A State-Aware Rate Optimization Scheme based on Congestion Discrimination over Wireless Networks

Haitao Zhao
College of Communication and Information Engineering/Nanjing University of Posts and Telecommunications, Nanjing, China
Email: zhaohtmail@gmail.com

Yuning Dong, Hui Zhang, and Xinxing Zhao
College of Communication and Information Engineering/Nanjing University of Posts and Telecommunications, Nanjing, China
National Mobile Communications Research Laboratory/Southeast University, Nanjing, China
Email: dongyn@njupt.edu.cn, zhhjoice@126.com, zhaobofu@gmail.com

***Abstract*—In this paper, a novel rate optimization scheme based on discrimination is proposed for multimedia transmission over heterogeneous IP networks. Via analysis wireless channel characteristics, the scheme can detect the nature of packet losses by sending large and small packets alternately, then adopt adaptive rate control strategies to increase the network throughput and decrease congestion packet loss rate. By means of updating factor and loss queue, this scheme can adapt to the changes of network states quickly and improve delivery quality of real-time multimedia stream. Compared with previous algorithms, simulation results show that the proposed scheme can improve the networks throughput of multimedia transmissions and reduce the congestion loss rate, in different network topology environments.**

***Index Terms*—Wireless networks, Congestion discrimination, Packet loss types, Rate control**

I. Introduction

With the rapid development of the wireless networks technology, many applications of wireless service are based on integration of wireless communication networks with Internet. In such wireless-wired hybrid IP networks, there are mainly two types of packet losses: 1) packet losses caused by congestions in wired networks; 2) packet losses by random bit errors or shadow fading in wireless networks. Besides, the principal reasons of packet losses in these two kinds of networks (wired and wireless) are different. Since the link quality is usually fairly good in wired IP networks, the packet losses are mainly due to congestion at the network nodes. However, in wireless networks, the packet losses are most probably due to random bit errors.

To improve the performance of wireless multimedia networks, lots of researches which focus on packet loss types (random channel error or congestion losses) have been reported. In the meantime, the widely used TFRC (TCP-Friendly Rate Control) [1] rate control protocol for real-time multimedia transmission often overly decreases the transmission rate and reduces data throughput of the wireless networks because it cannot distinguish the types of packet losses. The problem also degrades the performance of TCP that always decreases its sending window (or its transmission rate) whenever packet loss happens and causes the waste of network bandwidth and transmission performance degradation in wireless environment. The discrimination between congestion loss and wireless loss can affect the performance of many transport protocols like TCP and TFRC.

Currently there are mainly two schemes to distinguish the type of packet losses. One is the link-based algorithm that uses the information provided by the network layer. This algorithm marks random bit errors by adding explicit loss notification (e.g. ECN) in IP packet header [2], or sets up an agent between the wired and wireless BS to carry out transmission control and to collect the statistical information of wireless and wired channels respectively [3]. Since it needs amelioration of current network infrastructures, this type of algorithms is difficult to implement in near future. Another scheme is the end-system based source algorithm that detects the nature of packet loss errors by sending probe packets [4], or by heuristic approaches, such as Spike [5] that uses relative one-way trip time (ROTT) to differentiate packet losses. The source algorithm can be directly applied to existing networks and thus easy to implement. The Zigzag [6] algorithm based on ROTT and the number of the continuous lost packets can discriminate the cause of packet loss. The performances of these algorithms depend on network topologies and competition flows. The Biaz [7] algorithm only works well on some specific topologies (such as wireless last hop, WLH) with a single flow running. Its performance will dramatically degrade when many flows compete in a shared link. The Zigzag

Manuscript received Sept. 25, 2010; revised Nov. 16, 2010; accepted July 31, 2011.
Corresponding author: Haitao Zhao.

doi:10.4304/jcp.7.3.627-635

algorithm showed a relatively better performance across different scenarios. Park et al [8] proposed a packet loss discrimination method (Spld (statistical packet loss discrimination)) based on statistics of Piat, which improved the accuracy of the loss type classification in the scenario of multi-flow competition. In order to mitigate TCP unfairness, a cross-layer dual queue scheme [9] is designed to provide two separate queues at the AP and selects these queues with appropriate probabilities so that TCP per-flow fairness is improved. Due to the complexity of wireless networks, the performances of the above methods are greatly affected by topology of network and the competition flows, and when the network state changes frequently, the classification accuracy becomes worse and that in turn may cause low utilization of network bandwidth. Furthermore, the deterioration of performance is more obvious when the quality of the wireless link gets worse.

In this paper, we propose a novel end-to-end packet loss discrimination method using wireless channel model (EPLD-CM) that can differentiate congestion packet loss from wireless random bit error packet loss over hybrid networks. This work analyses the dynamic change characteristics of wireless communication network from the transport/application layer, and to achieve a better end-to-end QoS by making adaptive adjustment according to these changes. The proposed scheme shows to be more accurate than existing methods in estimating current network status by means of the wireless channel model and statistical analysis of large and small packets loss rates, and its performance basically not affected by the variation of network topology and the competition flows. The real-time multimedia transmission protocol can carry out performance optimization based on the packet loss discrimination results.

The rest of the paper is organized as follows. Section II describes the basic principles of the packet loss differentiation method by use of the wireless channel model and special packet transmission mode. Section III discusses updating of the algorithm to packet loss queue. The rate control mechanism and implementation details of EPLD-CM are described in Section IV. Section V presents the NS2 simulation results with comparisons to previous algorithms. Finally, we conclude this work in Section VI.

