Multiple Quality Control: A New Framework for QoS Control in Forward Link of 1xEV-DV Systems

Jeong-woo Cho[†], Taesoo Kwon[†], Changhoi Koo^{††}, DS Park^{††}, Daegyun Kim^{††} and Dong-ho Cho[†]

[†] Department of Electrical Engineering and Computer Science
Korea Advanced Institute of Science and Technology (KAIST)
373-1 Kusong-dong, Yusong-gu, Daejeon 305-701, Korea

^{††} Samsung Electronics

Abstract— One of main issues in 1xEV-DV systems is how to support and satisfy various QoS requirements of data services and voice service simultaneously. Another main issue in 1xEV-DV systems is how to increase throughput of data services without sacrificing voice service's QoS requirements. The main motivation for our proposed architecture, MQC (Multiple Quality Control), is to use per-stream buffers in physical layer and to fully utilize air resource. The problem of existing architecture is that PDUs from only one stream can be transmitted and those PDUs may not fill out the allocated time slots. In contrast to existing architecture, MQC can multiplex several streams into one PLP(Physical Layer Packet) because MQC utilizes per-service buffers in physical layer. We also introduce a multiplexing scheme that minimizes queueing delays of realtime traffic and maximizes cell throughputs.

I. INTRODUCTION

Currently, there are rapidly increasing demands on data services, such as web service, ftp service, and realtime video service in mobile communication systems. To support data services, 1xEV-DO [1], [2] and 1xEV-DV [3] systems are currently considered as major systems because conventional systems, such as IS-95B and cdma2000-1x, are not designed to support various data services and their bandwidths are quite limited. To overcome bandwidth limitation, 1xEV-DO system supports maximum 2Mbps in forward link and maximum 153.6kbps in reverse link. Moreover, 1xEV-DV systems, whose standardization is currently being under progress, would support quite more bandwidth and operate on the same RF carrier as current cdma2000-1x services, and support various data services concurrently.

One of the main issues in 1xEV-DV systems is how to support and satisfy various QoS requirements of data services and voice services simultaneously. According to TSG-S open requirement for 1xEV-DV systems, 1xEV-DV systems shall support "Mixed realtime and Non-realtime traffic on the same carrier and multiple(concurrent) packet data session". These requirements can give points of excellence compared with IS-2000 or 1xEV-DO.

Therefore, mixed and concurrent services, which consist of realtime and non-realtime traffic simultaneously, shall be provided as a part of target services for 1xEV-DV systems and protocol architecture for those services shall be designed and implemented. And also, to achieve better performance of 1xEV-DV than that of HSDPA being designed in 3GPP, multimedia or multiple-media services shall be supported in 1xEV-DV systems.

However, current system architecture including IS-2000 and 1xEV-DO do not have functionalities and protocols to support

services that require multiple QoS grades, because they have been designed to service single or multiple services with the same QoS level. Thus, to support multimedia and multiple services in 1xEV-DV, it is required that the new functionalities or architecture should be designed and adopted for 1xEV-DV systems.

In this paper, we introduce a new architecture for support of concurrent services that consist of realtime and non-realtime traffics. One of main problems in existing architecture is that PDUs from only one service can be transmitted at a time and those PDUs may not fill out the allocated time slots which are determined based on C/I measurements. To overcome this problem, MQC (Multiple Quality Control) architecture uses per-service buffers in physical layer and multiplexes several services' frames into one PLP (Physical Layer Packet). Main advantage of our architecture is to increase cell throughputs. We also introduce a new multiplexing scheme that not only increases cell throughputs but also minimizes delays of realtime traffic. Moreover, our architecture is very simple and can be easily implemented.

The remainder of this paper is organized as follows. In Section II, we describe forward link scheduling systems that is currently considered in 1xEV-DV systems. And, we will introduce our new architecture named MQC in Section III. Advantages of our proposed architecture and multiplexing scheme also will be introduced in Section III. In Section IV, we present a detailed simulation model and some simulation results to examine the properties of our proposed architecture and multiplexing scheme. Finally, we would make a conclusion in Section V.

