International Journal of Innovative Computing, Information and Control
Volume **16**, Number **5**, October **2020**
ICIC International ©2020 ISSN 1349-4198
pp. 1739–1755

RELIABLE BROADCAST FOR WIFI-BASED INTERNET OF THINGS THROUGH MULTI-ACCESS EDGE-DYNAMIC FEC ADAPTATION

Yingjie Duan[1,2], Hong Ni[1,2], Xiaoyong Zhu[1,2,*] and Xu Wang[1,2]

[1]National Network New Media Engineering Research Center
Institute of Acoustics, Chinese Academy of Sciences
No. 21, North 4th Ring Road, Haidian District, Beijing 100190, P. R. China
{ duanyj; nih; wangx }@dsp.ac.cn; *Corresponding author: zhuxy@dsp.ac.cn

[2]School of Electronic, Electrical and Communication Engineering
University of Chinese Academy of Sciences
No. 19(A), Yuquan Road, Shijingshan District, Beijing 100049, P. R. China

Received April 2020; revised July 2020

Abstract. *In WiFi-based IoT scenarios, reliable broadcast plays an important role in distributing data to surrounding devices. However, it has been greatly challenged due to nonperfect channel and network conditions. To improve the reliability of the broadcast, this paper presents an architecture which takes advantage of FEC dynamic adaptation scheme and Multi-access Edge Computing (MEC) providing services near the end devices. The proposal involves an element, called the Multi-access Edge-Dynamic FEC adaptation function (ME-DFEC), which includes Wireless Monitor module, Adaptive FEC Algorithm module and FEC Encoder module. Upon the element, by introducing a standard deviation threshold and an equilibrium parameter, the proposed Adaptive FEC Algorithm adjusts FEC rate adaptively according to the wireless channel and network traffic load conditions. The experiment results show that the data recovery probability of ME-DFEC outperforms the Legacy scheme by 59.89% and 45.08% at best under light and heavy network load conditions, respectively. It can also significantly improve packet delivery ratio and get a relatively high data recovery efficiency compared with the Sender-Based FEC scheme, the AP-Based SFEC scheme and the Legacy scheme. Moreover, ME-DFEC can significantly improve wired network resources utilization compared with the traditional Sender-Based FEC scheme.*

Keywords: FEC, Internet of Things (IoT), Reliable broadcast, MEC, Wireless access point, LT, WiFi

1. **Introduction.** The Internet of Things is an infrastructure that provides intelligent services by connecting many things and people in the Internet [1], which has been adopted in different fields and applications, such as smart healthcare, smart cities, smart manufacturing, smart agriculture [2], and smart transportation [3]. Mostly, billions of IoT devices (ranging from tiny sensors units to complex interactive systems) are connected to the transmission network through widely deployed wireless access technologies, like 802.11, ZigBee, Bluetooth, 802.15.4, LoRa, LTE, and NB-IoT [4]. Among them, 802.11 is one of the most widely used non-3GPP wireless access technologies and provides low-cost Internet access services, which has been widely deployed in enterprises, homes, commercial environments and so on [5]. In IoT scenarios using 802.11-based WLAN (WiFi), many applications rely on reliable broadcast to distribute data (e.g., local weather information, daily arrival and departure flights information, real-time traffic information, cricket score, inventory information) to the devices. However, due to the instability of the wireless

DOI: 10.24507/ijicic.16.05.1739

link, there may be interference or noise during communication. Channel error and/or AP buffer overflow can also lead to packet loss. What is more, broadcast does not support MAC-layer retransmissions, so it has a much higher packet loss probability than unicast [6]. All these factors may contribute to the unreliability of broadcast.

In wireless communications, there are generally two types of technologies for reliable communication: Automatic Repeat reQuest (ARQ) and Forward Error Correction (FEC) [7]. ARQ is a kind of error detection scheme, which guarantees the reliability of data transmission based on acknowledgement responses. However, there is growing standby power consumption in order to maintain the session for packet retransmissions when the number of devices increases. FEC is an error-correcting scheme that recovers lost source packets by adding redundant packets. In this way, errors can be detected and corrected at the receivers without retransmissions, which not only shortens the time required to recover the data packets but also reduces the transmission delay. Therefore, FEC technology is more suitable than ARQ technology in wireless broadcast environment [8].

Nevertheless, Static FEC (SFEC) scheme inevitably worsens the network congestion and degrades the network performance in the heavily-congested networks. Data source-based FEC architecture lowers the bandwidth utilization rate and throughput of wired links. Recently, applying MEC to IoT has been studied. ETSI has identified the Internet of Things as one of the key use cases for MEC [9,10]. As an emerging architecture, MEC has become a new paradigm to satisfy demand of IoT and localized computing since it can provide effective methods to overcome limitations of cloud-only models through extending computing, control, storage and network functions to the edge of the network [11]. In order to gain additional advantages of MEC through heterogeneous access technologies (such as 4G, 5G, WiFi and fixed connection), ETSI ISG officially changed the name of mobile edge computing to multi-access edge computing in 2017 [11,12]. After that, MEC servers can be deployed by the network operators at various locations within RAN and/or collocated with different elements that establish the network edge, such as BS, optical network units, radio network controller sites, routers, switches, and WiFi access points [13,14], while applications run on these servers through VMs [15]. Therefore, for IoT applications, it is a promising technology to integrate MEC with wireless access network. However, no one has considered improving broadcast reliability with the help of the technology so far.

In this paper, we focus on improving broadcast reliability for WiFi-based IoT. We design a new broadcast scheme by introducing MEC at the edge of network and propose an FEC adaptation algorithm. Based on this, the FEC adaptation function can be deployed at the edge of the network. The IoT devices can decode and recover the lost data packets. Simulation results demonstrate that the proposed scheme can abtain a balance between data recovery capability and recovery efficiency while improving wired network resources utilization. The main contributions of this paper are as follows.

1) We propose an architecture intended to generate FEC redundant packets adaptively near the IoT devices. This proposal involves a network element, denoted as Multi-access Edge-Dynamic FEC adaptation function (ME-DFEC), which is located within the wireless access network following the novel MEC principles [16]. ME-DFEC is designed as an MEC application instance, and integrates wireless monitoring and adaptive FEC capabilities.

2) We propose a new FEC adaptation algorithm. By introducing the standard deviation threshold σ of packet loss rate and the equilibrium parameter β, the redundancy rate is tuned on the basis of the conditions of wireless channel (as indicated by the packet loss rate) and network load (as indicated by the AP queue length).

3) We set up network simulation environment, and then deploy the function of ME-DFEC on a node in NS-2. We evaluate the performance of the proposed scheme based on extensive network simulations conducted within NS-2 eventually.

The remainder of this paper is organized as follows. Section 2 discusses the related work. In Section 3, we present the proposed architecture and introduce the functional modules of the ME-DFEC element in detail. Section 4 describes the details of Adaptive FEC Algorithm. Section 5 presents the simulation experiments with results and discussion. Finally, we conclude the paper in Section 6.

2. **Related Work.**

2.1. **MEC for IoT.**

Recently, how IoT can leverage MEC technology in various application scenarios has been studied. The surveys on recent MEC-enabled IoT application scenarios were provided in [13,17-20]. As stated in [9,13], the collaboration between IoT and MEC has three main benefits: firstly, reducing traffic through the infrastructure; secondly, lowering the latency for services and applications; thirdly, scaling network services diversely.

