_j$. Then, we have $\gamma_v \in \Gamma_v\{1, \ldots, N_L\}$ as $\{\gamma_v \in \Gamma_v | 0 \le \gamma_v \le N_L \mathbb{C}_{\{\boldsymbol{g}_\sigma[t] > 0\}}\}$ ($\mathbb{C}_{\{\cdot\}}$ denotes the characteristic function), where N_L represents the number of PDU sessions established in LERMS.

3.2. Concatenated Encoder Principle

In this section, we introduce a double-layer concatenated Algorithm 1 for processing data packets within PDU sessions, and the algorithm complexity is $\mathcal{O}(N \cdot (K + M) \cdot \log(K +$

$M)$). This encoder scheme combines the advantages of two distinct encoders to ensure reliable, efficient, and secure transmission across multiple disjoint PDU sessions.

Algorithm 1 Concatenated Encoder.

Input: Read $Ps = (Ps_1, Ps_2, \ldots, Ps_g)^T$ from the queue, $\Omega_d = (\Omega_1, \Omega_2, \ldots, \Omega_{\max})$, the PDU sessions number N_L;
Output: $\bar{P}c$

$Pc_{g\gamma\times 1} \leftarrow 0_{g\gamma}$; ▷ Raptor code encoding procedure
$G^{Raptor}_{g\gamma\times m} = G^{LT}_{g\gamma\times m} G^{pre}_{m\times g}$;
$Pc_{g\gamma\times 1} = G^{Raptor}_{g\gamma\times g} Ps_{g\times 1}$;
$\bar{\boldsymbol{RPc}} \leftarrow Pc_{g\gamma\times 1}$; ▷ Packet-level Raptor code encoder output $\bar{\boldsymbol{RPc}}$
Generate uniform random matrix $\boldsymbol{X} = \{X_1, X_2, \ldots, X_M\}$;
Split $\bar{\boldsymbol{RPc}}$ into K groups$(RPc_1, RPc_2, \ldots, RPc_K)$; ▷ Lagrange encoding procedure
for $i = 1, 2, \ldots, N$ **do**
  $LPc_i \leftarrow \sum_{j\in[K]} RPc_j \cdot \prod_{k\in[K+M]\setminus\{j\}} \frac{\alpha_i-\beta_k}{\beta_j-\beta_k} + \sum_{j=K+1}^{K+M} X_j \cdot \prod_{k\in[K+M]\setminus\{j\}} \frac{\alpha_i-\beta_k}{\beta_j-\beta_k}$; $\{\alpha_i\}_{i=1}^{K+M} \cap \{\beta_j\}_{j=1}^{K} = \emptyset$
  $\bar{P}c \leftarrow append[\bar{P}c, LPc_i]$
end for ▷ Fast Polynomial interpolation
return $\bar{P}c$;

As shown in Figure 4, the LERMS strategy incorporates a concatenated encoder scheme. The first-level encoder, known as the Raptor Encoder, primarily focuses on providing reliability by enhancing the decodability and dependability of data streams transmitted within PDU sessions. The second-level encoder, called the Lagrange Polynomial Encoder, is designed to offer resilient transmission over untrusted paths within the edge network. Through the implementation of the "Double-layer Concatenated Encoder" scheme, we aim to provide a robust encoding solution for PDU sessions. The concatenated encoder principle will be described in detail in two subsections.

3.2.1. Packet-Level Raptor Code Encoder

In this subsection, we present a packet-level Raptor encoding scheme for enhancing the robustness of PDU session transmissions in MPEN. We model these PDU sessions as packet erasure channels, where data packets can either be received entirely or erased. To improve transmission reliability, Ps will be transmitted after the Raptor encoder.

We encode Ps in PDU sessions using a Raptor packet encoder. The LERMS strategy selects g data packets for encoding at each time slot. Raptor-encoded data packets Pc are generated through two stages: an outer coder (pre-code) Φ and LT encoder [37] (inner code). The pre-code Φ is a (g, m) block code that generates m intermediate coded symbols from g Ps. The inner LT encoder generates $g\gamma$ data packets through $\xi(g, m, \Omega(x))$, where γ represents the encoding redundancy, which is the inverse of the code-rate. The LT encoding matrix is constructed from a predetermined degree distribution $\Omega(x) = \sum_{d=1}^{d_{\max}} \Omega_d x^d$, with the degree distribution following a probability distribution $\Omega_d = (\Omega_1, \Omega_2, \ldots, \Omega_{\max})$ and satisfying $\sum_{d=1}^{d_{\max}} \Omega_d = 1$. The relationship between the encoder's input and output is given by $Pc_{g\gamma\times 1} = G^{LT}_{g\gamma\times m} G^{pre}_{m\times g} Ps_{g\times 1}$, where G_{pre} and G_{LT} denote the outer and inner encoding matrices, respectively.

The Raptor coding scheme enhances the reliability of data transmission by mixing data packets. When transmitting over packet erasure channels, it offers higher robustness and resilience. Data transmissions do not require feedback, as the encoding redundancy can be determined based on the channel characteristics. The receiver only needs to receive slightly more than the number of source data packets g to complete decoding. Due to the encoding scheme's data mixing approach, even if some encoded data packets are erased, the entire source data block can still be recovered by continuing to receive encoded data packets.

3.2.2. Encoder for Lagrangian Polynomial Code Multipath Transmission

LCMT performs encoding operations on the output of Raptor encoder, denoted as RPc, to provide safe and reliable transmission over multiple paths, some of which may be untrusted, while protecting data privacy. Let us suppose an MPEN has N_L physically isolated paths. LCMT encodes RPc to generate LPc, which is the output of Lagrangian encoder. Each generation of LPc will be split into K groups, namely $(LPc_1, LPc_2, \ldots, LPc_k)$, and transmitted to the receiver. Through the reasonable coding method of LCMT, the system aims to tolerate the failure of S paths in the MPEN, malicious behavior of A paths, and collusion of T paths to steal data while still obtaining safe and reliable data transmission. If the data are safely received, we call this transmission scheme realizing the triplet (S, A, T).

To achieve this level of resilience, it is necessary to satisfy the following condition:

$$N_L \geq K + T + S + 2A \tag{6}$$

At this point, we can say that LCMT can realize the triplet (S, A, T). The significance of this result is that, by adding one path, the link failure resilience can be increased by 1 or the robustness of the malicious behavior path can be increased by 1/2. Furthermore, data privacy can be improved at the same time.

Let us take the transmission of $\{Pc\}$ as an example, where $K = 2$, $N = 6$, and $(S, A, T) = (1, 1, 1)$. In this case, $\{Pc\}$ is split into $\boldsymbol{Pc_1}$ and $\boldsymbol{Pc_2}$. The key point of LCMT is to select a uniform random matrix $\boldsymbol{X}$ and encode it through Lagrange interpolation polynomial $(\boldsymbol{Pc_1, Pc_2, X})$. The encoding process is given by the following equation:

$$\psi(x) \triangleq \boldsymbol{Pc_1}\frac{(x-2)(x-3)}{(1-2)(1-3)} + \boldsymbol{Pc_2}\frac{(x-1)(x-3)}{(2-1)(2-3)} + \boldsymbol{X}\frac{(x-1)(x-2)}{(3-1)(3-2)} \tag{7}$$

To transmit $\{Pc\}$, six different values $\{\alpha_i\}_{i=1}^{6}$ in the finite field $\mathbb{F}$ are determined such that $\{\alpha_i\}_{i=1}^{6} \cap \{1, 2\} = \emptyset$. Then, N_L PDU sessions transmit $\psi(\alpha_1), \psi(\alpha_2), \ldots, \psi(\alpha_6)$, where each path transmits the value after interpolation. In other words, the linear combination of $\boldsymbol{Pc_1}$ and $\boldsymbol{Pc_2}$ is hidden by $\xi\boldsymbol{X}$, where ξ is a nonzero value. Since $\boldsymbol{X}$ is uniformly random, the data privacy of $T = 1$ can be guaranteed. If there is one malicious path ($A = 1$) and one invalid path ($S = 1$), a Reed–Solomon decoder needs to be used at the receiver, and three additional shares of data are required (one additional copy for each invalid path and two additional shares for the malicious path). At the receiving end, Pc_1 and Pc_2 can be recovered by computing $\psi(1)$ and $\psi(2)$.

