SYNCHRONIZED END-TO-END MULTICAST IN REAL-TIME PACKET CELLULAR NETWORKS

Dongyan Xu
Baochun Li

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois

ABSTRACT

In this paper we present and evaluate protocols (SMP and SMP-MH) for synchronized end-to-end multicast in real-time packet cellular networks. Our protocols extend current delay jitter control mechanisms to achieve synchronized real-time packet delivery to mobile hosts in multiple cells. We show that for a multicast session using SMP and SMP-MH: (1) the end-to-end delay bounds of each packet to the subscribing mobile hosts are identical, with small and bounded jitters, and (2) a mobile host can experience smooth, brief, and lossless handoffs, and thus receive seamless Quality of Service (QoS).

1 INTRODUCTION

In packet cellular networks, to support real-time applications such as multimedia, command and control systems, etc., the network should provide seamless Quality of Service (QoS) to mobile hosts as they roam about. In this paper, the problem of real-time multicast in packet cellular networks is studied, the solution of which may be applied to military or civilian information dissemination to mobile users. We consider two QoS parameters: end-to-end packet delay and delay jitter, which are both *location-dependent* and thus should appear seamless to a mobile host. More specifically, during a multicast session, mobile hosts in different cells may observe varying end-to-end delays and delay jitters for the same packet. Consequently, a mobile host may suffer from duplication or loss of packets and large jitters when it moves from one cell to another, resulting in QoS fluctuations.

Related issues are considered in [1] and [2]. An optimal channel allocation algorithm for multicasting within a cell is presented in [2], which maximizes the transmission throughput to mobile hosts. However it does not look at the problem of multicast in a multicell environment. In [1], to enforce synchronized broadcast in cellular networks, protocols are developed to minimize the maximum time difference between the local delivery inception times of a broadcast message by a group of base stations. Handoff procedure is not discussed in detail and is assumed to be lower priority task during synchronized broadcasts.

In this paper, we take an *end-to-end* approach (i.e. from source to mobile hosts) to support synchronized real-time multicast. Protocol SMP is used during multicast channel establishment time to initialize the synchronization, and protocol SMP-MH will be executed by mobile hosts during handoffs. The rest of the paper is organized as follows. Section 2 describes the protocols, Section 3 demonstrates the performance of the protocols using a multimedia multicast session, and Section 4 concludes this paper.

2 PROTOCOL DESCRIPTION

2.1 REAL-TIME PACKET SCHEDULING

Many algorithms have been proposed for real-time packet scheduling at network switches. In this paper we do not assume any single scheduling algorithm used by switches in the cellular networks. We will only assume that every switch adopts an algorithm that belongs to the class of *Guaranteed Rate* (*GR*) scheduling algorithms [3] (many well-known algorithms belong to the *GR* class. Refer to [3] for its formal definition).

For a multicast session in cellular packet networks, in order to eliminate delay and delay jitter variations in different cells, we aim at making the arrivals of the same packet in all cells synchronized. Our first step towards synchronized multicast is to control the jitter on the path from the source to mobile hosts in one cell. Based on current delay jitter control mechanisms [4], we modify algorithms in the *GR* class to be *jitter controlled*. Consider a flow *f* that is associated with a rate r_f. Let p_f^j and l_f denote the *j*th packet of flow *f* and its length. Let $GRC^i(p_f^j)$ and $A^i(p_f^j)$ denote the *Guaranteed Rate Clock* value and arrival time of packet p_f^j at switch *i*. Then $GRC^i(p_f^j)$ is revised as:

$$GRC^1(p_f^0) = 0 \tag{1}$$

$$GRC^i(p_f^j) = E^i(p_f^j) + \frac{l_f}{r_f},\ i \geq 1 \ \text{ and } \ j \geq 1 \tag{2}$$

$$E^1(p_f^j) = max\{A^1(p_f^j), GRC^1(p_f^{j-1})\},\ j \geq 1 \tag{3}$$

$$\begin{aligned} E^i(p_f^j) &= GRC^{i-1}(p_f^j) + \alpha^{i-1} \\ &= A^i(p_f^j) + (GRC^{i-1}(p_f^j) + \beta^{i-1}) - \\ &\quad L^{i-1}(p_f^j),\ i \geq 2 \ \text{ and } \ j \geq 1 \end{aligned} \tag{4}$$