II. RELATION ANALYSIS BETWEEN WIRELESS CHANNEL MODEL AND PACKET LOSS TYPES

Previous research indicates that the loss probability caused by random bit errors is related to the packet size in wireless environment [10], namely, the larger the packet size, the higher the packet loss probability. The congestion packet losses (drop-tail router), however, are generally independent of the packet size [11]. Based on this observation, one can send small probing packets regularly to the channel to distinguish from the large-size video packets, and estimate current main cause of packet losses from their feedback information. However, channel bandwidth utilization will be reduced by these probing packets. Therefore, we use small-size video packets instead of probing packet to improve channel bandwidth utilization. A method for identifying packet loss reason from the packet loss rates of large and small packets is developed below.

In order to identify between congestion packet losses and wireless fading losses, the sender node sends large and small packets alternately and the statistics of the lost large and small packets over a period is calculated at the receiver end. For ease of analysis, let us define the following variables:

N^s = the number of lost small packets;
$N^s{}_c$ = the number of lost small packets due to congestions;
$N^s{}_b$ = the number of lost small packets due to bit errors;
N^l = the number of lost large packets;
$N^l{}_c$ = the number of lost large packets due to congestions;
$N^l{}_b$ = the number of lost large packets due to bit errors.

According to traditional communication theory, the wireless packet loss rate varies exponentially with the channel bit error rate (BER). In some cases such as uniform distribution with a very small BER, the variations of packet loss rate with BER can be approximated by a linear function [12]. For the wired channel, as mentioned above, the numbers of large and small congestion packet losses in the given period would approximately be equal, namely $N^l{}_c = N^s{}_c$. Thus, we have the following equations:

$$N^l = N^l{}_c + N^l{}_b \quad (1)$$

$$N^s = N^s{}_c + N^s{}_b \quad (2)$$

Let β denote the ratio of $N^l{}_b$ to $N^s{}_b$ in certain wireless channel condition. Obviously, β is a function of the channel bit error rate BER and can be expressed as:

$$\beta(BER) = N^l{}_b / N^s{}_b \quad (3)$$

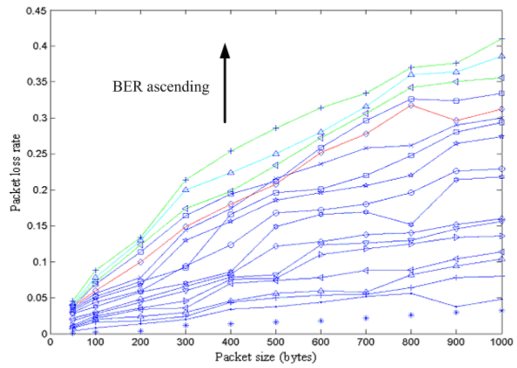
Thus, equation (1) can be rewritten as:

$$N^l = N^s{}_c + \beta(BER) \cdot N^s{}_b \quad (4)$$

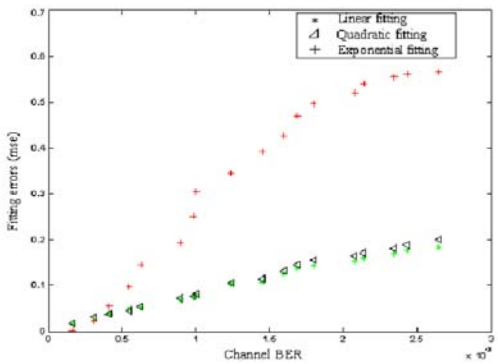
At the receiver end, one can obtain the statistical values of N^l and N^s over a time interval. If β is known, one can get the values of $N^l{}_c$ ($=N^s{}_c$), $N^l{}_b$ and $N^s{}_b$ by solving above equations, namely, the congestion loss rate and random erroneous loss rate of large and small packets respectively. One can then know current congestion level of the wired networks from $N^l{}_c$ and $N^s{}_c$, and fading condition of the wireless link from $N^l{}_b$ and $N^s{}_b$.

The problem now is how to know the value of β, and there seems very few works have been done on this issue. In [10], β was assumed to be a constant, dependent on the sizes of large and small packets. This however is not always true according to our simulation results (see below). Therefore, by taking this assumption in solving equations (2) through (4) for other variables, the applicable scope of the obtained values will probably be rather limited.

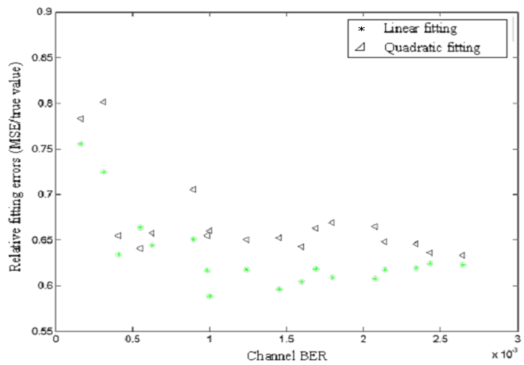
To analyze the relations between packet loss rates and the packet sizes, a group of simulation tests in wireless channels have been carried out by using a modified Jakes Rayleigh fading model [13] in different channel BERs (Bit Error Rates) and packet sizes with Matlab. The simulation results are shown in Fig. 1(a). Linear, quadratic and exponential curves are used to fit the obtained simulation data, and their fitting errors (MSE, Mean Square Error) are shown in Fig. 1(b), where the exponential fitting has the largest fitting errors. Fig. 1(c) depicts the relative fitting errors (MSE/true value) of the linear and quadratic functions. From these results one can see: 1) in certain BER and packet size ranges, the packet loss rates can be seen as linearly related to the packet sizes; 2) for either linear or quadratic fittings, their relative fitting errors are generally rather large (mostly above 60%).



