

Research Article

BSS: A Burst Error-Correction Scheme of Multipath Transmission for Mobile Fog Computing

Wentao Zhao ¹, Ping Dong ¹, Min Guo ², Yuyang Zhang ¹ and Xuehong Chen³

¹School of Electronic and Information Engineering, Beijing Jiaotong University, China

²School of Software Engineering, Beijing Jiaotong University, China

³Support Technology Institute China Industrial Control Systems Cyber Emergency Response Team, China

Correspondence should be addressed to Ping Dong; pdong@bjtu.edu.cn

Received 24 April 2020; Revised 23 May 2020; Accepted 16 June 2020; Published 30 June 2020

Academic Editor: Fuhong Lin

Copyright © 2020 Wentao Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the scenario of mobile fog computing (MFC), communication between vehicles and fog layer, which is called vehicle-to-fog (V2F) communication, needs to use bandwidth resources as much as possible with low delay and high tolerance for errors. In order to adapt to these harsh scenarios, there are important technical challenges concerning the combination of network coding (NC) and multipath transmission to construct high-quality V2F communication for cloud-aware MFC. Most NC schemes exhibit poor reliability in burst errors that often occur in high-speed movement scenarios. These can be improved by using interleaving technology. However, most traditional interleaving schemes for multipath transmission are designed based on round robin (RR) or weighted round robin (WRR), in practice, which can waste a lot of bandwidth resources. In order to solve those problems, this paper proposes a novel multipath transmission scheme for cloud-aware MFC, which is called Bidirectional Selection Scheduling (BSS) scheme. Under the premise of realizing interleaving, since BSS can be used in conjunction with a lot of path scheduling algorithms based on Earliest Delivery Path First (EDPF), it can make better use of bandwidth resources. As a result, BSS has high reliability and bandwidth utilization in harsh scenarios. It can meet the high-quality requirements of cloud-aware MFC for transmission.

1. Introduction

Mobile fog computing (MFC) extends cloud computing by adding a fog layer between vehicles and cloud, which has lower transmission delay and less traffic burden [1–3]. Unlike mobile edge computing (MEG), since MFC is able to be aware of the centralized cloud, it can jointly work with the centralized cloud [4]. However, during the communication between vehicles and fog layer, especially for delay-sensitive users, data loss greatly affects the quality of experience [5]. In order to meet the high-quality service requirements in mobile scenarios, it is important to construct vehicle-to-fog (V2F) communication with high reliability and bandwidth utilization for cloud-aware MFC. V2F communication requires a large amount of data to be quickly transmitted [6, 7]. It needs to use bandwidth resources with lower delay and higher tolerance for errors.

Considerable researches have proved that multipath transmission can significantly improve the throughput of V2F communication [8–10]. Scheduling algorithm is one of the significant factors that affect the bandwidth utilization of multipath transmission. A simple scheduling algorithm is round robin (RR) [11], which treats each path equally. Nevertheless, RR does not consider the practical status of each path. It can waste a lot of bandwidth resources of the entire transmission. Weighted round robin (WRR) pre-sets a weight on each path which indicates the status of bandwidth, packet loss, and delay. Therefore, WRR is more widely used than RR. Compared with RR and WRR, Earliest Delivery Path First (EDPF) [12] performs much better. This approach improves bandwidth utilization by estimating the path status in real-time. Earliest Completion First (ECF) [13] is one of the current scheduling algorithms based on EDPF, which considers all relevant information about

multipath transmission. It has higher bandwidth utilization and lower delay.

During the transmission process, especially in harsh scenarios, data loss often occurs. It can reduce the quality of transmission. Network coding (NC) is proved to be an effective scheme, which can significantly improve the reliability and bandwidth utilization of multipath transmission [14]. The NC scheme can be regarded as a packet-level error-correction scheme. It improves reliability by increasing redundancy and reduces the times of retransmission when encountering data loss. We previously proposed a BNNC scheme [15, 16] which works in the set of integers. Compared with traditional NC schemes, it improves the efficiency of encoding and decoding.

However, most researches about NC are designed for Additive white Gaussian noise (AWGN) channels, which have good performance to correct random errors [17]. In V2F communication, due to the hand-off of network or channel obstruction, there are also a large number of burst errors apart from random errors [18]. These schemes designed for AWGN exhibit poor performance in burst error scenarios. To improve the reliability of the burst error scenarios, interleaving is often used together with an error-correction scheme. Interleaving is a way of rearranging, which is usually employed to randomize the burst errors so that the error-correction scheme could be more effective. Huang et al. [19] first built a connection between interleaving and NC, which shows that interleaving can effectively deal with burst errors. Moon and Kim [20] proposed a Network-Adaptive Selection of Transport Error Control which can switch among different error schemes including forward error correction (FEC), interleaving, and automatic repeat request (ARQ).

Traditional interleaving schemes for multipath transmission are based on RR or WRR [21], which can waste a lot of bandwidth resources. Furthermore, because of the limitation of RR or WRR, these schemes can only maintain a systematic interleaving distance. It introduces extra coding delay and significantly affects the experience of cloud-aware MFC. Few studies have been performed well over the multipath transmission.

In this paper, we propose a novel burst error-correction scheme, Bidirectional Selection Scheduling (BSS), which has high reliability and bandwidth utilization of multipath transmission for cloud-aware MFC. BSS is a way of rearranging process rather than a way of coding schemes, so it is flexible enough to be compatible with most of the existing technology. BSS can implement the multipath interleaving process with high bandwidth utilization and maintain a path-level interleaving distance. The contributions of this paper can be summarized as follows:

- (i) We propose a novel scheduling algorithm, which implements the process of interleaving over multipath transmission
- (ii) We accomplish a complete burst error-correction scheme, BSS, of multipath transmission for cloud-aware MFC

- (iii) Numerous simulations have proved the feasibility of BSS. It has better reliability and bandwidth utilization than other NC scheme

The remainder of this paper is organized as follows. Section 2 reviews some related work about burst error-correction schemes. Section 3 introduces the implementation of BSS. Section 4 verifies the feasibility of BSS through simulations. Finally, Section 5 concludes the investigation.