Double-layer concatenated encoder processes downlink data packets from the edge DN to the UE, enabling their transmission through multiple disjoint PDU sessions. By leveraging both Fountain and Lagrange encoders, a concatenated encoder can efficiently and securely handle data packets within the PDU sessions, simultaneously improving transmission reliability and resilience against untrusted path transmissions.

3.3. Edge Network with Untrusted Paths

In this section, we analyze the transmission characteristics of MPEN, which is a multipath transmission network consisting of multiple physically disjoint PDU sessions. Our aim is to determine the performance level that can be achieved with the number N_L of multiple transmission paths through an analysis of encoding transmission characteristics.

The input and output of MPEN are denoted by $\mathbf{X}$ and $\mathbf{Y}$, respectively, while MPEN provides the transmission capability of the edge network. To begin, we define the parameters of the channel model. We assume that MPEN establishes N_L paths for the current transmission task, which are classified based on their behavioral characteristics. Specifically, we consider N paths that can be transmitted normally, F failed sessions that cannot be transmitted within a specified time, A malicious transmission paths, and T paths that may be compromised by Byzantine attackers or eavesdroppers.

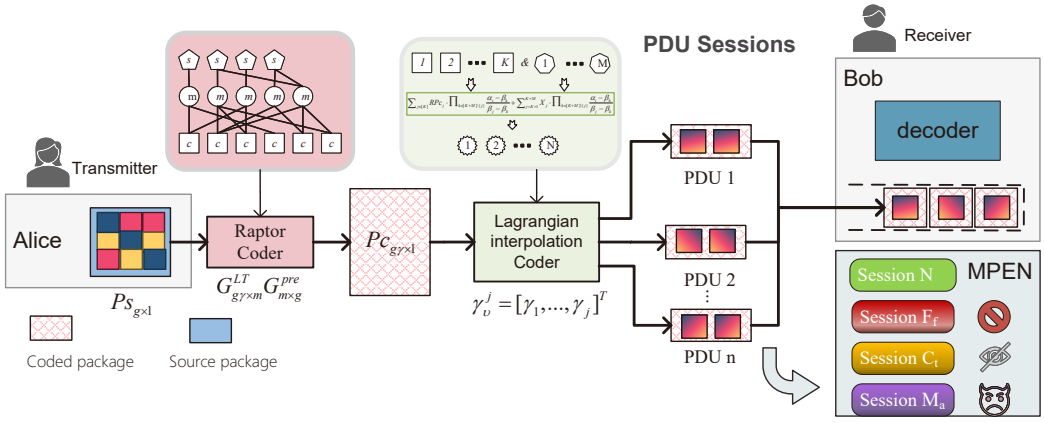


Figure 4. LERMS concatenated encoder scheme.

We assume that the input data stream **X** is evenly divided into K sub-packets $x_1, x_2, \ldots, x_K$, and these sub-packets are encoded into N packets as $\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_N$, where $K \leq N$. These encoded packets are distributed across N_L PDU sessions for transmission, and the receiver obtains the result $Y = y_1, y_2, \ldots, y_N$, where $N = N_L$ (Here, we assume that each PDU session has only one single behavioral feature) subject to the following constraints:

$$N_L = N + F + A + T \tag{8}$$

Let $\overrightarrow{\sum}$ denote the data collection operator and define the receiving vector $\mathbf{r}$, where its j-th element r_j represents the received data of the j-th PDU session. We define four types of receiving situations: normal transmission, represented by $\mathbf{r}_N$, where $\mathbf{r}_N = \overrightarrow{\sum}_{n=1}^{N} r_n = \overrightarrow{\sum}_{n=1}^{N} \tilde{x}_n$; failed transmission, represented by $\mathbf{r}_F$, where $\mathbf{r}_F = \overrightarrow{\sum}_{\mathrm{f}=1}^{F} r_{\mathrm{f}} = \overrightarrow{\sum}_{\mathrm{f}=1}^{F} (0 \cdot \tilde{x}_{\mathrm{f}})$; malicious transmission, represented by $\mathbf{r}_A$, where $\mathbf{r}_A = \overrightarrow{\sum}_{a=1}^{A} r_a = \overrightarrow{\sum}_{a=1}^{A} \epsilon_a(\tilde{x}_a)$; and Byzantine attack, represented by $\mathbf{r}_T$, where $\mathbf{r}_T = \overrightarrow{\sum}_{t=1}^{T} r_t = \overrightarrow{\sum}_{t=1}^{T} \epsilon_t(\tilde{x}_t)$. Here, we introduce the functions ϵ_a and ϵ_t, which, respectively, represent the influence of malicious transmission channels and Byzantine attack channels on the output.

We assume that the PDU session is a memoryless erasure channel, meaning that the output r_i depends only on the input x_i. Additionally, each data packet has a certain probability of loss, denoted by $\varepsilon_v^j = [\varepsilon_1, \ldots, \varepsilon_j]$, which represents the erasure probability of different sessions. This loss affects only the receiving result, so we have:

$$\mathbf{Y} = \overrightarrow{\sum}_{j=1}^{N_L} \varepsilon_j r_j \tag{9}$$

The above description about MPEN is fully in line with the 3GPP standard's definition of PDU session [38].

3.4. MPEN with Untrusted Path Reliable Function

In this subsection, we analyze the reliability of the two-layer concatenated encoder CMT in the edge network and propose a generalized reliability model. For the enumeration of all possible session state characteristics described in Figure 2, we first employ a $2^{N_L} \times N_L$ matrix $\mathbb{C}$:

$$\mathbb{C} = \begin{bmatrix} 1 & 1 & \ldots & 0 \\ \vdots & \vdots & \ldots & \vdots \\ 1 & 0 & \ldots & 0 \end{bmatrix}^{\mathrm{T}} \tag{10}$$

The value 0/1 of the elements in row i and column j indicates the i-th possibility of success/failure at the receiving end through the j-th PDU session. After decoding with the Reed–Solomon decoder [39], the malicious path is screened. Error correction and error

detection are performed, and abnormal data can be discarded directly; considering that the transmission of this kind of PDU session failed, at this time, the corresponding $c_{i,j}$ is set to 0 to exclude malicious data packets (The defense level against Byzantine attacks is determined during the PDU session establishment and will not be analyzed here).

