AN EFFICIENT TRANSPORT SCHEME FOR MULTIMEDIA OVER WIRELESS INTERNET

Fan Yang[1], *Qian Zhang*, *Wenwu Zhu*, *and Ya-Qin Zhang*
Microsoft research, China
5F, Bejing Sigma Center, No. 49, Zhichun Road
Haidian District, Beijing 100080, P. R. China

Abstract

Multimedia transmission over wireless Internet is a challenging work since the random errors frequently occurred in the wireless links heavily deteriorates the performance of traditional transport protocols such as TCP and UDP. This paper presents a new scheme for delivering multimedia over a connection of which only the last hop is wireless link. Simultaneously error control and congestion control are addressed in this wireless Internet platform. The packet loss caused by Internet congestion and wireless error can be differentiated in our scheme. Fading and high bit error rate (BER) on the wireless link as well as congestion in the wired network are separately controlled. Simulation results demonstrate that the proper congestion control and error control can improve throughput of the connection and the end-to-end quality of delivered media.

1. Introduction

Mobile Internet and Third Generation (3G) wireless multimedia applications and services promise a future world of universal wireless phones, global roaming and wireless Internet access. However, the widely used transport protocols in today's Internet do not fit well in wireless networks. This is mainly due to the special characteristics of wireless networks. Wireless networks usually have a bit error rate (BER) much higher than that of wired networks. The bit error rate in a wired link is normally less than 10^{-12}, while in wireless link it often varies from 10^{-5} to 10^{-3}. Moreover, wireless communication may confront with "fading". Packets transmitted to a mobile that is in a fading period will be lost.

The unreliability in wireless links results in low performances of traditional transport-layer protocols such as UDP and TCP.

- TCP protocol assumes congestion in the network to be the primary cause for packet losses and unusual delay. It will halve the transmitting rate while meet the packet lost. Unfortunately, packets are lost in wireless channel for error rather than congestion, thereby resulting in an unnecessary reduction in end-to-end throughput;
- Both the high BER and frequently occurred fading make packet loss ratio very high during a UDP connection.

Many works have been made to overcome the drawbacks of transport protocols over wireless networks. Some split schemes [1, 2, 3] use local retransmissions at the base station that shield the random errors over wireless link during a TCP connection. Some statistical methods [4, 5] are deployed to discriminate packet losses due to congestion from random errors thus makes current TCP congestion control mechanisms still perform well over wireless Internet. As for UDP, UDP-Lite [6] tries to bypass the checksum policy for some parts of a packet (e.g., payload of a packet) in order to decrease the packet lost ratio. This technique is called partial checksum. It leaves the task of correctness checking for the upper level applications. To deal with the fading problem, M-UDP [7] develops a mechanism for re-transmitting the lost packets after the fading ends. All these works had seldom addressed the following aspects:

- How to make a congestion control for an unreliable transport protocol over wireless Internet;
- Reporting the information of random errors to upper level applications so that they could adjust their error control mechanism.

In this paper we propose a new scheme that can efficiently deliver multimedia over wireless Internet. Packet losses due to congestion or random error can be distinguished in our scheme. Proper congestion control and error control are performed, which are suited for multimedia applications. Featured with retransmission mechanism, our scheme can also reduce the packet loss ratio caused by fading and random errors.

[1] This work is performed while he is a visiting student at Microsoft Research China. He is from State key laboratory of Novel Software Technology, Department of Computer Science, Nanjing University.

2. An End-to-End Architecture for Multimedia Delivery Over Wireless Internet

As mentioned above, to transmit multimedia over wireless Internet, high bit error rate and fading problem will confront over wireless links. In addition, packets loss may occur in the wired network. To efficiently deliver multimedia, the upper applications should be aware of the network conditions. In the meanwhile, appropriate error/fading control and congestion control should be taken.

A scheme for multimedia delivery over wireless Internet is proposed in this paper. Packet losses caused by congestion or by random errors can be discriminated, thus the congestion control and error control can be performed respectively. Network conditions include packet loss ratio and bit error rate (BER) / block error rate (BLER) are reported to multimedia application so that they can adjust their behavior. Retransmission mechanism is incorporated in our scheme to reduce the errors in a packet and the packet losses due to fading.

