PAPER *Special Section on Internet Technology V*

Effective Video Multicast over Wireless Internet: Rate Allocation and End-System Based Adaptation

Hao YIN[†a)], ***Nonmember***, **Chuang LIN**[†], ***Member***, **Jin-jun ZHUANG**[†], **Bo LI**[††], ***and*** **Qiang NI**[†††], ***Nonmembers***

SUMMARY With the rapid growth of wireless networks and great success of Internet video, wireless video services are expected to be widely deployed in the near future. As different types of wireless networks are converging into all IP networks, i.e., the Internet, it is important to study video delivery over the wireless Internet. This paper proposes a novel end-system based adaptation protocol called Wireless Hybrid Adaptation Layered Multicast (WHALM) protocol for layered video multicast over wireless Internet. In WHALM the sender dynamically collects bandwidth distribution from the receivers and uses an optimal layer rate allocation mechanism to reduce the mismatches between the coarse-grained layer subscription levels and the heterogeneous and dynamic rate requirements from the receivers, thus maximizing the degree of satisfaction of all the receivers in a multicast session. Based on sampling theory and theory of probability, we reduce the required number of bandwidth feedbacks to a reasonable degree and use a scalable feedback mechanism to control the feedback process practically. WHALM is also tuned to perform well in wireless networks by integrating an end-to-end loss differentiation algorithm (LDA) to differentiate error losses from congestion losses at the receiver side. With a series of simulation experiments over NS platform, WHALM has been proved to be able to greatly improve the degree of satisfaction of all the receivers while avoiding congestion collapse on the wireless Internet.
key words: *wireless Internet, multicast, video*

1. Introduction

The rapid development of the mobile wireless cellular networks and the deployment of new services are changing the ways we live. It is expected that the next generation (3G and beyond 3G) wireless cellular network will be fully capable of delivering multimedia content, in which video multicast might well be the next killer application. Video multicast serves as the basis for a large number of highly anticipated applications such as video conferencing, distance learning, online games and entertainment [1], [2]. Unfortunately, most of the research on video multicast is performed on Internet, as different types of wireless networks are converging into all IP networks, i.e., the Internet, it's important to study video multicast over the wireless Internet. Before video multicast can be deployed on wireless Internet, there are many critical issues that need to be examined, i.e. the intrinsic heterogeneity, large scale, error-prone nature, dramatic link status variation, users' mobility and scarce wireless spectrum. In addition, the video streams co-exist and share the resources with the TCP-based data traffic typically in wireless Internet. It is thus important for the video traffic to be *adaptive and friendly* with the TCP traffic [3]. Lack of bandwidth adaptability will lead to congestion collapse when the aggregate bandwidth of video traffic exceeds network capacity, whereas lack of TCP friendliness will result to compete unfairly with other adaptive traffic, such as TCP [4].

This paper proposes an end-system based adaptive protocol to support TCP-friendly video multicast in wireless Internet without the special support from core network equipments. In the traditional unicast environment, the sender collects the receiver's states via a feedback mechanism and adjusts its transmission rate accordingly. Such an approach faces three major problems in the context of wireless multicast: 1) How to avoid feedback implosion as there usually exists a large number of receivers in a multicasting application? 2) The single sending rate can not satisfy the conflicting requirements of a set of heterogeneous receivers, and layered video transmission has been shown as an effective approach to support heterogeneous receivers with varying bandwidth requirements [3]. While how should the scarce wireless bandwidth is allocated to different layers? 3) How to design a proper TCP-friendly congestion control protocol so that it can handle both the congestion loss and error loss in wireless IP network?

In this paper we extend our proposed Hybrid Adaptation Layered Multicast (HALM) protocol in [2], which is an end-system based adaptation protocol for layered video multicast over the Internet. In HALM, we develop a metric called Fairness Index for each receiver, and formulate the joint rate allocation into an optimization problem [2]. Then, we derive an efficient algorithm to solve the problem. Practical issues about scalable feedback and estimation of available network bandwidth for deploying the optimal algorithm in wireless Internet are solved by using our proposed Sender-Adaptive & Receiver-driven Layered Multicast (SARLM) Scheme [5] and TCP throughput formula [6]. One challenge for available bandwidth estimation based on TCP throughput is how to classify the congestion loss and error loss in wireless networks. We consult the end-to-end loss differentiation algorithm (LDA) [7] to solve the problem. By combining the advantages of HALM, SARLM and LDA, we propose a novel end-system based adaptation protocol for layered video multicast over wireless Internet,

Manuscript received June 30, 2004.
Manuscript revised September 19, 2004.
[†]The authors are with Computer Science Department, Tsinghua University, 100084, Beijing, China.
[††]The author is with Hong Kong University of Science and Technology, Hong Kong.
[†††]The author is with Planete Group, INRIA Sophia Antipolis, France.
a) E-mail: h-yin@mail.tsinghua.edu.cn
DOI: 10.1093/ietcom/e88–b.4.1395

which is called Wireless Hybrid Adaptation Layered Multicast (WHALM) protocol.