When transmitting through multiple PDU sessions, we assume that the maximum block error rate and bandwidth guaranteed by GBR QoS [40] for a set of PDU sessions are the same. Based on the delay model described in Section 2, the relationship between packet delivery ratio and transmission delay is established as the cumulative function of the delay probability distribution, called the delay reliability function [30]. Based on end-to-end network monitoring, the delay reliability function is available:

$$F_{LERMS}\left(T, \gamma_v^j, g\right) = \sum_{i=1}^{2^{N_L}} \chi_i \prod_{j=1}^{N_L} H_j(T, \gamma_j g) \tag{11}$$

We consider that N_L is at least greater than 3, and the relationship between delay and transmission reliability is given by $FLERMS$, where

$$\chi_i = \begin{cases} 1, & \text{if } \Sigma_{j=1}^{N_L} c_{i,j}\gamma_j \geq \gamma_d \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

g_i will exclude failed transmissions (i.e., exclude the output of malicious PDU sessions) to ensure that only the correct output of successful decoding is included, and γ_d is the threshold to ensure successful decoding, with a typical value of 1.05. Hj is defined as

$$H_j(T, \gamma_j g) = \begin{cases} F_j(T, \gamma_j g), & \text{if } c_{i,j}{=}1 \\ 0, & \text{if } c_{i,j}{=}0 \end{cases} \tag{13}$$

Among them, the product of $H_j(T, \gamma_j g)$ for $j = 1, \ldots, N_L$ appears in the form of a cumulative distribution function (CDF). In the default working mode, the completion of the last data transmission is regarded as the completion of the reliable transmission process.

4. Trade-Off Delay-Security for Variable Block-Length LERMS Strategy

4.1. The Markov Chain under LERMS Strategies Formulation

Based on the LERMS strategy, we uniformly sample data packets from multiple downlink service flows, and perform joint encoding and transmission. We probabilistically determine the sampling and transmission strategies of different service flows based on the current queue length, and transmit them on the edge network. Specifically, given the queue length $q_l[n]$, $l = 1, \ldots, L$, we establish the conditional probability $f_{\mathcal{Q}}^{\mathcal{G},\gamma}$ to determine the concatenated encoder coding block length $\boldsymbol{g}_\sigma[t]$, sample from L flows, and distribution and transmission strategy $\gamma_v^j[t]$ of J paths for the given queue length $q_l[n]$.

$$f_{\mathcal{Q}}^{\mathcal{G},\gamma} = \mathbb{P}\left\{\mathcal{G}[t] = \mathcal{G}, \gamma_v^j[t] = \gamma_v^j | \mathcal{Q}[t] = \mathcal{Q}\right\} \tag{14}$$

We denote the packet sampling size and buffer queue length of L server flows as vectors $\mathcal{G}[t] = [g_1[t], g_2[t], \ldots, g_L[t]]^T$ and $\mathcal{Q}[t] = [q_1[t], q_2[t], \ldots, q_L[t]]^T$, respectively. The transmission selection and distribution strategy are determined based on the current queue length $\mathcal{Q}[t]$. We denote specific queue lengths and sample sizes by vectors $\mathcal{Q}$ and $\mathcal{G}$.

Based on Equation (14), the strategy function of LERMS can be obtained:

$$\boldsymbol{S} = \left\{f_{\mathcal{Q}}^{\mathcal{G},\gamma} : \mathcal{Q} \in \mathcal{Z}^L, \mathcal{G} \in \mathcal{N}^L, 1 \leq \gamma_v < N_L, \gamma_j \in \{0,1\}\right\}. \tag{15}$$

Here, the value space of $\mathcal{Q}$ and $\mathcal{G}$ are $\mathcal{Z}^L$ and $\mathcal{N}^L$, respectively, obtained by taking the Cartesian product $\odot_{i=1}^L \mathcal{Q}(q_i)$ and $\odot_{i=1}^L \mathcal{G}(g_i)$. To ensure feasibility of the transmission strategies, we set the value of $f_{\mathcal{Q}}^{\mathcal{G},\gamma}$ to 0 for all infeasible strategies $\mathcal{G}$ and γ. In other

words, if a given combination of packet sampling sizes $\mathcal{G}$ and transmission strategies γ is infeasible, its corresponding probability value is forced to 0. To prevent the sender buffer from overflowing or underflowing, the system state q_l evolves based on Equation (5). We assume a temporary steady-state condition where no PDU session is being established or released, i.e., $\forall \mathcal{Q} \in \mathcal{Z}, \Sigma_{\mathcal{G}\in\mathcal{N}(\mathcal{Q})}\Sigma_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} f_{\mathcal{Q}}^{\mathcal{G},\gamma} = 1$.

Under the LERMS strategy, we consider the transmission process of downlink server flow in the edge network as a Markov chain, where the queue length $\mathcal{Q}[t]$ is the state value of the system. By analyzing the steady-state distribution of the Markov chain, we further analyze how to trade off between latency and reliability. Based on the given strategy $\boldsymbol{S}$, we first analyze the state transition probability of different queue lengths $\beta_{\mathcal{Q},\mathcal{Q}'} = \mathbb{P}\{q_l[t+1] = q_l' | q_l[t] = q_l\}$, where $\mathcal{Q}$ and $\mathcal{Q}'$ are the vectors of buffer queue lengths at two consecutive time slots. Specifically, the state transition probability $\beta_{\mathcal{Q},\mathcal{Q}'}$ can be expressed as

$$\beta_{\mathcal{Q},\mathcal{Q}'} = \sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})} \sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} f_{\mathcal{Q}}^{\mathcal{G},\gamma} \prod_{l=1}^{L} \sum_{n=0}^{N} \lambda_l^n \mathbb{C}_{\{\min\{(q_l-g_l)^*+n=q_l'\}} \tag{16}$$

where λ_l^n denotes the probability of n packet arrivals during single time slot for l-th flow. The value range of $\mathcal{Q}'$ is $\mathcal{Z}^L$.

Using $\beta_{\mathcal{Q},\mathcal{Q}'}$, we can determine the steady-state probability $\pi_{\boldsymbol{S}}(\mathcal{Q}')$ for different queue lengths, wherein $\mathcal{Q}$ belongs to $\mathcal{Z}^L$. We can then obtain the Markovian steady-state probability balance equation:

$$\sum_{\mathcal{Q}'\in\mathcal{Z}^L(\mathcal{Q})} \beta_{\mathcal{Q},\mathcal{Q}'}\pi_S(\mathcal{Q}') = \pi_S(\mathcal{Q}) \tag{17}$$

where $\mathcal{Z}^L(\mathcal{Q})$ is a subset of $\mathcal{Z}^L$, that contains all possible values of $\mathcal{Q}$ under $\boldsymbol{S}$. The collection branch is expressed as $\mathcal{Z}^L(\mathcal{Q}) = \{\mathcal{Q}' \in \mathcal{Z}^L | q_l - g_l \le q'_l \le q_l + n, \forall l\}$

4.2. Constrained Optimization Problem Construction

Based on the steady-state analysis of the state value $\mathcal{Q}$ within the edge network, we aim to construct a constrained optimization problem to balance the delay and reliability of multiple downlink flows transmission. Intuitively, joint encoding of multiple service flows' P_s not only improves the coding efficiency but also enhances the security compared to a single service flow. However, it also increases the corresponding queuing delay. Therefore, based on the LERMS strategy, we propose a safe and reliable multi-path transmission encoder strategy. This strategy can effectively utilize the edge network transmission resources and improve the transmission efficiency while satisfying the constraints of reliability functions.