The MEC-enabled IoT frameworks in [21-23] focused on behavior characteristics by monitoring student's locations and activities in smart school environments. In particular, [23] exploited the deep learning algorithms in MEC-enabled IoT smart classrooms for person recognition. According to [24], the second largest MEC use case is expected to occur in the retail businesses. Each store provides customers with WiFi connectivity by APs that connect to the MEC server. In the store, enabling MEC can provide high-speed mobile coverage and omit load balancing, policy engines or AP controllers required in the WAN [9]. In [25], a WiFi-based MEC architecture was proposed to solve the problem of joint resource assigning and computation offloading. The simulations in NS-3 have shown the advantages of method in latency and energy consumption. Despite the fact that the above issues in MEC-enabled IoT have been carefully studied, the problem of reliable transmission for broadcast remains to be solved.

2.2. **FEC.**

The basic principle of FEC is to encode redundant packets according to a certain algorithm and send them following with the data packets, so that the lost packets can be recovered without retransmissions [26]. The probability of successfully recovering data packets is related to the number of redundant packets injected in the process of transmission.

One of the FEC codes is the erasure code which has a subclass called fountain code. The fountain code has an important feature that the ratio between the number of source packets and the number of encoded output packets is flexible. Therefore, it is also known as ratelessness code and is recommended for reliable broadcast/multicast in wireless networks [27]. The first ratelessness erasure code is the Luby Transform (LT) code which selects source symbols according to a degree distribution randomly. Then the LT code implements bitwise eXclusive OR (XOR) operation on the source symbols [28]. In computer network communications, a symbol usually refers to a packet.

White Gaussian noise and short-term fast fading can be resisted by error-correcting codes (e.g., turbo codes) in the physical layer of wireless systems. However, the packet loss at the application, caused by other sources of damage (e.g., buffer overflow, slow fading, interference or congestion), cannot be recovered by these codes [29]. To overcome these shortcomings, Application-Layer packet-level Forward Error Correction (AL-FEC) has been proposed. In view of this, we consider applying AL-FEC to broadcast in this paper.

2.3. Reliable broadcast transmission for IoT wireless networks. There have been some error control researches for IoT wireless networks [29]. [30] was the first work to consider the availability of FEC in wireless sensor networks. [31] proposed a broadcast method to encode broadcast packets using erasure code for energy-harvesting networks. However, the FEC rate was constant in this method, which meant that the ratio between the number of code output packets and the number of source symbols was invariant. Therefore, this method was difficult to deal with the serious packet error or loss conditions flexibly. Then the probability of data recovery would go down. [32] proposed to add an ACK stage to estimate packet error rate of receivers after the broadcast stage. Then it calculated the expected number of encoded output packets of the next frame based on the estimated packet error probability. However, it did not consider the possibility of packet loss caused by queue congestion. In [26], an enhanced random early detection FEC scheme was proposed to optimize the quality of video transmissions over WLANs. Compared with most FEC schemes, this scheme adjusted FEC rate according to the path condition and network traffic load. However, the path condition was perceived by the number of packet retransmission times. So, it could only apply for unicast.

3. Proposed ME-DFEC Architecture.

The proposed ME-DFEC architecture, located in the wireless access network, exactly makes use of two technologies: MEC and reliable data transmission for IoT. It mainly consists of Wireless Monitor module, Adaptive FEC Algorithm module and FEC Encoder module. The overall network architecture of the solution is illustrated in Figure 1.

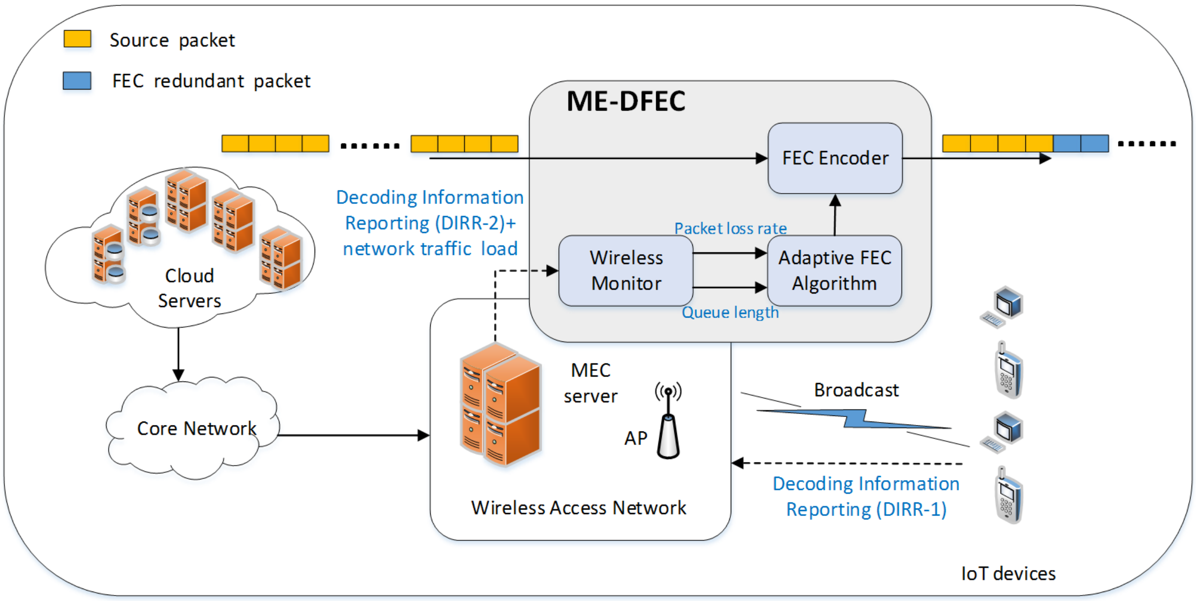


FIGURE 1. Integration of ME-DFEC in wireless access network

Compared with existing reliable broadcast schemes, we employ neither FEC adaptation scheme at the remote data source nor retransmissions. Instead, the goal of the proposed ME-DFEC is to deploy adaptive FEC adjustment functions at the edge of the network.

To achieve this, ME-DFEC relies on MEC to acquire high-bandwidth, low-latency and real-time access to wireless network resources and information. The MEC server is the core element in the MEC industry initiative, which can be integrated into AP in the case of WiFi-based IoT. According to cloud computing principles, third-party applications can be run on the MEC server over a common HW infrastructure. These applications range from lightweight monitoring instances to more complex applications, such as processing and modifying the traffic from/to end users at the wireless access network level. Therefore, following the initial technical architecture proposed by the ETSI MEC ISG, ME-DFEC element can be deployed as an application instance based on the MEC server to guarantee broadcast reliability for WiFi-based IoT.

Overall, the Wireless Monitor module retrieves network traffic load conditions and wireless channel conditions through periodically interacting with the MEC server. When ME-DFEC receives source packets from the Internet or application servers in the cloud, the Adaptive FEC Algorithm module derives the appropriate number of redundant packets. Then, the FEC Encoder module performs FEC encoding for a block. As long as the sum of received packets is not less than the number of source packets, the lost packets in this block can be recovered without retransmissions.

Next, we introduce the three main functional modules of ME-DFEC in detail.

The Wireless Monitor module is responsible for implementing MEC API to interact with the MEC server to monitor wireless channels and network traffic load conditions. Specifically, the network traffic load condition is perceived by acquiring AP queue length across layers, and the wireless channel condition is indicated by the packet loss rate which is estimated according to the collected decoding information. Decoding information contains the total of packets in an FEC block and the number of packets successfully decoded and recovered. And decoding information is reported by IoT devices to the AP at the Decoding Information Reporting Rate (DIRR-1), which is usually set to a low value (e.g., 1 block). However, this report granularity may result in high traffic in the interface between the Wireless Monitor module and MEC server in the scenarios with many users. Therefore, we define the coarse-grain report rate DIRR-2 (e.g., 3 blocks) in Figure 1.