β^i is the constant only dependent on the scheduling algorithm and switch *i*. $\alpha^i = \beta^i + \tau^{i,i+1}$, $\tau^{i,i+1}$ is the propagation delay from switch i to $i+1$. $E^i(p_f^j)$ is the *eligibility time* of packet p_f^j, i.e. the time when p_f^j is put into the ready queue of the scheduler at switch *i*. $L^{i-1}(p_f^j)$ in (4) is the actual time the last bit of p_f^j leaves switch $i-1$. From (3)(4), we have $A^i(p_f^j) \leq E^i(p_f^j)$. Between $A^i(p_f^j)$ and $E^i(p_f^j)$, p_f^j has to be buffered. Therefore to implement an algorithm belonging to the *jitter controlled GR* class, the packet scheduler should also include functions to:

- Timestamp any packet p_f^j with value $(GRC^i(p_f^j) + \beta^i) - L^i(p_f^j)$, so that this value can be passed on to the next switch.
- Extract the timestamp from any p_f^j coming from the previous switch, and hold p_f^j for the amount of time of the timestamp value before putting it into the ready queue at $E^i(p_f^j)$ (the source scheduler simply holds p_f^j until $E^1(p_f^j)$).

We have the following corollary on the end-to-end delay and delay jitter of packet p_f^j based on the results in [3].

Corollary 1 of [3] *If the scheduling algorithm at each of the K ($K \geq 3$) switches on the path of flow f belongs to the jitter controlled GR class, then the end-to-end delay of packet p_f^j is given by*

$$d_f^j \leq GRC^1(p_f^j) - A^1(p_f^j) + (K-1)\frac{l_f}{r_f} + \sum_{i=1}^{K} \alpha^i \tag{5}$$

$$\textit{and} \ \ d_f^j \geq GRC^1(p_f^j) - A^1(p_f^j) + (K-2)\frac{l_f}{r_f} + \sum_{i=1}^{K} \alpha^i - \beta^K \tag{6}$$

i.e. the delay jitter is bounded by $\frac{l_f}{r_f} + \beta^K$.

In our environment of real-time packet cellular networks, a scheduler with a *jitter controlled GR* class algorithm is created for each wired or wireless link. For a multicast session $\mathcal{M}$, a *Multicast Backbone Tree (MBT)* can be formed to link the source (we assume the source is a static host) and the base stations looking over the cells covering the multicast area. By using the *jitter controlled GR* class algorithms, the link schedulers can now guarantee that the delay jitters from the source to mobile hosts in the same cell, i.e. the *intra-cell* delay jitters, be bounded.

2.2 END-TO-END SYNCHRONIZATION

We have not yet considered the variations of end-to-end delays from the source to mobile hosts in *different* cells, i.e. the *inter-cell* delay jitter. We now further refine our solution.

During the multicast channel establishment phase, beginning from the leaves of *MBT* (they are base stations), any switch that has more than one outgoing links for this multicast can gather delay values (formal definition below) summed up and passed along from downstream switches in the *MBT*. Then it sets *additional holding time* for each outgoing link as the difference between the largest delay value and the one received from (the reverse of) this link. Finally it passes the largest delay value to its upstream switch in the *MBT*. After the operations are performed at each switch from the leaves back to the source, the end-to-end synchronization of packet delivery for $\mathcal{M}$ is set up.