As usually there are two main wireless network topologies for supporting multimedia applications: Networks with last-hop wireless links and networks with wireless backbones [7]. The wireless last-hop topology corresponds to cellular networks or satellite modems, and the wireless backbone topology corresponds to high-bandwidth backbones or wireless LAN network such as 802.11 [7]. Taking the characteristics of video multicasting applications into account, WHALM assumes a wireless network topology where the wireless link is the last hop.

The rest of the paper is organized as follows. Section 2 discusses the framework of WHALM. Section 3 describes the formulation of the optimal rate allocation problem and the integration of its scalable solution into WHALM. Section 4 discusses the available bandwidth estimation for wireless network at receiver side. Section 5 presents a new scalable feedback mechanism for our protocol. Section 6 evaluates the performance of WHALM through simulation experiments. Finally, Sect. 7 concludes the paper and discusses some future directions.

2. Overview of the WHALM

WHALM works on top of the Real-time Transport Protocol (RTP) [8]. The video stream is delivered by RTP and control messages are exchanged by an application-specific RTP Control Protocol (RTCP) [8].

2.1 Sender Functionality

In WHALM, the sender encodes the raw video into l cumulative layers using a layered coder. Layer 1 is the base layer and layer l is the least important enhancement layer. The layer rates are given by b_i ($i = 1, 2, \ldots, l$). Let c_j denote the cumulative layer rate up to layer j, that is, $c_j = \sum_{i=1}^{j} b_i$, $j = 1, 2, ..., l$, and ρ_l denote the rate vector, $\rho_l = (c_1, c_2, \ldots, c_l)$. This discrete set offers all the possible video rates that a receiver could receive. In particular, the maximum rate that a receiver with an expected bandwidth r can receive is given by

$$\Gamma(r, \rho_l) = \max\{c : c \leq r, c \in \rho_l\} \tag{1}$$

Note that there is a gap between this receiving rate and the expected bandwidth of a receiver. To minimize this gap, the sender also collects the reports of the expected bandwidths from the receivers. Assume the session size is N and the receivers' expected bandwidths are $\{r_1, r_2, \ldots, r_N\}$. The sender will adjust the layer rates based on the distribution of the expected bandwidths with a control period of T_{ctrl} seconds.

The sender multicasts two kinds of report packets to all receivers. One report packet is Sender Control Report (SCR), which is generated and multicast at the beginning of every control period to distribute the control parameters. SCR has the format of (ID, TS, ρ_l, I, λ, α, T_{ctrl}), where ID is RTP synchronization source identifier (SSRC) [8], TS is a timestamp of the sender's local time, ρ_l is the current rate vector, I is the identification number of the control period, λ and α are parameters of the gamma-distributed timer used by the receivers to avoid feedback implosion, T_{ctrl} is the interval size of the current control period. Another report packet is Sender Report (SR). SR is multicast every T_{SR} second, where $T_{SR} = T_{ctrl}/k$ for some integer $k > 1$ and is used by the receivers to estimate Round Trip Time (RTT). SR includes SSRC, a timestamp of the sender's local time and the response to receivers' requests. To reduce the control overhead, the sender does not give a response to each request but uses a batch process. A SR multicast at time $t + T_{SR}$ contains the SSRCs of the receivers whose RR packets (described in 2.2) arrive in the time slot $[t, t + T_{SR}]$ and their delays t_i^{delay}(the interval between their arrival time and $t + T_{SR}$). Upon receiving the SR, these receivers can use the information contained in it to calculate their RTTs. Detail algorithm can be found in [2] and we have depicted these cases in Fig. 1.

2.2 Receiver Functionality

A receiver decides whether to join a higher layer, stay at or leave the current layer at the beginning of each control period based on the rate vector in SCR and its expected bandwidth. To be friendly with TCP, a receiver directly uses the following TCP throughput formula [6] to calculate its expected bandwidth and feedbacks it to the sender:

$$B = \frac{s}{RTT\sqrt{\frac{2p}{3}} + RTO\left(3\sqrt{\frac{3p}{8}}\right)p(1 + 32p^2)} \tag{2}$$

This gives the TCP throughput B in bytes/sec, as a function of the packet size s, round-trip time RTT, steady-state loss event rate p, and the TCP retransmit timeout value RTO. The receiver dynamically monitors these parameters to calculate the bandwidth and selectively feedbacks it to the sender. Specifically, it uses a scalable feedback scheme, detailed in Sect. 5, to decide whether and when to generate report packets in a control period. A feedback packet, named as RR, contains the SSRC of the receiver, the expected bandwidth, and the timer setting z_i of the receiver, which is used by the sender to estimate the number of receivers. It also serves as a request for RTT estimation.

2.3 Workflow of the Protocol

The sender adjusts its sending rates once every control period of T_{ctrl} seconds, which varies from 2 s to 15 s according to the number of receivers. The control process works as follow.

