

TCP-Friendly Wireless Multimedia Flow Control Using ECN Marking

Seong-jun Bae
Dept. of EECS
KAIST
Taejon 305-701, Korea
mythox@netsys.kaist.ac.kr

Song Chong
Dept. of EECS
KAIST
Taejon 305-701, Korea
song@ee.kaist.ac.kr

Abstract— In a wireless network packet losses can be caused by not only network congestion but also unreliable error-prone wireless links. Therefore, flow control schemes which use packet loss as a congestion measure cannot be directly applicable to a wireless network because there is no way to distinguish congestion losses from wireless losses. In this paper, we extend so-called TCP-friendly flow control scheme, which was developed for the control of multimedia flows carried over UDP in the presence of TCP flows, to a wireless environment by using ECN (Explicit Congestion Notification) marking capability with RED (Random Early Detection) routers. By early detecting congestion, explicitly informing multimedia sources of congestion using ECN marking and calculating TCP-friendly rate based on ECN-marked packet probability instead of packet loss probability, we were able to effectively separate the effect of wireless losses from flow control and thus prevent throughput degradation of multimedia flows traveling through wireless links. In addition, we refine well-known TCP throughput model developed for TCP-friendliness of multimedia flows such that the refined model provides more accurate estimate of TCP flows' throughput when both ECN marking and RED queue management are used. Through simulations we show that the proposed scheme indeed improves the quality of delivered video significantly while maintaining TCP-friendliness for the case of wireless MPEG-4 video.

I. INTRODUCTION

With rapid growth of emerging demand and deployment of wireless infrastructures like IMT-2000 networks and wireless LANs, much of IP traffic including multimedia traffic is forced to undergo wireless links. Such a change in networking environments brings us a necessity to refine conventional flow control schemes for not only non-real-time elastic traffic but also real-time multimedia traffic such that the schemes can cope with unreliable error-prone wireless links. In this paper we focus on issues in multimedia flow control with a particular emphasis on video transmission over wireless IP networks.

The quality of networked multimedia applications is more sensitive to delay and delay jitter than loss which is experienced by packets because interruption in playback annoys users more seriously than degradation of picture quality. Therefore, most of multimedia applications use UDP as its transport layer protocol because UDP incurs no retransmission delay and jitter. However, UDP itself provides no flow control scheme so that sources cannot adapt its transmission rate to time-varying available bandwidth which depends on network loading. Therefore, it is necessary to have an application-layer flow control scheme for the transmission of multimedia over UDP. In order to facilitate this necessity, the RTP/RTCP protocol [9], which enables measurement and

This work was supported by LG Electronics Ltd., Korea.

calculation of available network bandwidth, has been introduced.

What is the available bandwidth for a multimedia flow? The answer to this question seems not clear at first glance because underlying TCP flows are flow-controlled so that they inherently try to consume most out of available network bandwidth. However, from an inter-flow fairness standpoint, one can make a reasonable assumption to this problem that multimedia flows must consume the same bandwidth as TCP flows at least in a long-term average sense. This is the key idea behind the so-called TCP-friendly approach to multimedia flow control [1]. In the TCP-friendly flow control, multimedia flows send data at the long-term average rate of underlying TCP flows, which is estimated based on a TCP throughput model. Depending on assumptions, the throughput of a TCP flow can be modeled in different forms [3]–[5] but it is basically a function of packet loss probability P_L and round trip time R of the flow, that is,

$$Y = f(P_L, R) \quad (1)$$

where Y is the long-term average throughput of a TCP flow and is decreasing with respect to both P_L and R . This equation implies that each multimedia source must measure P_L and R along its path in order to send data at the same rate Y as TCP flows provided that the throughput function $f(P_L, R)$ is known.

In a wireless network packet losses can be caused by not only network congestion but also unreliable error-prone wireless links. Therefore, flow control schemes which use packet loss probability P_L as a congestion measure cannot be directly applicable to a wireless network because there is no way to distinguish congestion losses from wireless losses. The same is true for TCP-friendly flow control scheme based on the equation (1) since it uses P_L as the congestion measure. In order to get around this problem, we modify TCP-friendly flow control scheme with the help of ECN (Explicit Congestion Notification) marking capability [6] and RED (Random Early Detection) queue management scheme [7] such that the new scheme uses ECN-marked packet probability P_M as a congestion measure instead of packet loss probability P_L .