Given multicast session $\mathcal{M}$ and its *MBT* with source *s*, the multicasting area is covered by a set of cells *C*. We assume that *C* is *connected* in that a mobile host can move between any two locations in the area without crossing a cell $c \notin C$. r_m amount of bandwidth has been reserved on each link (wired or wireless) for $\mathcal{M}$. All packets are of size l_m. Functions $parent()$, $children()$, and $outlinks()$ map a switch to its parent switch, set of children switches, and set of outgoing links in the *MBT*, respectively. Function $b()$ maps any cell in *C* to its base station. Function $enable\_jitter\_control(link\_id, additional\_holding\_time)$ is called at each switch to set the link scheduler(s) to *jitter control* mode. The first parameter identifies the link, and the second parameter is the time to hold packets for *inter-cell* jitter control, in addition to the holding time for *intra-cell* jitter control described in 2.1. We now describe the *Synchronized Multicast Protocol* (SMP) in Figure 1.

When establishing multicast channel for $\mathcal{M}$:

for each leaf switch x
  $d_x^0 \leftarrow \frac{l_m}{r_m} + \alpha_x^0$; // superscript 0 indicates a wireless link
  $d_x^{max} \leftarrow d_x^0$;
  send d_x^{max} up to $parent(x)$;
  $enable\_jitter\_control(0, 0.0)$;

for each non-leaf switch y
  if exists $c_y \in C$ such that $y = b(c_y)$
    $d_y^0 \leftarrow \frac{l_m}{r_m} + \alpha_y^0$;
  else
    $d_y^0 \leftarrow 0$;
  receive d^{max} values from switches in $children(y)$;
  for each $i \in outlinks(y)$
    $d_y^i \leftarrow \frac{l_m}{r_m} + \alpha_y^i + (d^{max}$ value received from the reverse of i);
  $d_y^{max} \leftarrow max_{i \in outlinks(y) \cup \{0\}} d_y^i$;
  if exists $c_y \in C$ such that $y = b(c_y)$
    $enable\_jitter\_control(0, (d_y^{max} - d_y^0))$;
  for each $i \in outlinks(y)$
    $enable\_jitter\_control(i, (d_y^{max} - d_y^i))$;
  if $y \neq s$
    send d_y^{max} up to $parent(y)$;

Figure 1: Protocol SMP

During the multicast transmission, each link scheduler works in the jitter control mode. The calculation of $GRC^{(y,i)}(p_m^j)$ and $E^{(y,i)}(p_m^j)$ for packet p_m^j by the scheduler of link i from switch y (link denoted as (y,i)) is revised as follows. The *additional holding time* $(d_y^{max} - d_y^i)$ in (9) (10) is already set by SMP. $Pl(y)$ denotes the link from $parent(y)$ to y.

$$GRC^{(s,i)}(p_m^0) = 0 \tag{7}$$

$$GRC^{(y,i)}(p_m^j) = E^{(y,i)}(p_m^j) + \frac{l_m}{r_m},\ j \geq 1 \tag{8}$$

$$\begin{aligned} E^{(s,i)}(p_m^j) =& max\{A^s(p_m^j), GRC^{(s,i)}(p_m^{j-1})\} + \\ =& (d_s^{max} - d_s^i),\ j \geq 1 \end{aligned} \tag{9}$$

$$\begin{aligned} E^{(y,i)}(p_m^j) =& GRC^{Pl(y)}(p_m^j) + \alpha^{Pl(y)} + (d_y^{max} - d_y^i) \\ =& A^y(p_m^j) + ((GRC^{Pl(y)}(p_m^j) + \beta^{Pl(y)}) - \\ & L^{Pl(y)}(p_m^j) + (d_y^{max} - d_y^i),\ y \neq s \end{aligned} \tag{10}$$

Protocol SMP only requires one bottom-up pass in the *MBT* to initialize the synchronization, and can be used in conjunction with the bandwidth reservation protocol (such as *RSVP*[6]) for $\mathcal{M}$. To show that both intra-path and inter-path delay jitters are controlled by SMP, we first have the following lemma.