At the beginning of a control period, the sender adjusts its sending rates based on the receivers' bandwidth distribution. It also multicasts a SCR to deliver the new rate vector, the duration of the current control period and the parameters of the gamma-distributed timer (λ and α) used by receivers

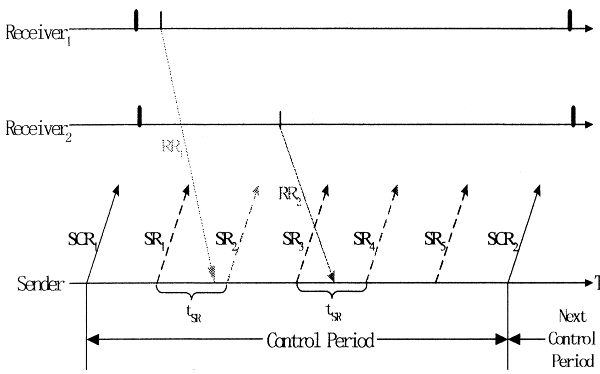


Fig. 1 Timing diagram for the control loop.

to avoid feedback implosion.

Upon receiving SCR, each receiver decides and then performs its join/leave layer actions. It also generates a gamma distributed random timer to decide whether and when to send feedback in the current control period. It also collects network status from the SR packets and data packets sent by the sender to estimate its available bandwidth.

Based on the feedback packets, the sender estimates the number of receivers, N. The size of N determines the number of feedbacks (denoted by n) needed to collect in next control period. Given the limited control bandwidth and n, the sender can derive the duration of next control period. The sender also calculates the new λ and α for the next feedback loop based on n.

Figure 1 shows the control process. Note that receiver$_i$' RR packet not only feedbacks its expected bandwidth, but also serves as a request for closed-loop RTT estimation. Each SR serves as the response for all the RR packets that have arrived at the sender since last SR was multicast.

3. Sender-Based Dynamic Rate Allocation

We define a *Fairness Index* $F(.)$ to help establish the optimization objective for rate allocation. The fairness index $F(.)$ for a receiver with expected bandwidth r is as follows:

$$F(r, \rho_l) = \frac{\Gamma(r, \rho_l)}{r} \tag{3}$$

This definition can be used to assess the satisfaction of a receiver when there is performance loss incurred by a mismatch between the discrete set of possible rates and the expected bandwidth. Since the expected bandwidth is estimated as the throughput of a TCP connection over the same path, this index also reflects the degree of fairness between video traffic and TCP traffic. Specifically, the fairness index of 1 is optimal in that it allows the receiver to fully exploit the expected bandwidth, that is, to fairly share the bandwidth with TCP connections.

For a multicast session, a natural optimization objective is to maximize the expected fairness index, $\overline{F}(r, \rho_l)$, for all the receivers by choosing an optimal rate vector. We state the optimization problem as follows:

$$\text{Maximize } \overline{F}(r, \rho_l) = \frac{1}{N}\sum_{i=1}^{N} F(r_i, \rho_l), \tag{4}$$
$$\text{Subject to } l \le \text{L}, \quad 0 < c_{i-1} < c_i, i = 2, 3, \ldots, l.$$

where L is the maximum number of layers that the sender can manage.

The complexity of this problem can be further reduced by considering some characteristics of a practical layered codec. First, there are only a finite number of possible rates for any given source. These rates, called *operational rates* [9], depend only on the compression algorithm and source features. Secondly, to avoid the undesired situation where a receiver cannot join any layer, the base layer should adapt to the minimum expected bandwidth. However, the dynamic range of a layered coder is limited which usually places a lower bound to the rate of the base layer. Taking these two characteristics into account, we assume there are M operational points. The set of operational rates is given by $\pi = \{R_1, R_2, \ldots, R_M\colon R_i < R_{i+1}\}$, and R_1 is the lower bound for the base layer rate. We can then re-formulate the optimization problem as follows:

$$\text{Maximize } \overline{F}(r, \rho_l) = \frac{1}{N}\sum_{i=1}^{N} F(r_i, \rho_l), \tag{5}$$
$$\text{Subject to } l \le \text{L},$$
$$c_1 = \max_j\{R_j : R_j \le \min_i\{r_i : r_i \ge R_1\}\},$$
$$c_i \in \pi,\ c_{i-1} < c_i,\ i = 2, 3, \ldots, l.$$