The Adaptive FEC Algorithm module is in charge of making decisions on the number of FEC redundant packets for the next block according to packet loss rate and the AP queue length which are input by Wireless Monitor module periodically. To do this, a standard deviation threshold is introduced to indicate the fluctuation degree of the wireless channel condition, and an equilibrium parameter is introduced to balance the proportion of packet loss rate and AP queue length in determining the number of FEC redundant packets. For the purpose of the channel aware estimations, the MEC server provides low-level information to the Adaptive FEC Algorithm in detail. As the most wireless technology-dependent part of ME-DFEC, the Adaptive FEC Algorithm module should be designed for the specific deployment.

The FEC Encoder module implements FEC encoding based on the decisions of the Adaptive FEC Algorithm module. As a result, the output of the encoding process is composed of source packets and FEC redundant packets. Finally, the encoded blocks are delivered towards the IoT devices.

4. **Adaptive FEC Algorithm.**

This section describes the Adaptive FEC Algorithm presented in Section 3 in detail. And the pseudocode is illustrated as Algorithm 1.

4.1. **Estimation of packet loss rate and its standard deviation.**

In Wireless Monitor module, the packet loss rate of each device is calculated in accordance with the decoding information. And let p_i be the packet loss rate of the i-th device during a feedback period. If the decoding information of m in s IoT devices is received in the AP coverage area, the current packet loss rate p can be determined as

$$p = \frac{\sum_{i=0}^{m} p_i}{m}, \quad 0 < m \le s \tag{1}$$

Algorithm 1. Adaptive FEC Algorithm

(1) Initialization: $r = 0$, $\beta = const$, $\sigma = const$, $\alpha = 0.9$, $\gamma = 0.9$, $th_{low} = 25$, $th_{high} = 40$
(2) When a block arrives: calculate the current packet loss rate p by Equation (1), then estimate average packet loss rate p_{avg} by Equation (2) and update the current standard deviation plr_{dev} by Equation (3)
(3) Calculate r based on the weighted moving average packet loss rate by Equation (6); Update Num_{FECplr} by Equation (7)
(4) Update the current weighted moving average $q_{len,i}$ of AP queue length by Equation (4); Update $Num_{FECqlen}$ by comparing $q_{len,i}$ with two threshold values th_{low} and th_{high} by Equation (8)
(5) Determine the final number of redundant packets Num_{FEC}:
(6) **if1** ($q_{len,i} \geq th_{low}$)
(7) **if2** ($plr_{dev} > \sigma$ and $(p - p_{avg}) > 0$)
(8) Update Num_{FEC} by choosing the minimum from ($Num_{FECplr}, Num_{FECqlen}$)
(9) **else**
(10) Update Num_{FEC} by assigning a greater weight value to Num_{FECplr} by Equation (9)
(11) **end if2**
(12) **else if1** ($q_{len,i} < th_{low}$)
(13) **if3** ($plr_{dev} > \sigma$ and $(p - p_{avg}) > 0$)
(14) Update Num_{FEC} by choosing the maximum from ($Num_{FECplr}, Num_{FECqlen}$)
(15) **else**
(16) Update Num_{FEC} by assigning a greater weight value to $Num_{FECqlen}$ by Equation (10)
(17) **end if3**
(18) **end if1**
(19) return Num_{FEC}

Then we estimate and smooth packet loss rate using EWMA calculation model [33] as average packet loss rate as follows:

$$p_{estimation,i} = \gamma \times p + (1 - \gamma) \times p_{estimation,i-1} \tag{2}$$

where γ $(0 < \gamma < 1)$ is a weighting factor. Note that $p_{estimation,i}$ is computed as the current weighted moving average of p in implementing the scheme.

Since the standard deviation reflects the dispersion degree of data, we count the standard deviation plr_{dev} of detected packet loss rate and use it to judge the fluctuation degree of wireless link condition. In Equation (3), p is the current packet loss rate of the wireless link calculated in Equation (1), and p_{avg} ($p_{avg} = p_{estimation,i}$) is the average packet loss rate of the wireless link, which has been recorded in Equation (2) by EWMA.

$$plr_{dev} = fabs(p - p_{avg}) \tag{3}$$

4.2. **Weighted moving average of queue length.** Let $q_{current\_i}$ be the queue length when i-th queue length information reaches Wireless Monitor module. As shown in Equation (4), it responds by computing the current weighted moving average (i.e., $q_{len,i}$) of the queue length,

$$q_{len,i} = \alpha \times q_{current\_i} + (1 - \alpha) \times q_{len,i-1} \tag{4}$$

where α $(0 < \alpha < 1)$ is a weighting factor.

4.3. **Adjustment of FEC redundant packets.** In Adaptive FEC Algorithm module, we propose an FEC rate adaptation algorithm to derive the number of the encoded output packets required for the next block. Compared with existing adaptive FEC algorithms for reliable broadcast, we take both wireless channel conditions and network traffic load conditions into account, and we are the first to introduce the standard deviation threshold σ of packet loss rate to represent the fluctuation degree of the wireless channel condition. In addition, we propose the equilibrium parameter β to balance the proportion of Num_{FECplr} and $Num_{FECqlen}$ in determining Num_{FEC} under different network load conditions.

Firstly, Num_{FECplr} and $Num_{FECqlen}$ are calculated by the estimated packet loss rate and the AP queue length, respectively.

The process of estimating wireless channel packet loss rate is described in Section 4.1. Let n and k be the number of the encoded output packets and the number of source packets respectively, then $n = k + r$. Next, we assume the wireless packet loss rate is p with using weighted moving average, the number of packets successfully received by the devices needs to be greater than or equal to k so that data can be recovered. Hence, the expected relationship between k and n is expressed as

$$n(1-p) \geq k \tag{5}$$

The minimum value of $n - k$ (i.e., r) which is required for the successful recovery of source packets is

$$r = n - k = \left\lceil \frac{kp}{1-p} \right\rceil \tag{6}$$

where $\lceil \cdot \rceil$ is a ceiling operator.

The maximum of r is limited to k in order to avoid network congestion or excessive energy consumption. Thus, the value of Num_{FECplr} can be determined as

$$Num_{FECplr} = \begin{cases} \left\lceil \dfrac{kp}{1-p} \right\rceil, & 0 \leq r \leq k \\ k, & r > k \end{cases} \tag{7}$$

Then we compute the value of $Num_{FECqlen}$ by comparing the result in Equation (4) with two threshold values of AP queue length, named th_{low} and th_{high}. The value of $Num_{FECqlen}$ is determined as

$$Num_{FECqlen} = \begin{cases} k, & q_{len} < th_{low} \\ k * \dfrac{th_{high} - q_{len}}{th_{high} - th_{low}}, & th_{low} \leq q_{len} \leq th_{high} \\ 0, & q_{len} > th_{high} \end{cases} \tag{8}$$

Finally, the number of redundant packets Num_{FEC} is determined based on Num_{FECplr} and $Num_{FECqlen}$ in Equation (9) or (10). The symbol *round* indicates the result is rounded.