Lemma 1 *For a multicast session $\mathcal{M}$ to an area covered by a set of connected cells C, if each link is scheduled by an algorithm belonging to the jitter controlled GR class, and SMP is used during the multicast channel establishment phase, then the end-to-end delay $d_{m\to c}^j$ of packet p_m^j to mobile hosts in any cell $c \in C$ is bounded by*

$$D_m^j - (\frac{l_m}{r_m} + \beta^{(b(c),0)}) \leq d_{m\to c}^j \leq D_m^j \tag{11}$$

where $(b(c), 0)$ denotes the wireless link from the base station of c, and D_m^j is defined as

$$\begin{aligned} D_m^j =& GRC^{(s,I)}(p_m^j) - A^s(p_m^j) + (K_{c_{max}} - 1)\frac{l_m}{r_m} \\ &+ \sum_{(y,I)\ on\ path\ P} \alpha^{(y,I)} \end{aligned} \tag{12}$$

$K_{c_{max}}$ is the number of switches on path P, which is from s to (any mobile host in) cell c_{max}, and $c_{max} \in C$ is chosen such that $K_c \frac{l_m}{r_m} + \sum_{(y,i)\ from\ s\ to\ c} \alpha^{(y,i)}$ has the largest value when $c = c_{max}$. Link (s, I) denotes the first link on P.

With Lemma 1 it is easy to derive the following theorem on the synchronized arrivals of the same packet at mobile hosts in any two adjacent cells.

Theorem 1 *Let $A^c(p_m^j)$ be packet p_m^j's arrival time at a mobile host in cell c. Under the conditions described in Lemma 1, for any two adjacent cells $c, c' \in C$, it holds that*

$$|A^c(p_m^j) - A^{c'}(p_m^j)| \leq \frac{l_m}{r_m} + max\{\beta^{(b(c),0)}, \beta^{(b(c'),0)}\} \tag{13}$$

From Theorem 1, we know that the arrival times of the same packet to any two adjacent cells differ at most by a small constant $\frac{l_m}{r_m} + max\{\beta^{(b(c),0)}, \beta^{(b(c'),0)}\}$, independent of the paths from *s* to the cells. For the rest of the paper, we make a further assumption that the *GR* class algorithms for the wireless links are chosen such that the β values are smaller than $\frac{l_m}{r_m}$. Hence we have:

$$\begin{aligned} |A^c(p_m^j) - A^{c'}(p_m^j)| \leq& \frac{l_m}{r_m} + max\{\beta^{(b(c),0)}, \beta^{(b(c'),0)}\} \\ <& \frac{2l_m}{r_m} \end{aligned} \tag{14}$$

2.3 HANDOFF PROCEDURE

In this section we show how a mobile host subscribing to a real-time multicast session $\mathcal{M}$ can achieve smooth (no large jitters), brief, and lossless handoffs when it moves from cell to cell, if $\mathcal{M}$ has been set up as synchronized by SMP. Suppose a mobile host *MH* is moving from cell *c* to a neighboring cell *c'*. When *MH* receives strong enough beacons from the new base station $b' = b(c')$, *MH* will contact *b'*, which will in turn acknowledge *MH* with the identifier of the wireless channel for $\mathcal{M}$ in cell *c'*, and the value of $\beta^{(b',0)}$. All the interactions are performed via a separate control channel between *b'* and *MH*.

Now *MH* can receive multicast data on both the old wireless channel from $b = b(c)$, and the new one from *b'*. We aim at making the duration of this transient state as short as possible. Base on Theorem 1, we can show that *MH* will be in this state of listening to both channels for at most $\frac{l_m}{r_m} + max\{\beta^{(b,0)}, \beta^{(b',0)}\}$ amount of time, then it quits the old channel and completes the handoff procedure with no packet loss.