Suppose that TCP flows are ECN-capable (we call them ECN-TCP flows) and react to ECN-echoed ACKs in the same way as they react to dropped packets, halving congestion window [6]. Then, the long-term average throughput of an ECN-TCP can be expressed by (1) by substituting P_{LM} for P_L where P_{LM} denotes the probability that a packet is either dropped or ECN-marked. If we further assume that RED routers in the network are tuned appropriately such that no packet is lost at the routers and ECN-TCP flows do not undergo wireless links, then the long-term average of an ECN-TCP can be expressed by (1) by substituting P_M for P_L because no packet is lost and thus the ECN-TCP source will see only ECN-echoed ACKs. Therefore, the throughput of

an ECN-TCP would be given by $f(P_M, R)$ if it experiences neither wireless loss nor congestion loss. In this end, if a multimedia flow can measure P_M as experienced by concurrent ECN-TCP flows and its own R and send data at the rate $f(P_M, R)$ no matter it experiences wireless loss or not, its throughput would be as much as that of an ECN-TCP flow experiencing no loss and be greater than that of an ECN-TCP flow experiencing loss. In order to enable a multimedia source to measure P_M and R , we adopt the RTP/RTCP protocol and propose both sender and receiver algorithms to measure them.

We also refine well-known Floyd's TCP throughput model [4] to take into account the dormant period followed by each congestion window reduction in the congestion avoidance phase. We show that this dormant period plays a significant role as the number of TCP flows multiplexed in a link increases and the refined model predicts TCP throughput more precisely than Floyd's model particularly in such a situation.

Multimedia packets can be lost in wireless hops. We assume that there exists a separate link-layer error control in the wireless link to recover this loss, which is beyond the scope of this paper.

II. TCP-FRIENDLY WIRELESS MULTIMEDIA FLOW CONTROL : TF-WMFC

A. ECN-TCP Throughput Model

Two widely-used TCP-Reno throughput model are Padhye's model [5] and Floyd's model [4]. In fact Padhye's model is a generalization of Floyd's model, taking into account the timeouts incurred by consecutive packet losses, and provides a better throughput estimate particularly when the TCP flows travel through drop-tail routers. However, if the routers are equipped with RED queue management scheme, the probability of timeout occurrence due to consecutive packet losses falls dramatically off. For this reason, we use Floyd's model given in the below as the starting point of our study, assuming that routers in the network are equipped with RED queue management scheme.

$$Y = f(P_L, R) = \frac{K}{\sqrt{P_L}} \cdot \frac{s}{R} \quad (2)$$

where Y is the long-term average throughput of a TCP-Reno flow, P_L and R denote packet loss probability and round-trip time of the flow respectively, s is maximum segment size of TCP and K is a constant resulting from the modeling. In our study we let $K = 1.22$ assuming periodic loss [4].

An ECN-capable RED router marks ECN-bit in incoming packets' IP header in a probabilistic manner when it detects congestion. At the same time an ECN-TCP reacts to these ECN-marked packets in the same way it does to dropped packets, halving its congestion window. This implies that the long-term average throughput of an ECN-TCP would be estimated by (2) by substituting P_{LM} for P_L where P_{LM} denotes the probability that a packet is either dropped or ECN-marked. If we suppose that RED routers in the network are tuned appropriately such that no packet is lost at the routers and ECN-TCP flows do not travel through wireless links, then the long-term average of an ECN-TCP would be estimated by $f(P_M, R)$ using ECN-marked packet probability P_M instead of P_L because no packet is lost and thus the ECN-TCP source will see only ECN-echoed ACKs. Therefore, we conclude that the throughput of an ECN-TCP would be given by $f(P_M, R)$ if it experiences neither wireless loss

nor congestion loss. In this end, we conjecture that if a multimedia flow can measure P_M as experienced by concurrent ECN-TCP flows and its own R and send data at the rate $f(P_M, R)$ no matter it experiences wireless loss or not, its throughput would be as much as that of an ECN-TCP flow experiencing no loss and be greater than that of an ECN-TCP flow experiencing loss. Through simulations in Section III, we show that this conjecture is true.