A scalable algorithm for this problem with time complexity O(LM^2) and auxiliary storage space O(LM) is derived in [2]. As the complexity does not depend on the number of receivers, the algorithm is highly scalable. Moreover, it relies only on the bandwidth distribution of all the receivers, therefore sampling can be used to reduce collection time for bandwidth reports. From the statistical theory, let n be the number of samples needed to calculate the expected fairness index within confidence interval ε and confidence level $1 - \alpha$. The smallest n that satisfies:

$$P(|\overline{F}_A^n - \overline{F}_A| < \varepsilon) \ge 1 - \alpha, \tag{6}$$

can be calculated by:

$$n_0 = \left(\frac{Z_{\frac{\alpha}{2}} S}{\varepsilon}\right)^2 \quad \text{and} \quad n = \frac{n_0}{1 + \dfrac{n_0}{N}} \tag{7}$$

where $\overline{F}_A$ is the average fairness based on the distribution of all the receivers, $\overline{F}_A^n$ is the one based on n reports, $Z_{\alpha/2}$ is the upper $\alpha/2$ percentage point of the standard normal distribution, and S is the standard deviation of the fairness indices, which can be estimated once every control period. Given a fixed average control bandwidth (e.g., 20 kbps), the interval time needed to collect the feedbacks (i.e., the control period T_{ctrl}) can be determined by n. Note that the more the receivers, the more samples needed to be collected, and the longer control period, and vice-versa. Since a very short or long control period may result in inaccurate bandwidth reports, and a short control period also may cause highly oscillative adaptation behavior, we confine T_{ctrl} in the range of [2 s, 15 s].

4. Available Bandwidth Estimation in Wireless IP Networks

As is depicted in Sect. 2.2, each receiver estimates its available bandwidth using a TCP throughput formula (see equation (2)) which needs the estimation of *RTT*, *RTO* and *p*. But how to estimate *RTT*, *RTO* and *p* in wireless IP networks bring great challenges [10]. First of all, the varying wireless environment results in dramatic fluctuation of the end-to-end RTT over wireless Internet. Thus the bandwidth estimation counted on RTT may be inaccurate and fluctuate greatly. To solve this problem, the receivers use the following RTT estimation to measure the "average" round trip time during a period of time:

$$\overline{RTT_n} = \overline{RTT_{n-1}} * \alpha + RTT_n * (1 - \alpha) \tag{8}$$

where α is a weighting parameter that is set to 0.9 in our protocol, $\overline{RTT_{n-1}}^*$ is the average round trip time at the $(n-1)$-th measurement interval, and RTT_n^* is the estimated round trip time at the n-th measurement interval. As a result, the bandwidth estimation performs more smoothly.

Another parameter RTO can be estimated from RTT. Practically, the simple heuristic of RTO = max {1,4RTT} works reasonably well to provide fairness with TCP [2].

Moreover, in wireless IP networks, the end-to-end packet loss can be caused by either congestion loss in the wired network or the erroneous loss in the wireless part. Traditional TCP and TCP-friendly protocols treat every loss event as a signal of congestion and correspondingly reduce the transmission rate. However, this rate reduction is unnecessary if the loss is due to the error in wireless network. In WHALM we incorporate a LDA (loss differentiation algorithm) called spike scheme [7] at the receiver side to classify the loss type and estimate the packet loss rate using only congestion loss. Since only congestion losses are used as congestion signals, and wireless losses do not restrict the sending rate, WHALM can achieve considerable performance optimization.

Spike scheme uses the Relative One-way Trip Time (ROTT) to identify the state of the current connection. ROTT is a measure of the time a packet takes to travel from the sender to the receiver. Since the sending and receiving times are measured at the sender and receiver separately and there exists skewness between the two clocks, thus the name "relative."

Spike scheme works as follows. On receipt of a packet with sequence number *i*, if the connection is currently not in the spike state, and the ROTT for packet i exceeds the threshold Bspikestart, then the state enters the spike state. Otherwise, if the connection is currently in the spike state and the ROTT for packet i is less than a second threshold Bspikeend, the state changes out of the spike state. When

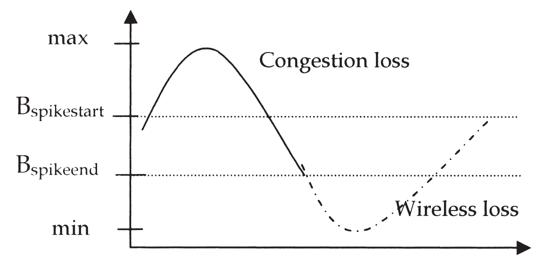


Fig. 2 Spike scheme.

the receiver detects a loss because of a gap in the sequence number of received packets, if the connection is in the Spike state, losses are assumed to be due to congestion. Otherwise, losses are assumed to be wireless, which is illustrated in Fig. 2.

5. Scalable Feedback Mechanism

In WHALM the receivers need to send bandwidth feedbacks and RTT estimation request to the sender, thus a feedback control mechanism is needed to suppress feedbacks. Several solutions exist for implosion avoidance based on hierarchy, parameterized and randomly delayed timers. Nonnenmacher et al. suggested a truncated exponentially distributed timer, but it requires a multicast feedback channel for every receiver, which leads to additional overheads. We only use a unicast feedback channel from receiver back to the sender. We introduce a devised Gamma-distributed timer from 0 to T_{ctrl} for each receiver, which outperforms other distributed timers when the number of users is large. The density of the Gamma distributed timer is:

$$f_{Z_i}(z_i) = \begin{cases} \dfrac{1}{(e^{\lambda} - 1)} \cdot \alpha \dfrac{\lambda}{T_{ctrl}^{\alpha}} z_i^{\alpha-1} e^{\frac{\lambda}{T_{ctrl}^{\alpha}} z_i^{\alpha}}, & 0 \le z_i \le T_{ctrl} \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

Where T_{ctrl} is the control period, λ and α are factors related to the number of receivers.