2. Related Work

In this section, we mainly focus on some different types of burst error-correction schemes.

2.1. NC Scheme by Adding Redundancy. NC scheme was firstly designed to improve the throughput of multicast networks. FMTCP applies NC to MPTCP, which improves the reliability of multipath transmission by adding redundancy [22]. NC (n, k) can be regarded as a packet-level error-correction scheme. At the sending end, the original k packets are encoded to n packets by adding r redundant packets. It is possible to recover the original packets as long as k packets are received rather than all n packets at the receiving end. The error-correction capability of NC is related to the length of redundancy r .

To improve the capability of burst error-correction, increasing the length of redundancy r is a useful method. Our previous research shows that simply increasing redundancy can increase the overall system overhead and reduce bandwidth utilization. Merging the original i coding groups of NC (n, k) into one big coding group NC $(n \times i, k \times i)$ can increase the redundancy without losing extra bandwidth utilization. It can improve the capability of burst error-correction by i times.

However, this type of scheme itself is a single-path solution rather than a multipath solution. The error-correction capability of these schemes is unstable under different multipath scheduling algorithms. And the main problem is the performance. The increasing of redundancy can greatly increase the computational complexity, though the total encoding length does not change. Furthermore, in some cases, it may expand the impact of packet loss. If the length of burst errors is too long to the new redundancy $i \times r$, this scheme can cause the loss of all $i \times k$ original packets.

2.2. Interleaving Coding Scheme. Path-dependent models [23–26] are usually used with interleaving to improve the performance in burst errors, which is called interleaving coding. Huang et al. [19] first built a connection between interleaving and network coding, which shows that interleaving can effectively deal with burst errors. These models send the information packets to different paths and then encode those packets on each path [27]. Interleaving coding disrupts the order of packets on each path and converts burst errors into random errors.

This type of scheme can be seen as a combination of many single-path transmissions. Each path maintains its coding scheme which means different paths have different lengths of

TABLE 1: Analysis of different solutions.

Scheme	NC scheme by adding redundancy	Interleaving coding scheme [28]	Path interleaving scheme [21]	BSS
Computational complexity	High	Low	Low	Low
Burst error constraint	System level	Path level	System level	Path level
Robust in multipath transmission	×	×	×	√
Robust if there are long errors	×	×	×	√
Robust if there is a poor path	×	×	×	√

redundancy and interleaving distance. Compared with systematic interleaving distance, it has lower average delay.

However, this type of scheme does not take full advantage of the characteristics of the multipath transmission. Even though most paths are very healthy, it can seriously affect the bandwidth utilization of the entire system, once a long length of burst errors occurs on one path.

2.3. Path Interleaving Scheme. Multipath transmission techniques have been designed to improve the throughput of end-to-end transmission [28]. In addition, because the state of each path is independent, multipath transmission can spread packets on each path, thereby reducing the packet loss rate caused by a certain path. Path interleaving schemes consider the independence of each path in multipath transmission. They scatter the packets of the same coding group on different paths to reduce the correlation of transmission [29].

Giovambene et al. [30] proposed a Path-Based Network Coding (PBNC) scheme. It focuses on the distribution of data packets on multipath, which can improve the performance in independent path scenarios. This type of scheme first encodes the information packets, then sends both information packets and redundancy packets over multiple paths. Since the packets of the same coding group may be scattered on different paths, it is effective in dealing with packet loss on one path. Tsai et al. has proposed multipath transmission combining forward error-correction and path interleaving, which breaks up the packets on the respective path through the RR [21]. It has higher reliability than most schemes in burst scenarios.

In fact, the RR algorithm can be dragged down by the worst path. In addition, this scheme only maintains a systematic distance of interleaving. For each path, an excessively long interleaving distance can degrade the performance of the system, and an excessively short interleaving distance can reduce the reliability of the system.

3. Bidirectional Selection Scheduling Scheme

Before the introduction of BSS, we summarize the limitation of these solutions in Table 1 which are proposed in Section 2. Increasing the redundancy length of NC scheme is not a multipath solution. Furthermore, it has a computation issue and low robust which can cause a huge impact of long burst errors. Both interleaving coding schemes and path interleaving schemes have a major issue in performance. Extra average coding delay and a poor path can seriously drag down overall throughput.

In this paper, we propose an improved adaptive burst error-correct scheme, BSS, which applies to multipath transmission for cloud-aware MFC. BSS does not add extra computational overhead. It is very robust in both long burst errors and separate poor path. Compared with most of the burst error-correction schemes, it has higher reliability and bandwidth utilization for multipath transmission.

3.1. The Network Topology of BSS. The process of transmission is symmetrical. In this paper, we introduce BSS by taking the uplink traffic from users to fog layer and centralized cloud. Figure 1 shows the network topology of BSS deployed in V2F communication. The topology is composed of user layer, multipath network, fog layer, and centralized cloud. In the user layer, during the motivation process, users forward data to the Internet through mobile router (MR) to the different base-stations. MR has the function of accessing heterogeneous networks of different operators, so the model can be regarded as concurrent multipath transmission [31]. In the fog layer, fog nodes are responsible for recovering the original data from multipath network and processing with virtual machines (VM). During the data transmission from the user layer to the fog layer, there may be burst errors. In order to reduce the impact of burst errors on the system, BSS is designed as an end-to-end burst error-correction scheme.