In the congestion avoidance phase of an ECN-TCP there is a dormant period (where congestion window is kept constant) followed by each congestion window reduction as shown in Fig. 1. Through simulations we found that this dor-

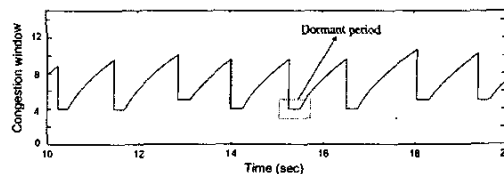


Fig. 1. Typical trace of congestion window in an ECN-TCP

mant period plays a significant role as the number of ECN-TCP flows multiplexed in a link increases and, moreover, the Floyd's throughput model (2) which does not take this period into account exhibits poor performance in such a situation. We consider this dormant period in modeling and obtain the following refined model.

$$Y = f(P_M, R) = \frac{s}{P_M \left(\sqrt{\frac{2}{3P_M}} + \frac{25}{36} + \frac{7}{6} \right) \cdot R} \quad (3)$$

This refined model predicts ECN-TCP throughput more precisely than the Floyd's model particularly as the number of ECN-TCP flows multiplexed in a link increases, as will be shown in Section III Here we omit the derivation of the refined model due to the limited space.

B. Overview of ECN-Based TF-WMFC

Fig. 2 shows the overall framework of proposed ECN-based TF-WMFC scheme. For the implementation of the proposed scheme, we adopt the RTP/RTCP protocol. The sender estimates P_M and R being experienced by its flow based on feedback information sent by the receiver and then computes the TCP-friendly rate Y using the ECN-TCP throughput model (3). On the other hand, the receiver keeps monitoring arriving packets, collects information including number of packets received and number of ECN-marked packets received in a certain time interval and feeds this information back to the sender by sending a backward control packet called RTCP receiver report. For the estimation of R , the sender sends the receiver a forward control packet called RTCP sender report with stamping its current time on it and the receiver returns this timestamp to the sender via a RTCP receiver report. One of the key differences between ECN-TCP operation and ECN-based TF-WMFC operation is that in the former ECN-marked packet is immediately echoed back to the sender via the corresponding ACK whereas in the latter the receiver collects the information on total number of packets received and total number of ECN-marked packets received in a certain interval and feeds this statistics back to the sender via a RTCP receiver report.

Routers in the network are ECN-capable and equipped with RED queue management scheme, i.e., the routers mark

the ECN bit on incoming packets in the forward direction with a certain probability depending on the level of congestion.

Packets can be corrupted in the wireless hop due to high bit errors as the channel status degrades. For the sake of simplicity we assume in this paper that both ECN-marked packets and unmarked packets suffer wireless loss with the same probability so that the ECN-marked packet probability P_M remains unaffected as packets travel through the wireless hop.

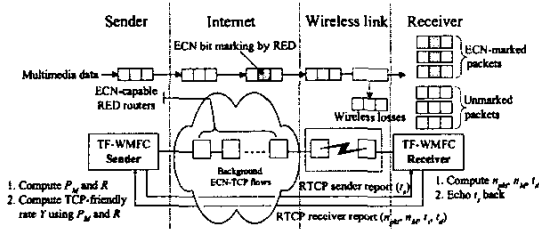


Fig. 2. Overview of ECN-based TF-WMFC

C. Detailed Behavior of Sender and Receiver

1). *Sender Behavior:* Upon every epoch of Δ , the sender updates its TCP-friendly rate $Y(k)$ using the following equations.