At the beginning of a control period, the sender multicasts a SCR packet. SCR contains the parameters of the gamma-distributed timer, i.e., T_{ctrl}, λ and α. Upon receiving the SCR, receiver i schedules a gamma distributed random timer $z_i \sim [0, T_{ctrl}]$. Only the receivers that get the timer between (0,0+c) can send feedback, other receivers are suppressed by this way, where c is the receiver-sender delay. Figure 3 shows the Gamma distributed timer setting of z_i. When the timer z_i expires, receiver i sends a RR packet back to the sender.

At the end of a control period, the sender computes N^*, the estimated number of the receivers,

$$N^* = \frac{X(1 - F_Z(m))}{F_Z(m + c) - F_Z(m)} \tag{10}$$

using the knowledge about the timer settings z_i of all the receivers that have returned RR packets during current control period. The sender then computes the new λ and α for the next control period based on N^* and n, the desired number

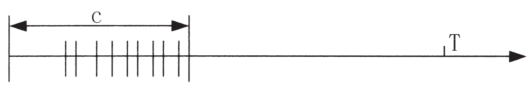


Fig. 3 Gamma distributed timer setting z_i.

of RRs:

$$\lambda = 1.1 \cdot \ln N^* + 0.8 \tag{11}$$

$$\alpha = \frac{\ln\left[\frac{1}{\lambda}\ln\left[\frac{n(e^{\lambda}-1)+N^*}{N^*}\right]\right]}{\ln\left(\frac{c}{T_{ctrl}}\right)} \tag{12}$$

In this way the sender can control the number of feedback packets during each control period and thus avoid feedback explosion.

6. Simulation Results

We simulate WHALM and HALM using network simulator ns-2 [11]. We take the network topology that is used in [2] and modify it to evaluate the effectiveness of our protocol by replacing some wired receivers with wireless ones. Figure 4 illustrates the network topology. There is a WHALM sender and $6m$ receivers belonging to six LANs, each LAN having m receivers. The first three LANs are wireless LANs, where the receivers are connected to the switches through wireless links. The rest LANs are wired LANs whose bottleneck links are (SW0, SWi), $i = 4, 5, 6$, respectively. A TCP connection modeled as a FTP flow shares (SW0, SWi), $i = 1, 2, \ldots, 6$, with the corresponding video streams. The cumulative layer rates of the video source are initialized to $\{128, 384, 896\}$ kbps, and the lower bound of base layer rate is 100 kbps.

It is important to accurately estimate available bandwidths of receivers. We also calculate the optimal allocation based on the exact and instant bandwidth distribution of all the receivers. Assume the expected fairness index under this allocation is F^*, and the one under a practical algorithm (in WHALM and HALM) is F', the accuracy of the practical algorithm is defined as [2]:

$$Accuracy = \frac{F'}{F^*} \tag{13}$$

Figure 5(a) illustrates the accuracy comparison of WHALM and HALM.

It can be seen that WHALM achieves better accuracy than HALM. This improvement results from: (1) HALM collects feedbacks in a fixed control period of 15 s, which is long enough for sampling up to 5000 receivers. Such a long collection time is unnecessary when the number of receivers is smaller. Moreover, it causes serious skewness between a receiver's current expected bandwidth and its recent report in a highly dynamic environment such as wireless network. While in WHALM, the control period varies with the number of receivers to reduce the skewness. (2) WHALM incorporates spike scheme which differentiates between congestion and wireless losses, thus it uses relatively accurate packet loss rates when estimating the available bandwidths.