In figure 2, we describe protocol SMP-MH, which is executed by mobile hosts during handoffs. Variable $next\_packet\_number$ identifies the next packet *MH* is expecting. We also assume that mechanism already exists to discard duplicated packets based on current $next\_packet\_number$ value.

We have the following theorem on properties of SMP-MH.

Theorem 2 *Given a multicast session $\mathcal{M}$ with its MBT set up using SMP, for a mobile host entering cell c' from cell c, the execution time of protocol SMP-MH is bounded by $\frac{l_m}{r_m} + max\{\beta^{(b,0)}, \beta^{(b',0)}\}$. No packet will be lost.*

Theorem 2 indicates that the handoff procedure using SMP-MH is brief and lossless for mobile hosts. Furthermore, despite the possible need to buffer and re-order one packet, the actual delivery time of the buffered packet to the upper layer of mobile host still falls into its expected jitter controlled arrival time interval. Therefore the handoff is smooth with no extra jitter incurred for any packet.

Immediately after MH has received packet p_m^j from b (denoted as $b.p_m^j$) via the old channel, begin tuning in to both the old and new channels.
$next\_packet\_number$ is now $j+1$.
begin a timer with time-out period $\frac{l_m}{r_m} + max\{\beta^{(b,0)}, \beta^{(b',0)}\}$
case the next packet received is:
$b.p_m^{j+1}$: $next\_packet\_number++$;
break;
$b'.p_m^{j-1}$: **break**;
$b'.p_m^{j}$: **break**;
$b'.p_m^{j+1}$: $next\_packet\_number++$;
break;
$b'.p_m^{j+2}$: buffer $b'.p_m^{j+2}$;
wait for $b.p_m^{j+1}$;
deliver $b'.p_m^{j+2}$ after $b.p_m^{j+1}$;
$next\_packet\_number =$
$next\_packet\_number + 2$;
break;
none until time-out: **break**;
quit the old channel and clear the timer

Figure 2: Protocol SMP-MH

3 PERFORMANCE EVALUATION

In this section we will present the performance results obtained from the simulation of a multimedia multicast session $\mathcal{M}$ shown in Figure 3. $\mathcal{M}$'s multicasting area is covered by cells numbered 1 to 12. s is the source host. We use an MPEG video trace with frame rate of 15 *frames/s* to generate source traffic. The required bandwidth r_m for this MPEG video stream is 1.2 Mbps. *Virtual Clock* [5] enhanced with jitter control is used as the scheduling algorithm ($\beta^i = \frac{l_m}{R}$ for link i with bandwidth R). Packet size is $1KB$.

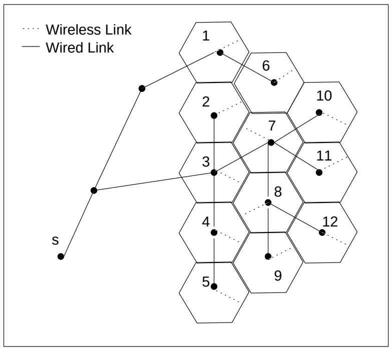

Figure 3: Experiment Setup: A Multicast Session

Figure 4 compares the QoS of $\mathcal{M}$ when SMP is used (Figure 4), and when SMP is not used (Figure 5). In both cases, we record the end-to-end delays of the first 100 MPEG frames (i.e. the delay of the last packet in each frame) observed in 4 of the cells (#4, 5, 8, and 9). Figure 4 indicates that the same frame's arrivals are well synchronized. On the