$$Y(k) = \frac{s}{P_M(k) \left(\sqrt{\frac{2}{3P_M(k)} + \frac{25}{36}} + \frac{7}{8} \right) \cdot R(k)}, \quad (4)$$

$$P_M(k) = (1 - \alpha)P_M(k-1) + \alpha P_{M_s}, \quad (5)$$

$$R(k) = (1 - \beta)R(k-1) + \beta R_s, \quad (6)$$

where P_{M_s} and R_s are the latest sample of ECN-marked packet probability and round-trip time which were obtained by the feedback information contained in the latest RTCP receiver report whereas $P_M(k)$ and $R(k)$ are the EWMA (Exponentially Weighted Moving Average) of P_{M_s} and R_s with $0 < \alpha < 1$ and $0 < \beta < 1$. By selecting appropriate values for α and β , one can sufficiently smooth out the instantaneous fluctuation of P_{M_s} and R_s , thereby achieving slowly-varying $Y(k)$ which may significantly improve the receiving multimedia quality. In our simulation study in Section III, we found that a good choice for these parameters is $\alpha = 0.01$, $\beta = 0.05$ and $\Delta = 0.1$ sec.

On the other hand, upon every arrival of RTCP receiver report, the sender updates P_{M_s} and R_s as follows. Suppose that the sender receives i -th receiver report at time t_i and $R(k)$ is the latest value of the EWMA of round-trip time at the give time t_i .

$$P_{M_s} = \frac{\min\{n_M, \frac{t_i - t_{i-1}}{R(k)}\}}{n_{pkt}}, \quad (7)$$

$$R_s = (t_i - t_s) - t_d \quad (8)$$

where n_{pkt} and n_M are the number of packets and the number of ECN-marked packets, which arrived at the receiver between the generation of $(i-1)$ -th and i -th receiver reports.

Recall that these values are fed back to the sender via the i -th receiver report. The i -th receiver report also contains t_s and t_d . t_s is the latest timestamp which was stamped by the sender on the latest RTCP sender report and echoed back to the sender by the receiver via the i -th receiver report, and t_d is the time spent by the receiver between the receipt of the sender report and the generation of the i -th receiver report. Thus, the round-trip time sample R_s can be calculated by subtracting $t_i - t_s$ by t_d as in (8).

If we follow the definition of P_M , the sample P_{M_s} must be calculated by $P_{M_s} = \frac{n_M}{n_{pkt}}$. However, in our calculation

of P_{M_s} in (7), we upper-bound n_M by the quantity $\frac{t_i - t_{i-1}}{R(k)}$ which is interpreted as the number of round-trip times that can be contained in the interval between the arrival of $(i-1)$ -th receiver report and the arrival of i -th receiver report. This is because in accordance with the ECN-TCP protocol an ECN-TCP sender reacts to at most one ECN-marked packet in each round-trip time, halving its congestion window, although the sender may receive multiple ECN-echoed ACKs in a single round-trip time. Therefore, if the sender were the ECN-TCP sender, the number of congestion window reduction in an interval $t_i - t_{i-1}$ would be less than or equal to $\frac{t_i - t_{i-1}}{R(k)}$. In order to see the same statistics as ECN-TCP senders, ECN-based TF-WMFC senders should take into account this behavior of ECN-TCP senders. For this reason, we upper-bound n_M by $\frac{t_i - t_{i-1}}{R(k)}$ in calculating P_{M_s} .

The sender sends the receiver a RTCP sender report periodically with an interval T . As discussed, the sender report contains the timestamp indicating its generation time. In our simulation study in Section III we let $T = 1.0$ sec to minimize the bandwidth consumed by the sender reports.

2). *Receiver Behavior:* The receiver generates a RTCP receiver report periodically with an interval T and sends it to the sender. The receiver report generation interval can be different from the sender report generation interval but in our study we make them equal.

Upon receipt of a data packet, the receiver increases n_{pkt} by 1 and if the packet is ECN-marked, increases n_M by 1 as well.

Upon receipt of a sender report, the receiver reads the timestamp t_s contained in the sender report and stores it until a new RTCP receiver report is generated to carry this timestamp to the sender.

Upon every epoch of T , the receiver sends a receiver report to the sender. Each receiver report contains information including the current n_{pkt} , n_M , t_s and the time during which the receiver stores the timestamp t_s , that is, t_d .

3). *Initial Sender Behavior:* Initially, the sender is not aware of ECN-marked packet probability $P_M(k)$ and round-trip time $R(k)$. These quantities can be known only after transmitting some portion of data to collect statistics. In our scheme we introduce a TCP-like rate adjustment scheme for the initial ramp-up of transmission rate.