(a)



(b)



(c)

Figure 1. Simulation tests of the correlation of packet loss rates with packet sizes under different channel BERs. (a) Original simulation measurements; (b) Fitting MSE errors of three different forms; (c) Relative fitting errors of the linear and quadratic functions.

Let us analyze the possible values of β. Let G denote packet length, and assume the random bit error of a wireless channel is subject to uniform distribution and any two bit error events are uncorrelated. The packet loss error rate P of wireless link can thus be computed as

$$P = 1-(1-BER)^{G} \quad (5)$$

If BER is very small, by Taylor expansion in BER = 0 and omitting higher order terms, the linear (first-order) and second-order approximation of equation (5) can be obtained as follows,

Linear approximation (first order): $P \approx BER * G$ (6)

Second-order approximation:

$$P \approx BER * G(1 - BER * G/2) \quad (7)$$

If the linear approximation is used, we have,

$$P^{l}/P^{s} = G^{l}/G^{s} = b \quad (8)$$

Where P^l and P^s are the loss rate of large and small packets respectively; G^l and G^s are the lengths of large and small packet respectively; b denotes the ratio of the large packet length to small packet length. β in this condition is a constant (=b), which is frequently used in previous literatures. However, when BER is not small enough, the above equation is not applicable. At this time, one can consider second-order approximation and the ratio can be obtained from equation (7) as,

$$P^{l}/P^{s} = b\frac{2 - BER * b * G^{s}}{2 - BER * G^{s}} = b \cdot \alpha(BER) \quad (9)$$

Where $\alpha(BER) = \dfrac{2 - BER * b * G^{s}}{2 - BER * G^{s}}$ (10)

When $b = 2$, the above equation becomes,

$$\alpha(BER) = \frac{1 - BER * G^{s}}{1 - BER * G^{s}/2} \quad (11)$$

From above analysis, we can see that, when BER is not small enough, P^l/P^s is not a constant but determined by α. In most practical settings, the reasonable value scope of α in equation (11) is 0.6~1 on condition that $BER < 4/7G^{s} = 0.571/G^{s}$ (In fact, this is not a necessary condition). For example, when G^s=4000bits(500Bytes), then $BER < 1.4\times10^{-4}$, and when G^s = 800bits(100Bytes), $BER < 7.1\times10^{-4}$. That is to say, the bit error rate is kept in a small or medium value range. Otherwise, the second-order approximation (equation (7)) may not be applicable. α(BER) monotonically decreases for a given value G with the increase of BER.

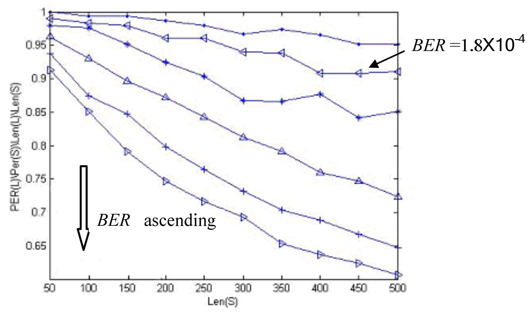


Figure 2. $Y = P^l/P^s/b$ versus G^s under different BER conditions when $b = G^l/G^s = 2$.

Fig. 2 presents a group of emulative curves of $Y = P^l/P^s/b$ by using the modified Jakes Rayleigh fading model. As shown in the figure, the Y value changes very little with the packet size and closes to 1 when $BER < 1.8\times10^{-4}$. With the further increase of bit error rate BER, the Y value shows a larger variation with the packet size.

On the other hand, let us define the following variable,

$$\begin{aligned}\Delta N &= N^l - N^s = N^l{}_c + N^l{}_b - N^s{}_c - N^s{}_b \\ &= N^l{}_b - N^s{}_b = \Delta N_b \quad (\text{Since } N^l{}_c = N^s{}_c)\end{aligned} \quad (12)$$

From equation (12), we can know that the difference between the number of lost large and small packets is equal to the difference between the number of lost large and small packets due to bit errors. The simulation curves (Fig. 1(a)) show that the wireless packet loss rate P increases linearly with packet size under certain BER ranges, namely,

$$\Delta P \approx k \cdot \Delta G \quad (13)$$

Where, $\Delta P = P^l - P^s$, $\Delta G = G^l - G^s$, and k(> 0) is a BER-related constant. Since ΔP can be calculated from equation (12), and ΔG is known in advance, constant k can be obtained, that reflects the degree of link bit errors. Fig. 3 presents the emulative curve of $\Delta P/\Delta G$ under different BER conditions (b=2).

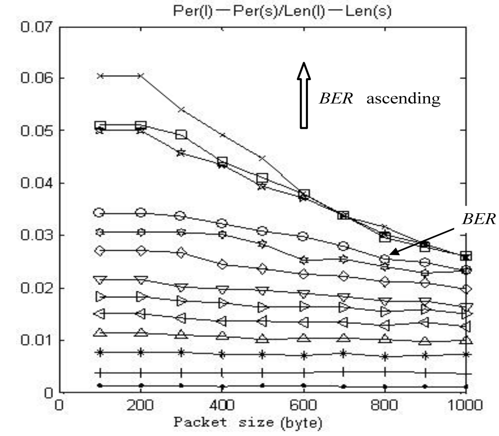


Figure 3. The slope of packet loss rate curve versus packet size under different channel BERs ($\Delta P/\Delta G$ vs. G^s) (b=2).