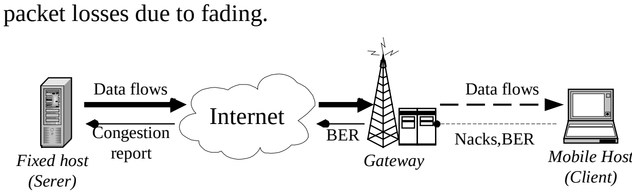


Figure 1. System architecture for media over wireless Internet.

To distinguish the packet losses caused by different reasons, an intermediate node (gateway) at the edge between wired network and wireless link is introduced in our scheme. Application-level error control and congestion control are performed on the sender side based on the feedbacks from the gateway and the mobile host. Wired network and wireless link are treated differently in this scheme, thus requiring an agent running at the intermediate node interacts with the two ends respectively. Considering the delay constraint, packet retransmissions are only performed between the gateway and mobile host.

The system architecture of our scheme is illustrated in Figure 1. During a connection, the fixed host (server) sends data to the mobile host via a gateway. The gateway buffers the data first, and then relays them to the mobile host (client). The mobile host calculates the BER/BLER and informs the server periodically. While a corrupted packet is received, the mobile host would send a negative acknowledgement (NACK) to the gateway if its delay constraint can be satisfied. On receiving this negative acknowledgement, the gateway retransmits the corresponding packet if it is still in the buffer. The gateway also monitors the wired network conditions, and sends the congestion information back to the sender at a regular interval. The TCP-friendly transport protocol we proposed [8] is adopted in the wired network part for the purpose of congestion control.

Three agents are running at the server (fixed host), the client (mobile host), and the gateway, respectively.

2.1 Components in the fixed host (server)

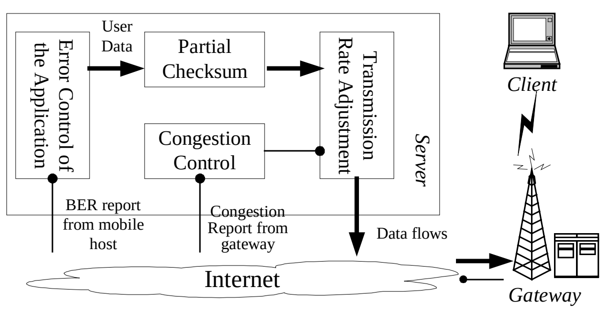


Figure 2. Block diagram of server-side behaviors.

Figure 2 depicts the behaviors of the fixed host. Error resilient media bitstream is protected by certain error control scheme. In the meanwhile, partial checksum policy is applied to the bitstream while performing packetization. Then these packets are sent to the mobile host. Based on the congestion report from the gateway, server adjusts its sending rate. Different level error control schemes are performed on the server side according to the feedback from the mobile host.

2.2 Components in the mobile host (client)

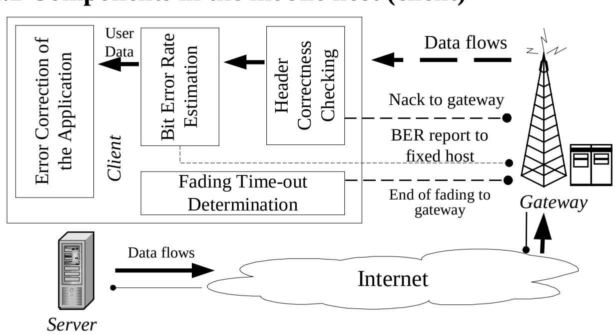


Figure 3. Block diagram of client-side behaviors.

The components of the client are briefly shown in Figure 3. On receiving a packet, the mobile host first checks its correctness. If the packet were corrupted, the receiver would send a negative acknowledgement within the delay constraint on the client side. The client is also responsible for notifying the gateway at the end of fading. At the same time, the BER/BLER over the wireless link is calculated in the mobile host and reports it to the fixed host periodically.