Suppose that the sender updates $R(k)$ based on (6) and (8) upon every epoch of Δ . Until a RTCP receiver report containing nonzero n_M arrives, the sender keep updating its transmission rate upon every epoch of round-trip time $R(k)$ as follows.

$$Y = \begin{cases} 2Y & \text{if } Y < \frac{W_{th}}{R(k)} \\ Y + \frac{s}{R(k)} & \text{if } Y \geq \frac{W_{th}}{R(k)} \end{cases} \quad (9)$$

where $R(k)$ is the latest round-trip time and $\frac{W_{th}}{R(k)}$ is the threshold between the multiplicative increase and the additive increase in adjusting the transmission rate Y .

Upon arrival of a receiver report containing nonzero n_M , the initial sender behavior terminates.

III. SIMULATION RESULTS

In order to evaluate the performance of the proposed TF-WMFC scheme, we implement the scheme onto the *ns* simulator [8] and adopt both real-time MPEG-4 codec and MPEG-4 FGS (Fine-Grained Scalable) codec [10] as adaptive video applications. The real-time MPEG-4 codec is for live video applications where adaptive encoding is carried out in real time so as to match the output rate of the encoder to the TCP-friendly rate $Y(k)$. In contrast, the MPEG-4 FGS codec is for stored video applications where a video is pre-encoded into two layers (base layer and enhancement layer) such that the output rate can be controlled by selectively sending enhancement-layer data.

A. Accuracy of ECN-TCP Throughput Model

First, we examine the accuracy of refined ECN-TCP throughput model (3) as compared to that of Floyd's model (2) when ECN-marked packet probability P_M is used as the congestion measure. Consider a wired network scenario with a single bottleneck link in Fig. 3 where all ECN-TCP flows share the 20ms-long link and routers are equipped with ECN-capable RED queue management scheme. The number of ECN-TCP flows and the bottleneck bandwidth are varied respectively from 8 to 128 and from 32 Mbps to 128 Mbps. We set packet size to be 1,000 bytes. We change the RED param-

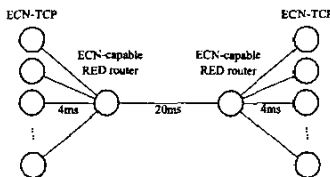


Fig. 3. A wired network scenario for the study of refined ECN-TCP model

eters according to the bottleneck bandwidth, say C [Mbps], as similarly in [7] (see Table I).

Parameter	Value [packets]	Parameter	Value
min_{th}	$5C/16$	max_p	1.0
max_{th}	$50C/16$	w_q	0.002
$queue_{size}$	$400C/16$		

TABLE I
RED parameters used in simulations

Fig. 4 shows the ECN-TCP throughput predicted by both refined model and Floyd's model as normalized by the average of ECN-TCP flows' actual throughput. For the throughput prediction, we measure P_M as seen by the ECN-TCP source and apply this probability to the equations (3) and (2). The refined model significantly outperforms Floyd's model in predicting actual throughput of ECN-TCP flows particularly as the number of ECN-TCP flows multiplexed increases

and/or the bottleneck bandwidth decreases. This is because the dormant period followed by each congestion window reduction plays a significant role in determining throughput of ECN-TCP and our model takes this effect into account.

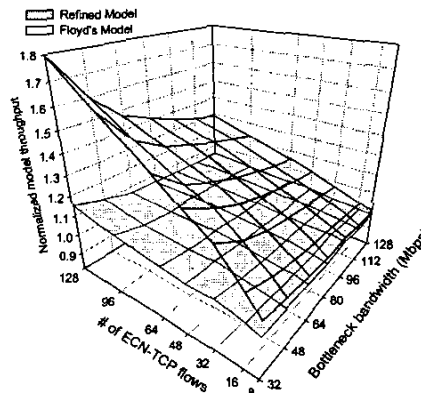


Fig. 4. Accuracy of ECN-TCP throughput models

Next we study the same wired network scenario as in Fig. 3 but we add ECN-based TF-WMFC flows to ECN-TCP flows with the ratio of the number of ECN-based TF-WMFC flows to that of ECN-TCP flows being 1. Then, we control the TF-WMFC flows as proposed in previous section and see how they share the bottleneck bandwidth with concurrent ECN-TCP flows. For the simulation of TF-WMFC flows, we set the parameters as in Table II.