II. FORWARD LINK SCHEDULING ARCHITECTURE IN 1xEV-DV SYSTEMS

In this section, we describe the details of scheduling procedures in forward link of 1xEV-DV systems [4]. The main scheduling operation is to select an MS(Mobile Station) to be served, and to decide data rate, the number of allocated time slots, and the number of PDUs (Packet Data Unit) to be transmitted. Currently, basic time slot length is set to 1.25ms and basic PDU size is set to 384bits. Existing architecture that is currently considered has only one Tx (Transmission) buffer in physical layer as shown in Figure 1. In upper layers, MUX&QoS entities serve the realtime traffic without server switching until the buffer for the realtime traffic is empty. That is, the upper layer scheduling only provides the prioritized transmission control.

Because there are no per-service Tx buffers in the physical

Fig. 1. Existing Tx buffer filling structure in physical layer

layer, PDUs from realtime traffic and non-realtime traffic can be mixed in the Tx buffer as shown in Table 1. To support strictly prioritized transmission, the physical layer should know the pointers to PDUs from realtime traffic. Furthermore, in order to support prioritized services, the physical layer should be able to access them randomly. That is, there should be some hardware such as CAM (Content Addressable Memory) that is hard to be implemented with low cost so that each PDU can be read with random access by physical layer scheduler.

Periodically, BS (Base Station) measures C/I values at every time slot and reports the measured C/I values to upper layers. When BS sends PDUs to an MS (Mobile Station), upper layer decides data rate based on C/I values reported to upper layers. This data rate is decided as shown in Table II. We will call this data rate **R1**. For example, if the measured C/I value is 7.0dB, **R1** would be 921.6kbps.

Final data rate **R2** and the number of allocated time slots **TS** are decided based on **R1** and the number of PDUs in buffer as shown in Table II. On account of space consideration, the cases when number of PDUs in buffer is 4, 5, 8 and 9 are not shown in Table II. Shaded parts in Table II indicate the cases when the final data rate **R2** is smaller than **R1**. It should be remarked that **R2** is more likely equal to **R1** as number of PDUs in buffer increases. In this table, each **R2** also can be calculated as follows:

$$\mathbf{R2} = \frac{\text{No. of PDUs to send} \times 384\text{bits}}{\text{TS} \times 1.25\text{ms}} \quad (1)$$

TABLE I
C/I VALUE TO **R1** MAPPING TABLE

Measured C/I value	R1 (kbps)
$-\infty < C/I \leq -9.0\text{dB}$	38.4
$-9.0\text{dB} < C/I \leq -6.0\text{dB}$	76.8
$-6.0\text{dB} < C/I \leq -3.0\text{dB}$	153.6
$-3.0\text{dB} < C/I \leq -2.0\text{dB}$	230.4
$-2.0\text{dB} < C/I \leq 0.0\text{dB}$	307.2
$0.0\text{dB} < C/I \leq 1.0\text{dB}$	384
$1.0\text{dB} < C/I \leq 2.0\text{dB}$	460.8
$2.0\text{dB} < C/I \leq 4.0\text{dB}$	614.4
$4.0\text{dB} < C/I \leq 6.0\text{dB}$	768
$6.0\text{dB} < C/I \leq 8.0\text{dB}$	921.6
$8.0\text{dB} < C/I \leq 10.0\text{dB}$	1228.8
$10.0\text{dB} < C/I \leq 11.0\text{dB}$	1536
$11.0\text{dB} < C/I \leq 12.0\text{dB}$	1843.2
$12.0\text{dB} < C/I \leq 14.0\text{dB}$	2457.6

