Multi-path Scheduling Algorithm for Real-Time Video Applications in Next-Generation Wireless Networks

Juan Carlos Fernandez*, Tarik Taleb†, Kazuo Hashimoto, Yoshiaki Nemoto, and Nei Kato
Graduate School of Information Sciences, Tohoku University, Sendai, Japan
*carlos@it.ecei.tohoku.ac.jp, †taleb@aiet.ecei.tohoku.ac.jp

Abstract

In wireless mobile networks, ensuring quality of service (QoS) for real-time video applications is a challenge due to resource limitations. Nowadays, such a problem can be alleviated by exploiting the features of mobile computers equipped with multiple wireless interfaces, which can be used simultaneously for transmission/reception of data belonging to a single application. This increases the throughput but is tied to the packets reordering issue. This paper proposes a scheme for scheduling packets through different paths and minimizing the reordering delay at the receiver side. We consider a QoS negotiation system where active users have a specific amount of bandwidth negotiated with the network by each available interface. To allocate such negotiated bandwidth to users, the network implements a time slot division-based strategy. A new scheduling mechanism is developed by taking into consideration such QoS system. Its performance is evaluated and compared against the most popular scheduling schemes via simulations. The results show that our algorithm outperforms the former scheduling algorithms.

1. Introduction

Real-time video applications are highly sensitive to delay, jitter and bandwidth restrictions. These problems become more significant in wireless mobile networks as to the bottleneck for most wireless communications is the last hop; from the access point to the mobile terminals.

Mobile terminals, equipped with multiple interfaces using different wireless technologies, are able to maintain simultaneous connections through these interfaces when the coverage areas of these technologies partially overlap. Such capacity allows mobile terminals to increase the streaming bandwidth by distributing the load over multiple network paths.

If the bandwidth of a single interface is not enough to meet the real-time video application rate, the user may consider using two or more interfaces to ensure the quality of the video application. However, such transmission of packets (of the same application across multiple interfaces) makes packets of the same application experience different latencies as each wireless link may have different capacity and different propagation delay. This results in out-of-order reception at the final destination. Consequently, some packets of the real-time video application experience delays higher than their timers and ultimately get discarded. In this context, the contribution of this paper consists in an enhanced version of the Earliest Delivery Path First scheme [1] to cope with the packet reordering issue in multi-path video transmission.

The paper is organized as follows. Section 2 presents some research work pertained to Multi-path scheduling algorithms. Section 3 describes the considered QoS architecture. Section 4 describes the proposed scheme and how it efficiently distributes the video packets on the available network paths to minimize the reordering delay. Section 5 shows the performance evaluation of our scheduling mechanism. Finally, Section 6 concludes the paper.

2. Related work

The current Internet infrastructure was not designed with the needs for simultaneous accesses to two or more different technologies in mind. However, with the growing demand for mobility support, coverage areas of different access points are overlapping. Thus, mobile hosts can simultaneously use multiple communication channels to increase their throughputs. To achieve this goal, many research works have been done allowing mobile nodes to obtain multiple Care-of-Addresses (CoAs) [2] [3], and keeping senders always informed of these CoA registrations directly from the mobile nodes [4].

978-1-4244-1841-1/08/$25.00 ©2008 IEEE

In multi-path video streaming, many schemes have been proposed to address the issue of out-of-order delivery to the final destination due to the heterogeneous characteristics of wireless links. For TCP applications, such disorder in packet reception results in the transmission of duplicate acknowledgments (DupAcks), which is assumed as an indication of network congestion. As a remedy, work in [5] presents a buffer management policy solution. For UDP applications, transmission of data via multiple paths results in an additional delay at the receiver side due to the packet reordering issue. Consequently, some of the packets of the real-time video applications experience delays higher than their timers and ultimately get discarded. To cope with this issue, there are several scheduling strategies such as Weighted Round Robin (WRR), Weighted Interleaved Round Robin (WIRR), Surplus Round Robin (SRR) and the most recently developed Earliest Delivery Path First algorithm (EDPF). Other approaches aim at minimizing the distortion perceived by the end user by considering the selection of inter-dependent video packets to be transmitted [6] [7] [8].

To guarantee QoS to mobile users, some negotiation systems have been proposed for wireless networks [9] [10] [11]. These systems easen the scheduling operation as the access network guarantees certain amount of bandwidth to mobile users during their connection. Thus, knowledge on the bandwidth of each path is available: no monitoring of any kind is required. In the following section we describe our envisioned QoS negotiation architecture.