Parameter	Value	Parameter	Value
α	0.01	T	1.0 sec
β	0.05	s	1000 bytes
Δ	0.1 sec	W_{th}	64k bytes

TABLE II
TF-WMFC parameters used in simulations

Fig. 5 shows the average throughput of ECN-based TF-WMFC flows as normalized by the average throughput of ECN-TCP flows. When the proposed TF-WMFC scheme uses the refined throughput model, the normalized average throughput of TF-WMFC flows remains between 0.9 to 1.2, which implies that TF-WMFC flows equally share the bottleneck bandwidth with ECN-TCP flows, i.e., TF-WMFC flows are ECN-TCP friendly. In contrast, if the TF-WMFC scheme uses the Floyd's model, the performance degrades particularly as the number of ECN-TCP flows multiplexed increases and/or the bottleneck bandwidth decreases.

B. Performance of TF-WMFC over Wireless Hop

In the previous subsection we showed that the throughput of wired TF-WMFC flows is almost equal to that of wired ECN-TCP flows in a long-term average sense. In this subsection we show that the throughput of TF-WMFC flows does not depend on whether or not they travel through a wireless hop suffering wireless loss. By combining these two results, we will then conclude that the throughput of a TF-WMFC flow would be as much as that of a wired ECN-TCP flow, no matter it experiences wireless loss or not.

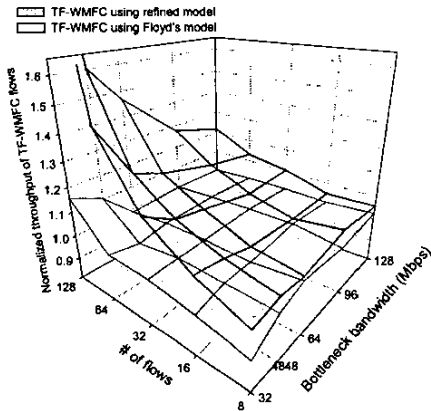


Fig. 5. Average throughput of TF-WMFC flows (normalized by the average throughput of ECN-TCP flows)

In contrast, if a multimedia flow is controlled by the so-called *loss-based* TFRC (TCP Friendly Rate Control) scheme in [1] or its variants, it is quite obvious that its throughput would significantly degrade when it travels through a wireless hop because the TFRC scheme reacts to wireless loss, reducing its sending rate.

In order to quantify this throughput difference between ECN-based TF-WMFC flows and loss-based TFRC flows, we also implement a variant of TFRC scheme where $P_L(k)$ and $R(k)$ are known to the sender using the RTP/RTCP protocol and the sender calculates the TCP-friendly rate by substituting $P_L(k)$ for $P_M(k)$ in the equation (4). Throughout the paper we call this scheme as the loss-based TFRC scheme for the sake of convenience, although several variants can exist depending on detailed implementation.

Consider the simulation scenario in Fig. 6 where 4 wired ECN-based TF-WMFC flows, 4 wireless ECN-based TF-WMFC flows and 8 ECN-TCP flows share the 20ms-long, 16 Mbps bottleneck link. We assume that packets are lost in the wireless hop according to the Bernoulli trials with a certain loss probability.

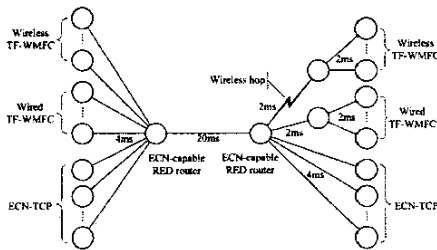


Fig. 6. A network scenario for the study of TF-WMFC over wireless hop

In Fig. 7 we plot the individual and the average throughput of wireless TF-WMFC flows with respect to increasing wireless loss probability, as normalized by the average throughput of wired TF-WMFC flows. The normalized throughput of wireless TF-WMFC flows stays in the vicinity of 1 irrespective of wireless loss probability, which implies that the throughput of wireless TF-WMFC flows is not only independent of wireless loss probability but also almost equal to that of wired TF-WMFC flows and consequently that of wired

ECN-TCP flows.