As shown in Equation (9), when the current weighted moving average of the queue length is greater than or equal to th_{low}, we consider the network load is heavy. At this point, if plr_{dev} is greater than σ and packet loss rate is increasing, Num_{FEC} should choose the minimum from Num_{FECplr} and $Num_{FECqlen}$ to avoid exacerbating network congestion. Otherwise, Num_{FECplr} shoud be given a greater weight value.

$$Num_{FEC} = \begin{cases} \min(Num_{FECplr}, Num_{FECqlen}), & (plr_{dev} > \sigma) \text{ and } (p - p_{avg}) > 0 \\ round(\beta * Num_{FECplr} + (1-\beta) * Num_{FECqlen}), & \text{otherwise} \end{cases} \tag{9}$$

If the current weighted moving average of the queue length is less than th_{low}, the network load is considered to be relatively light and more attention should be paid to the wireless channel state. In this case, the number of encoded output redundant packets is finally determined as Equation (10). If plr_{dev} is greater than σ and packet loss rate is increasing, Num_{FEC} should choose the maximum from Num_{FECplr} and $Num_{FECqlen}$ to generate as many FEC redundant packets as possible. Otherwise, $Num_{FECqlen}$ should be assigned a greater weight value.

$$Num_{FEC} = \begin{cases} \max(Num_{FECplr}, Num_{FECqlen}), & (plr_{dev} > \sigma) \text{ and } (p - p_{avg}) > 0 \\ round((1-\beta) * Num_{FECplr} + \beta * Num_{FECqlen}), & \text{otherwise} \end{cases} \quad (10)$$

5. Simulation and Result.

5.1. Simulation environment and parameters. The simulation topology is shown in Figure 2. Since this paper aims at evaluating how ME-DFEC enhances the reliability of IoT broadcast using WiFi, we deployed the function of ME-DFEC on a node and set four nodes as broadcast receivers in NS-2 simulator [34]. A unicast background traffic flow (CBR over UDP) was also introduced to increase network traffic load. Simulation parameters and their values are shown in Table 1. Additionally, in the initialization phase

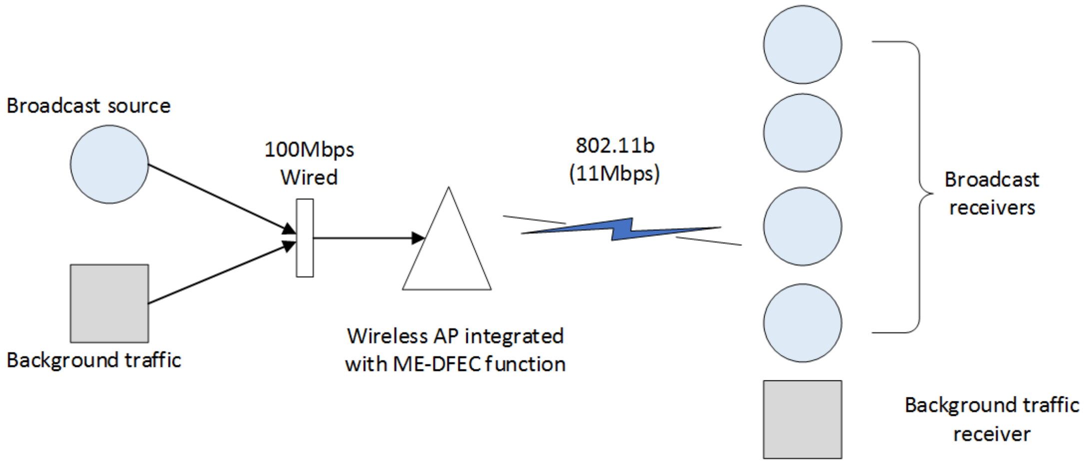


FIGURE 2. Simulation topology

TABLE 1. Wireless network configuration

Parameter	Value
Simulation area/m^2	1237×751
Channel type	WirelessChannel
Radio-propagation model	TwoRayGround
MAC type	802.11b
Antenna model	OmniAntenna
Simulation duration	100s
The number of source packets within each FEC block (k)	4-6
The maximum queue length	50 packets
The upper threshold for queue length (th_{high})	40 packets
The lower threshold for queue length (th_{low})	25 packets

of Algorithm 1, we set the values of α, γ, th_{high} and th_{low} according to [26]. And the values of β and σ are set through the experiments in Section 5.2. In performing simulations, the transmission units of 1000 bytes were broadcasted to the receivers by a base station node (i.e., AP) and were decoded at the receivers. The FEC Encoder module utilized LT code in the application layer. Besides, it was assumed that no packets were lost in the wired segment of the delivery path.

We integrated Gilbert-Elliott channel model (GE) into NS-2 to simulate packet loss over the network. GE is widely used due to its simplicity and mathematical traceability, which uses two-state Markov chain to get a good approximation of wireless channel and represent the packet loss [35]. Loss occurs with low probability p_G in the "good" (G) state, while it happens with high probability p_B in the "bad" (B) state. p_{GB} and p_{BG} represent the probabilities of the transition from G state to B state and the transition from B state to G state, respectively. The probabilities of the transition from G state to G state and the transition from B state to B state are $p_{GG} = 1 - p_{GB}$ and $p_{BB} = 1 - p_{BG}$, respectively. The steady state probabilities of being in G state and B state are $\pi_G = \frac{p_{BG}}{p_{BG}+p_{GB}}$ and $\pi_B = \frac{p_{GB}}{p_{BG}+p_{GB}}$, respectively. The average packet loss rate can be calculated as $p = p_G\pi_G + p_B\pi_B$. The GE module parameter values are shown in Table 2. Figure 3 shows that the packet loss rate simulated by the GE is in good agreement with the true value.

TABLE 2. Parameters setting for GE model

Parameter	Value
The probability of the transition from G state to G state (p_{GG})	0.96
The probability of the transition from B state to B state (p_{BB})	0.94
The packet loss probability in the "good" state (p_G)	0.01
The packet loss probability in the "bad" state (p_B)	0.05-0.4

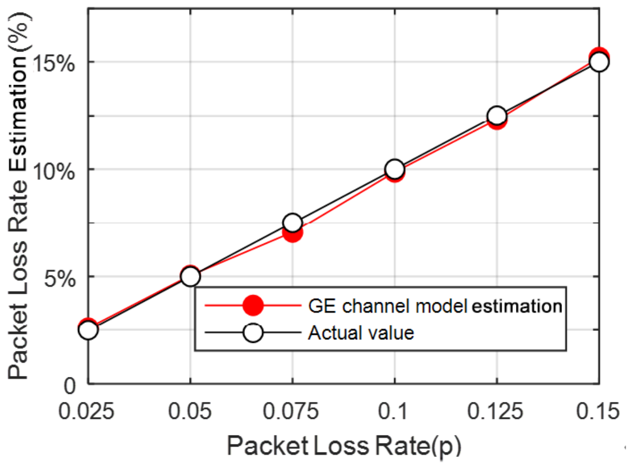
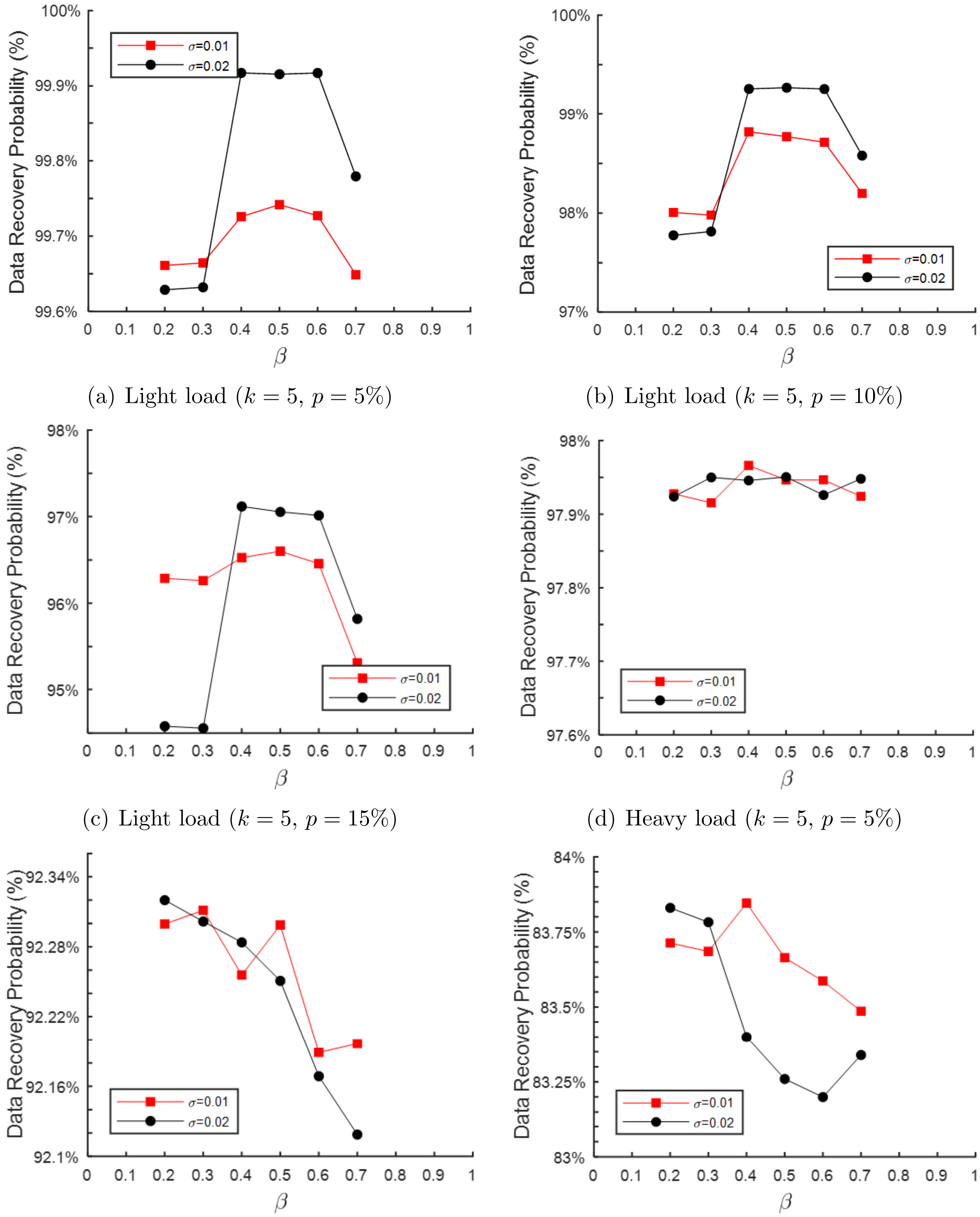