III. MULTIPLE QUALITY CONTROL ARCHITECTURE AND PROPOSED MULTIPLEXING SCHEME

One mobile station may want to use voice service and data service simultaneously. For example, someone may want to enjoy both of voice communication and web service concurrently. It should be reminded that voice services are delay-sensitive and voice frames that are delayed for more than a certain threshold are useless while data services are not so sensitive to delays. To minimize delays of voice frames, they should be served with highest priority. But, if only voice frames are served without filling allocated time slots, those unallocated time slots are wasted. The first step to fill the wasted time slots is to know the pointers to all voice frames and data frames in physical layer. The most easiest way to know the pointers to voice frames and to reduce implementation overhead is to buffer voice frames and data frames in different buffers and to serve them in First-Come First-Served manner. The second step is to place buffer in physical layer. If per-service buffers are not placed in physical layer, data rate and the number of allocated time slots based on C/I measurements are reported to upper layers with a delay and upper layers send incorrect number of frames to physical layer. Main features of MQC (Multiple Quality Control) architecture are to place per-service buffers in physical layer and to serve realtime traffic with highest priority. These features are helpful in view of increasing cell throughputs and serving realtime traffic with highest priority. Furthermore, it should be reminded that there is no need to report C/I measurements to upper layers in MQC architecture because data rate and the number of allocated time slots are decided in physical layer.

MQC is well suited to multiplex realtime and non-realtime frames. By the word “multiplexing”, we mean that realtime traffic and non-realtime traffic are encoded into a single PLP (Physical Layer Packet) and transmitted at once. A simple way to multiplex is just to select the largest number of realtime frames available in the corresponding buffer and to fill the remaining space with non-realtime frames. But, this type of multiplexing which we call **Full Mux** cannot minimize the queueing delays of realtime frames because allocated time slots are fully exhausted and each MS is served with longer delays. On the other hand, we can give up multiplexing to decrease the queueing delays of realtime frames. When no multiplexing scheme is used and

TABLE II
R1 TO R2 AND TS MAPPING TABLE FOR INITIAL TRANSMISSION

No. of PDUs in buffer	R1 (kbps)	R2 (kbps)	No. of PDUs to send	TS	R1>R2	No. of PDUs in buffer	R1 (kbps)	R2 (kbps)	No. of PDUs to send	TS	R1>R2
≥ 10	38.4	38.4	1	8	No	2, 3	38.4	38.4	1	8	No
	76.8	76.8	2	8	No		76.8	76.8	2	8	No
	153.6	153.6	4	8	No		153.6	153.6	2	4	No
	230.4	230.4	6	8	No		230.4	153.6	2	4	Yes
	307.2	307.2	8	8	No		307.2	307.2	2	2	No
	384	384	10	8	No		384	307.2	2	2	Yes
	460.8	460.8	6	4	No		460.8	307.2	2	2	Yes
	614.4	614.4	8	4	No		614.4	614.4	2	1	No
	768	768	10	4	No		768	614.4	2	1	Yes
	921.6	921.6	6	2	No		921.6	614.4	2	1	Yes
	1228.8	1228.8	8	2	No		1228.8	614.4	2	1	Yes
	1536	1536	10	2	No		1536	614.4	2	1	Yes
	1843.2	1843.2	6	1	No		1843.2	614.4	2	1	Yes
	2457.6	2457.6	8	1	No		2457.6	614.4	2	1	Yes
	3072	3072	10	1	No		3072	614.4	2	1	Yes
6, 7	38.4	38.4	1	8	No	1	38.4	38.4	1	8	No
	76.8	76.8	2	8	No		76.8	76.8	1	4	No
	153.6	153.6	4	8	No		153.6	153.6	1	2	No
	230.4	230.4	6	8	No		230.4	153.6	1	2	Yes
	307.2	307.2	4	4	No		307.2	307.2	1	1	No
	384	307.2	4	4	Yes		384	307.2	1	1	Yes
	460.8	460.8	6	4	No		460.8	307.2	1	1	Yes
	614.4	614.4	4	2	No		614.4	307.2	1	1	Yes
	768	460.8	6	4	Yes		768	307.2	1	1	Yes
	921.6	921.6	6	2	No		921.6	307.2	1	1	Yes
	1228.8	1228.8	4	1	No		1228.8	307.2	1	1	Yes
	1536	921.6	6	2	Yes		1536	307.2	1	1	Yes
	1843.2	1843.2	6	1	No		1843.2	307.2	1	1	Yes
	2457.6	1843.2	6	1	Yes		2457.6	307.2	1	1	Yes
	3072	1843.2	6	1	Yes		3072	307.2	1	1	Yes

there are realtime frames, only realtime frames are transmitted to support prioritized realtime service. Non-realtime frames are transmitted only when there are no realtime frames. By using this type of multiplexing which we call **No Mux**, the delays of realtime frames are minimized. But, cell throughputs can be decreased because **No Mux** does not ever try to fill the remaining space with non-realtime frames.