2.3 Components in the gateway

How the gateway interacts with the server and the client is illustrated in Figure 4. The gateway caches every packet received from the server and passes it to the client. It calculates the packet loss ratio during a specified period and then informs the congestion information to the server. On receiving the NACKs or end of fading notifications from the mobile host, the gateway first finds out what packets it should retransmit and performs packets retransmission based on certain weighted scheduling algorithm.

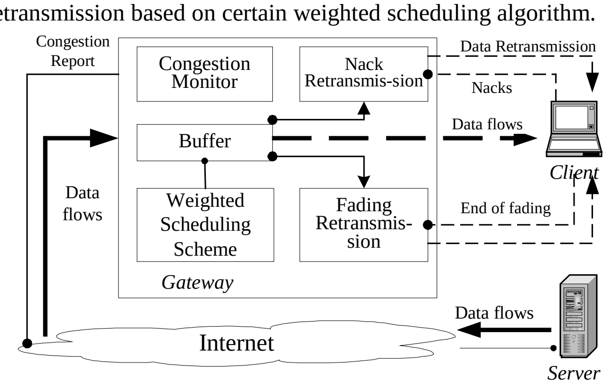


Figure 4. Block diagram of behaviors in the gateway.

Three major functionalities of our scheme include error control, fading retransmission, and congestion control. The error control involves with the server and client, of which the server adjusts its error protection level based on the BER/BLER report from the client. Fading retransmission is accomplished by the collaboration between the gateway and the mobile host. While congestion control can be performed by adjusting the sending rate on the server side based on the feedback from the gateway.

3. Error Control

To overcome the high BER/BLER over the wireless links, an error control mechanism is designed as follows.

Partial checksum

A wireless link usually has a much higher bit error rate than that of wired network. Based on the checksum policy, traditional UDP will discard every packet that contains errors, even in a single bit error case. The multimedia applications, however, can tolerate certain level of errors. Thus it is inefficient if those corrupted packets cannot be delivered to applications. With this in mind, the partial checksum policy is incorporated in our end-to-end architecture. It allows users to implement checksum for certain parts of packet user defined. Thus, when some errors occur in the non-partial-checksum parts of a packet, it can still be delivered to the upper application. How to handle this corrupted packet is left to the application. By using partial checksum, packet loss ratio can be decreased dramatically.

BER/BLER report generation

The mobile host should estimate bit error rate periodically and report it to the server. On receiving the BER/BLER information, the server informs it to the upper-level application. If the feedback information indicates a high BER/BLER, the application enhances the error control and vice versa. Note that the error control is performed at application level, because different applications usually have different error control mechanisms. For example, the application can perform channel-adaptive UEP (unequal error protection) according to the channel and media characteristics [9]. The following is the details of calculating the BER/BLER of a wireless link. The packets transmitted over the wireless link only have their header checksumed.

Suppose the probability of a single bit error is b, the length of the packet header is hl. If no dependency is considered, the packet error probability (P_e) can be simply calculated using the following equation: $P_e = \frac{i}{n}$, where n is the number of the transmitted packets during a specified period of time, and i is the number of the corrupt packets. Taking the error dependency into consideration, other similar formula can be used. Since the relationship between BER (b) and P_e can be formulated by $P_e = 1-(1-b)^{hl}$, we obtain b as $b = 1-(1-P_e)^{1/hl}$.

Handling corrupted packet

When mobile host receives a corrupted packet, it may send a negative acknowledgement to the gateway if the delay constraint for that packet can be satisfied. It is known that multimedia often is delay sensitive, if the corresponding delay bound is exceeded, there is no need to send request for the retransmission. Notice that the sequence number of the corrupted packet is sent with the NACK. On receiving this negative acknowledgement, the gateway should retransmit the packet if it can be found in the buffer.