FIGURE 3. Packet loss rate simulated by GE model without FEC recovery

5.2. **Experiments on the setting of values σ and β.** In the Adaptive FEC Algorithm module, the values of the parameter σ and β play an important role in the performance of the algorithm. In this subsection, the setting of values σ and β in Equation (9) and Equation (10) is evaluated in terms of the data recovery probability. Let p_{res} be the data recovery probability, and then it can be expressed as

$$p_{res} = \sum_{i=k}^{n} \binom{n}{i} p^{n-i}(1-p)^{i} \tag{11}$$

β is set to 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1 and σ is set to 0.01, 0.02. Then the performance of the algorithm is evaluated when the packet loss rate is 5%, 10% and 15%, under light load conditions (average queue length <= 6 packets) and heavy load conditions (average queue length = 36.05 packets), respectively. Some of the results are



(a) Light load ($k = 5$, $p = 5\%$)
(b) Light load ($k = 5$, $p = 10\%$)
(c) Light load ($k = 5$, $p = 15\%$)
(d) Heavy load ($k = 5$, $p = 5\%$)
(e) Heavy load ($k = 5$, $p = 10\%$)
(f) Heavy load ($k = 5$, $p = 15\%$)

FIGURE 4. Experiments on the setting of values σ and β

shown in Figure 4. From Figures 4(a)-4(c), when σ is 0.02 and β is 0.4, the data recovery probability can achieve the peak under light load conditions. From Figures 4(d)-4(f), under heavy load conditions, when σ is 0.02 and β is 0.2, the data recovery probability can achieve the peak at the packet loss rate of 10% and 15%, and it is very close to the peak at the packet loss rate of 5%. Hence, the value of σ is always set to 0.02, and the value of β is set to 0.4 and 0.2 when the AP queue length is less than or equal to th_{low} and greater than th_{low}, respectively.

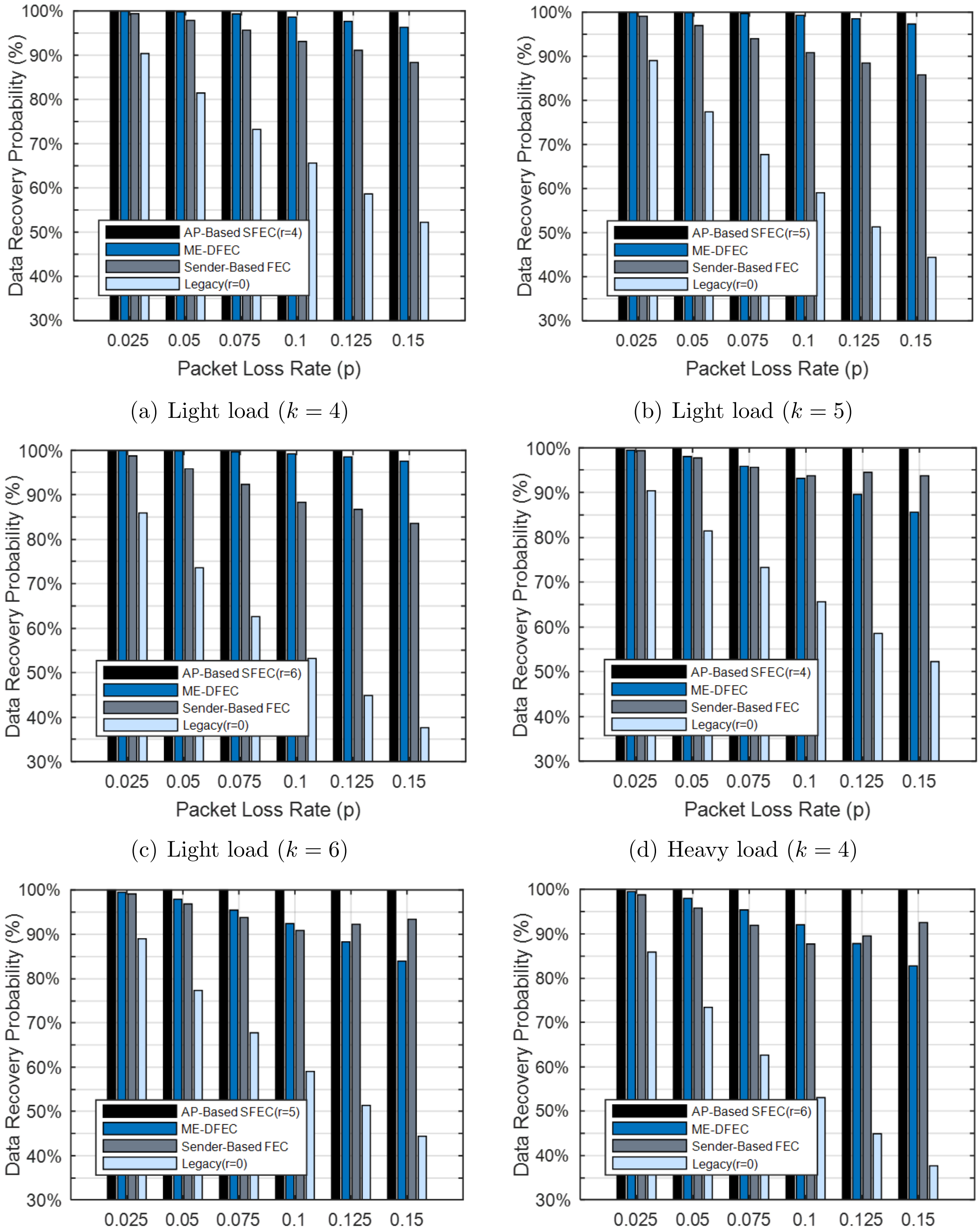
5.3. **Simulation results and analysis.** Through 30 times simulation experiments, we have obtained the simulation results, averaged over all receivers. We compared ME-DFEC with the AP-Based SFEC scheme, the Sender-Based FEC scheme and the Legacy scheme (no FEC) at the aspect of Data Recovery Probability (DRP), Packet Delivery Ratio (PDR) and Recovery Efficiency (RE) under light and heavy network load conditions, respectively. We also compared ME-DFEC with the Sender-Based FEC scheme for the impact on the wired network traffic. For the AP-Based SFEC scheme, the number of redundant packets is always equal to the number of the source packets in a block. The Sender-Based FEC scheme determines FEC rate based on the average packet loss rate fed back from AP and encodes FEC packets at the data source.

