APPLICATION QOS MANAGEMENT FOR THE REVERSE LINK OF A 3G NETWORK

Patrick Hosein

Ericsson Wireless Communications Inc., 5012 Wateridge Vista Dr., San Diego, CA 92121, US, patrick.hosein@ericsson.com

Abstract - Third Generation (3G) wireless networks have been designed to provide high speed data services on both forward and reverse links. In addition to increased data capacity they also support QoS (Quality of Service) for mobile users. Users may run multiple applications (service instances) simultaneously and QoS guarantees must be assured for one or more of them. In this paper we address the problem of providing QoS guarantees on the reverse link of a 3G (CDMA) network. We present a flexible, low overhead framework for efficient management of the reverse link resources to satisfy the QoS guarantees of each user’s applications. We show how the QoS demands of all service instances of a particular user can be converted and combined into a single resource demand, namely the data rate of the corresponding reverse link channel.

Keywords - CDMA, QoS, 1XEV-DV, 1XEV-DO, WCDMA

I. INTRODUCTION

Third Generation networks provide users with significantly higher data throughput on both the forward and reverse channels. However, these presently provide best effort services to users. Recent enhancements to the 1XEV-DV standard (see [1]) allow for the provision of QoS guarantees to mobile users. Similar features are being included in 1XEV-DO (see [2]) and WCDMA. These enhancements allow operators to provide guarantees on one or more application attributes such as goodput, jitter, delay and error rate (see [3], [4], [5]). Users may run multiple QoS enabled applications simultaneously. Each is supported by a corresponding service instance (distinguished by its service identification number).

We focus on the reverse link although the proposed framework can also be applied to the forward link. We also focus on the 1XEV-DV version of these QoS enhancements but similar frameworks can be used for the other 3G systems. We assume that each active user (i.e. one with data for transmission) is provided with a single reverse link traffic channel (the Reverse Packet Data Channel or R-PDCH) over which data from multiple service instances are multiplexed. These mobile channels can be either time shared or code shared.

In the 1XEV-DV Revision D standard each mobile reports information (buffer occupancy and power headroom) for each service instance (SI) to the Base Station (BS). These reports can be triggered by certain events (such as queue length thresholds) or be performed periodically. The standard also allows for these reports to be done independently for each SI. The BS uses this information to decide on the appropriate rate to assign to the mobile.

Note that the net result is the determination of the appropriate rate by the BS. However, such a determination can be more accurately done at the mobile since it has the most recent measurements. Furthermore, the mobile can combine the requested rates of all SIs and take into account other factors such as power headroom and instead request a total rate for all SIs. We show how each of the QoS attributes (goodput, jitter, delay and error rates) can be converted into a corresponding rate request. These are then used as the request rate for the corresponding SI.

We first present the measurements that must be taken at the mobile for our approach. We then show how these can be used to determine the service rate needed for each application to maintain its QoS guarantee. The mobile’s requested rate then becomes the sum of these per application rates. Next we present and address various practical issues. Finally, illustrative examples are used to demonstrate the effectiveness of the approach.

II. MODEL AND TERMINOLOGY

Assume that the IP packets of each SI are split into multiple RLP (Radio Link Protocol) frames and placed at the tail of its RLP buffer. The RLP frames in this buffer are served in a First In First Out (FIFO) fashion. If the first transmission of a RLP frame fails then it is re-transmitted up to a maximum of two re-transmissions. These re-transmissions are combined with earlier transmissions (soft combining) to increase its probability of success. The transmission power of each transmission is varied so that the residual FER (after all transmissions) is at most 1%. Note that the 1% failures are retransmitted at the RLP layer but with a fresh rounds of physical layer transmissions.

The physical layer frame consists of one or more RLP frames. The number of data bits per physical layer frame depends on the reverse link rate of the mobile. If the first transmission of a physical layer frame is unsuccessful then it is re-transmitted up to a maximum of two re-transmissions. These re-transmissions are combined with earlier transmissions (soft combining) to increase its probability of success. The transmission power of each transmission is varied so that the residual FER (after all transmissions) is at most 1%. Note that the 1% failures are retransmitted at the RLP layer but with a fresh rounds of physical layer transmissions.