3. System architecture

The components of the architecture are depicted in Fig. 1. The network is divided into a number of domains administrated by different Internet service providers. Each domain consists of a QoS Global Server (QGS), an AAA (Authentication, Authorization, and Accounting) server, a number of Base Stations (BSs), and a population of mobile users, termed henceforth as Mobile Stations (MS).

QGS, introduced also in the Dynamic Service Negotiation Protocol [9], basically functions as a Policy Decision Point defined in the Policy Framework presented in [12]. It performs service level negotiation and is responsible for maintaining global information about the available resources in the whole domain. Based on this information, it admits or rejects a service level request. BSs are responsible for applying different service levels to MSs and for controlling the traffic flow of all MSs in their coverage areas. BSs inform QGS of their local resource availability and receive SLS of mobile users for traffic conditioning. A detailed description of the QoS negotiation procedure can be found in [10] [13][14].

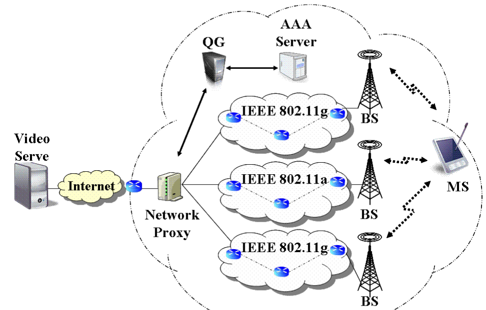

Fig. 1. The envisioned architecture for QoS negotiation

4. Scheduling algorithm

The efficiency of multi-path video streaming is tied to the packed scheduling strategies, which aim at offering high quality of service in real-time multimedia applications. Earliest Delivery Path First (EDPF) [1] is the most notable scheduling algorithm. It bases its scheduling on a prior knowledge of the available bandwidth at each interface. The key idea behind EDPF algorithm lies on the estimation of the delivery time of the next packet through each path. Using this estimation, EDPF transmits the packets via the path with the earliest delivery time.

The considered QoS negotiation system implements a time-slotted approach for bandwidth allocation at the BSs. Each MS is allocated a specific period of time to use the wireless channel. At any given time, only one MS is allowed to transmit/receive data through a particular BS. The length of the time-slot allocated to a given MS through a particular BS, corresponds to the bandwidth agreed for the MS divided by the total bandwidth of the wireless link. As a result, the time-slot size varies from an MS to another. The BS has knowledge on the specific beginning and ending times of the time-slot for each MS attached to it. Using these two parameters, the network proxy can make an accurate estimation of the delivery time of the next packet for the MS through each available path.

We developed an enhanced version of EDPF dubbed Time-Slotted Earliest Delivery Path First (TS-EDPF), which uses the above mentioned two parameters for an accurate computation of the delivery time of the next packet.

The delivery time through each path is estimated by computing two components; the first component computes the time at which the transmission can begin at the BS on the path, and the second component computes the packet transmission time from such a BS.

The time at which the transmission can begin at the BS is denoted as

$$S_i^l = MAX(a_i + D_l, A_l) \tag{1}$$

where a_i, D_l, and A_l denote the time at which packet i arrives at the proxy, the delay from the proxy to the BS along path l, and the time instants when path l will be available for next transmission, respectively.

The delay on a path l from the proxy to the BS denoted by D_l in Eq. (1) should include the total sum of queuing delay, transmission time, and link propagation delay to the next entity for all entities along the path from the Network Proxy (including itself) to the BS. In this way, we can estimate the packet delivery time more accurately.

We should now ensure that the time, at which the transmission can begin at a BS, is within the slot time assigned to the MS by the BS. Let $[X_l, Y_l]$ be the time-slot period for the MS through path l, and S_i^l be the time at which transmission of packet i can begin at the BS on path l. Furthermore, let $\Gamma(S_i^l, l)$ be the function that returns the next valid time at which the transmission can commence at the BS on path l based on the time-slot $[X_l, Y_l]$.

$$\Gamma\left(S_i^l, l\right) = \begin{matrix} S_i^l & if \quad S_i^l \in \left[X_l, Y_l\right] \\ X_i^{'} & otherwise \end{matrix} \tag{2}$$

where X'_l is the starting time of the subsequent time-slot. The packet transmission time for packet i across link l is computed as follow

$$T_i^l = \frac{L_i}{B_l} \tag{3}$$

where L_i and B_l denote the size of packet i and the bandwidth of path l.