In the constrained optimization problem, we aim to minimize the weighted sum of queuing delays for multiple users while satisfying reliability and system bandwidth constraints. The queuing delay D_μ^S is determined based on Little's Law:

$$D_\mu^S = \sum_{l=1}^{L} \frac{\mu_l}{\lambda_l} \sum_{\mathcal{Q}\in\mathcal{Z}^L} \sum_{q'\in\mathcal{Z}} q'\pi_S(\mathcal{Q})\mathbb{C}_{\{q_l=q'\}} \tag{18}$$

The weight coefficients are represented as $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_L]^T$, where $\boldsymbol{\mu}$ adheres to the conditions of non-negativity and sums up to 1. We can further compute the reliability and bandwidth using the following equations:

$$R^S = \sum_{\mathcal{Q}\in\mathcal{Z}^L} \sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})} \sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} F_{LERMS}(\boldsymbol{g}_\sigma, \gamma) f_{\mathcal{Q}}^{\mathcal{G},\gamma}\pi_S(\mathcal{Q}) \tag{19}$$

$$W^S = \sum_{\mathcal{Q}\in\mathcal{Z}^L} \sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})} \sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} W_S^j(\boldsymbol{g}_\sigma, \gamma_v^j) f_{\mathcal{Q}}^{\mathcal{G},\gamma}\pi_S(\mathcal{Q}) \tag{20}$$

We define the optimization variable as $x_{\mathcal{Q}}^{\mathcal{G},\gamma} = f_{\mathcal{Q}}^{\mathcal{G},\gamma}\pi_S(\mathcal{Q})$, the optimization problem can be formulated as follows:

$$\min_{\{x_{\mathcal{Q}}^{\mathcal{G},\gamma}\}} \quad \sum_{\mathcal{Q}\in\mathcal{Z}^L}\sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})}\sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} D_\mu(\mathcal{Q})x_{\mathcal{Q}}^{\mathcal{G},\gamma} \tag{21a}$$

$$\text{s.t.} \quad \sum_{\mathcal{Q}\in\mathcal{Z}^L}\sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})}\sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} F_{LERMS}(g_\sigma,\gamma)x_{\mathcal{Q}}^{\mathcal{G},\gamma} \geq r^{\text{th}} \tag{21b}$$

$$\sum_{\mathcal{Q}\in\mathcal{Z}^L}\sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})}\sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} W_S^j(g_\sigma,\gamma_v^j)x_{\mathcal{Q}}^{\mathcal{G},\gamma} \leq W_j^{\text{th}} \tag{21c}$$

$$\sum_{\mathcal{Q}'\in\mathcal{Z}^L(\mathcal{Q})}\sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q}')}\sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} x_{\mathcal{Q}'}^{\mathcal{G},\gamma}\prod_{l=1}^{L}\lambda_l^{q_l-q_l'-g_l} = \sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})}\sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} x_{\mathcal{Q}}^{\mathcal{G},\gamma}, \quad \forall\mathcal{Q}\in\mathcal{Z}^L \tag{21d}$$

$$\sum_{\mathcal{Q}\in\mathcal{Z}^L}\sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})}\sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} x_{\mathcal{Q}}^{\mathcal{G},\gamma} = 1 \tag{21e}$$

$$x_{\mathcal{Q}}^{\mathcal{G},\gamma} \geq 0, \qquad \forall\mathcal{Q}\in\mathcal{Z}^L,\ \mathcal{G}\in\mathcal{N}^L,\ \gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G}) \tag{21f}$$

where $D_\mu(\mathcal{Q}) = \Sigma_{l=1}^{L}\frac{\mu_l}{\lambda_l}q_l$.

By solving the optimization problem in Equation (21), we can obtain the minimum average queuing delay under the constraints of reliability and system bandwidth in an untrusted network environment. This allows us to determine the optimal trade-off between latency and reliability. We define the optimal solution of Equation (21) as ${x^*}_{\mathcal{Q}}^{\mathcal{G},\gamma}$, and we will use the optimal strategy S^* to determine the steady-state probability $\pi_{S^*}(\mathcal{Q})$:

$$\pi_{S^*}(\mathcal{Q}) = \sum_{\mathcal{G}\in\mathcal{N}^L(\mathcal{Q})}\sum_{\gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})} {x^*}_{\mathcal{Q}}^{\mathcal{G},\gamma} \tag{22}$$

among them, we define the lerms optimal strategy as f^*

$${f^*}_{\mathcal{Q}}^{\mathcal{G},\gamma} = \begin{cases} \frac{{x^*}_{\mathcal{Q}}^{\mathcal{G},\gamma}}{\pi_{S^*}(\mathcal{Q})} & \text{if } \pi_{S^*}(\mathcal{Q}) > 0 \\ \mathbb{C}_{\{\boldsymbol{g}=\boldsymbol{g}_\sigma^{\max}{}_{\mathcal{Q}}\}} & \text{if } \pi_{S^*}(\mathcal{Q}) = 0, \end{cases} \tag{23}$$

where we define $\boldsymbol{g}_\sigma{}_{\mathcal{Q}}^{\max} = \arg\max_{g\in\mathcal{N}^L}$. In summary, the optimal strategy S^* obtained from the solution of the optimization problem can be used to determine the transmission strategy $\mathcal{G}[t]$ and $\gamma_v^j[t]$ based on the current system state $\mathcal{Q}[t]$ using the conditional probability $\{{f^*}_{\mathcal{Q}}^{\mathcal{G},\gamma} : \mathcal{G}\in\mathcal{N}^L(\mathcal{Q}), \gamma_v^j\in\Gamma_v^{N_L}(\mathcal{G})\}$.

4.3. Matrix-Based Solving Methods

Considering the exponential growth of the value range of the cache queue $\mathcal{Q}$ and transmission policy with the increase in the number of business flows L and the number of paths N_L, this subsection proposes a matrix-based approach to obtain the optimal trade-off for untrusted MPEN transmission. Firstly, we rewrite the linear programming problem in Equation (21) and then, using the unified matrix constraints in Algorithm 2, algorithm complexity is $\mathcal{O}(L(Z\cdot\mathcal{N}+2))$, and we automatically generate the LP problem and solve for the LERMS optimal strategy for downlink transmission of multiple service flows.

We represent the optimization variable $x_{\mathcal{Q}}^{\mathcal{G},\gamma}$ as a column vector, denoted by $\mathbf{x}$ with an index corresponding to the optimization variable $x_{\mathcal{Q}}^{\mathcal{G},\gamma}$. The dimension of $\mathbf{x}$ is given by

$$\prod_{l=1}^{L} \left(|\mathcal{Z}||\mathcal{N}||\Gamma_v^{N_L}|\right)^{l-1}(|\Gamma|(|\mathcal{N}|q_l + g_l) + \gamma_v^j) + 1 \tag{24}$$

where $|\cdot|$ denotes the number of elements in the set. We can express Equation (21) in matrix form as follows:

Algorithm 2 Algorithm to constraints matrix for Equation (21).

Input: Number of server flows, L; Peak flow rate, N_Λ; The upper bounds of $g[t]$, B; Number of PDU sessions, N_L; The probability distribution of Ps, $\boldsymbol{\lambda_l} = [\lambda_l^0, \lambda_l^1, \ldots, \lambda_l^{N_\Lambda}]^T$, $l = 1, \ldots, L$. Reliability function $F(\boldsymbol{g}_\sigma, \gamma)$.

Output: Reliability vector, $\boldsymbol{R}$; PDU Session Aggregate Maximum Bit Rate vector $\boldsymbol{W}$; Delay vector, $\boldsymbol{D}_\mu$; Matrix for constraints, $\mathbf{M}$.