As shown in Fig. 3, the slope of packet loss rate curve decreases with packet length increase when BER is relatively high; while the curve slope almost keeps stable and uncorrelated to packet length when BER is small, namely, the packet loss rate varies linearly with packet length under small BER conditions. Therefore, one can estimate the current level of BER from k within the scope of $BER < 1.8\times10^{-4}$.

Based on above analyses, we propose the following computation strategy:

1) When the packet loss rate rises to an unacceptable level, the rate control mechanism will decrease the sending rate. Thus, (1) If this high loss rate is due to congestions of the wired link, then as discussed above, the loss rate will drop considerably within several RTTs; (2) If however, the loss rate doesn't show any obvious drop within the time interval, one may regard present packet losses most probably due to the bit errors of the wireless link rather than network congestions.

2) If present packet losses are indeed mainly due to bit errors of the wireless link, we may assume the number of congestion packet losses $N^s{}_c$ in (2) and (4) much less than the number of erroneous packet losses $N^s{}_b$, and $N^l{}_c << N^l{}_b$. Equations (2) and (4) can then be simplified as:

$$N^s \approx N^s{}_b \quad (14)$$

$$N^l \approx \beta(BER) * N^s{}_b \quad (15)$$

From above equations, one can estimate the value of β, and at the same time identify the main cause of current packet losses. Note that in our method, there is no assumption of the linear correlation of the erroneous packet loss rate with the packet size.

3) A linear prediction with error correction method is adopted to estimate the value of β. If present packet losses are mainly due to bit errors of the wireless link, then

$$\beta_t = d\beta_{t-1} + (1-d)\Delta\beta_t \quad (\beta_0 \text{ is the initial value}) \quad (16a)$$

Otherwise, $\beta_t = \beta_0$ (16b)

Where $\beta_0 = G^l/G^s$; $\Delta\beta_t = N^l/N^s$ is the prediction error corrective value; d is a weighting coefficient ($0 \le d \le 1$) that reflects the weights in different time and is set up 0.9 in our NS2 simulation; t＝1， 2， ⋯， denotes sampling time with sampling interval $\Delta t = q\bullet$RTT (q>0, a constant).

As mentioned above, β is no longer a constant when wireless random bit errors are high. Therefore, we adopt the linear prediction with error correction (corrective value $\Delta\beta_t$) method to gradually track the value of β as shown in equation (16a). The weighting coefficient d determines tracking speed of β_t to the real value of β that is not a constant, and the selection of d should compromise between tracking speed and stability of β. Equation (16b) represents the case of low BER of wireless link, where packet losses are mainly due to network congestion and β can be approximated by a constant (see equation (8)).

III. Updating to Packet Loss Queue

A loss queue of 32 packets is used in EPLD-CM scheme. If the length of queue is too long, the response to the change of the network state will be slow, although the sending rate can be kept relatively stable. On the other hand, if the queue is too short, the sending rate may become fluctuating. Further experiments show that, the packets congestion losses rates are relatively small and the queue cannot update quickly under various channel states. In order to solve this problem, we introduce an update factor M to the EPLD-CM algorithm. When M continuous packets are received, we alternately insert one lost packet into queue, whose type is opposite to the current receiving packet. Inserting large and small packets alternately can reduce the response time to the change of network states, since the EPLD-CM algorithm can make correct judgment of current network states.

Following factors are considered in choosing M:

1) M is an odd number, insert large and small packets alternately.

2) $M > 1/per_B$, the loss queue of bad state will not be influenced by inserted M packets.

3) M is not too big; the queue can adapt the variety of state in time.

As is discussed above, the M is a function of per_B and the time t of keeping and is given by $M = f(per_B, t)$. Subject to:

$$M \in [M_{\min}, M_{\max}] \quad \text{(M is an odd number)} \quad (17)$$

$$M_{\min} = \max[1/per_B, 3] \quad (18a)$$

$$M_{\max} = \frac{at * n}{w} \quad (18b)$$

$$a = \min[3w/nt, \pi_G/\pi_B] \quad (18c)$$

The terms per_B and t represent packet loss rate at channel bad states and duration of state respectively. a describes weight with M in duration of state. N denotes the number of sending packets per second and W represents the length of loss queue. π_G, π_B characterizes the ratio of good states or bad states to total time. In experiment, we can calculate the range of M by equation (17) and select a value of M. Because of $per_B > 1/3$ in experiment, we consider 3 as a minimum value of M to guarantee packet of the loss queue all updated. In simulations, the loss queue size is 32, the number of transmitted packets per second is 25, a=2.3 and duration time of state is 10. We can calculate the value of M over the above parameters and the result is 18. We select the odd numbers of the scope 3, 5, 9, 17 and ∞ which do not adopt updating factor as the value of M and simulate in Wireless Last Hop (WLH). We set bandwidth of wired shared link 260kbps. The simulation results are shown in Tab. I. The table shows the ThroughPut (TP) and Congestion Error Rate (CER) for various M values. RER denote Random Error Rate, and PER denote total Packet Error Rate in the table. From Tab.1 we can see that the EPLD-CM algorithm has higher throughput and lower packet loss rate when M is 5. M=5 is used in the following.

TABLE I. RELATION OF UPDATING FACTOR M AND THE EPLD-CM ALGORITHM PERFORMANCE.