We deploy the encoding and decoding schemes in MR and fog nodes, respectively. In MR, users' data are mapped into packets through a mapping module. Network encoder implements NC, which adds redundant packets to improve the reliability of random errors. Then, the information packets and redundant packets are transmitted through BSS which is deployed on a bidirectional selector. The bidirectional selector implements the interleaving process through the mutual selection mechanism of paths and packets. It improves the reliability of burst errors. Furthermore, network interface controller (NIC) sender can access heterogeneous networks of different operators to achieve parallel transmission. Fog nodes recover the received packets to the source data through receive buffer, deinterleaver, network decoder, and mapping module and then process source data with virtual machines. Finally, fog nodes only need to forward the important results to the centralized cloud. Furthermore, the link management modules are deployed both in MR and fog nodes. They can detect the status of the entire network transmission in real-time and dynamically adjust the parameters of the bidirectional selector.

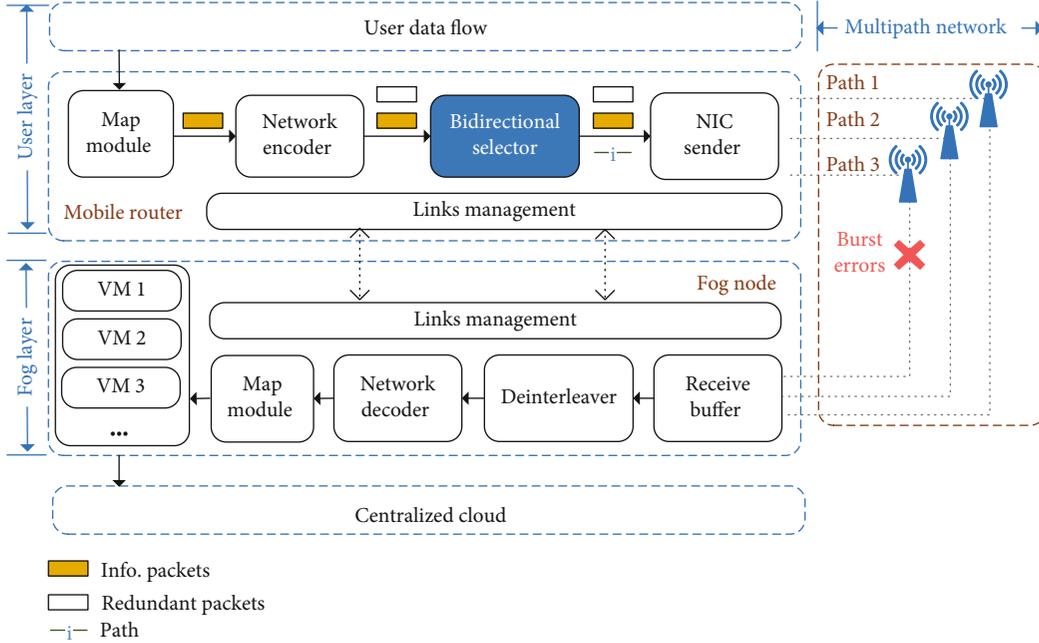


FIGURE 1: The topology of Bidirectional Selection Scheduling scheme.

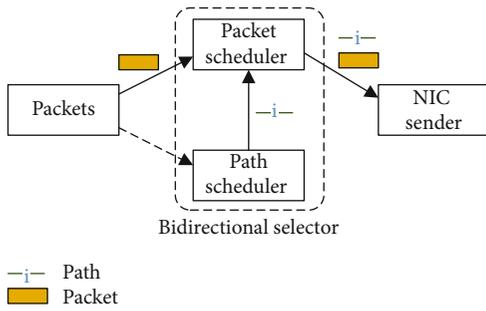


FIGURE 2: Design of bidirectional selector.

3.2. Design of Bidirectional Selector Module. BSS implements the process of interleaving over multipath transmission. Interleaving technology is to disperse the original continuous and related packets, thereby converting continuous burst errors into random errors, and then relying on the error-correction capability of NC to recover the data. Unlike traditional multipath interleaving schemes based on the RR algorithm, BSS can work with any path scheduling algorithm.

The traditional scheme of multipath transmission consists of a first input first output buffer and a path scheduler. The path scheduler only serves the first packet in the buffer. Its essence is a pathfinding process of the first packet, which guarantees that the first packets can always be sent on the most suitable path. BSS adds a module of packet scheduler and redesigns the structure of buffer which can provide packet selection service. Packet scheduler cooperates with the path scheduler to complete the bidirectional selection of packets and paths. Therefore, BSS can ensure that packets are sent not only in the most suitable path but also at the most suitable time.

As shown in Figure 2, in order to implement BSS, we use a bidirectional selector to customize the degree of discreteness of

packets. The bidirectional selector mainly consists of a packet scheduler and a path scheduler. The packet scheduler can temporarily store the packets to be sent. Packets first enter into the send buffer in the packet scheduler for temporary storage and at the same time trigger the path scheduler to get the appropriate path. After obtaining the path, BSS makes the second selection for the appropriate packet to be sent in the send buffer. This process disrupts the order of packets on each path and implements the process of interleaving.

3.3. Mathematical Analysis of BSS. Interleaving distance is the most important mathematical parameter of the capability of burst error correction. Considering the characteristics of multipath transmission and NC, we make a few assumptions:

- (i) The status of each path is independent
- (ii) The packets of different network coding groups are independent