In the following description the subscript $i$ is used to identify the SI while the index $n$ denotes the control period.
On a periodic basis (with period $\tau$ which is assumed to be a multiple of a frame duration), the mobile counts the number of information bits, $b_i(n)$, that departed the original transmission buffer for this service instance ($i$) and information bits). The (filtered) information bit departure rate is then estimated by

$$\mu_i(n) = \alpha_\mu \frac{b_i(n)}{\tau} + (1 - \alpha_\mu)\mu_i(n-1)$$

where $\alpha_\mu$ is the corresponding filter constant and $\mu_i(0) = 0$.

Next note that the number of information bits that arrive during this period is given by $b_i(n) + q_i(n) - q_i(n-1)$. Therefore the (filtered) information bit arrival rate is given by

$$\lambda_i(n) = \alpha_\lambda \frac{b_i(n) + q_i(n) - q_i(n-1)}{\tau} + (1 - \alpha_\lambda)\lambda_i(n-1)$$

where $\alpha_\lambda$ is the corresponding filter time constant and $\lambda_i(0) = 0$.

At the beginning of control period $n$ we assume that SI $i$ requests a rate $\tilde{r}(n)$ and is assigned (after some delay) a rate $r_i(n)$ by the BS for the subsequent control period. The difference between $\tau$ and $\tilde{r}$ is due to the finite number of rates supported in the reverse direction.

We assume that once an RLP frame departs the original transmission queue it eventually arrives successfully at the BS. In other words we assume an unlimited number of RLP re-transmissions. Under this assumption, this information bit departure rate, $\mu_i$, equals the goodput of the channel since it is the rate of successfully received frames. If we denote the overhead incurred (due to physical and RLP layer re-transmissions) in transmitting an RLP frame over the air interface by $\rho$ then we have

$$\mu_i = (1 - \rho)r_i$$

Therefore at the end of period $n$ we can estimate $\rho$ by:

$$\rho(n) = \alpha_\rho \left(1 - \sum_i \mu_i(n) \frac{R(n)}{R_i(n)}\right) + (1 - \alpha_\rho)\rho(n-1)$$

(1)

where $\alpha_\rho$ is the filter constant, $\rho(0) = 0$ and $R(n) = \sum_i r_i(n)$ is the reverse link rate. This overhead estimate will be used by all SIs and hence the goodput of SI $i$ during period $n$ is estimated by

$$g_i(n) = (1 - \rho(n))r_i(n)$$

At each decision point (i.e., every $\tau$ seconds) the objective is to determine the service rate, $\tilde{r}(n+1)$, that should be applied in the subsequent interval so that at the end of the interval the expected value of the concerned QoS attribute equals the desired value. Note that this is computed for each SI and hence the request that is sent to the BS is obtained by summing over all SIs.

### III. Rate Determination for Each QoS Attribute

In this section we compute the rate that is required in the subsequent control period in order to maintain the corresponding QoS attribute near its desired value. More precisely, the rate is chosen so that the expected value of the attribute at the end of the subsequent period equals the desired value. Whenever a single SI is being analyzed we drop the subscript $i$ for simplicity.

#### A. Goodput Guarantee

Many applications require some minimum goodput for acceptable performance. In this subsection we determine the rate that should be requested at each control interval in order to maintain the application’s goodput at the desired value. Note that simply providing a constant rate corresponding to the desired goodput may not be sufficient because of errors on the channel. If the channel frame error rate increases then the rate must also be increased in order to maintain the specified goodput.

Denote the desired minimum goodput of the SI by $g$. Note that although this is the minimum desired goodput we will assume that the SI is provided with at most this amount. If more resources than are necessary are provided then the resulting system capacity (number of supported users) will be reduced. Recall that an estimate of the goodput is given by $\mu_i(n)$ and that $\tilde{r}$ is used to denote the desired or requested rate. We can approximate the overhead in the subsequent period with its present value. Since we wish the goodput to be $g$ in the subsequent interval we have

$$g = (1 - \rho(n))\tilde{r}(n+1) \quad \text{hence} \quad \tilde{r}(n+1) = \frac{g}{1 - \rho(n)}.$$  

Note that as the FER increases, $\rho(n)$ (given in 1) increases since more re-transmissions (at both physical and RLP layers) are necessary. This results in an increase in the requested rate to compensate for the errors and maintain the same goodput.

#### B. Jitter Guarantee

In the case of jitter guarantees we assume that the total delay (queuing and transmission) should be maintained at some specified target value $d_T$. Therefore the objective becomes the determination of the rate for the next interval such that the expected value of the delay at the end of the interval equals $d_T$.