4. Dealing with Fading

Neither the gateway nor the mobile host can know the beginning of fading. However the mobile host can observe the end of fading. A timer to indicate fading is running on the mobile host side during a connection. Every reached packet resets this timer. If the timer runs out, the mobile host considers the fading happening. The mobile host should notify the gateway immediately as soon as it comes out of fading. In the out-of-fading notification, mobile host should tell the gateway which packet has been lastly received before fading and which packet has been firstly received after fading, so that the gateway can find out what packets in the buffer should be retransmitted.

Since the gateway cannot be noticed when the fading happens, it continuously transmits the packets to the mobile host. But it can deduce that fading has happened when it receives the end-of-fading notification from mobile host. On receiving that, the gateway retransmits the packets in its buffer according to indications in the end-of-fading notification packet.

Figure 5 describes the flows of processing retransmission. As mentioned above, the gateway relays the packets from the server to the client. In the meanwhile, it puts every packet that has been transmitted into a buffer. Suppose the fading happens after the gateway has relayed packet 2 and ends when the receiver get packet 7. So the mobile host knows the fading has happened and sends an “eofn” (end of fading notification) to the gateway with the sequence no 2 and 7 in it to indicate the beginning and ending of the fading, respectively. On getting the “eofn”, the gateway first finds the packets specified the “eofn” in its buffer, and then retransmits those packets (3, 4, 5, 6 in this case) to the mobile host. During the retransmission phrase, the gateway should cache the packets (8, 9, … in this case) from the server simultaneously. After the retransmission phrase, the gateway should first deliver the packets cached in the buffer during the retransmission phrase.

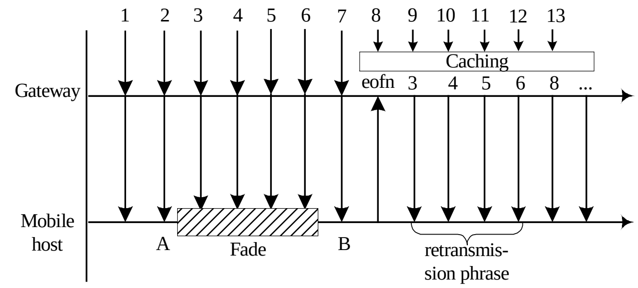


Figure 5. Fading retransmission procedure.

5. Congestion Control

The fixed host adjusts its sending rate in response to the feedback information from the gateway, which indicates the packet loss ratio caused only by congestion. Considering the multimedia delivery requirement, the sending rate should be smoothly adjusted. Our proposed TCP-friendly transport protocol [8] is used here for congestion control. On receiving the feedback information, the server estimated the available bandwidth by the round trip time, packet loss ratio and retransmission time out. The sending rate is smoothly adjusted by the estimated available bandwidth and historical information.

The gateway estimates the packet loss ratio using the method described in [8]. After certain period of time, it reports the congestion information to the fixed host. Note that the packet loss ratio is not that between the server and the gateway, but that between the server and the client. As described above, the gateway is the last hop of the connection between the fixed host and the mobile host. If the bandwidth between the fixed host and the gateway is larger than that between the gateway and the mobile host, packet loss will happen at the wireless link between the gateway and the mobile host. So it would be appropriate to estimate the packet loss ratio between the fixed host and the mobile host.

Based on the client’s feedback packet, the server can adjust the current round trip time (RTT) using the following equation:

$$RTT = \alpha \times \overline{RTT} + (1-\alpha) \times M,$$

Where $\overline{RTT}$ is the current round trip time, RTT is the estimated round trip time, α is a weighting parameter, and M can be calculates as follows.

$$M = now - ST - \Delta RT + 2 \times psize / rate,$$

where *now* is the timestamp indicating the time at which the congestion information was received in the server, ST is the time at which the server is about to send the packet to the mobile host, ΔRT is the time interval between the packet that is being sent to mobile host and the congestion information that is about to be sent to sender, *psize* is the packet size, and *rate* is the bandwidth of the wireless link. Note that the calculation of ΔRT is different from that in [8]. It’s not only the interval during which a packet stays in the gateway, but also the time spent in the last wireless link.