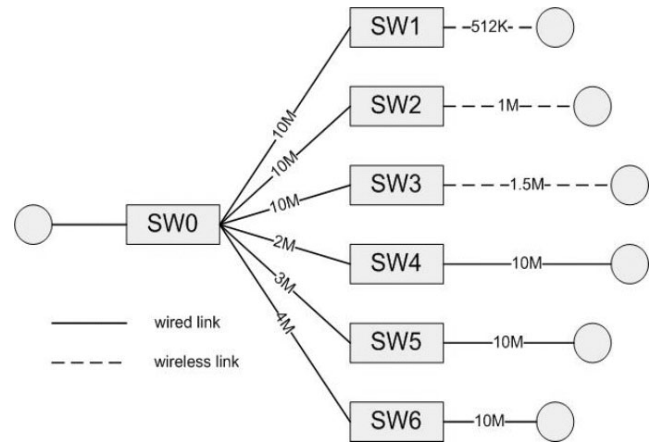
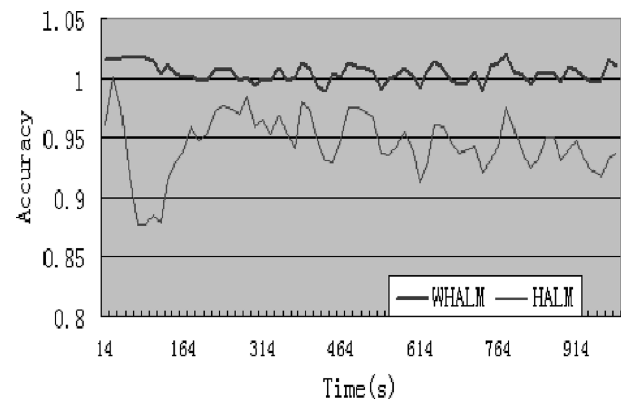
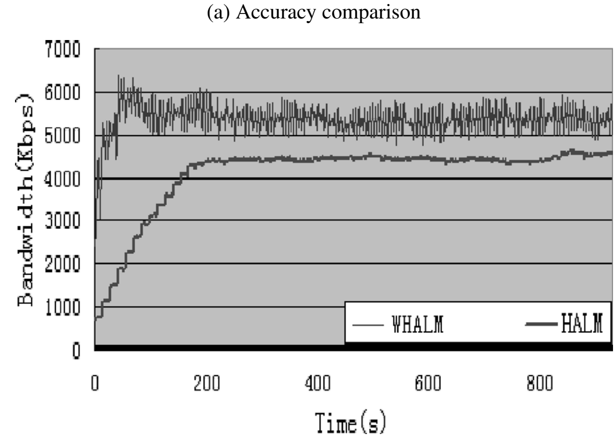


Fig. 4 Topology of simulation experiments.



(a) Accuracy comparison

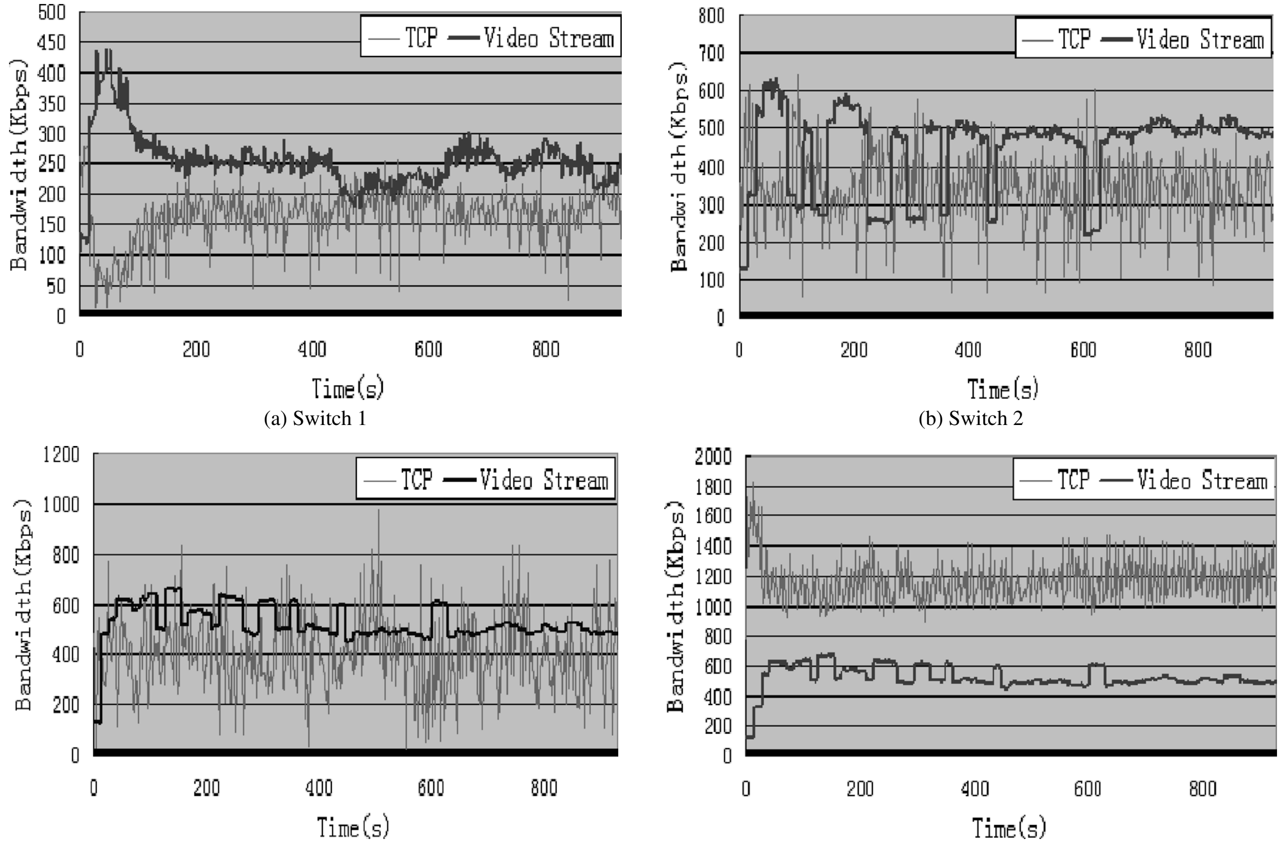


(b) Throughput comparison

Fig. 5 Performance comparison of WHALM and HALM.