M	TP(bps)	TP (%)	RER(%)	CER (%)	PER (%)
3	246.226K	94.70	4.94	0.37	5.30
5	248.276K	95.49	4.40	0.19	4.59
9	238.436K	91.71	6.23	0.65	6.88
17	237.608K	91.39	7.52	0.91	8.43
∞	243.328K	93.59	4.70	1.54	6.24

IV. ADAPTATION RATE CONTROL SCHEME

A. *Packet Loss Discrimination*

For timely response to the change of network status, a finite length history record based large and small packet loss statistics is used. In a sliding window, N^s and N^l record the lost number of small and large packets respectively. When a new packet loss occurs, the oldest recorded one will be removed from the record queue, and the statistical value will be updated correspondingly. The length of lost packets record queue represents a compromise between response time and control stability. In order to track the change of the network states effectively, we introduce another statistic P_{th} in equation (19) to help determine the current network states.

$$P_{th} = \frac{N^s{}_c + N^l{}_c}{N^s + N^l} \quad (19)$$

The statistic P_{th} reflects the ratio of the number of congestion packet losses to the number of total lost packets. Therefore, the value of P_{th} also reflects the congestion/wireless packet loss status of current networks. The larger the value is, the more serious the congestion packet losses.

When the receiver detects a packet loss event, it calculates the P_{th} first, then judges whether it is a congestion packet loss by comparing the value of P_{th} with a given threshold Tsh ($0 \le \text{Tsh} \le 1$). When P_{th} is greater than Tsh (P_{th}>Tsh), we judge that current packet loss type is congestion loss and record the loss packets; otherwise current packet loss type is judged as random loss and we do not record the loss packets. At every certain time, the receiver end feeds back the loss rate P_c (the ratio of the recorded loss packets to the total received packets) and round trip time to the sender. Thus, the packet loss rate P_c obtained from the RTCP feedback packets at the sender reflects only the congestion packet loss rate, not including the wireless packet losses. And the calculated sending rate will be fit the current network situation well. We generally set Tsh=0.8 after running a number of simulation tests. In this way, we can track current network status more quickly, and at the same time achieve good control stability.

B. *EPLD-CM Behavior Description*

To satisfy the TCP friendliness requirement, an adaptive rate control based on TCP throughput model is used in this paper. The sending rate R is calculated by equation (20) [6].

$$R = \frac{LDU}{RTT\sqrt{\frac{2P_c}{3}} + T_{out}(3\sqrt{\frac{3P_c}{8}})P_c(1+32P_c^2)} \quad (20)$$

Where, LDU denotes length of data unit, RTT denotes round trip time, T_{out} denotes retransmission timeout time which is set T_{out} = 4 $RTTs$ in our simulation, P_c denotes packet loss rate. Equation (20) works well in wired IP environment, but in wireless environment, the performance of conventional TCP is often unsatisfactory. Therefore, we modify this model a bit by removing the number of lost packets due to wireless bit errors in the calculation of P_c. Thus the capability to differentiate the packet loss types will be essential.

The proposed EPLD-CM algorithm is based on RTP/UDP transport layer protocol, and works in following steps:

A. Initialization: after setting the initial sending rate, the parameters of α, b, d, q, G^l, G^s, and the length of record queue of packet losses, the algorithm enters a slow start stage. When the media streaming gets into a stable state when packet loss occurs, the receiver end computes the statistical values of N^l, N^s, RTT, and P_c (recording the congestion packet loss rate only).

B. The receiver end calculates the value of P_{th} by equation (19) for every packet loss event, and estimates the reason of the current packet loss. If P_{th}>Tsh, the packet loss is judged as the congestion loss, otherwise, the wireless packet loss. If it is a congestion packet loss, it will be used to update the packet loss rate P; otherwise it will not be. At every sampling time point t, β_t is updated by equation (16) according to the nature of the current packet loss. At every certain time interval (every 4 RTT times), the receiver informs the sender the RTT and packet loss rate P by RTCP feedback packets.

C. If there are packet losses occurred during the feedback period, when the sender receives the RTCP feedback packets, rate control scheme [14] will be carried out according to the equation (20). The whole adaptive rate control scheme is described in Algorithm 1.

Algorithm 1: The adaptive rate control scheme
1. Initialization
2. Updating N^s, N^l
3. for {receive a ACK packet} do
4. Compute P_{th} by equations (19) and (1)-(4)
5. if P_{th}>Tsh
6. The current packet loss is the congestion loss, and update the packet loss rate P_c
7. else then
8. The current packet loss is the wireless loss, and increase MAC retransmission count
9. end else
10. end if
11. At a certain time interval, updating RTT and P_c, and computer R by equation (20)
12. $0.75\times R_{current} < R < 1.25\times R_{current}$
13. The next transmission rate $R_T = R$
14. if $R_s > 1.25\times R_{current}$
15. The next transmission rate $R_T = 1.2\times R_{current}$
16. if $R_s < 0.75\times R_{current}$
17. The next transmission rate $R_T = 0.9\times R_{current}$
18. end if
19. end if
20. end if
21. end for

V. Simulation Results

To achieve these goals, we evaluate these algorithms via simulation using NS2 in this section. We study the performance and differentiation accuracy of the EPLD-CM under networks with wireless last-hop links.