5.3.1. *Data Recovery Probability (DRP).* As shown in Figure 5, the DRP achieved by the AP-Based SFEC scheme is very close to 1 because of quantities of redundant packets injected. In the Legacy scheme, the device cannot recover the original data for a single packet lost, and the DRP decreases sharply as packet loss rate increases, which is an exponential function of the packet reception rate $(1-p)$. Therefore, when $k > 4$, DRP decreases faster as packet loss rate increases than it does when $k = 4$. By contrast, the other two schemes can adaptively generate additional encoded output packets, thus DRP decreases slowly with the packet loss rate increasing. Furthermore, when $k = 6$ and $p = 30\%$, ME-DFEC can improve DRP by up to 45.08% than the Legacy scheme under heavy load conditions, and the proportion of improvement can be up to 59.89% under light load conditions.

From Figures 5(a)-5(c), under light load conditions, thanks to the reasonable setting of the values of σ and β in Section 5.2, the DRP of ME-DFEC is very close to that of the AP-Based SFEC scheme and more robust to packet loss rate than the Sender-Based FEC scheme and the Legacy scheme. Under heavy load conditions, as shown in Figures 5(d)-5(f), ME-DFEC consistently has a greater DRP than the Sender-Based FEC scheme and the Legacy scheme, except when the packet loss rate is greater than 10%, it is slightly lower than that of the Sender-Based FEC scheme. This is because the Sender-Based FEC scheme still injects more FEC redundant packets as the packet loss rate increases when the network is congested, whereas the proposed ME-DFEC does not.

Overall, given the same packet loss rate and number of source packets, the DRP of ME-DFEC is lower under heavy load conditions than under light load conditions. This is because we consider the queue length of the AP when adjusting FEC rate. According to Equation (9), when the queue is nearly full, less redundant packets are added in order to avoid network overload caused by the FEC redundant packets.

5.3.2. *Recovery Efficiency (RE).* In this simulation, RE is defined as the ratio of the total number of packets recovered during transmission to the total number of redundant packets sent. As shown in Figure 6, under the packet loss rate of 15%, $k = 6$ and heavy load, ME-DFEC recovers 690 lost data packets by injecting 2917 redundant packets. Therefore, the RE value is 23.65%. By contrast, under the same conditions, the RE values of the AP-Based SFEC scheme and the Sender-Based FEC scheme are 11.50% and



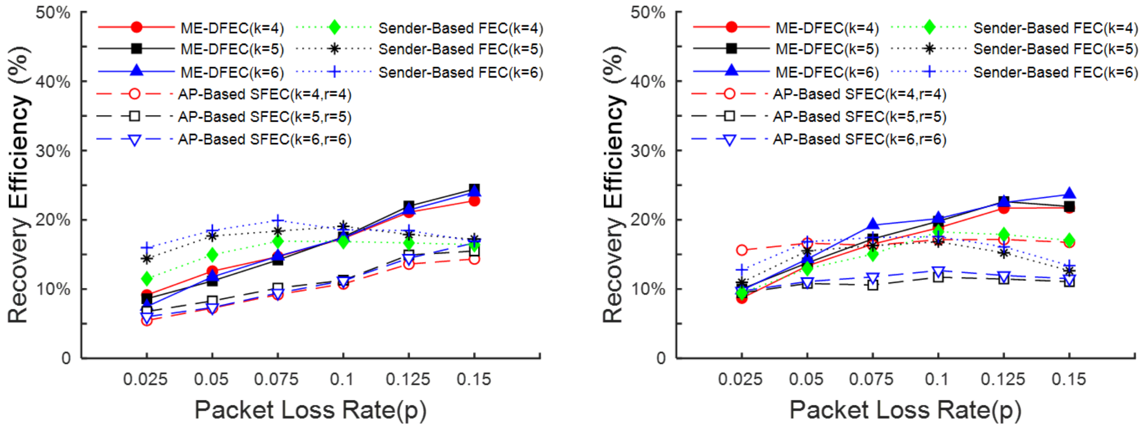
(a) Light load ($k = 4$) (b) Light load ($k = 5$)

(c) Light load ($k = 6$) (d) Heavy load ($k = 4$)

(e) Heavy load ($k = 5$) (f) Heavy load ($k = 6$)

FIGURE 5. Variation of Data Recovery Probability (DRP) with packet loss rate

13.38%, respectively. Whereas under the packet loss rate of 15%, $k = 6$ and light load, the RE values of ME-DFEC, AP-Based SFEC scheme, and Sender-Based FEC scheme are 23.98%, 16.68% and 16.89%, respectively. Note that the RE of the Legacy scheme is always equal to 0.



(a) Light load (b) Heavy load

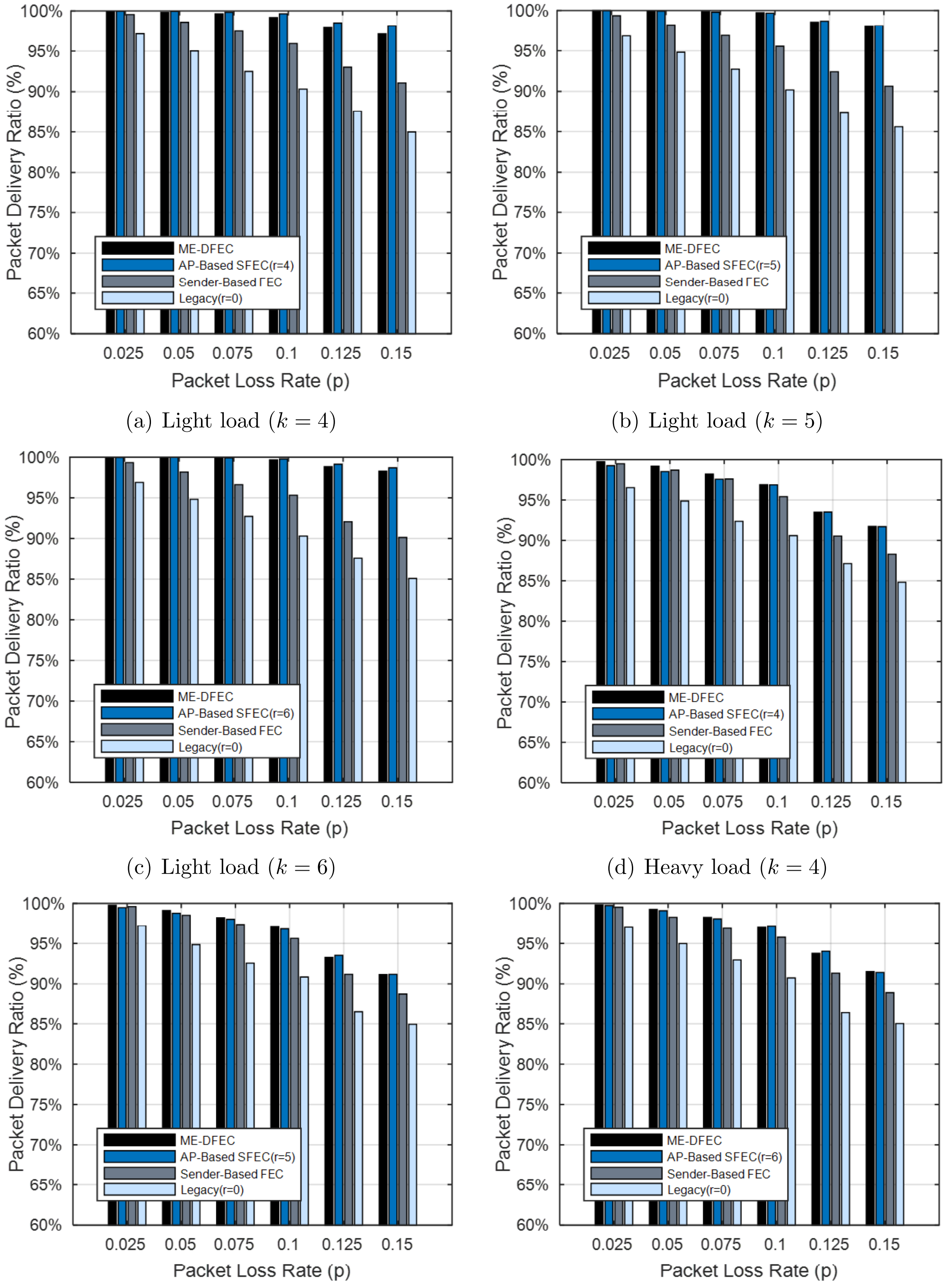
FIGURE 6. Variation of Recovery Efficiency (RE) with packet loss rate