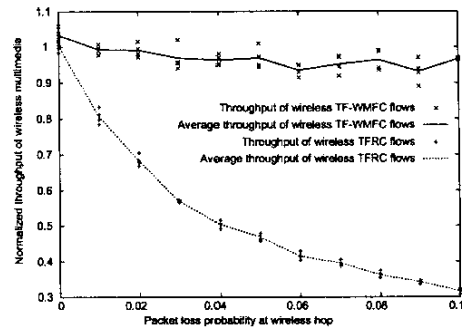
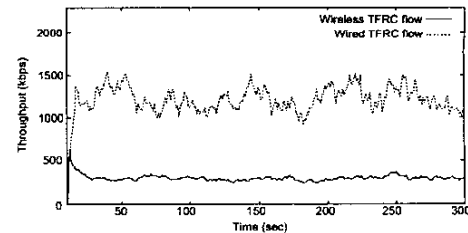


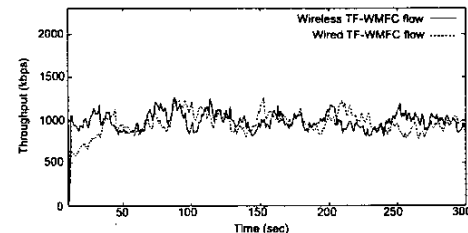
Fig. 7. Throughput of wireless multimedia flows under ECN-based TF-WMFC scheme and loss-based TFRC scheme

For comparison, we replace 8 ECN-based TF-WMFC flows by 8 loss-based TFRC flows correspondingly and simulate the same network scenario. In Fig. 7 we also plot the individual and the average throughput of wireless TFRC flows with respect to increasing wireless loss probability, as normalized by the average throughput of wired TFRC flows. In contrast to the TF-WMFC case, the normalized throughput of wireless TFRC flows drastically falls down as the wireless loss probability increases, which clearly shows that the ECN-based TF-WMFC scheme is superior to the loss-based TFRC scheme in the wireless environment.

In Fig. 8 we also compare the throughput traces of wireless and wired multimedia flows. Under the ECN-based TF-WMFC scheme the throughput of a wireless flow agrees with that of a wired flow (see Fig. 8 (b)) whereas under the loss-based TFRC scheme the throughput of a wireless flow is much lower than that of a wired flow (see Fig. 8 (a)).



(a) Loss-based TFRC scheme



(b) ECN-based TF-WMFC scheme

Fig. 8. Throughput traces of wireless and wired multimedia flows

C. Wireless MPEG-4 Video over TF-WMFC

In this subsection we quantify the improvement on video quality that can be achieved by the proposed ECN-based TF-WMFC scheme when the video is delivered over a wireless link. The comparison is made with the loss-based TFRC scheme. We consider both real-time MPEG-4 codec and MPEG-4 FGS codec which are the most popular video compression techniques today particularly for wireless adaptive video applications. The real-time MPEG-4 codec is for live video applications whereas the MPEG-4 FGS codec is for stored video applications.

Consider the same simulation scenario shown in Fig. 6 with the packet loss probability at the wireless hop being 10%. First, we experiment with live video transmission under both ECN-based TF-WMFC and loss-based TFRC schemes respectively. The *Terminator-2* QCIF video sequence is used as the test sequence and encoded on-line using the real-time MPEG-4 encoder. Fig. 9 shows the PSNR trace and a sample frame of the test video decoded at the receiver. We found that the ECN-based TF-WMFC scheme significantly outperforms the loss-based TFRC scheme, improving the PSNR by 5.0dB on average. Next, we experiment with