Fig. 4 shows a typical wireless last hop network, where the last hop to the receiver is a wireless link with bandwidth 150Kb and 10ms delay, and the other parts of the network are wired links with similar parameter settings as in [6]. The bandwidth of the shared wired link is 260Kb when one or two flows are transmitted. When multi-flows are transmitted simultaneously, the bandwidth is 86% of the aggregated total bandwidth of the wireless links. The delay of shared link is 60ms. The bandwidth of all the wired LAN links is 10Mb, and the delay is 1ms. So when multiple flows were transmitted in parallel, the wired shared link is the bottleneck. When only one media flow in the network, we set the capacity between routers base station1 and base station2 roughly twice the wireless link capacity so the bottleneck is the wireless link. This topology simulates a cellular network system and satellite Direct-TV system. For simplicity, wireless nodes are static in the simulation.

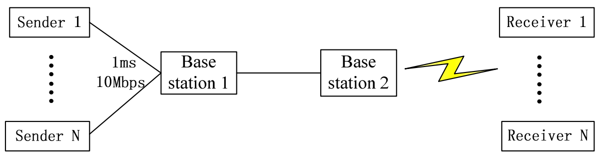


Figure 4. A typical WLH simulation topology.

The modified Gilbert wireless channel model [13] is used in our experiments to simulate the wireless loss patterns, with wireless channel bytes error rate = 0.0001, 0.0002, 0.0004 and 0.0008. The IEEE 802.11b is used at the wireless MAC/PHY layer. We then study the EPLD-CM under various scenarios of competing traffic where multiple flows use the same EPLD-CM. All the data are the average of ten runs. We compared the EPLD-CM algorithm with TCP, Spld, Zigzag and GOD algorithms in three major performance metrics: throughput, congestion packet loss rate, and wireless loss misclassification rate M_w (M_w is defined the percentage of misclassifying wireless loss as congestion loss) for one and four flows respectively. We also assessed the TCP-friendliness of the proposed algorithm, and reported part of the results in our previous publication [15]. GOD is a hypothetic god algorithm, which can differentiate the congestion and wireless packet losses correctly and is used only for comparison. The packet size is 762 bytes for TCP, Zigzag and Spld. The large packet of EPLD-CM is 1016 bytes and the small packet size of EPLD-CM is 508 bytes. Therefore, b=2 and the average packet size of EPLD-CM is 762 bytes. The packet loss history length is 32. Each simulation runs for 400 seconds.

A. For one flow only

In this case, there is only one of EPLD-CM, Spld, Zigzag, TCP and GOD flow running in the network from Sender1 to Receiver1. The bandwidth of wired shared link is 260kbps and the size of queue in base station1, base station2 is 12 packets. Therefore, the wireless link between base station2 and Receiver1 is the bottleneck. Because the bandwidth of wireless link is only 150kbps

(<<260kbps), the congestion loss occurs in base station2 inevitably.

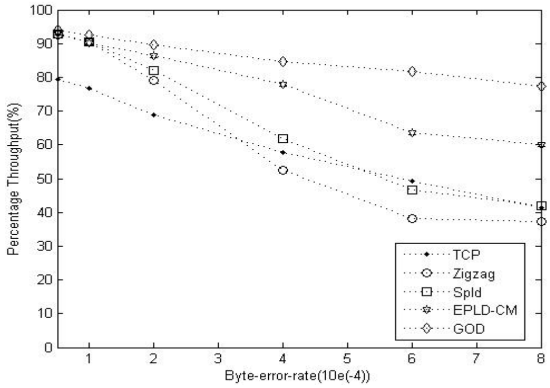


Figure 5. Throughput of one flow with WLH topology.

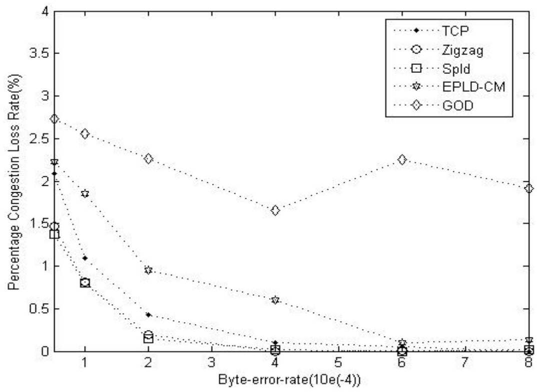


Figure 6. Congestion packet loss rate of one flow with WLH topology.

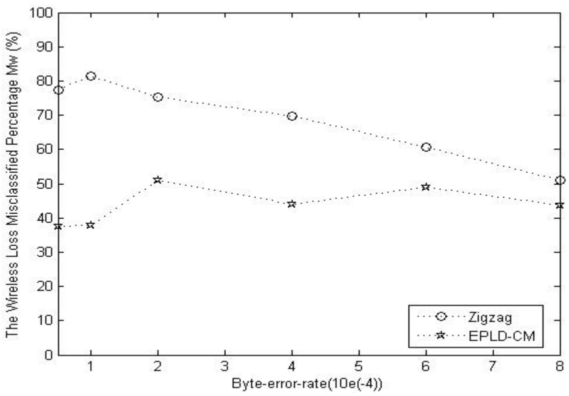


Figure 7. The wireless loss misclassified as congestion loss percentage of one flow with WLH topology.