The selection of packet to path mainly depends on the implementation of the path scheduling algorithm, so we only need to analyze the selection algorithm of the path to packet. Multipath transmission allows that packets, which are encoded by NC (n, k) , can be sent through multiple paths concurrently. Although the amount of the same NC group is smaller on each path, the packet loss constraint of each path is not changed. The packet loss constraint of burst errors is not greater than the length of redundancy $r = n - k$. Interleaving can improve it by the rearranging process. Therefore, the burst packet loss constraints C_{burst} for each path are

$$C_{burst} = \begin{cases} r, & \text{without interleaving,} \\ r \times d, & \text{with interleaving,} \end{cases} \quad (1)$$

where d means the minimum distance between adjacent packets after interleaving. Formula (1) shows that C_{burst} is only related to the length redundancy r and the distance of interleaving d . Since r is determined by network coding, the key is how to design and implement the distance of interleaving d . We define the length of burst errors as s and the burst loss rate of each packet as $B\text{Loss}$. When interleaving is not used, the probability of successfully recovering a group of encoded packets:

$$\beta_0 = (1 - B\text{Loss})^{s-k+1}. \quad (2)$$

When using interleaving technology and $D \geq s$, burst errors become to random errors. The probability of successful decoding is

$$\beta_1 = \sum_{i=k}^n C_n^i \alpha^{n-i} (1 - \alpha)^i, \quad (3)$$

where α means the overall packet loss rate. If $d \leq s$, the burst errors become to $\lceil s/d \rceil$. The probability of successful decoding is

$$\beta_2 = (1 - B\text{Loss})^{\lceil s/d \rceil - k + 1}. \quad (4)$$

We can see that formula (2) is a special case of formula (4) when $d = 1$.

In general, we observe the path for a period of time and get the overall packet loss rate α . Formula (3) and formula (4) can be expressed as

$$\beta = \begin{cases} (1 - \alpha)^{\lceil s/d \rceil - k + 1/s}, & d < s, \\ \sum_{i=k}^n C_n^i \alpha^{n-i} (1 - \alpha)^i, & d \geq s. \end{cases} \quad (5)$$

The formula (5) shows that as long as $d \geq s$, the probability of successful recovering can reach the highest value. Therefore, it is necessary to satisfy that any consecutive d packets in each path are independent.

3.4. Implementation of the Core Algorithm. BSS provides a path-level interleaving distance. Each path can maintain its own interleaving distance to reduce the performance issue impact of interleaving. The core algorithm of BSS is shown in Algorithm 1. Steps 1 to 3 first initialize the interleaving distance of each path with the feedback from link management. Then, each path initializes a collision buffer with the same length to the interleaving distance of the path. BSS needs to monitor the status of packets in the send buffer. When the send buffer is not empty, the path scheduler gives the most suitable path at this time according to the configured path scheduling algorithm. Then, BSS triggers the packet scheduler to look for the packet that meets the interleaving distance of the selected path. After obtaining the appropriate packet and path, BSS can complete the operation of sending and updating the send buffer and collision buffer. In the worst case, there is no suitable packet in

```

1: Foreach Path in Multipath do
2:   UpdateInterleavingDistance(Path);
3:   InitPathBuffer(Path);
4: End foreach
5: While SendBuffer.isNotEmpty() do
6:   Path = PathScheduler();
7:   If timeout() then
8:     Packet = GetFirstPacket();
9:   Else
10:    Packet = PacketScheduler(Path);
12:  End if
13:  If Packet.isNull() then
14:    Break;
15:  End if
16:  Send(Packet, Path);
17:  Update(SendBuffer);
18:  Update(Threshold);
19:  Update(PathBuffer);
20: End while

```

ALGORITHM 1: Core algorithm of BSS scheme.

the send buffer, when it needs. At this time, it is possible to send directly or continue to wait. Since interleaving distance reflects burst errors, ideally, as long as the interleaving distance is sufficiently accurate, packets are always lost at this time. In practical terms, we generally set a timeout threshold. Packets with a waiting time exceeding the threshold can be sent immediately.

4. Performance and Discussion

In order to quantitatively analyze the buffer errors in V2F, we simulate the communication between vehicles and cloud-aware MFC through NS3. Firstly, we introduce the burst error model used in the simulation. Secondly, we introduce the design of simulation for V2F communication including simulation topology and parameters. Finally, we analyze and compare the performance of different schemes under burst error scenarios, which proves the feasibility of BSS.

4.1. Burst Error Model. We use a two-state Markov model, Gilbert-Elliot model, to describe the burst error property. This model is proposed by Gilbert and Elliot [32]. The path state is divided into two states: good (G) and bad (B). In the G state, the path presents high reliability, which has a low loss probability. However, in the B state, the packets are hardly received.

In order to simplify the burst error model, we assume that the packet loss rate in the G state is 0 and the packet loss rate in the B state is 100%. Furthermore, we set the initial state to G. We uniformly define the first subscript to represent the original state and the last one to represent the converted state. The transition probability from G to B is P_{gb} , and the transition probability from B to G is P_{bg} . Therefore, we can change the burst error model to simulate different scenarios by modifying the values of P_{gb} and P_{bg} .

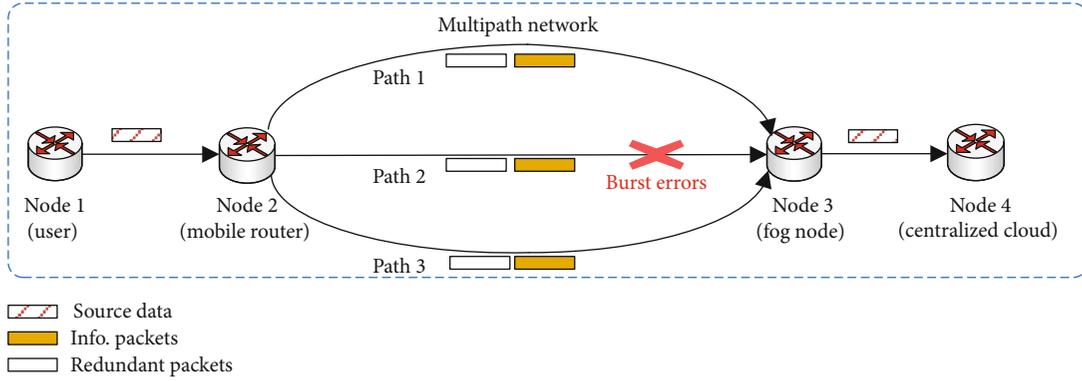


FIGURE 3: Simulation topology.