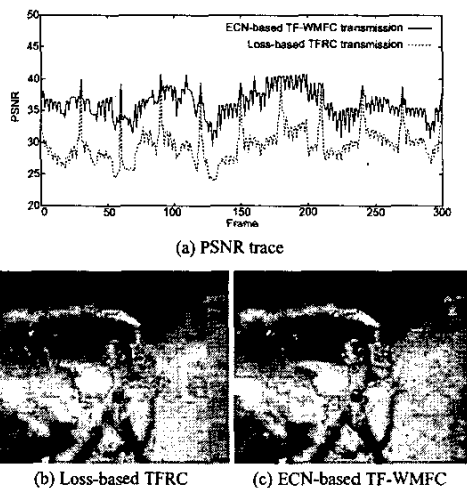


Fig. 9. Quality of real-time MPEG-4 video over wireless

stored video transmission under both schemes respectively. The *Stefan* QCIF video sequence is pre-encoded off-line by the MPEG-4 FGS encoder, stored at the stream DB and retrieved on-line by the FGS server at the given TCP-friendly rate for transmission. Fig. 10 shows the PSNR trace and a sample frame of the test video decoded at the receiver. Again, the ECN-based TF-WMFC scheme significantly outperforms the loss-based TFRC scheme, improving the PSNR by 8.7dB on average.

IV. CONCLUSION

The contribution of this paper is two folds. First, we refine the well-known Floyd's TCP throughput model by taking into account the dormant period followed by each congestion window reduction in the congestion avoidance

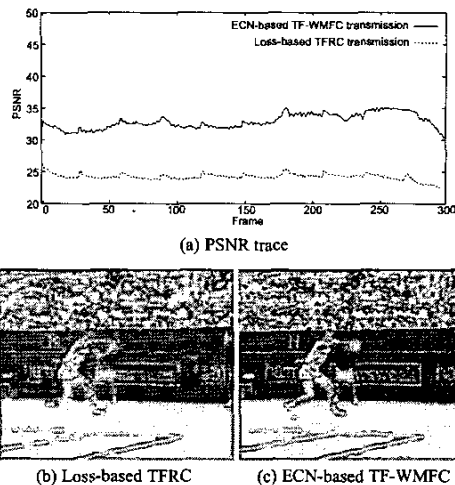


Fig. 10. Quality of MPEG-4 FGS video over wireless

phase. We show that this refined model predicts the long-term average throughput of a TCP flow more precisely than Floyd's model, thereby serving as a good basis for TCP-friendly multimedia flow control. Second, in order to avoid the throughput degradation of a multimedia flow due to wireless loss, we propose an ECN-based TCP-friendly flow control scheme (called ECN-based TF-WMFC) where instead of packet loss, ECN-marked packet is used as a congestion indicator and the TCP-friendly rate is computed using the refined TCP throughput model. We show that the long-term average throughput of a TF-WMFC flow is always equal to that of concurrent wired ECN-TCP flows no matter it experiences wireless loss or not. Through simulation experiments with MPEG-4 video, we demonstrate that the ECN-based TF-WMFC scheme significantly improves the quality of delivered video in a wireless environment, compared with the conventional loss-based TCP-friendly flow control scheme.

REFERENCES

- [1] M. Handley, J. Padhye, S. Floyd, J. Widmer, "TCP Friendly Rate Control (TFRC) : Protocol Specification," Internet Draft, May 2001.
- [2] S. McCreary, K. Claffy, "Trends in Wide Area IP Traffic Patterns," <http://www.caida.org/outreach/papers/AIX0005/>
- [3] T. Ott, J. Kemperman, and M. Mathis, "The Stationary Behavior of Ideal TCP Congestion Avoidance Algorithm," <ftp://ftp.bellcore.com/pub/tjo/TCPwindow.ps>
- [4] J. Mahdavi and S. Floyd, "TCP-friendly Unicast Rate-Based Flow Control," Note sent to end2end-interest mailing list, Jan. 1997.
- [5] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Throughput: A Simple Model and Its Empirical Validation," *SIGCOMM 98*, 1998.
- [6] K. K. Ramakrishnan, S. Floyd, D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP," RFC 2481, Jan. 1999.
- [7] S. Floyd, V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking*, Vol. 1, No. 4, Aug. 1997.
- [8] The Network Simulator - ns-2, <http://www.isi.edu/nsnam/ns/>
- [9] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," RFC1889, January, 1996.
- [10] H. M. Radha, M. V.D. Schaar, Y. Chen, "MPEG-4 Fine-Grained Scalable Video Coding Method for Multimedia Streaming Over IP," *IEEE Transactions On Multimedia*, Vol. 3, No. 1, Mar. 2001.