Figure 5(b) shows throughput comparison between WHALM and HALM. Here the throughput is the cumulative one of all the receivers. Evidently WHALM achieves higher throughput than HALM. This is because wireless losses do not restrict the sending rate in WHALM.

Figure 6 shows the bandwidth distribution between the competing video streams using WHALM and TCP flows at switches 1–4. Compared to TCP flows, WHALM not only



(a) Switch 1

(b) Switch 2

(c) Switch 3

(d) Switch 4

Fig. 6 Bandwidth distribution between WHALM and TCP at switches 1–4.

exhibits smoother behavior but also has a higher bandwidth usage in wireless environments. At the same time, experimental results also show that the TCP friendliness is preserved.

7. Conclusion

We propose WHALM, an end-system based TCP friendly adaptation protocol for layered video multicast over wireless Internet in this paper. WHALM integrates an optimal rate allocation algorithm and a scalable feedback mechanism to dynamically adapt to the heterogeneous requirementes from the receivers. In addition it efficiently avoids the well-known performance degradation in wireless Internet by incorporating a loss differentiation scheme. After a series of simulation experiments over NS platform, we can conclude the following unique advantages for WHALM: 1) By developing a metric called Fairness Index for each receiver, and formulating the joint rate allocation into an optimization problem, WHALM greatly improves the degree of satisfaction for receivers with heterogeneous bandwidth requirements; 2) WHALM not only maintains high bandwidth utility but also preserves smoothness and TCP friendliness of output rate, which are very important for the QoS improvement of multimedia transmission and stability of the network. 3) WHALM is feasible to be deployed in wireless Internet for adopting our proposed scalable feedback control mechanism and end-to-end loss differentiation algorithm.

Our future work is to conduct more simulations and real experiments with advanced layered coding algorithms. This also enables more extensive and realistic comparisons with other layered multicast protocols over wireless Internet. Other potential work includes how to improve the error resilience of video multicast over wireless channel by using error control algorithms, and how to improve the accuracy of the estimation of available network bandwidth.

Acknowledgment

This work was supported in part by a grant from the Projects of Development Plan of the State Key Fundamental Research (No. 2003CB314804), the National Natural Science Foundation of China (No. 60372019), and China Postdoctoral Science Foundation (No. 2003034152).

References

[1] S. McCanne, V. Jacobson, and M. Vetterli, “Receiver-driven layered multicast,” Proc. ACM SIGCOMM 96, Aug. 1996.

[2] J.-C. Liu, B. Li, and Y.Q. Zhang, “An end-to-end adaptation protocol for layered video multicast using optimal rate allocation,” IEEE Trans. Multimed., vol.6, no.1, pp.87–102, Feb. 2004.

[3] B. Li and J.-C. Liu, “Multi-rate video multicast over the Internet: An overview,” IEEE Networks Special Issue on Multicast, vol.17, no.1,

pp.24–29, Jan./Feb. 2003.
[4] W. Tan and A. Zakhor, “Real-time Internet video using error resilient scalable compression and TCP-friendly transport protocol,” IEEE Trans. Multimed., vol.1, no.2, pp.172–186, June 1999.
[5] Q. Ni, Q. Zhang, and W.W. Zhu, “SARLM: Sender-adaptive and receiver-driven layered multicast for scalable video.” IEEE ICME 2001 (IEEE International Conference on Multimedia and Expo), Japan, Aug. 2001.
[6] M. Handley, S. Floyd, J. Padhye, and J. Widmer, “TCP friendly rate control (TFRC): Protocol specification,” RFC 3448, Jan. 2003.
[7] S. Cen, P.C. Cosman, and G.M. Voelker, “End-to-end differentiation of congestion and wireless losses,” IEEE Trans. Netw., vol.11, no.5, pp.703–717, Oct. 2003.
[8] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A transport protocol for real-time applications,” RFC 1889, Jan. 1996.
[9] G. Schuster and A. Katsaggelos, Rate-Distortion Based Video Compression, Kluwer Academic Publishers, 1997.
[10] F. Yang, Q. Zhang, W. Zhu, and Y.-Q. Zhang, “End-to-end TCP-friendly streaming protocol and bit allocation for scalable video over wireless Internet,” IEEE J. Sel. Areas Commun., vol.22, no.4, pp.777–790, May 2004.
[11] “The Network Simulator (NS),” http://www.isi.edu/nsnam/ns/

Hao Yin received the B.S., M.E., and Ph.D. degree from Huazhong University of Science and Technology, China in 1996, 1999, and 2002, respectively, all in electrical engineering. In the summer of 2001, he was an Intern with Microsoft Research, Asia, where he conducted research on performance measurement and estimation of wireless network. From 2001 to 2002, he was a CTO in Wuhan Teamswan Digital Technology Co., Ltd, where he worked on multimedia distance learning system and implemented two famous distance learning platforms. He is currently an assistant professor in Computer Science and Technology Department, Tsinghua University, Beijing, China. His research interests span broad aspects of performance evaluation for Internet and wireless network, image/video coding, multimedia communication over wireless network, and security. He has published over 30 papers in refereed journal and conference.