Let E_i^l be the time at which transmission of packet i can finish at the BS on path l.

$$E_i^l = \Gamma(MAX[a_i + D_l, A_l], l) + T_i^l \tag{4}$$

Similarly, we should ensure that the transmission of any packet i at the BS is completed within the time interval $[X_l, Y_l]$. Let $\Theta(E_i^l, l)$ denote the function that returns the next valid time at which the transmission of packet i can finish at the BS on path l based on the time-slot $[X_l, Y_l]$. $\Theta(E_i^l, l)$ can be expressed as follows:

$$\Theta(E_i^l, l) = \begin{matrix} E_i^l & if \quad E_i^l \in \left[X_l, Y_l\right] \\ X_l^{'} + T_i^l & otherwise \end{matrix} \tag{5}$$

The delivery time of packet i, through path l, can be then computed as follows:

$$d_i^l = \Theta\left(\Gamma\left(MAX\left(a_i + D_l, A_l\right), l\right) + T_i^l, l\right) \tag{6}$$

This algorithm estimates the delivery time of a packet through each available path, and sends the packet via the path with the earliest delivery time.

5. Experimental evaluation

This section presents and discusses the performance of our TS-EDPF scheduling algorithm. The goal of this evaluation is to demonstrate three good features of TS-EDPF: i) the efficient use of the overall negotiated bandwidth through each interface, ii) the minimization of the reordering delay, and iii) the minimization of packets loss rate. As comparison terms, we use Weighted Round Robin (WRR), Weighted Interleaved Round Robin (WIRR), and the original Earliest Delivery Path First algorithm (EDPF).

As previously mentioned, in our QoS architecture MSs negotiate with the network the amount of bandwidth that they can use during the current session. An MS with multiple interfaces should negotiate the bandwidth to use through each of these interfaces. After each successful bandwidth negotiation, the network proxy is informed of 1) the pair (BS, MS) involved in the negotiation, and 2) the pair (X, Y) that represents the beginning and end times of the time-slot assigned to the MS. An important feature of our algorithm is that the network proxy does not need to know the quantity of the negotiated bandwidth for the MS because our architecture uses a Time Slot Division strategy to guarantee the QoS to MSs. Thus, the bandwidth of each MS through the wireless link l used to calculate the transmission time of a packet (variable B_l in Eq. 3) is equal to the total bandwidth of this link, which depends on the wireless technology of the BS associated to this link. The network proxy has knowledge of the link bandwidth. It should be reminded that in a Time Slot Division system, each MS uses the total bandwidth of the link during a short period of time.

Several simulations were performed using the Network Simulator (NS2) [15]. We consider one MS equipped with three interfaces that correspond to different wireless technologies supported by the same service provider in a single domain as shown in Fig. 1. The MS has an aggregate bandwidth of 640 Kbps to receive a video streaming from a video server. The maximal playback delay and the maximal transmission delay are set to 70 ms and 300 ms, respectively.

A. Time-slot interval of 1s, no background traffic

Time-slot interval of 1s means that the MS will receive the service from each BS once in one second, the specific time of the service are given by the time-slot assigned to the MS during the negotiation process.

Fig. 2-a shows the actual playback time of the first three hundred packets delivered by the evaluated algorithms. The TS-EDPF scheme outperforms clearly all the other schemes. Fig. 2-b shows the playback delay experienced by packets, that indicates the time for which a packet resides in the buffer awaiting the arrival of preceding packets. Notice that among the first three hundred packets only five packets arrived in out-of-order in case of TS-EDPF. However, in case of the original EDPF, a quite number of packets arrived out of order and this resulted in a longer reordering delay compared to TS-EDPF.

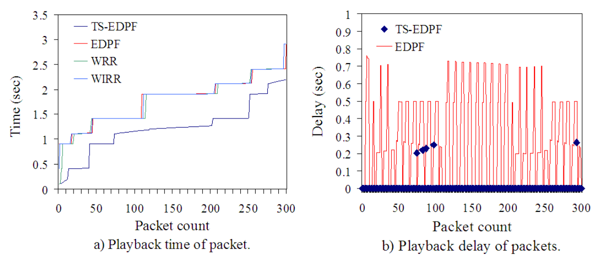

Fig. 2. Playback of the scheduling algorithms.

Table 1. Comparison among scheduling schemes in case Δ = 1s.