TABLE 2: Analysis of different solutions.

Network parameters	Description
Number of paths	3
Error model	Gilbert-Elliot model
Network coding scheme	BNNC (4,3)
Scheduling algorithm	ECF

TABLE 3: Analysis of different solutions.

Paths	Bandwidth	RTT	Average loss	RTT
Paths 1	5 Mbps	24 ms	0.1%	×
Paths 2	12 Mbps	44 ms	1%, 5%, and 10%	✓
Paths 3	10 Mbps	40 ms	0.5%	×

4.2. Design of Simulation for V2F Communication. The simulation topology is shown in Figure 3. We use four nodes to form a minimum multipath transmission model for cloud-aware MFC. Node 1 (user) sends the source data to node 2 (mobile router) which is deployed on the vehicle. Node 2 encodes and interleaves the source data; then, it sends both information and redundant packets to node 3 (fog node) through three parallel paths. After receiving enough packets, node 3 recovers the original data and forwards the results to node 4 (centralized cloud).

It is worth noting that the paths between node 2 and node 3 may experience burst errors. We set the encoding scheme and decoding scheme on node 2 and node 3, respectively, to reduce the impact of burst errors on the system. In order to simulate different scenarios, we set different path states and change burst error models to verify the feasibility of BSS.

Table 2 shows the simulation parameters. Node 1 generates source packets. BNNC (n, k) is used to be the error-correction scheme, and it is also used as the benchmark. We fix the parameter (n, k) to $(4, 3)$. In order to simulate the burst errors, we use the Gilbert-Elliot model to generate packet loss. As shown in Table 3, we set three different paths. There is no burst error on Path 1 and Path 3. We simulate three different burst packet loss rates on Path 2, which are 1%, 5%, and 10%. Based on formula (5), we set the interleaving distance of each path equal to the continuous packet loss length on it.

4.3. Validation and Analysis. Firstly, Figure 4 compares the decoding failure ratio between BSS and NC when Path 2 has burst errors. Decoding failure ratio means the probability of a coding group that cannot be decoded; that is, more than $r = n - k$ packets of the same group are lost in transmission. When the average length of burst errors is equal to 1, which can be seen as an ideal random error scenario, both of the two schemes hardly have decoding failure and information lost, because one single packet loss in one coding group can always be corrected by NC. When the burst error length is greater than 1, the error model may generate continuous packet loss in one coding group exceeding packet loss constraints r . NC exhibits poor reliability with the growth of the average length of burst errors. Since BSS uses interleaving, it converts continuous burst errors to random errors. Although the decoding failure ratio of BSS also increases when the average length is equal to 2, its subsequent changes are very slow. When in the burst error scenarios (the average length of burst error is greater than 1), the decoding failure ratio of BSS is much lower than NC, or even only half of NC. It indicates that BSS has higher reliability than NC in different scenarios. Reliability has a great impact on the performance of the entire system. The lower decoding failure ratio also represents fewer retransmissions, which means a higher bandwidth utilization. Therefore, BSS is far superior to other schemes in reliability, and it can also improve overall performance greatly.

Secondly, Figure 5 shows the real-time throughput of BSS, BNNC, and path interleaving scheme when Path 2 has 5% average packet loss and the average length of burst error is 5. It indicates that the real-time throughput of NC fluctuates sharply and shows a downward trend of average throughput. However, the real-time throughput of both BSS and path interleaving are relatively stable. It indicates that both BSS and path interleaving are less affected by burst errors. It is noting that path interleaving cannot make full use of the bandwidth of each path. With the setting of Table 3, the RR algorithm limits the bandwidth of each path to less than 5 Mbps which is equal to the bandwidth of Path 1. Compared with NC and path interleaving, despite the burst errors, the real-time throughput of BSS was maintained at a high value, which means BSS has a higher bandwidth utilization.

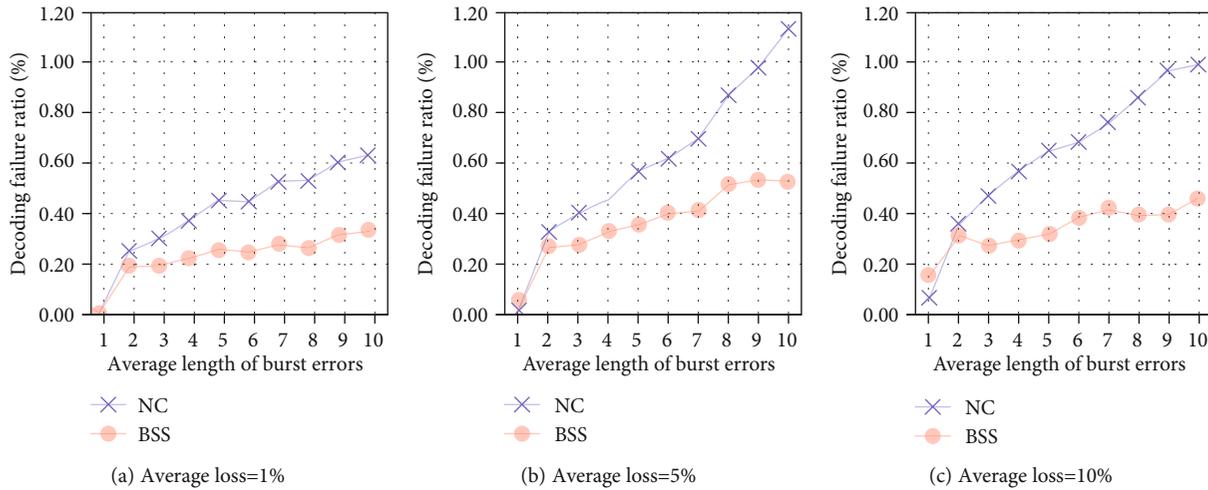


FIGURE 4: Comparison of decoding failure ratio (note: Path 2 has burst errors).