As expected, the AP-Based SFEC scheme does not consider packet loss rate and network load when adjusting FEC rate, so the extremely great DRP obtained in the previous section is actually at the expense of RE. Since the Sender-Based FEC scheme ignores the impact of network load, the RE is not improved but reduced when the packet loss rate increases. However, no matter whether the network load condition is light or heavy, ME-DFEC can consistently achieve better recovery efficiency than the other three schemes when packet loss rates are higher. This shows that, in terms of recovery efficiency, ME-DFEC has more prominent advantages under higher packet loss rates.

5.3.3. *Packet Delivery Ratio (PDR).* In this simulation, PDR is defined as the ratio of the number of source packets delivered to the receivers to the total number of source packets transmitted by the data source. This indicator directly measures the reliability of broadcast. For instance, under light load conditions, when the packet loss rate is 15% and $k = 6$, the number of source packets transmitted by the data source is 10611. The reveivers can receive 8973 source packets and recover 1455 packets by the ME-DFEC. Thus, the PDR gained by the ME-DFEC is 98.28%, which is 13.19% greater than the Legacy scheme under the same conditions.

From Figure 7, we can learn that the packet delivery ratios obtained by the Sender-Based FEC scheme and the Legacy scheme reduce significantly with the packet loss rate increasing. However, there is no obvious decrease under ME-DFEC since ME-DFEC determines the number of FEC redundant packets by taking account of wireless channel and network load conditions. Overall, the packet delivery ratios of ME-DFEC are always greater than those of the Sender-Based FEC scheme and the Legacy scheme under light or heavy load conditions. In addition, ME-DFEC can obtain almost the same PDR as the AP-Based SFEC scheme. The comparisons on PDR confirm the ability of ME-DFEC to improve broadcast transmission reliability and quality by adaptively adjusting FEC rate for the next block.

5.3.4. *Wired network traffic.* We evaluate the impact of ME-DFEC and the Sender-Based FEC scheme on the wired network traffic in terms of the total amount of wired network traffic towards the receivers and the backhaul traffic from AP to the data source. Figure 8 presents the reduction of the wired network traffic under the ME-DFEC compared with the Sender-Based FEC scheme. It can be seen the wired network traffic reduction increases gradually as the packet loss rate increases. Moreover, when the packet loss rate is greater



(a) Light load ($k = 4$)
(b) Light load ($k = 5$)
(c) Light load ($k = 6$)
(d) Heavy load ($k = 4$)
(e) Heavy load ($k = 5$)
(f) Heavy load ($k = 6$)

FIGURE 7. The Packet Delivery Ratio (PDR) for different packet loss rates

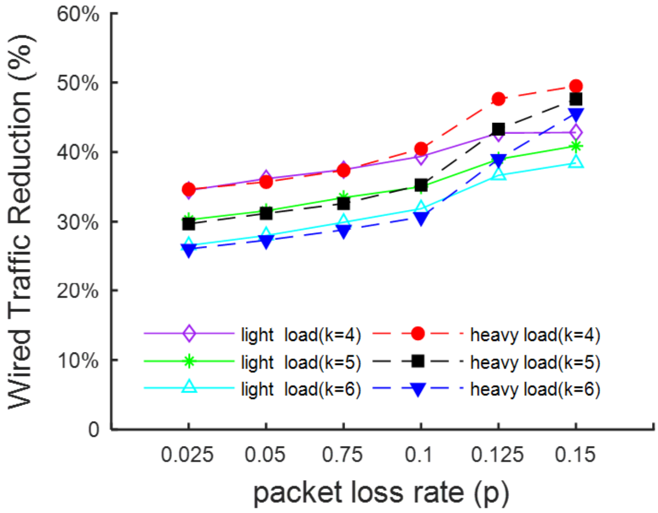


Figure 8. The wired network traffic reduction for different packet loss rates

than 10%, the wired traffic reduction is greater under heavy load conditions than under light load conditions. This is because the Sender-Based FEC scheme must produce more redundant packets to compensate for the packet loss under heavily-congested network load conditions.

According to the numerical results, ME-DFEC can reduce the total wired network traffic by 42.82% with the packet loss rate of 15% and $k = 4$ under light load conditions, and by 49.47% under heavy load conditions. In fact, this represents the additional benefits of ME-DFEC providing dynamic FEC adjustment at the edge of network.

5.4. **Analysis and discussion.** Among the four schemes, owing to quantities of redundant packets injected, the DRP and the PDR achieved by the AP-Based SFEC scheme are the highest. However, that is achieved at the expense of RE. The DRP and the PDR achieved by Legacy scheme are very low as a result of no FEC redundant packets. In particular, the RE is 0. For the Sender-Based FEC scheme, the PDR is always lower than ME-DFEC. Simultaneously, it has a lower DRP than ME-DFEC in most cases. This is because the former only depends on the packet loss rate to adjust the FEC rate. What is more, ME-DFEC has a higher RE than other three schemes when packet loss rate is higher whether the network load is light or heavy.

In summary, ME-DFEC can obtain the highest PDR and relatively high DRP and RE. In other words, a good balance between data recovery capability and recovery efficiency can be achieved under ME-DFEC. Besides, ME-DFEC can dramatically reduce wired traffic than traditional Sender-Based FEC scheme as shown in Figure 8. From the perspective, adaptive FEC encoding at the edge of networks by ME-DFEC can improve wired network resources utilization.

6. **Conclusions.** In this paper, we proposed an architecture to enhance the reliability of broadcast in WiFi-based IoT wireless networks. The proposal introduced an element, denoted as ME-DFEC, which was placed within the wireless access network to generate FEC redundant packets adaptively. Upon the element, by introducing a standard deviation threshold and an equilibrium parameter, the Adaptive FEC Algorithm module calculated the expected number of FEC redundant packets based on the network traffic and wireless channel conditions reported by the Wireless Monitor module. Finally, The FEC Encoder module performed FEC encoding dependent on the decisions of the Adaptive FEC Algorithm module.