To overcome the defects of **Full Mux** and **No Mux**, we propose a new multiplexing scheme that can be helpful in view of reducing queueing delays of realtime traffic as well as increasing cell throughputs. Main idea of our proposed multiplexing scheme, **New Mux** is to multiplex realtime and non-realtime frames conditionally. Scheduler uses Table II twice for a transmission. Then, scheduler compares the number of allocated time slots, **TS** when only realtime frames are transmitted with that when both realtime and non-realtime frames are multiplexed and transmitted. It should be noted that “Number of PDUs in buffer” in Table II for the case only realtime frames are transmitted is always smaller than or equal to that for the case realtime and non-realtime frames are multiplexed and transmitted. If **TS** for the case of multiplexing is larger than that for the case of no multiplexing, **TS** for the case of no multiplexing is selected. In other cases, **TS** for the case of multiplexing is se-

lected. That is, scheduler decides to transmit only voice frames if transmitting voice frames with data frames increases the number of allocated time slots. Therefore, the queueing delays of realtime frames are not increased and maintained the same as that for the case that **No Mux** is used. There are two factors that can affect cell throughputs. **First**, because **New Mux** does not multiplex non-realtime frames for the case allocated time slots **TS** are increased, **New Mux** can reduce cell throughputs. **Secondly**, **New Mux** waits until the number of non-realtime frames in buffer is sufficiently large and thus can reduce the occurrence of the case that **R2** is smaller than **R1** as shown in Table II. Second factor may be helpful in increasing cell throughputs. Inferring from the second factor, we can see that cell throughputs of **New Mux** can be increased if load of non-realtime traffic is high.

IV. SIMULATION RESULTS AND DISCUSSION

To verify our proposed MQC architecture and multiplexing scheme, we used a single cell configuration where a hexagonal cell inscribed in a circle of 5000m radius is employed. BS is located at center and MSs are not permitted to be located within a radius of 35m from the center. We used round robin scheduling where each MS is served sequentially. AWGN channel model

that has average FER (Frame Error Rate) of 0.01 is used for wireless link. Each MS has two buffers, where one is used for real-time traffic and the other is used for non-realtime traffic. Path loss and intra-cell interference are also considered. There are 3 kinds of traffic model we used for simulations. For packet voice model, voice frame size is set to 320bits. Inter-arrival times of voice frames are exponentially distributed with a mean value of 20ms so that average arrival rate of voice frames is set to 16kbps. For web traffic model, we used HTTP model and TCP model described in [5]. For full-loading traffic model, a buffer is always filled with 30 PDUs (11520bits). When some frames are served, the buffer is filled with new PDUs so that the number of PDUs in the buffer is always 30. Through simulation of **Full Mux** and **New Mux**, we want to show that MQC can increase cell throughputs and concurrent services are now possible. Each simulation was done for 400,000,000 time slots and positions of all MSs are relocated within the cell for each 800,000 time slots.

In the first simulation, we used packet voice model for the first buffer and web traffic model [5] for the second buffer. Cell throughputs and queueing delays of each buffer are shown in Figure 2, 3 and 4.