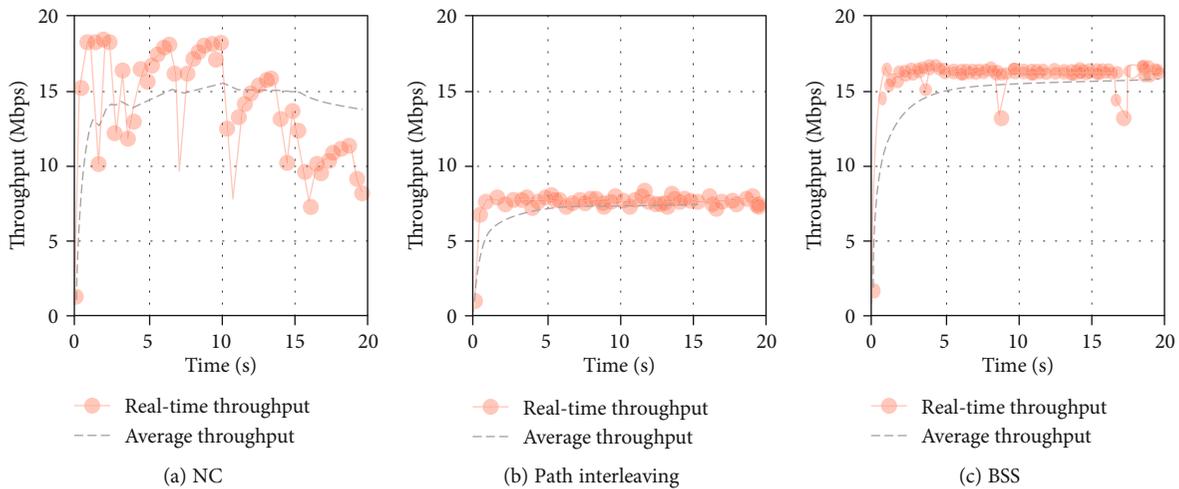


FIGURE 5: Comparison of real-time throughput (note: Path 2 has burst errors. The average packet loss rate is 1% and the average length of burst error is 5).

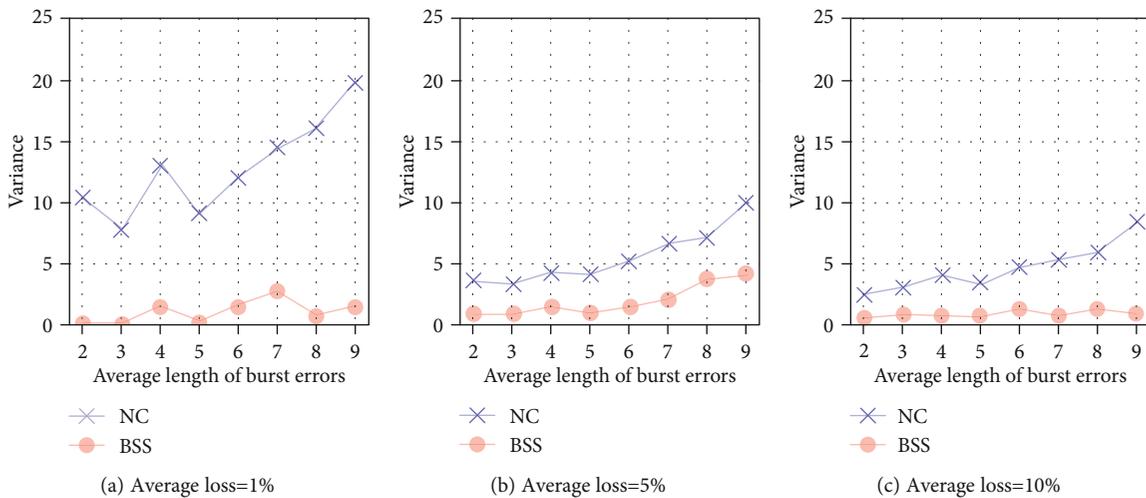


FIGURE 6: Comparison of variance in burst error scenarios with different lengths of burst errors (note: Path 2 has burst errors).

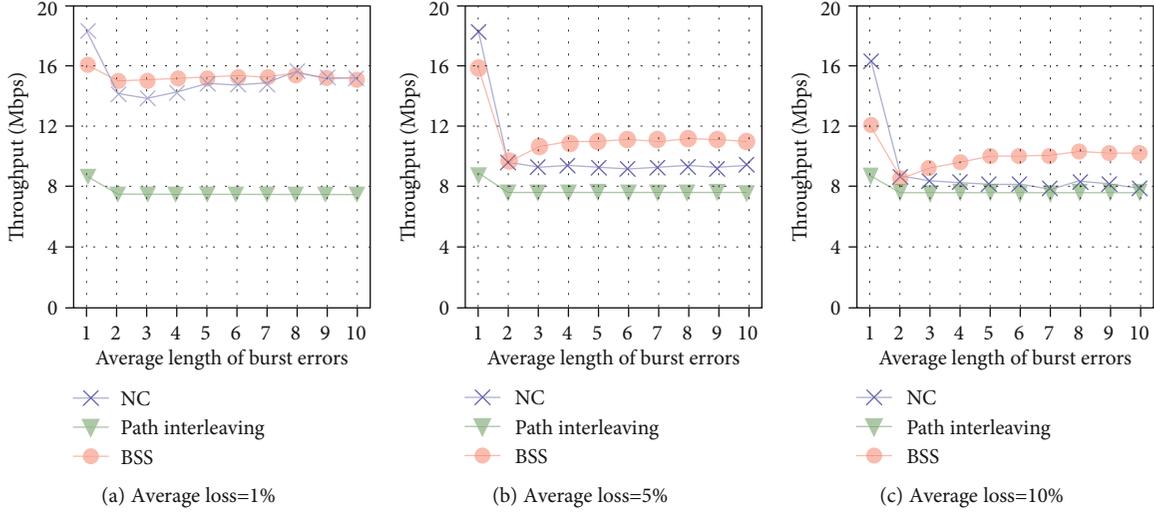


FIGURE 7: Comparison of average throughput with different lengths of burst errors (note: Path 2 has burst errors).