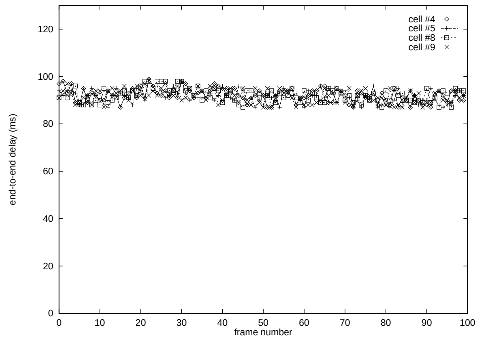

Figure 4: End-to-End Delays of the First 100 MPEG Frames: SMP used

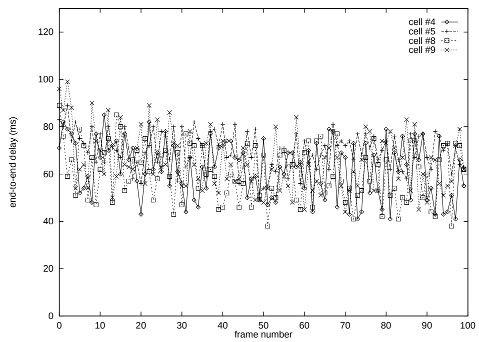

Figure 5: End-to-End Delays of the First 100 MPEG Frames: SMP not used

contrary, Figure 5 shows that the arrival times of the same frame vary substantially. A mobile host may even find the video some frames backward or forward after a handoff.

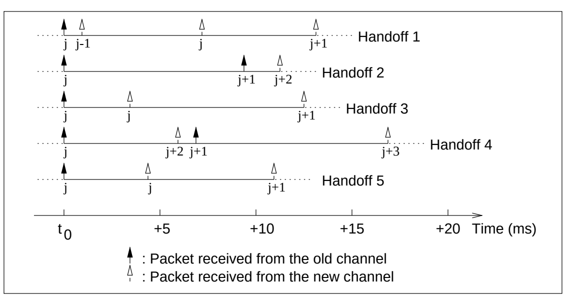

Figure 6: Packet Arrivals During Handoffs

We also simulate a subscribing mobile host moving from one cell to another, whose handoffs are controlled by SMP-MH. Figure 5 shows five of the handoffs it experiences after $\mathcal{M}$ has been initialized by SMP. t_0 denotes the time instance it begins to execute SMP-MH. i denotes sequence number of the packet received right before t_0 (handoff 4 reflects the case when a packet $(i+2)$ has to be buffered). The results show that SMP-MH guarantees smooth, short, and lossless handoffs.

4 CONCLUSION

We have described protocols SMP and SMP-MH to support end-to-end synchronized real-time multicasting in packet cellular networks. Both theoretical and experimental results indicate that the protocols can provide seamless QoS (delay and delay jitter) to mobile hosts, whose handoff procedures are smooth, brief, and lossless. One problem, which also exists in the current delay jitter control schemes, is that the actual end-to-end delays during a session are all close to the worst case delay bound. For future work, we will consider the possibility of using QoS-based multicast routing in order to make the *MBT* more balanced, thus reducing the uniform end-to-end delay bound for the multicast session initialized by SMP.

References

[1] D.O. Awduche and A. Ganz. Synchronized Broadcast in Cellular Networks. *Proceedings of 2nd Telecommunications R&D Conference in Massachusetts*, March 1996.

[2] K. Brown and S. Singh. The Problem of Multicast in Mobile Networks. *Proceedings of IEEE 5th International Conference on Computer Communications and Networks*, October 1996.

[3] P. Goyal and H.M. Vin. Generalized Guaranteed Rate Scheduling Algorithms: A Framework. *IEEE/ACM Trans. Networking*, 5, August 1997.

[4] H. Zhang. Providing End-to-End Performance Guarantees Using Non-Work-Conserving Disciplines. *Computer Communications: Special Issue on System Support for Multimedia Computing*, 18, October 1995.

[5] L. Zhang. VirtualClock: A New Traffic Control Algorithm for Packet Switching Networks. *Proceedings of ACM SIGCOMM'90*, August 1990.

[6] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. RSVP: A New Resource ReSerVation Protocol. *IEEE Network Magazine*, September 1993.