Fig. 2. Cell throughput in first simulation

Fig. 3. Queueing delays of frames in the 1st buffer in first simulation

Fig. 4. Queueing delays of frames in the 2nd buffer in first simulation

Because load of traffic increases as number of MS increases, cell throughput increases accordingly. As shown in Figure 2, we can see that **Full Mux** increases cell throughput. On the other hand, cell throughput of **No Mux** is smaller than any other scheme. It should be reminded that MQC architecture allows easy implementation of multiplexing so that **Full Mux** and **New Mux** can increase cell throughput. Our proposed **New Mux** scheme minimizes queueing delays and shows the same queueing delays as **No Mux** as shown in Figure 3. Queueing delays of non-realtime traffic in Figure 4 are mainly related to cell throughput. As cell throughput becomes higher, the queueing delay of non-realtime traffic becomes shorter. **New Mux** maintains the queueing delays of non-realtime traffic that is smaller than that of **No Mux** but larger than that of **Full Mux**.

We can see **New Mux** minimizes the queueing delays of real-time traffic and increases cell throughputs. But, cell throughput of **New Mux** is somewhat lower than that of **Full Mux**. It should be remarked that web traffic model has too long off-times about 30s that correspond to reading times. To say in other words, load of web traffic model described in [5] is underestimated because an MS will not wait 30s additionally after downloading web pages for about 50s as shown in Figure 4. Furthermore, a multiplexing scheme should not increase the queueing delays of real-time traffic to support truly concurrent services even if there is heavy non-realtime traffic. If an MS experiences longer queueing delays of voice frames when the MS uses data services such as web and ftp, the MS will not use data services during voice service.

Considering these two factors, we have done second simulation where traffic load of second buffer is somewhat high. We used full-loading traffic model instead of web traffic model for the second buffer. Results are shown in Figure 5, 6 and 7.

When number of MS is relatively small, in case of **No Mux**, average number of PDUs in realtime buffer is very small and only small number of voice frames are transmitted in many cases. As number of MS and load of packet voice increase, the frequency that only voice frames are transmitted increases and cell throughput decreases accordingly in case of **No Mux**. However, cell throughputs of **Full Mux** and **New Mux** are not

Fig. 5. Cell throughput in second simulation

Fig. 6. Queueing delays of frames in the 1st buffer in second simulation

Fig. 7. Queueing delays of frames in the 2nd buffer in second simulation

so changing as number of MS varies because they use multiplexing. As shown in Figure 5, cell throughput of **New Mux** is now comparable to or even larger than that of **Full Mux** because **New Mux** reduces the occurrence of the case that **R2** is smaller

than **R1** as shown in Table II. Furthermore, **New Mux** maintains smaller queueing delays of realtime traffic which are the same as those of **No Mux** as shown in Figure 6. Enlarged cell throughput of **New Mux** also minimizes the queueing delays of non-realtime traffic as shown in Figure 7.

V. CONCLUSIONS

To control QoS requirements of realtime and non-realtime traffic in forward link of 1xEV-DV systems, we have proposed MQC architecture whose main feature is to place per-service Tx buffers in physical layer. By exploiting per-service buffers, MQC allows easy implementation of prioritized services. MQC also increases cell throughput by multiplexing several services' frames into one PLP (Physical Layer Packet). To increase cell throughputs and to minimize the queueing delays of realtime traffic simultaneously, we have proposed a new multiplexing scheme which multiplexes realtime and non-realtime traffic conditionally. Using proposed scheme, realtime traffic is affected not at all by non-realtime traffic and strictly prioritized services are supported. Thus, we believe that the support of truly concurrent realtime and non-realtime services is now possible in 1xEV-DV systems.

REFERENCES

- [1] 3GPP2 C.S0024, "cdma2000 High Rate Packet Data Air Interface Specification," Version 2.0, October 27, 2000.
- [2] 3GPP2 S.R0023, "High-Speed Data Enhancements for cdma2000 1x-Data Only," Version 2.0, December 5, 2000.
- [3] 3GPP2 S.R0026, "High-Speed Data Enhancements for cdma2000 1x-Integrated Data and Voice," Version 1.0, October 17, 2000.
- [4] Changhoi Koo, DS Park, DG Kim and BS Bae, "MQC Architecture and Upper Layer for 1xEV-DV," Submitted to L3QS Forward Link Framework Proposal.
- [5] WG5 Evaluation AHG, "1xEV-DV Evaluation Methodology – Addendum (V4)," Contribution 3GPP2-C50-20010709-021R1, July 12, 2001.