Chuang Lin is a professor and the head of the Department of Computer Science and Technology, Tsinghua University, Beijing, China. He received the Ph.D. degree in Computer Science from Tsinghua University in 1994. In 1985–1986, he was a Visiting Scholar with the Department of Computer Sciences, Purdue University. In 1989–1990, he was a Visiting Research Fellow with the Department of Management Sciences and Information Systems, Graduate School of Business, University of Texas at Austin. In 1995–1996, he visited the Department of Computer Science, Hong Kong University of Science and Technology. His current research interests include computer networks, performance evaluation, logic reasoning, and Petri net theory and its applications. He has co-authored more than 150 papers in research journals and IEEE conference proceedings in these areas and has published three books.

Jin-jun Zhuang received her B.S. degree in Computer Science from National University of Defence Technology, China in 1999. She is currently a postgraduate with Computer Science and Technology department in Tsinghua University, Beijing, China.

Bo Li received his B.Eng. (summa cum laude) and M.Eng. degrees in the Computer Science from Tsinghua University, Beijing in 1987 and 1989, respectively, and the Ph.D. degree in the Electrical and Computer Engineering from University of Massachusetts at Amherst in 1993. Between 1993 and 1996, he worked on high performance routers and ATM switches in IBM Networking System Division, Research Triangle Park, North Carolina. Since 1996, he has been with the Department of Computer Science, Hong Kong University of Science and Technology, where he is now an associated professor and Co-Director for the ATM/IP Cooperate Research Centre, a government sponsored research center. Since 1999, he has also held an adjunct researcher position at the Microsoft Research Asia, Beijing, China. His recent research focuses are on adaptive video multicast, routing in optical networks, resource management in wireless cellular systems, and content replication. He has published 60 some journal papers and held several patents in above areas. He has been on editorial board for IEEE Transactions on Wireless Communications, IEEE Transactions on Vehicular Technology, ACM/Kluwer Journal of Wireless Networks (WINET), IEEE Journal of Selected Areas in Communications (JSAC)—Wireless Communications Series, ACM Mobile Computing and Communications Review (MC2R), Elsevier Ad Hoc Networks, SPIE/Kluwer Optical Networking Magazine (ONM), KICS/IEEE Journal of Communications and Networks (JCN). He served as a guest editor for IEEE Communications Magazine Special Issue on Active, Programmable, and Mobile Code Networking (April 2000), ACM Performance Evaluation Review Special Issue on Mobile Computing (December 2000), and SPIE/Kluwer Optical Networks Magazine Special Issue in Wavelength Routed Networks: Architecture, Protocols and Experiments (January/February 2002), IEEE Journal on Selected Areas in Communications Special Issue on Protocols for Next Generation Optical WDM Networks (October 2000), Special Issue on Recent Advances in Service-Overlay Network (January 2004), and on Quality of Service Delivery in Variable Topology Networks (3rd Quarter of 2004), and ACM/Kluwer Mobile Networks and Applications (MONET) Special Issue on Energy Constraints and Lifetime Performance in Wireless Sensor Networks (4th Quarter of 2004). In addition, He has been involved in organizing over 40 conferences, esp. IEEE Infocom since 1996. He was the Co-TPC Chair for IEEE Infocom 2004.

Qiang Ni received the B.S., M.S. and Ph.D. degrees from Hua Zhong University of Science and Technology (HUST), Wuhan City, China in 1993, 1996 and 1999 respectively. From 1999 to 2001, he was a post-doctoral research fellow in the multimedia and wireless communication laboratory, HUST. He did his research internship at wireless and networking group of Microsoft Research Asia during the year of 2000. In 2001, he joined INRIA, France, where he was an expert research staff engineer at Planète group. He is currently a Senior Research Scientist at the Hamilton Institute, National University of Ireland, Maynooth. Since 2002, he has been active as a voting member for the IEEE 802.11 wireless LAN Standard Working Group. He has served as Technical Program Committee member for the IEEE Vehicular Technology Conference, Integrated Heterogeneous Wireless Networks Symposium, 2003, 2005 IEEE Sarnoff Symposium, Princeton, USA, and IEEE WiOPT'05 (Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks), Italy. His current research interests include communication protocol design and analysis for wireless LAN/PAN/MAN and 4G networks, vertical handover and mobility management in mobile networks, and multimedia transmission over hybrid wired/wireless networks.