Moreover, the management of buffer at the gateway should also be taken into account. The buffer caches all data from the fixed host in the order that the gateway receives them. When retransmission is needed, transmitting the packets locally (over wireless link) can reduce the delivery time and packet loss ratio. During the retransmission of packet, the gateway should cache the packets from server at the same time. Considering the different media priorities, the weighted scheduling scheme can be adopted here to control the buffer so as to improve the end-to-end quality.

6. Simulation

6.1 Simulation environment

To demonstrate effectiveness of our scheme, we use network simulator version 2 (NS2) to study the transport scheme. Figure 6 illustrates the network topology of the simulation. A single shared bottleneck link with the bandwidth of 0.75Mbps is used in this topology. The dash line represents a wireless link. It has the bandwidth of 0.2Mbps and delay of 100ms. All links are configured as droptail links. Except for the wireless link and bottleneck link, the bandwidth of all other links is 10Mbps. To characterize the wireless environment, we add two error models to the wireless link to respectively simulate the bit error rate and fading. In the simulation, the background traffic consists of three infinite-duration TCP-Taheo connections.

For simplification, the delay constraint of a packet has not been checked when re-transmitting the packet. In addition, the weighted scheduling algorithm for the buffer management has not been applied. Instead, FIFO algorithm is used to control the buffer in the gateway. The buffer size is chosen largely enough to accommodate the packets sent from the fixed host during fading and packet retransmission phase.

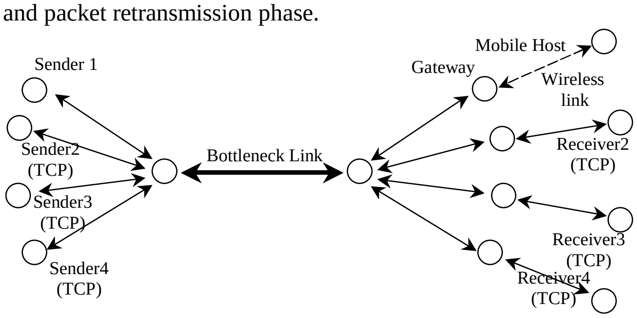


Figure 6. Simulation topology.

To demonstrate the effectiveness of partial checksum and congestion control, in this simulation, we tested: (1) TCP Tahoe; (2) UDP; (3) our scheme without partial checksum (denoted as WMSTFP/PC); (4) our scheme without the partial checksum and the congestion control (denoted as WMSTFP/PC&CC); (5) our scheme (denoted as WMSTFP).

To define a fair sending rate for the schemes (2) and (4), we first run the simulation of our scheme, then calculate the average sending rate during the simulation based on the throughput of our scheme, and finally use this rate for schemes (2) and (4) in the simulation.

The metrics of the performance are throughput (shown in number of packets received during simulation time) and packet loss ratio (shown in packet received rate).

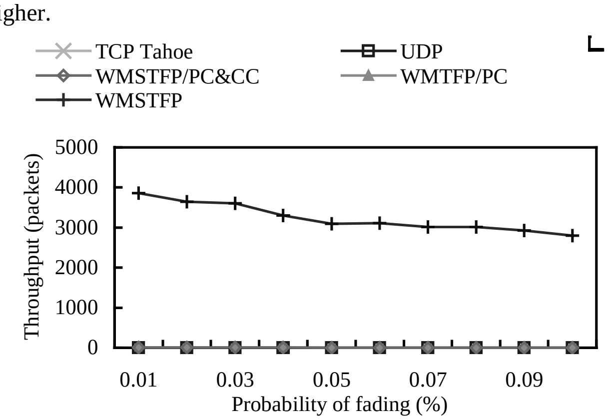
6.2 Simulation results

With the packet length of 1000 bytes and packet header length of 40 bytes, we measured the throughput and packet loss ratio for different schemes under different fading probability and bit error rate during 300 seconds. Here the fading is defined as not receiving any packet within 0.4 second.