$\boldsymbol{g}_\sigma \leftarrow 0$, $\dot{\boldsymbol{g}}_\sigma \leftarrow \mathbf{1}_{|\mathcal{Z}|} \otimes [0, 1, \ldots, B]^T$, $W \leftarrow 0$; ▷ Generate $\boldsymbol{R}$, $\boldsymbol{W}$.
for $l = 1$ to L **do**
  $\boldsymbol{W} \leftarrow \boldsymbol{W} \otimes \mathbf{1}_{|\mathcal{N}|\times|\mathcal{Z}|}$;
  $\boldsymbol{g}_\sigma \leftarrow \boldsymbol{g}_\sigma \otimes \mathbf{1}_{|\dot{\boldsymbol{g}}|} + \mathbf{1}_{|\boldsymbol{g}_\sigma|} \otimes \dot{\boldsymbol{g}}_\sigma$;
end for
$\boldsymbol{W} \leftarrow \boldsymbol{W} \otimes [0, 1, \ldots, B]^T$, $\boldsymbol{R} \leftarrow \mathbf{0}_{|\boldsymbol{g}_\sigma|} \otimes \mathbf{1}_{|\Gamma|}$;
for $\{\gamma_{v\_i}^j\}$, $i = 1$ to 2^{N_L} **do**
  $\boldsymbol{R} \leftarrow \boldsymbol{R} + F(\boldsymbol{g}_\sigma, \gamma_{v\_i}^j) \otimes \mathbf{e}_{|\Gamma|,i}$;
end for
$\boldsymbol{D}_\mu \leftarrow 0$ ▷ Generate delay vector, $\boldsymbol{D}_\mu$.
for $l = 1$ to L **do**
  $\boldsymbol{D}_l \leftarrow \frac{\mu_{L-l+1}}{\lambda_{l-l+1}}[0, 1, \ldots, Z]^T \otimes \mathbf{1}_{|\mathcal{N}|}$;
  $\boldsymbol{D}_\mu \leftarrow \boldsymbol{D}_\mu \otimes \mathbf{1}_{|\boldsymbol{D}_l|} + \mathbf{1}_{|\boldsymbol{D}_\mu|} \otimes \boldsymbol{D}_l$;
end for
$\boldsymbol{D} \leftarrow \boldsymbol{D} \otimes \mathbf{1}_{|\Gamma|}$;
$\dot{\boldsymbol{M}} \leftarrow 1$, $\ddot{\boldsymbol{M}} \leftarrow 1$, $\tilde{\boldsymbol{M}} \leftarrow 1$; ▷ Generate delay vector, $\boldsymbol{M}$.
for $l = 1$ to L **do**
  for $q = 1$ to Z **do**
    $\dot{\boldsymbol{M}}_{l,q} \leftarrow \mathbf{1}_{|\mathcal{Z}|,|\mathcal{N}|}$, $\ddot{\boldsymbol{M}}_{l,q} \leftarrow \mathbf{1}_{|\mathcal{Z}|,|\mathcal{N}|}$;
    for all $g \in \mathcal{N}(q)$ **do**
      $\boldsymbol{M}_{l,q,g} \leftarrow \begin{bmatrix} \mathbf{0}_{q-g,N_\Lambda+1}; \\ \mathrm{diag}(\mathbf{1}_{1,N_\Lambda+1}); \\ \mathbf{0}_{Z-N_\Lambda+g-q,N_\Lambda+1} \end{bmatrix}$;
      $\dot{\boldsymbol{M}}_{l,q}(:, g+1) \leftarrow \boldsymbol{M}_{l,q,g}\boldsymbol{\lambda}_{L+1-l}$;
      $\ddot{\boldsymbol{M}}_{l,q}(:, g+1) \leftarrow \boldsymbol{M}_{l,q,g}\boldsymbol{\lambda}_{N_\Lambda+1}$;
    end for
    $\dot{\boldsymbol{M}}_l \leftarrow [\dot{\boldsymbol{M}}_l, \dot{\boldsymbol{M}}_{l,q}]$, $\ddot{\boldsymbol{M}}_l \leftarrow [\dot{\boldsymbol{M}}_l, \ddot{\boldsymbol{M}}_{l,q}]$;
  end for
  $\dot{\boldsymbol{M}} \leftarrow (\dot{\boldsymbol{M}} \otimes \ddot{\boldsymbol{M}}_l)(\ddot{\boldsymbol{M}} \otimes \dot{\boldsymbol{M}}_l)$, $\ddot{\boldsymbol{M}} \leftarrow \ddot{\boldsymbol{M}} \otimes \ddot{\boldsymbol{M}}_l$;
  $\tilde{\boldsymbol{M}} \leftarrow \tilde{\boldsymbol{M}} \otimes (\mathrm{diag}([\mathbf{1}_{|\mathcal{Z}|}]^T) \otimes ([\mathbf{1}_{|\mathcal{N}|}]^T))$
end for
$\boldsymbol{M} \leftarrow \dot{\boldsymbol{M}} - \tilde{\boldsymbol{M}}$

$$\min_{x \geq 0} \quad \boldsymbol{D}_{\mu}^{T}\mathbf{x} \tag{25a}$$

$$\text{s.t.} \quad \boldsymbol{R}^{T} \geq r^{\text{th}} \tag{25b}$$

$$\boldsymbol{W}^{T} \leq W_{j}^{\text{th}} \quad \forall j \in \{1, 2, \ldots, N_L\} \tag{25c}$$

$$\boldsymbol{M}\mathbf{x} = \mathbf{0} \tag{25d}$$

$$\mathbf{1}^{T}\mathbf{x} = 1 \tag{25e}$$

where $\mathbf{0}$ and $\mathbf{1}$ are zero and one vectors, respectively, $f_{\mathcal{Q}}^{\mathcal{G},\gamma}$ is the joint encoding and transmission strategy for the given queue length $\mathcal{Q}$, $\beta_{\mathcal{Q},\mathcal{Q}'}$ is the state transition probability for different queue lengths, $\pi_S(\mathcal{Q})$ is the steady-state probability for the given queue length $\mathcal{Q}$, and μ_l is the weight coefficient for user l. D_l and B_l represent the queuing delay and bandwidth for user l, respectively. The maximum values of queuing delay and bandwidth are denoted as $D_{\max}$ and $B_{\max}$, respectively. We define the following vectors: Object delay vector $\boldsymbol{D}_{\mu}$; Reliability vector, $\boldsymbol{R}$; Bandwidth vector $\boldsymbol{W}$. We also construct Equation (21d) by the matrix $\boldsymbol{M}$.