Algorithm	Buffer size (pkts)	Longest playback delay (ms)	BW (%)	Disorder delivery ratio (%)	Longest trans. delay (ms)	Packet loss ratio (%)
TS-EDPF	1	205	99.75	0.01	500	89.2
EDPF	78	758	98.99	38.67	1003	98.3
WRR	83	737	98.94	36.21	1001	99.4
WIRR	82	787	98.75	54.77	1008	100

Table 1 shows more detailed results. The buffer size reflects the largest number of packets that were queued in the buffer awaiting playback. The bandwidth ratio indicates how much bandwidth the end-terminal could indeed use out of the agreed bandwidth. The disorder delivery ratio indicates the proportion of packets that arrived in an out-of-order manner. The results of the table demonstrate that the proposed TS-EDPF scheme outperforms the three other schemes in terms of the overall quantifying parameters. Indeed, the proposed scheme ensures high utilization of the network resources while minimizing the number of packets received out of order and thus reducing the associated reordering delay. However, due to the time-slotted approach for bandwidth allocation at the BSs, packets that arrive to the BS in a time different than the time-slot assigned to the destination node are buffered to wait for such time-slot in the next interval of time. Thus their transmission time is dramatically increased, and as a result, many packets are discarded as it is indicated by the high packet loss ratio in the table.

B. Time-slot interval of 0.1s, no background traffic

To mitigate the delay introduced by the time-slotted approach, the time-slot interval is reduced to 0.1s, which means that the MS will receive the service from each BS once in one hundred milliseconds, ten times in one second. The specific time of the service is given by the time-slot assigned to the MS during the negotiation process divided by ten.

Table 2. Comparison in case Δ = 0.1s and no background traffic.

Algorithm	Buffer size (pkts)	Longest playback delay (ms)	BW (%)	Disorder delivery ratio (%)	Longest trans. delay (ms)	Packet loss ratio (%)
TS-EDPF	1	22	99.89	0.6	253	0
EDPF	14	71	98.79	43	310	6.2
WRR	15	79	99.82	12	306	7.4
WIRR	15	77	99.8	41	305	8.5

Table 2 shows the results when the time-slot interval is set to 0.1s. The results of the table indicate that all schemes achieve fairly high throughput.

TS-EDPF shows the best performance: the disorder delivery ratio is 0.6%, the average playback delay is 0.1ms, the average buffer size is only one packet, and zero packet loss. This good performance is attributable to the time-slot based policy enforcement strategy adopted by the TS-EDPF and lacking in the other three schemes.

C. Time-slot interval of 0.1s, with background traffic

The proposed scheme is now compared in a more realistic case; in the presence of high data traffic. The algorithms are compared in three different scenarios, namely Scenario 1, Scenario 2, and Scenario 3. The number of intermediate routers within the path from the network proxy to the BS in Scenario 1, 2, and 3 are set to one, three and five, respectively. The key idea is to evaluate the performance of TS-EDPF when the delivery time of the packet could be affected by the queuing delay at the routers due to the background traffic, and then it may differ from the delivery time estimated by the network proxy.

Fig. 3-a shows how the background traffic affects the estimation of the delivery time of packets through different paths, TS-EDPF is the most sensible

algorithm to the inaccurate estimation of the delivery time. Indeed, for TS-EDPF the buffer size gets increased from one when there is no background traffic, to 4, 23, and 26 in Scenarios 1, 2, and 3, respectively. That means, the higher the number of intermediate routers along the path, the higher the buffer size for TS-EDPF. The buffer size for TS-EDPF takes the largest value.

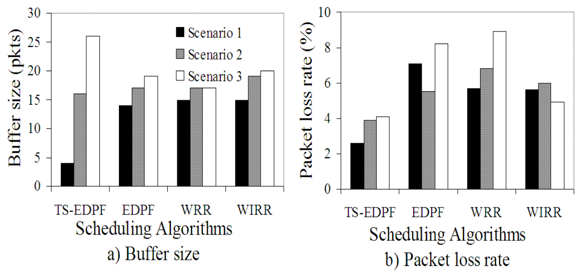

Fig. 3. Comparison among the four scheduling algorithms in case of $\Delta = 0.1$s and background traffic.