The simulation results are shown in Fig. 5 through Fig. 7. Fig. 5 shows the throughput with bytes error rate for each type of flows. Fig. 6 shows congestion packet loss rate with bytes error rate for each type of flows. From Fig. 5, we see that the wireless environment becomes worse and worse with the increase of bytes error rate and the average throughput of all the flows has been decreased. TCP had comparatively low throughput. It reacts to wireless losses as congestion losses, unduly reducing its sending rate. Zigzag is more conservative in that it classifies some wireless losses as congestion losses. The Spld algorithm made few mistakes on wireless losses and were designed for this kind of topology. Because of this, they have the same slightly higher congestion loss, while Zigzag have less congestion since it misclassify more wireless losses (see Fig. 7), and therefore reduce sending rate (see Fig. 6). EPLD-CM almost fully utilizes the bottleneck bandwidth and misclassified less congestion losses. Throughput of EPLD-CM is obviously higher than other algorithms except GOD, and its congestion packet loss rate is lower than that of others.

B. Four flows competing

In this case, there are four flows running in the network topology, such as EPLD-CM, Spld, Zigzag, TCP and GOD. We set the bandwidth of wired shared link 520kbps and the queue size in base station1, base station2 is 18. Flow 1 starts at the 1st second from Sender1 to Receiver1, and Flow 2 starts at the 10th second from Sender2 to Receiver2, and Flow 3 starts at the 20th second from Sender3 to Receiver3, Flow 4 is done in a similar way. Therefore, the four streams compete for bandwidth at the common link, and congestion can happen both at the wired shared link as well as at the last wireless link. All flows stop at the 400th second. The bottleneck is now the wired shared link between base station1 and base station2.

The simulation results are shown in Fig. 8 through Fig. 10. From Fig. 8 we see that the average throughput of TCP decreases with bytes error rate. The reason is that flows will experience wireless error at wireless channels. With the increase in bytes error rate, it is likely that wireless loss will be deteriorated and unduly reduce their sending rate. Spld maintains high throughput regardless of the number of competing flows. However, they have high congestion loss because congestion losses at the shared bottleneck link become misclassified as wireless (see Fig. 9). The Zigzag scheme has low throughput and high congestion loss rate. Once a large ROTT is measured due to high buffer levels at both locations, it can no longer correctly gauge individual buffer levels. The high ROTT measured previously will make the scheme miss congestion loss in such case (see Fig. 10). The throughput of the EPLD-CM algorithm is higher than other algorithms except GOD, however, the bandwidth efficiency is also improved. As is shown in Fig. 9, the congestion loss rate of the EPLD-CM algorithm is lower than others with the increase of bytes error rate.

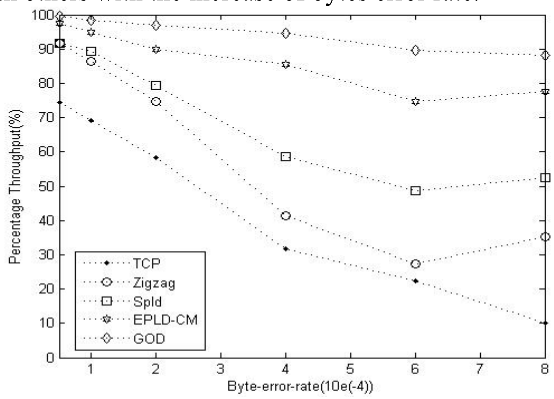


Figure 8. Throughput of four flows with WLH topology.

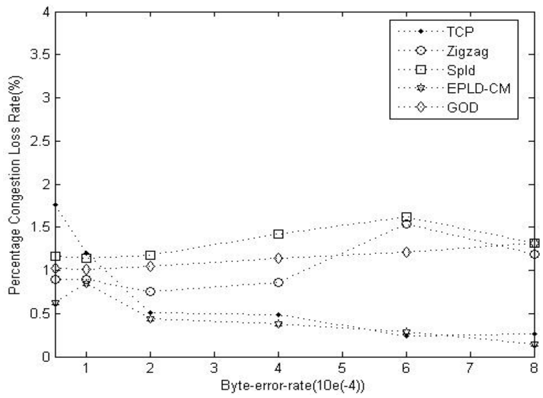


Figure 9. Congestion packet loss rate of four flows with WLH topology.

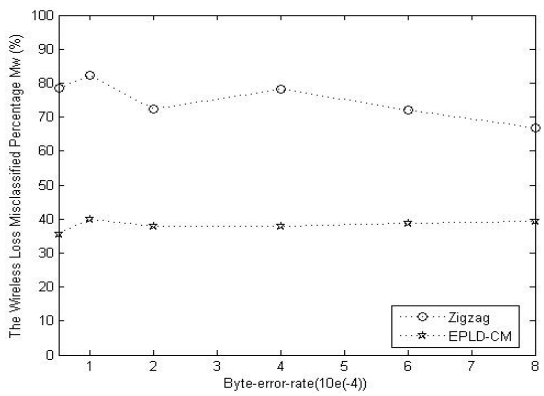


Figure 10. The wireless loss misclassified as congestion loss percentage of four flows with WLH topology.

VI. CONCLUSION

This paper proposes an end-to-end packet loss discrimination method based on channel model over wireless multimedia transmission networks. This scheme can detect the network status and differentiate packet loss types (wireless bit errors or congestion loss) using a wireless channel model. Furthermore, a special packet sending scheme with different packet sizes is adopted. It can adapt to the dynamic change of the networks and control the sending rate effectively. NS2 simulation results have shown that the proposed algorithm is effective in increasing packet delivery ratio and reducing packet loss rate.

The congestion loss is a global problem while the wireless loss is a special problem in wireless links. As the limited available information in upper layers, the control mechanisms of lower layers (for instance MAC layer) can be considered. Therefore, the across layer design is our direction of further studies.