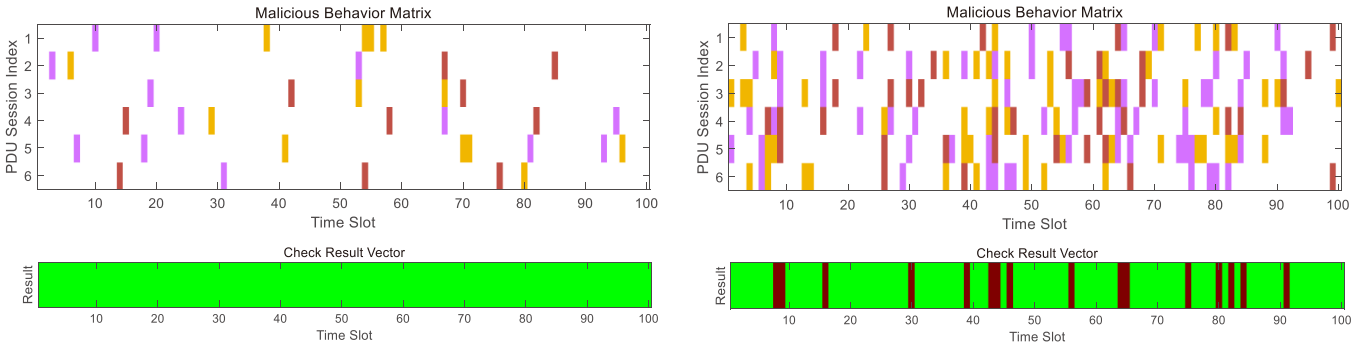
By Algorithm 2, we can automatically obtain Equation (25), where we only need to determine $\boldsymbol{D}_{\mu}$, $\boldsymbol{R}$, $\boldsymbol{W}$ and $\boldsymbol{M}$. In Algorithm 2, as shown in lines 10 to 28, we generate the feature matrices for each server flow, namely $\boldsymbol{D}_{\mu}$, $\ddot{\boldsymbol{M}}$, and $\tilde{\boldsymbol{M}}$ for the l-th flow. We then construct the target matrix by the Kronecker product $\otimes$ of these matrices. We define $\mathbf{1}_k$ and $\mathbf{0}_k$ as column vectors containing all ones and all zeros, respectively, with k items. We also define the sampling vector $\mathbf{e}n, k$ as an N-dimensional column vector, where the k-th item is 1. By applying Algorithm 2, we effectively transform Equation (21) as well as Equation (25) into an LP problem in matrix form, allowing us to solve for the optimal strategy for aggregated transmission of downlink multi-service flows in MPEN.

5. Numerical Results

In this section, we validate the effectiveness of the LERMS strategy in improving low-latency and highly reliable transmission capabilities in an untrusted edge network environment. To conduct the evaluation, we set up an experimental mobile network and analyze the performance of the proposed LERMS scheme. The core network of LERMS is implemented by enhancing the Free5GC platform [41]. The RCMEN core network is deployed on a laptop equipped with an I7-11800 processor and 16 GB of memory, while the UPF is deployed on a desktop computer with an I7-10700 processor and 32 GB of memory. We simulate the transmission of data by modifying the provided script in the Free5GC.

Firstly, in Figures 5a,b we consider the simulation test of the secure transmission capability of the double-layer encoder. Considering establishing six PDU sessions according to the same situation as described in Section 3.2.2, that is, N_L = 6. We will randomly add malicious behavior, including transmission failure link(red), malicious tampering link(purple), eavesdropping link(yellow), i.e., $(S, A, T) = (1, 1, 1)$. For the concatenated encoder scheme encoding, we use the Reed–Solomon decoder to receive the $\bar{P}c$. To decode, we will identify the decoding result, which can solve the Ps as recognition success, that is, we do not consider the potential safety hazards of Byzantine attackers and eavesdroppers, which is directly regarded as a system capability in the PDU session establishment phase. There is no need to pass the experimental analysis; as shown in Figure 5a, our encoding strategy can provide reliable transmission capabilities in an untrusted network environment. The system has $(S, A, T) = (1, 1, 1)$ protection capability. As can be seen in Figure 5b, we conducted further tests to evaluate the maximum safety capability. We found that excluding malicious attacks, the maximum capability to recover Ps is $(2, 0, 2)$. However, as malicious attacks can tamper with data packets, and every tampering of a data packet requires two more data packets for error correction. Thus, in the check matrix, we found that in the time slot corresponding to red, the number of malicious tampering and transmission

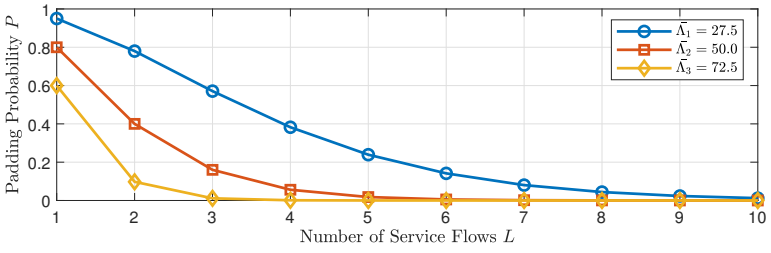
failures exceeds the range defined by Equation (6). Therefore, it is not possible to provide transmission capability in an untrusted transmission network with $N_L = 6$.



(**a**) (S, A, T)= (1, 1, 1) (**b**) (S, A, T)= (2, 2, 2)

Figure 5. Untrusted edge network with concatenated encoder check results. In the malicious behaviors matrix: link failures are represented in red, malicious tampering in purple, and eavesdropping in yellow.

We analyze the effectiveness of the SOMD-JE strategy, with the aim of coding and ensuring transmission efficiency as well as data encoding security and CMT. The shortest coding block length of *Ps* is set to 100, and we assume that the arrival probabilities of data packets of different service flows i.i.d, as shown in Figure 6b, while the relationship between the number of aggregated service flows and the probability of padding occurrence is given in Figure 6a. It can be seen that with small $\bar{\Lambda}$, the padding probability is high, and the transmission efficiency of the edge network is low at this time. As the number of aggregated flows L increases, $\mathbb{P}$ decreases. Therefore, aggregated data flows can significantly reduce the padding probability $\mathbb{P}$; therefore, joint encoding improves transmission efficiency.



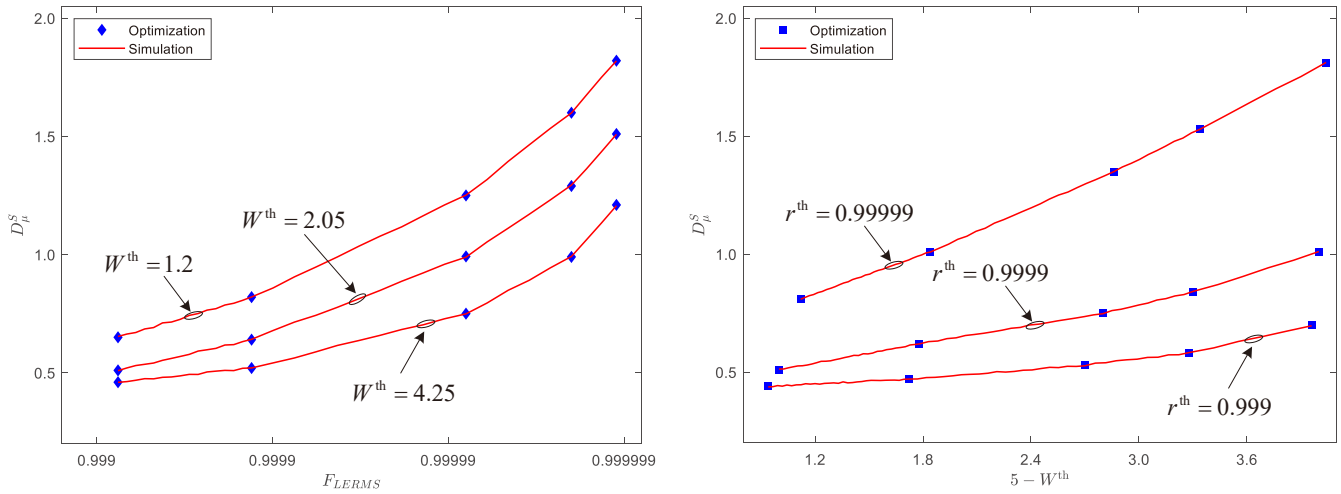
(**a**)

	$\bar{\Lambda}$	$\boldsymbol{\lambda} = [\lambda_{25}, \ldots, \lambda_{100}]$
Λ_1	27.5	[0.3, 0.15, 0.1, 0.05]
Λ_2	50.0	[0.2, 0.2, 0.2, 0.2]
Λ_3	72.5	[0.1, 0.15, 0.3, 0.4]

(**b**)

Figure 6. Effectiveness test of SOME-JE strategy to improve LERMS transmission efficiency. (**a**) Relationship between aggregation number L and padding probability $\mathbb{P}$. (**b**) Simulation parameter settings.