Simulation results demonstrate that ME-DFEC can obtain higher packet delivery ratio and get relatively high data recovery efficiency and data recovery probability compared with the Sender-Based FEC scheme, the AP-Based SFEC scheme and the Legacy scheme. Moreover, ME-DFEC can significantly improve wired network resources utilization compared with traditional Sender-Based FEC scheme. In the future, we will combine FEC rate adaptation with PHY-rates adaptation to improve broadcast reliability and efficiency further. Besides, we intend to extend the proposed scheme to reliable multicast transmission.

Acknowledgment. This work is partially supported by Strategic Leadership Project of Chinese Academy of Sciences: SEANET Technology Standardization Research System Development (Project No. XDC02070100).

REFERENCES

[1] M. A. López Peña and I. Muñoz Fernández, SAT-IoT: An architectural model for a high-performance fog/edge/cloud IoT platform, *Proc. of the 2019 IEEE the 5th World Forum on Internet of Things (WF-IoT)*, Limerick, Ireland, pp.633-638, 2019.

[2] H. Hejazi, H. Rajab, T. Cinkler and L. Lengyel, Survey of platforms for massive IoT, *Proc. of the 2018 IEEE International Conference on Future IoT Technologies (Future IoT)*, Eger, pp.1-8, 2018.

[3] E. Husni and G. Boy, Car driver attitude monitoring system using fuzzy logic with the Internet of Things, *ICIC Express Letters*, vol.12, no.11, pp.1115-1122, 2018.

[4] R. Muñoz et al., Integration of IoT, transport SDN, and edge/cloud computing for dynamic distribution of IoT analytics and efficient use of network resources, *Journal of Lightwave Technology*, vol.36, no.7, pp.1420-1428, 2018.

[5] J. Sheth and B. Dezfouli, Enhancing the energy-efficiency and timeliness of IoT communication in WiFi networks, *IEEE Internet of Things Journal*, vol.6, no.5, pp.9085-9097, 2019.

[6] D. Kim and J. Kim, Broadcast packet collision and avoidance method in Wi-Fi based broadcasting system, *Proc. of the International Conference on Information Network*, Bali, pp.108-113, 2012.

[7] J. Nonnenmacher and E. W. Biersack, Reliable multicast: Where to use forward error correction, *Proc. of the IFIP 5th International Workshop on Protocols for High Speed Networks*, pp.134-148, 1996.

[8] H. Weatherspoon and J. D. Kubiatowicz, Erasure coding vs. replication: A quantitative comparison, *Proc. of the 1st International Workshop on Peer-to-Peer Systems (IPTPS)*, 2002.

[9] P. Porambage et al., Survey on multi-access edge computing for Internet of Things realization, *IEEE Communications Surveys & Tutorials*, vol.20, no.4, pp.2961-2991, 2018.

[10] Y. C. Hu et al., Mobile edge computing: A key technology towards 5G, *ETSI White Paper*, Sophia Antipolis, France, no.11, pp.1-16, 2015.

[11] R. Ullah, S. H. Ahmed and B. Kim, Information-centric networking with edge computing for IoT: Research challenges and future directions, *IEEE Access*, vol.6, pp.73465-73488, 2018.

[12] S. Kekki et al., MEC in 5G networks, *ETSI White Paper*, Sophia Antipolis, France, no.28, pp.1-28, 2018.

[13] Q. V. Pham et al., A survey of multi-access edge computing in 5G and beyond: Fundamentals technology integration and state-of-the-art, *arXiv Preprint arXiv:1906.08452*, 2019.

[14] D. J. Hong and L. Wang, Overview of mobile edge computing technology, *Journal of Network New Media*, vol.8, no.5, pp.60-63, 2019.

[15] G. Premsankar, M. Di Francesco and T. Taleb, Edge computing for the Internet of Things: A case study, *IEEE Internet of Things Journal*, vol.5, no.2, pp.1275-1284, 2018.

[16] M. Patel et al., Mobile-edge computing introductory technical white paper, *White Paper*, https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf, 2014.

[17] D. Sabella et al., Mobile-edge computing architecture: The role of MEC in the Internet of Things, *IEEE Consumer Electronics Magazine*, vol.5, no.4, pp.84-91, 2016.

[18] S. Wang et al., A survey on mobile edge networks: Convergence of computing, caching and communications, *IEEE Access*, vol.5, pp.6757-6779, 2017.

[19] S. Shahzadi, M. Iqbal, T. Dagiuklas and Z. U. Qayyum, Multi-access edge computing: open issues, challenges and future perspectives, *Journal of Cloud Computing*, vol.6, no.1, 2017.

[20] W. Yu et al., A survey on the edge computing for the Internet of Things, *IEEE Access*, vol.6, pp.6900-6919, 2018.

[21] A. Y. N. Pratama, A. Zainudin and M. Yuliana, Implementation of IoT-based passengers monitoring for smart school application, *Proc. of the 2017 International Electronics Symposium on Engineering Technology and Applications (IES-ETA)*, Surabaya, pp.33-38, 2017.

[22] Z. Samia, D. R. Khaled and Z. Warda, Multi-agent systems and ontology for supporting management system in smart school, *Proc. of the 2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, Tebessa, pp.1-8, 2018.

[23] A. Pacheco et al., A smart classroom based on deep learning and osmotic IoT computing, *Proc. of the 2018 Congreso Internacional de Innovación y Tendencias en Ingeniería (CONIITI)*, Bogota, pp.1-5, 2018.

[24] *The Business Case for MEC in Retail: A TCO Analysis and its Implications in the 5G Era*, https://builders.intel.com/docs/networkbuilders/thebusiness-case-for-mec-in-retail-a-tco-analysis-and-its-implications-in-the-5g-era.pdf, Accessed in 2018.

[25] B. Dab, N. Aitsaadi and R. Langar, A novel joint offloading and resource allocation scheme for mobile edge computing, *Proc. of the 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, USA, pp.1-2, 2019.

[26] C. Lin, C. Shieh and W. Hwang, An access point-based FEC mechanism for video transmission over wireless LANs, *IEEE Trans. Multimedia*, vol.15, no.1, pp.195-206, 2013.

[27] D. J. C. MacKay, Fountain codes, *IEE Proceedings – Communications*, vol.152, no.6, pp.1062-1068, 2005.

[28] M. Luby, LT codes, *Proc. of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, Vancouver, BC, pp.271-280, 2002.

[29] A. Badr, A. Khisti, W. Tan and J. Apostolopoulos, Perfecting protection for interactive multimedia: A survey of forward error correction for low-delay interactive applications, *IEEE Signal Processing Magazine*, vol.34, no.2, pp.95-113, 2017.

[30] R. Kumar, A. Paul, U. Ramachandran and D. Kotz, On improving wireless broadcast reliability of sensor networks using erasure codes, *Proc. of the 2nd International Conference on Mobile Ad-hoc and Sensor Networks*, pp.155-170, 2006.

[31] C. Kuan, G. Lin, H. Wei and R. Vannithamby, Reliable multicast and broadcast mechanisms for energy-harvesting devices, *IEEE Trans. Vehicular Technology*, vol.63, no.4, pp.1813-1826, 2014.

[32] S. Y. Jeon, J. H. Ahn and T. Lee, Data distribution in IoT networks with estimation of packet error rate, *Proc. of the 2016 10th International Conference on Next Generation Mobile Applications, Security and Technologies (NGMAST)*, Cardiff, pp.94-98, 2016.

[33] P. E. Maravelakis and P. Castagliola, An EWMA chart for monitoring the process standard deviation when parameters are estimated, *Computational Statistics & Data Analysis*, vol.53, no.7, pp.2653-2664, 2009.

[34] *NS-2 Simulator*, https://www.isi.edu/nsnam/ns/ns-build.html.

[35] M. Zorzi, R. R. Rao and L. B. Milstein, On the accuracy of a first-order Markov model for data transmission on fading channels, *Proc. of IEEE ICUPC'95*, pp.211-215, 1995.