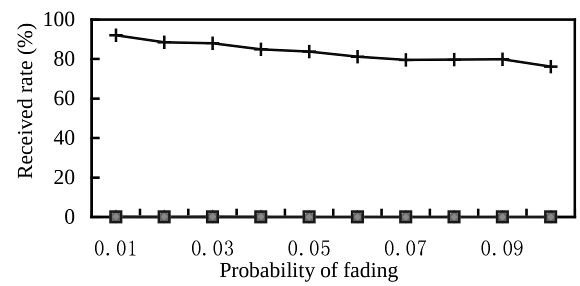
Figure 7 shows the performance comparison results under different BER (10^{-3} to 10^{-5}). The fading probability varies from 0.01 to 0.1 in this case. It can be seen at under high BER (10^{-3}), the throughput and received rate of all the other schemes except ours are extremely low while our scheme still has about 75%-91% received rate and the throughput of about 3000-3800 packets. This is due to the partial checksum policy of the packets in our scheme. The packet loss ratio can be calculated using equation $P_{lost} = 1-(1-b)^{length}$, where P_{lost} is the packet loss ratio, *length* is the length in a packet with checksum policy. If BER is 10^{-3}, the packet loss ratio for packets without partial-checksum (*length* is 8000 bit) is greater than 99.96%, while the packet loss ratio for packets with partial-checksum (*length* is 320 bit) is only 27.40%. With the aid of re-transmitting the corrupted packets, our scheme can achieve 90% packet received rate and about 3800 packets throughput. One thing needs to emphasize is that with the partial checksum policy all packets received on the mobile host side may not be correct. Whether they can be used is up to the application's error control and error correction mechanism.

As the BER is low enough (e.g., 10^{-5}), it can be seen that the probability of fading is the dominating factor affecting throughput and received rate. With the increasing of fading probability, the received rate of UDP falls down quickly. While the other three schemes with fading retransmission mechanism fall a little bit lower.

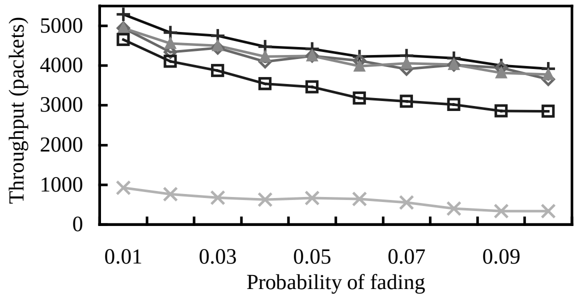
Figure 8 shows the performance comparison results with varying BER. The fading probability is fixed to 0.05 in this case. It can be seen that the throughput falls down when BER increases higher because the overhead must spend to retransmit the corrupted and lost packets. The packet received rate falls when BER increases, because higher BER makes the possibility of dropping NACK packets, and end of fading notification packets higher.



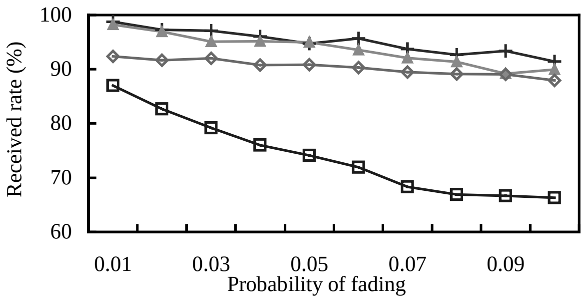
(a)



(b)



(c)



(d)

Figure 7. Comparison results of five schemes.
(a) Throughput comparison at BER 10^{-3}. (b) Received rate comparison at BER 10^{-3}. (c) Throughput comparison at BER 10^{-5}. (d) Received rate comparison at BER 10^{-5}.

7. Conclusion

This paper addresses how to efficiently deliver multimedia over wireless Internet. A transport scheme is proposed that can achieve this goal by:

- Deploying a split architecture to distinguish the congestion and error status in wireless Internet connections;
- Reducing the fading effect in the wireless link through retransmitting cached data in the gateway near base-station;
- Improving both the throughput and the received rate from end to end using partial checksum and proper congestion control;
- Leaving decision on how to correct the corrupted packet to application so that media characteristics can be considered to achieve better efficiency;
- Applying differential scheduling scheme to different prioritized media in the gateway so that end-to-end quality can be improved.