The LERMS strategy is designed to achieve an optimal trade-off between the average queuing delay, available bandwidth, and reliability, with ε_j set to 0.1 for all sessions, resulting in an upper limit of reliability of $1 - (0.1)^6 = 0.999999$. As shown in Figure 7, the resulting trade-off curve between latency and reliability is a segmented broken line that matches our theoretical analysis. Figure 7a demonstrates that as the reliability value F_{LERMS} increases, the required queuing delay D_{μ}^{S} also increases. Moreover, higher available bandwidth Wth can lead to lower queuing delays D_{μ}^{S} at a given reliability level r. Figure 7b shows that a higher reliability threshold rth requires a higher average queuing delay D_{μ}^{S}. These results illustrate the effectiveness of our LERMS strategy in achieving the optimal trade-off between delay and reliability.



(**a**) Impact of different *W*th on the result (**b**) Impact of different *r*th on the result

Figure 7. Optimal delay–reliability trade-off curves.

6. Conclusions and Future Directions

In this paper, we propose a low-latency and highly reliable transmission service for downlink traffic subscribed to edge services in untrusted edge network environments. To address potential failures, malicious tampering, and eavesdropping in the edge network, we introduce an encoder based on Lagrangian interpolation and Raptor double-layer cascading to fully utilize the multipath transmission resources of the edge network and provide secure CMT capabilities. Additionally, we design a variable block length encoding strategy that considers the accumulation of randomly arriving data packets at the sending end and selects an appropriate encoding block length based on queue length state information. We model this decision-making process as a Markov chain and obtain the optimal delay–reliability trade-off through matrix operation methods.

Effective communication and computing management is critical in the context of edge networks, where data is strongly related to the server flow. Currently, CMT is the primary focus, but it is essential to integrate distributed transmission and computing to meet the core requirements of edge networks. By distributing service flows based on computing requirements, we can enable the edge network to be more responsive to the service flow. This approach further enhances the core computing requirements of the edge network for corresponding scenarios, eliminates the single-point vulnerability of central cloud computing, and provides elastic edge network transmission and computing services. Our future research directions include exploring the integration of transmission and computing, multi-session transmission, and multi-DN computing in the edge network.

Author Contributions: Z.G.: Conceptualization, methodology, software, writing—original draft; X.J.: Formal analysis, project administration,supervision; W.Y.: Supervision, visualization; M.X.: Formal analysis, writing—review and editing; Y.Z.: Project administration; D.Z.: Supervision; Z.C.: Supervision; L.W.: Supervision; All authors have read and agreed to the published version of the manuscript.

Funding: Project supported by the National Key Research and Development Program of China (Nos. 2020YFB1806607 and 2022YFB2902204).

Data Availability Statement: The 5G Core Network that support this study are available from https://github.com/free5gc/free5gc (accessed on 1 February 2021).

Acknowledgments: We would like to thank the editors and the anonymous reviewers for their efforts.

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

Appendix A. Multiple PDU Sessions Transmission Procedure for Aggregation Data of Downlink Multi-Service Flow

In the scenario of multiple-device single-owners service flow aggregation transmission, we propose a procedure to support data aggregation over multiple PDU sessions. This is in addition to the existing physical isolation multi-PDU session establishment [27]. Currently, the core network does not support the joint transmission of multiple service flows. In this context, we propose the use of multi-PDU sessions to transmit the downlink data flow of edge computing services. This supports the transmission of aggregated traffic streams through physically isolated PDU sessions on a per-session basis.

1. The DN initiates the PDU session establishment request by sending a PDU Session Establishment Request message, which includes the requested DNN (identifier of the data flow from the core network to the UE), PDU session type, etc. The proposed procedure establishes a group of physically isolated multi-PDU sessions and assigns a session ID to this group of PDU sessions.
2. All devices within the same aggregated service flow must be assigned the same session ID.
3. Anchor UPF-A can identify the aggregated service flow through a specific session ID, facilitating unified management within the core network.
4. The local DN (MEC nodes) must perform the operation of inserting tags. Before the service flow is aggregated, the data owner is identified by adding a unique tag to the data packet.
5. After the device receives the aggregated data packet and completes decoding, it distinguishes the data according to the tag and extracts the data it needs. The rest of the data can be used to verified untrusted paths or discarded directly.

References

1. *Technical Specification (TS) 23.501*, version 17.6.0; System Architecture for the 5G System (5GS); 3GPP: Antibes, France, 2022.
2. Wu, H.; Xiang, Z.; Nguyen, G.T.; Shen, Y.; Fitzek, F.H. Computing meets network: Coin-aware offloading for data-intensive blind source separation. *IEEE Netw.* **2021**, *35*, 21–27. [CrossRef]
3. Feng, L.; Li, W.; Lin, Y.; Zhu, L.; Guo, S.; Zhen, Z. Joint computation offloading and URLLC resource allocation for collaborative MEC assisted cellular-V2X networks. *IEEE Access* **2020**, *8*, 24914–24926. [CrossRef]
4. Zhan, K. Sports and health big data system based on 5G network and Internet of Things system. *Microprocess. Microsyst.* **2021**, *80*, 103363. [CrossRef]
5. Javed, A.R.; Shahzad, F.; ur Rehman, S.; Zikria, Y.B.; Razzak, I.; Jalil, Z.; Xu, G. Future smart cities requirements, emerging technologies, applications, challenges, and future aspects. *Cities* **2022**, *129*, 103794. [CrossRef]
6. Adhikari, M.; Hazra, A. 6G-enabled ultra-reliable low-latency communication in edge networks. *IEEE Commun. Stand. Mag.* **2022**, *6*, 67–74. [CrossRef]
7. Ranaweera, P.; Jurcut, A.; Liyanage, M. MEC-enabled 5G use cases: A survey on security vulnerabilities and countermeasures. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–37. [CrossRef]
8. Tian, W.; Wang, G. State estimation in mobile edge computing with unreliable communications. *IEEE Commun. Lett.* **2020**, *25*, 1149–1152. [CrossRef]
9. *Technical Report (TR) 21.917*, version 17.0.1; Summary of Rel-17 Work Items; 3GPP: Antibes, France, 2023.
10. Deng, C.; Liu, D.; Yektakhah, B.; Sarabandi, K. Series-fed beam-steerable millimeter-wave antenna design with wide spatial coverage for 5G mobile terminals. *IEEE Trans. Antennas Propag.* **2020**, *68*, 3366–3376. [CrossRef]
11. Khan, B.S.; Jangsher, S.; Ahmed, A.; Al-Dweik, A. URLLC and eMBB in 5G Industrial IoT: A survey. *IEEE Open J. Commun. Soc.* **2022**, *3*, 1134–1163. [CrossRef]
12. Hui, H.; Ding, Y.; Shi, Q.; Li, F.; Song, Y.; Yan, J. 5G network-based Internet of Things for demand response in smart grid: A survey on application potential. *Appl. Energy* **2020**, *257*, 113972. [CrossRef]
13. Abd EL-Latif, A.A.; Abd-El-Atty, B.; Venegas-Andraca, S.E.; Mazurczyk, W. Efficient quantum-based security protocols for information sharing and data protection in 5G networks. *Future Gener. Comput. Syst.* **2019**, *100*, 893–906. [CrossRef]
14. Wu, Y.; Duong, T.Q.; Swindlehurst, A.L. Safeguarding 5G-and-beyond networks with physical layer security. *IEEE Wirel. Commun.* **2019**, *26*, 4–5. [CrossRef]
15. Kim, H. 5G core network security issues and attack classification from network protocol perspective. *J. Internet Serv. Inf. Secur.* **2020**, *10*, 1–15.
16. Fang, L.; Zhao, B.; Li, Y.; Liu, Z.; Ge, C.; Meng, W. Countermeasure based on smart contracts and AI against DoS/DDoS attack in 5G circumstances. *IEEE Netw.* **2020**, *34*, 54–61. [CrossRef]