ACKNOWLEDGMENT

This work was supported in part by National Natural Science Foundation of China (No.60972038, 61001077, and 61101105), the Natural Science Foundation of the Higher Education Institutions of Jiangsu Province (No.10KJB510013), and the Scientific Research Foundation for Nanjing University of Posts and Telecommunications (No. NY211007)).

REFERENCES

[1] Neng-Chung Wang, Jong-Shin Chen, Yung-Fa Huang, and Chi-Lun Chiou, “Performance Enhancement of TCP in Dynamic Bandwidth Wired and Wireless Networks,” Wireless Personal Communications, vol. 47, no. 3, pp. 399-415, Nov. 2008.

[2] H.-J. Lee, H.-J. Byun, and J.-T. Lim, “TCP-friendly congestion control for streaming real-time applications over wireless networks,” IET Commun., vol. 2, no. 1, pp. 159–163, 2008.

[3] Tu-Chih Wang, Hung-Chi Fang, and Liang-Gee Chen, “Low-delay and error-robust wireless video transmission for video communications,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 12, no. 12, pp. 1049-1058, Dec. 2002.

[4] A Lahanas, et al, “Improving TCP performance over networks with wireless components using ‘probing devices’,” International Journal of Communication Systems, vol. 15, pp. 495-511, 2002.

[5] Y. Tobe, Y. Tamura, A. Molano, S. Ghosh, and H. Tokuda, “Achieving moderate fairness for UDP flows by path-status classification,” in Proc. 25th Annu. IEEE Conf. Local Computer Networks (LCN 2000), Tampa, FL, Nov. 2000, pp. 252–261.

[6] Song Cen, et al, “End-to-End Differentiation of Congestion and Wireless Losses,” IEEE/ACM Transactions on Networking, vol. 11, no. 5, pp. 703-717, October 2003.

[7] S. Biaz and N. Vaidya, “Discriminating Congestion Losses from Wireless Losses using Inter-arrival Times at the Receiver,” in Proc. IEEE Symp. Application-Specific Systems and Software Engineering and Technology, Richardson, TX, Mar. 1999, pp. 10–17.

[8] Min Kyu Park, Kue-Hwan Sihn, Jun Ho Jeong, “A Statistical Method of Packet Loss Type Discrimination in Wired-Wireless Networks,” in Proc. IEEE CCNC, 2006, pp. 458-462.

[9] Juho Ha, Eun-Chan Park, Kyung-Joon Park, Chong-Ho Choi, “A Cross-layer Dual Queue Approach for Improving TCP Fairness in Infrastructure WLANs,” Wireless Personal Communications, vol. 51, no. 3, pp. 499-516, Nov. 2009.

[10] C. L. Lee, et al, “On the Use of Loss History for Performance Improvement of TCP over Wireless Networks,” IEICE Trans. Commun. , vol. E85-B, no. 11, pp. 2457-2467, 2002.

[11] S. Floyd, et al, “Router mechanisms to support end-to-end congestion control,” LBL Technical Report, 1996, URL ftp://ftp.ee.lbl.gov/papers/collapse.ps.

[12] D. Eckhardt, et al, “Improving wireless LAN performance via adaptive local error control,” in Proc.6th IEEE International Conference on Network Protocols, 1998.

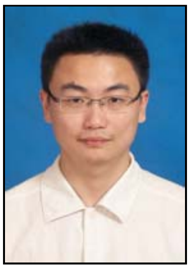
[13] Y. Li, Y. L. Guan, “Modified Jakes model for simulating multiple uncorrelated fading waveforms,” in Proc. IEEE ICC, vol.1, June 2000, pp. 46-49.

[14] Z. W. Zhou, Y. N. Dong, “Research on a transmission control algorithm of the real-time video stream AVTC over IP network,” Computer Research and Development, vol.41, no.5, pp. 812-820, 2004. (in Chinese)

[15] Yu-ning Dong and Meng-yue Chen, “Real time video transmission control in wireless-wired IP networks,” in Proc. IEEE Wireless Communications and Networking Conference (WCNC2007), Hong Kong, Mar. 2007, pp. 3687-3691.

Hai-tao Zhao received his Ph.D and M.S. degrees in signal and information processing from Najing University of Posts and Telecommunications, Nanjing China, in 2011 and 2008. Now he is currently working at Department of Communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China. The main research direction is wireless multimedia communication. (E-mail: zhaohtmail@gmail.com).

Yu-ning Dong received his Ph.D degree in Engineering from Southeast University, China, in 1988, and M.Phil degree from Queens University of Belfast, U.K, in 1998. He worked as British Postdoctoral research at Imperial College University of London, U.K from 1992-1993. He has been engaged in teaching and researching on Multimedia Communications and IP networks. He is full professor now at Nanjing University of Posts and Telecommunications. His research interests include multimedia communications and wireless Mesh networks. (E-mail: dongyn@njupt.edu.cn).



Hui Zhang received his Ph.D. degree in signal and information processing from Nanjing University of Posts and Telecommunications, China, in 2008. He is currently with the Jiangsu Provincial Key Lab of Wireless Communications, Nanjing University of Posts and Telecommunications. His current research interests include ad hoc networks, wireless mesh networks, network coding. (E-mail: zhhjoice@126.com).

Xin-xing Zhao Received the B.Sc degree in 2008 from Huaihai Institute of Technology and now is pursuing his M.Sc degree in the Department of Communication Engineering, Nanjing University of Posts and telecommunications, P.R. China. His main research interest is in the area of wireless Networks. (E-mail: zhaobofu@gmail.com).