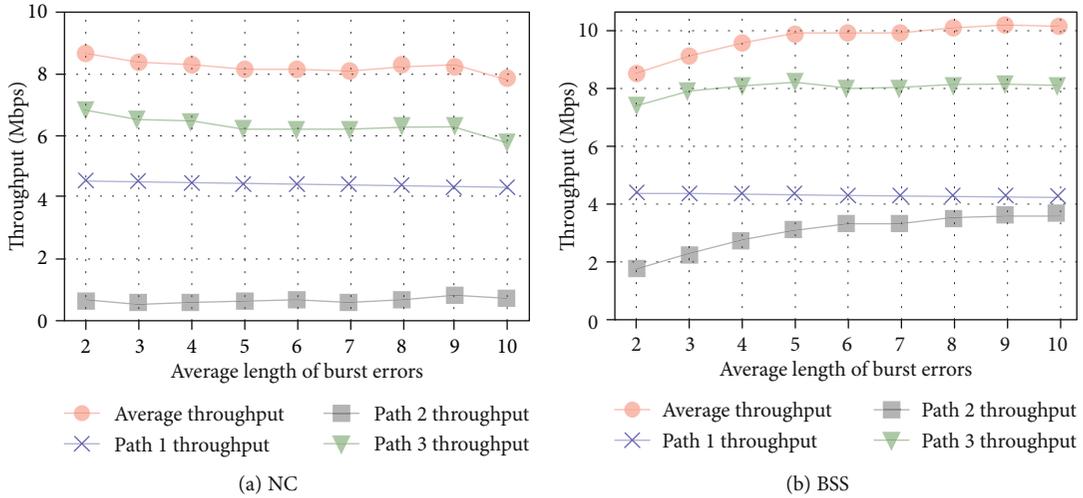


FIGURE 8: Comparison of average throughput on each path in burst error scenarios with different lengths of burst errors (note: Path 2 has burst errors and the average packet loss rate is 5%).

Finally, we compare the bandwidth utilization of BSS and NC in terms of statistical indicators. Figure 6 shows the variance between BSS and NC. In the simulation, the variance of NC increases with the length of burst errors, but BSS is relatively stable and maintains a low value. Compared with NC, BSS has a lower variance, which means that its real-time throughput is more stable and smooth. Especially in the case of low packet loss rate, BSS is much better than NC. Figure 7 shows the average throughput of NC, path interleaving, and BSS with different lengths of burst errors. Similar to the decoding failure ratio, NC performs better when the average length of burst errors is equal to 1 which can be seen as an ideal random error scenario. However as the average length of burst errors increases to 2, the throughput of NC reduces rapidly. When the average loss is equal to 10%, the average throughput of NC is even similar to path interleaving which has been proven to lose a lot of bandwidth resources. In the case of high continuous packet loss, compared to NC, BSS can increase throughput by up to nearly 20%. Figure 8 shows

the average throughput of different paths in burst error scenarios (average length of burst error is greater than 1) when the average packet loss rate is 5%. Although Path 1 and Path 3 still maintain high throughput, Path 2 drags down the average throughput of NC. Furthermore, since Path 2 is in a state of packet loss for a long time, TCP reduces the transmission speed of this path, and most path scheduling algorithms lower the weight of this path at the same time, thereby greatly reducing the bandwidth utilization of this path. In the same situation, no matter what the average length of burst errors is, BSS can make better use of bandwidth resources of Path 2. It indicates that in the burst error scenarios, BSS performs much better than NC.

5. Conclusion

In this paper, we propose a novel burst error-correction scheme of V2F communication for cloud-aware MFC. The scheme uses a Bidirectional Selection Scheduling

algorithm to implement the multipath interleaving process. It can maintain an adaptive path-level interleaving distance. Finally, we did a lot of simulations. The results show that BSS has a lower packet loss rate and higher bandwidth utilization in burst error scenarios. In the case of high continuous packet loss, compared to NC, BSS can reduce the decoding failure ratio by 50% and increase throughput by up to nearly 20%. Therefore, it is concluded that BSS can cope with burst error scenarios for cloud-aware MFC. In the future, we will focus analyze the status of multipath burst errors and optimize the interleaving distance of the scheme.

Data Availability

The authors confirm that the data supporting the findings of this study are available within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61872029 and in part by the Beijing Municipal Natural Science Foundation under Grant 4182048.