Simulation results show that our scheme performs well comparing with other schemes. Both the throughput and received rate of our scheme change smoothly and stand at a relatively high level when BER and fading probability varies.

8. Reference

[1] A.Bakre and B.R.Badrinath, "I-TCP: Indirect TCP for Mobile Hosts", In *Proc. 15th International Conf. on Distributed Computing Systems (ICDCS)*, May 1995.

[2] Balakrishnan, H., Seshan, S., Amir, E., and Katz, R., "Improving TCP/IP Performance over Wireless Networks, "Proc. *1st ACM Conf. on Mobile Computing and Networking (Mobicom)*, Berkeley, CA, November 1995.

[3] K. Ratnam and I. Matta, WTCP: an Efficient Transmission Control Protocol fro Networks with Wireless Links, *Technical Report NU-CCS-97-11*, Northeastern University, July 1997.

[4] S.Biaz and N.Vaidya, "Discriminating Congestion Losses from Wireless Losses using Inter-Arrival Times at the Receiver", *Technical Report 98-014*, CS Dept., Texas A&M University, June 1998.

[5] N.K.G. Samaraweera,"Non-congestion packet loss detection for TCP error recovery using wireless links", *IEE proc.-commun.*, Vol.146, No. 4, August 1999.

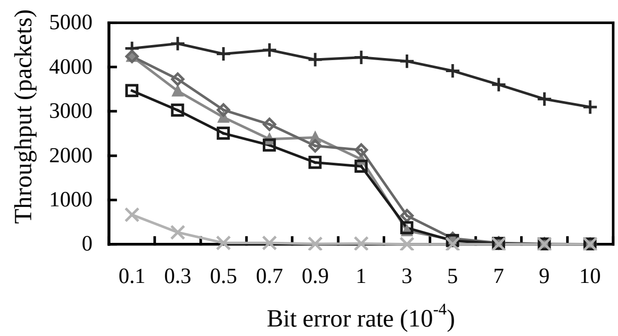
[6] L.Larzon, M.Degermark, and S.Pink, "UDP Lite for Real Time Multimedia Applications", In *Proceedings of the QoS mini-conference of IEEE International Conference of Communications (ICC'99)*, Vancouver, Canada, June 1999.

[7] K.Brown and S.Singh, "M-UDP: UDP for Mobile Cellular Networks", *ACM SIGCOMM Computer Communication Review*, 26(5), Oct. 1996.

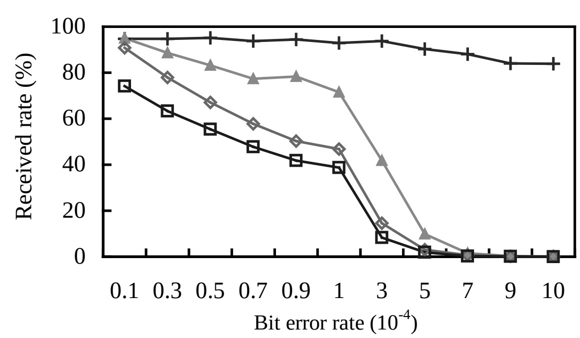
[8] Q. Zhang, W. Zhu, and Y.-Q. Zhang. "Resource allocation for audio and video streaming over Internet", *IEEE ISCAS'2000*, May 2000.

[9] G. J. Wang, Q. Zhang, W. Zhu, and Y.-Q. Zhang. "Channel-Adaptive Error Control for Scalable Video over Wireless Channel", *IEEE MoMuc 2000*, Oct, 2000.

[10] K. Ramakrishnan and S. Floyd, "A Proposal to add Explicit Congestion Notification (ECN) to IP", *IETF RFC 2481*, January 1999.



(a)



(b)

Figure 8. Comparison results of five schemes with the fading probability at 0.05.
(a) Throughput comparison. (b) Received rate comparison.