17. Yoshizawa, T.; Baskaran, S.B.M.; Kunz, A. Overview of 5g urllc system and security aspects in 3gpp. In Proceedings of the 2019 IEEE Conference on Standards for Communications and Networking (CSCN), Granada, Spain, 28–30 October 2019; pp. 1–5.
18. Chen, R.; Li, C.; Yan, S.; Malaney, R.; Yuan, J. Physical layer security for ultra-reliable and low-latency communications. *IEEE Wirel. Commun.* **2019**, *26*, 6–11. [CrossRef]
19. Shrivastava, V.K.; Baek, S.; Baek, Y. 5G evolution for multicast and broadcast services in 3GPP release 17. *IEEE Commun. Stand. Mag.* **2022**, *6*, 70–76. [CrossRef]
20. Zhang, S.; Wang, Y.; Zhou, W. Towards secure 5G networks: A Survey. *Comput. Netw.* **2019**, *162*, 106871. [CrossRef]
21. Park, S.; Cho, H.; Park, Y.; Choi, B.; Kim, D.; Yim, K. Security problems of 5G voice communication. In Proceedings of the Information Security Applications: 21st International Conference, WISA 2020, Jeju Island, Republic of Korea, 26–28 August 2020; Revised Selected Papers 21; Springer: Berlin/Heidelberg, Germany, 2020; pp. 403–415.
22. Zhang, J.; Yang, L.; Cao, W.; Wang, Q. Formal analysis of 5G EAP-TLS authentication protocol using proverif. *IEEE Access* **2020**, *8*, 23674–23688. [CrossRef]
23. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
24. Chen, H.; Abbas, R.; Cheng, P.; Shirvanimoghaddam, M.; Hardjawana, W.; Bao, W.; Li, Y.; Vucetic, B. Ultra-reliable low latency cellular networks: Use cases, challenges and approaches. *IEEE Commun. Mag.* **2018**, *56*, 119–125. [CrossRef]
25. Arianpoo, N.; Aydin, I.; Leung, V.C. Network coding as a performance booster for concurrent multi-path transfer of data in multi-hop wireless networks. *IEEE Trans. Mob. Comput.* **2016**, *16*, 1047–1058.
26. Ha, J.; Choi, Y.I. Support of a multi-access session in 5g mobile network. In Proceedings of the 2019 25th Asia-Pacific Conference on Communications (APCC), Ho Chi Minh City, Vietnam, 6–8 November 2019; pp. 378–383.
27. Guo, Z.; Ji, X.; You, W.; Xu, M.; Zhao, Y.; Cheng, Z.; Zhou, D. Delay optimal for reliability-guaranteed concurrent transmissions with raptor code in multi-access 6G edge network. *Comput. Netw.* **2023**, *228*, 109716. [CrossRef]
28. Yu, Q.; Li, S.; Raviv, N.; Kalan, S.M.M.; Soltanolkotabi, M.; Avestimehr, S.A. Lagrange coded computing: Optimal design for resiliency, security, and privacy. In Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics, PMLR, Naha, Japan, 16–18 April 2019; pp. 1215–1225.
29. Abbas, R.; Shirvanimoghaddam, M.; Huang, T.; Li, Y.; Vucetic, B. Novel design for short analog fountain codes. *IEEE Commun. Lett.* **2019**, *23*, 1306–1309. [CrossRef]
30. Ström, E.G.; Popovski, P.; Sachs, J. 5G ultra-reliable vehicular communication. *arXiv* **2015**, arXiv:1510.01288.
31. Zhao, Y.; Wang, Q.; Qi, X.; Feng, L.; Gao, J.; Yu, P. Research on 5G Multipath Concurrent Transmission System and End to End Delay Measurement. In Proceedings of the 11th International Conference on Computer Engineering and Networks, Dalian, China, 21–22 October 2023.
32. Wang, M.; Liu, J.; Chen, W.; Ephremides, A. Joint queue-aware and channel-aware delay optimal scheduling of arbitrarily bursty traffic over multi-state time-varying channels. *IEEE Trans. Commun.* **2018**, *67*, 503–517. [CrossRef]
33. Stavrou, E.; Pitsillides, A. A survey on secure multipath routing protocols in WSNs. *Comput. Netw.* **2010**, *54*, 2215–2238. [CrossRef]
34. Dhelim, S.; Aung, N.; Kechadi, M.T.; Ning, H.; Chen, L.; Lakas, A. Trust2Vec: Large-scale IoT trust management system based on signed network embeddings. *IEEE Internet Things J.* **2022**, *10*, 553–562. [CrossRef]
35. Beimel, A. Secret-sharing schemes: A survey. In Proceedings of the Coding and Cryptology: Third International Workshop, IWCC 2011, Qingdao, China, 30 May–3 June 2011; Proceedings 3; Springer: Berlin/Heidelberg, Germany, 2011, pp. 11–46.
36. Zhang, P.; Pang, X.; Kumar, N.; Aujla, G.S.; Cao, H. A reliable data-transmission mechanism using blockchain in edge computing scenarios. *IEEE Internet Things J.* **2020**, *9*, 14228–14236. [CrossRef]
37. Luby, M. LT codes. In Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science, Vancouver, BC, Canada, 19 November 2002; pp. 271–271.
38. *Technical Specification (TS) 29.244*, version 18.0.1; Interface between the Control Plane and the User Plane Nodes; 3GPP: Antibes, France, 2022.
39. Halbawi, W.; Azizan, N.; Salehi, F.; Hassibi, B. Improving distributed gradient descent using reed-solomon codes. In Proceedings of the 2018 IEEE International Symposium on Information Theory (ISIT), Vail, CO, USA, 17–22 June 2018; pp. 2027–2031.
40. Mannweiler, C.; Gajic, B.; Rost, P.; Ganesan, R.S.; Markwart, C.; Halfmann, R.; Gebert, J.; Wich, A. Reliable and deterministic mobile communications for industry 4.0: Key challenges and solutions for the integration of the 3GPP 5G system with IEEE. In Proceedings of the Mobile Communication-Technologies and Applications, Osnabrueck, Germany, 15–16 May 2019; 24. ITG-Symposium. VDE; pp. 1–6.
41. jay16213, free5gc-org, turtle11311. free5GC. 2022. Available online: https://github.com/free5gc/free5gc (accessed on 3 May 2023).