The packet loss rate for the evaluated schemes is shown in Fig. 3-b. The queuing delay at the routers, due to the background traffic, augments the packet loss rate of TS-EDPF, yet in a moderate manner. It also affects the packet loss rate of EDPF, WRR, and WIRR algorithms. Indeed, in some scenarios the loss rate increases whereas in other scenarios the loss rate decreases. This unstable performance of the schemes, in terms of packet loss rate, is attributable to the time-slot based policy enforcement strategy that is missing in these algorithms. Effectively, these algorithms deliver packets in an out of order manner, and any change in the overall path delay may improve or deteriorate the packet loss rate.

In general, the buffer size, when TS-EDPF is used, takes the largest value compared to the other evaluated algorithms and when there are five intermediate routers along the path to the BSs. This intuitively increases the packet loss. However, even in the presence of such increment, TS-EDPF is still the most outstanding scheduling algorithm with the lowest packet loss rate.

6. Concluding remarks

The use of multiple interfaces for wireless communication allows users to access several network infrastructures at the same time. This improves the performance of real-time video applications by aggregating the available bandwidth of these interfaces. Delivery of packets through multiple heterogeneous channels introduces the packet reordering issue. We proposed an enhancement to the EDPF scheme to minimize the delay due to packet reordering in multi-path wireless environments. Evaluation of the proposed scheme is performed via simulations and comparison against most popular scheduling algorithms was made. The results show that our algorithm outperforms the most popular scheduling schemes and represents an efficient strategy to deliver real-time video packets under the considered QoS system.

References

[1] K. Chebrolu and R. Rao, “Bandwidth Aggregation for Real-Time Applications in Heterogeneous Wireless Networks,” *IEEE Transactions on Mobile Computing*, Vol. 5, No. 4, pp. 388-403, Apr. 2006.

[2] W. Fritsche and F. Heissenhuber, “Mobile IPv6: Mobility support for Next Generation Internet,” *IPv6 Forum*, White Paper, 2000.

[3] C. Perkins, “IP Mobility support for IPv4,” Network Working Group, RFC 3344, Aug. 2002.

[4] R. Vadali, J. Li, Y. Wu, and G. Cao, “Agent-Based Route Optimization for mobile IP,” *IEEE Commun. Mag.*, Vol. 43, No. 12, pp. 156-163, Dec. 2005.

[6] K. Chebrolu, B Raman, and R. Rao, “A network layer approach to enable TCP over multiple interfaces,” *Wireless Networks*, Vol. 11, No. 5, September 2005.

[7] D. Jurca and P. Frossard, “Distortion Optimized Multipath Video Streaming,” *in Proc. of IEEE Packet Video Workshop (PV'04)*, California, USA, Dec. 2004.

[8] D. Jurca and P. Frossard, “Video Packet Selection and Scheduling for Multipath Streaming,” IEEE Transaction on Multimedia, Vol. 9, No. 3, pp 629-641, Apr.2007.

[9] L. Magalhaes and R. Kravets, “MMTP - Multimedia Multiplexing Transport Protocol,” *in Proc. of 1st ACM Workshop on data Communications in Latin America and the Caribbean (SIGCOMM-LA 2001)*, San Jose, Costa Rica, Apr. 2001.

[10] J.-C. Chen, A. McAuley, V. Sarangan, et al., “Dynamic Service Negotiation Protocol (DSNP) and Wireless DiffServ,” *in Proc. of ICC 2002*, New York, USA, Apr. 2002.

[11] J.C. Fernandez, T. Taleb, N. Ansari, K. Hashimoto, N. Kato, and Y. Nemoto, “Dynamic QoS Negotiation for Next Generation Wireless Communications Systems,” *in Proc. of WCNC 2007*, Hong Kong, Mar. 2007.

[12] T. Taleb, J.C. Fernandez, Y. Nemoto, and N. Kato, A Bandwidth Aggregation-aware QoS Negotiation Mechanism for Next-Generation Wireless Network,” *in Proc. of Globecom 2007*, Washington DC, Nov. 2007.

[13] R. Yavatkar, D. Pendarakis, and R. Guering, “A Framework for Policy-Based Admission Control,” Network Working Group, RFC 2753, Jan. 2000.

[14] T. Taleb, A. Nafaa, L. Murphy, K. Hashimoto, N. Kato, and Y. Nemoto, “Towards Efficient Service-level QoS Provisioning in Large-scale 802.11-based Networks,” *IEEE Network Mag.* Vol. 21, No. 5, pp. 42-48, Sep. 2007.

[15] UCB/LBNL/VINT: Network Simulator - ns (version 2). http://www.isi.edu/nsnam/ns/