References

- [1] S. Li and W. Sun, "Utility maximisation for resource allocation of migrating enterprise applications into the cloud," *Enterprise Information Systems*, pp. 1–33, 2020.
- [2] C. Gong, F. Lin, X. Gong, and Y. Lu, "Intelligent cooperative edge computing in the internet of things," *IEEE Internet of Things Journal*, p. 1, 2020.
- [3] H. Hui, C. Zhou, S. Xu, and F. Lin, "A novel secure data transmission scheme in industrial internet of things," *China Communications*, vol. 17, no. 1, pp. 73–88, 2020.
- [4] F. Lin, Y. Zhou, X. An, I. You, and K.-K. R. Choo, "Fair resource allocation in an intrusion-detection system for edge computing: ensuring the security of internet of things devices," *IEEE Consumer Electronics Magazine*, vol. 7, no. 6, pp. 45–50, 2018.
- [5] L. F. Bittencourt, J. Diaz-Montes, R. Buyya, O. F. Rana, and M. Parashar, "Mobility-aware application scheduling in fog computing," *IEEE Cloud Computing*, vol. 4, no. 2, pp. 26–35, 2017.
- [6] L. Liu, Z. Chang, X. Guo, S. Mao, and T. Ristaniemi, "Multiobjective optimization for computation offloading in fog computing," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 283–294, 2018.
- [7] P. Dong, T. Zheng, X. Du, H. Zhang, and M. Guizani, "SVCC-HSR: providing secure vehicular cloud computing for intelligent high-speed rail," *IEEE Network*, vol. 32, no. 3, pp. 64–71, 2018.
- [8] Y. Zhang, P. Dong, S. Yu, H. Luo, T. Zheng, and H. Zhang, "An adaptive multipath algorithm to overcome the unpredictability of heterogeneous wireless networks for high-speed railway," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 12, pp. 11332–11344, 2018.
- [9] Z. Haitao, D. Yi, Z. Mengkang, W. Qin, S. Xinyue, and Z. Hongbo, "Multipath transmission workload balancing optimization scheme based on mobile edge computing in vehicular heterogeneous network," *IEEE Access*, vol. 7, pp. 116047–116055, 2019.
- [10] L. Li, K. Xu, T. Li et al., "A measurement study on multi-path TCP with multiple cellular carriers on high speed rails," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, pp. 161–175, Budapest, Hungary, August 2018.
- [11] M. Li, A. Lukyanenko, Z. Ou et al., "Multipath transmission for the internet: a survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2887–2925, 2016.
- [12] K. Chebrolu, B. Raman, and R. R. Rao, "A network layer approach to enable tcp over multiple interfaces," *Wireless Networks*, vol. 11, no. 5, pp. 637–650, 2005.
- [13] Y. Lim, E. M. Nahum, D. Towsley, and R. J. Gibbens, "ECF: an MPTCP path scheduler to manage heterogeneous paths," in *Proceedings of the 2017 ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, pp. 33–34, Illinois, USA, June 2017.
- [14] G. Giambene, D. K. Luong, V. A. Le, T. de Cola, and M. Muhammad, "Transport layer performance combining multipath and network coding in mobile satellite networks," *International Journal of Satellite Communications and Networking*, vol. 35, no. 6, pp. 583–603, 2017.
- [15] Y. Zhang, P. Dong, Y. Yu et al., "A bignum network coding scheme for multipath transmission in vehicular networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 206–212, Abu Dhabi, United Arab Emirates, December 2018.
- [16] Y. Zhang, P. Dong, X. Du, H. Luo, T. Zheng, and M. Guizani, "Bnnc: improving performance of multipath transmission in heterogeneous vehicular networks," *IEEE Access*, vol. 7, pp. 158113–158125, 2019.
- [17] S. Yu, M. Liu, W. Dou, X. Liu, and S. Zhou, "Networking for big data: a survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 531–549, 2017.
- [18] Y. Zhang, T. Zheng, P. Dong, H. Luo, and Z. Pang, "Comprehensive analysis on heterogeneous wireless network in high-speed scenarios," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 4259510, 12 pages, 2018.
- [19] Z. Huang, X. Wang, X. Chen, and H. Kan, "Network coding with interleaving," in *2007 International Conference on Parallel Processing Workshops (ICPPW 2007)*, pp. 43–43, Xian, China, September 2007.
- [20] S. Moon and J. Kim, "Network-adaptive selection of transport error control (naste) for video streaming over wlan," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 4, pp. 1440–1448, 2007.
- [21] M. F. Tsai, N. K. Chilamkurti, S. Zeadally, and A. Vinel, "Concurrent multipath transmission combining forward error correction and path interleaving for video streaming," *Computer Communications*, vol. 34, no. 9, pp. 1125–1136, 2011.
- [22] Y. Cui, L. Wang, X. Wang, H. Wang, and Y. Wang, "FMTCP: a fountain code-based multipath transmission control protocol," *IEEE/ACM Transactions on Networking*, vol. 23, no. 2, pp. 465–478, 2015.

- [23] M. Wu, S. Makharia, H. Liu, D. Li, and S. Mathur, "Iptv multicast over wireless lan using merged hybrid arq with staggered adaptive fec," *IEEE Transactions on Broadcasting*, vol. 55, no. 2, pp. 363–374, 2009.
- [24] S. Mao, S. Lin, Y. Wang, S. S. Panwar, and Y. Li, "Multipath video transport over ad hoc networks," *IEEE Wireless Communications*, vol. 12, no. 4, pp. 42–49, 2005.
- [25] P. Djukic and S. Valaee, "Reliable packet transmissions in multipath routed wireless networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 5, pp. 548–559, 2006.
- [26] V. R. Gandikota, B. R. Tamma, and C. Murthy, "Adaptive fec-based packet loss resilience scheme for supporting voice communication over ad hoc wireless networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 10, pp. 1184–1199, 2008.
- [27] X. Zhang and B. Li, "Optimized multipath network coding in lossy wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 5, pp. 622–634, 2009.
- [28] C. de Alwis, H. K. Arachchi, V. De Silva, A. Fernando, and A. Kondo, "Robust video communication using random linear network coding with pre-coding and interleaving," in *2012 19th IEEE International Conference on Image Processing*, Orlando, FL, USA, October 2012.
- [29] S. Kafaie, Y. Chen, O. A. Dobre, and M. H. Ahmed, "Joint inter-flow network coding and opportunistic routing in multi-hop wireless mesh networks: a comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1014–1035, 2018.
- [30] G. Giambene, D. K. Luong, T. de Cola, V. A. Le, and M. Muhammad, "Analysis of a packet-level block coding approach for terrestrial-satellite mobile systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8117–8132, 2019.
- [31] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratt, and S. Banerjee, "MAR: a commuter router infrastructure for the mobile internet," in *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, pp. 217–230, Boston, MA, USA, June 2004.
- [32] J. Ebert and A. Willig, "A Gilbert-Elliot bit error model and the efficient use in packet level simulation," Technical Report TKN 99-2002, Berlin, Germany, 1999.