The expected queue size at the end of the interval that corresponds to this delay is given by $d_T\mu(n+1)$. The expected queue dynamics during the interval is given by:

$$q(n) + \lambda(n)\tau - \mu(n+1)\tau = d_T\mu(n+1)$$

We estimate $\mu(n+1)$ by

$$\mu(n+1) = (1 - \rho(n))\tilde{r}(n+1)$$
Substituting above and solving for \( \tilde{r}(n+1) \) we get

\[
\tilde{r}(n+1) = \frac{q(n) + \lambda(n) \tau}{(1 - \rho(n))(\tau + d_T)}. \tag{2}
\]

Note that the rate increases with increasing queue length, arrival rate and overhead as expected.

C. Delay Guarantee

In the case of a maximum delay guarantee, the delay should be kept below some specified value \( d_{\text{max}} \). As long as the queuing delay is below this value then it is best to maintain the rate close to the arrival rate so that the queuing delays do not increase. As the delay starts approaching \( d_{\text{max}} \) then the rate should be increased accordingly.

This problem is similar to the case of jitter guarantee. In this case we can assume a target value of \( d_T = 0 \). However, unlike the jitter case, it is not vital to get the delay to this value within a single interval. In fact the targeted time in the future at which we would like the delay to go to zero should depend on how close the present delay is to the maximum value. If it is very close to the maximum then the delay should be brought down quickly (say within one interval). If the delay is already close to zero then it should be brought to zero more slowly.

We assume a target time (i.e., time to bring the delay to zero) of \( d_{\text{max}}/d(n) \) where \( d(n) \) represents the present estimate of the queuing delay. We can write \( d(n) = q(n)/\mu(n) \) and hence this target time is given by \( d_{\text{max}} \mu(n)/q(n) \).

Substituting this value in 2 and using the fact that \( d_T = 0 \) we obtain

\[
\tilde{r}(n+1) = \frac{q(n^2) + \lambda(n) \mu(n) d_{\text{max}}}{(1 - \rho(n)) d_{\text{max}} \mu(n)}. \]

Again note that the requested rate varies with the parameters in an intuitive fashion.

D. FER Guarantee

We assume that the application requires some minimal IP packet loss rate. This can be provided by an appropriate number of RLP re-transmissions if packet transmission delay guarantees are not needed. However, if packet latency is also a concern then FER reduction should be done at the physical layer since less delay is incurred. Since soft recombining is performed at the physical layer, the probability of success of one additional physical layer re-transmission is much higher than the probability of success of the first physical layer transmission in a subsequent RLP round. Furthermore, the additional delay incurred in the physical layer case is much smaller than that in the RLP re-transmission case.

The physical layer FER can be reduced by increasing the maximum number of H-ARQ re-transmissions. However, this also results in increased delay. To maintain the same delay, the power assigned for each frame transmission can be increased (power boosting) so that the increased SNR of the received signal is increased thus reducing the probability of error. This approach is recommended in the 1XEV-DV standard.

IV. SOME PRACTICAL ISSUES

In this section we address some practical issues and show how they can be handled within our proposed framework.

A. Feedback Delay

In the above formulation we assumed zero feedback delay. In practice it takes some time for the mobile to determine the desired rate and send a request to the BS. The BS then has to determine what rate to assign and send a grant message back to the mobile. This takes on the order of 40ms. We propose that the estimates of \( \mu, \lambda, q, \rho \) etc. be computed every \( \tau \) seconds as previously explained. However, knowing that the new rate will take effect some time \( T \) in the future, we predict the queue length \( \hat{q} \) at that point in time in the future by \( \hat{q} = \max\{0, q + (\lambda - \mu)T\} \) where \( \lambda \) and \( \mu \) are the estimates made when the rate request is being computed (i.e., we assume they do not change over the period of time \( T \)). We then use \( \hat{q} \) in the formulas instead of \( q \) and perform the same computations as before.

B. Request Rate to Grant Rate Mapping

Recall that the mobile computes the desired rates \( \tilde{r}_i \) for each SI \( i \) and the total rate is the sum of these. Since the reverse link only supports a finite set of rates (namely 19.2, 40.8, 79.2, 156.0, 309.6, 463.2, 616.8, 924.0, 1231.2, 1538.4, 1845.6 kbps) then the mobile’s desired rate has to be mapped to one of these. The obvious solution is to map to the nearest supportable rate.

Since consecutive supported rates differ by a factor of two, the resulting control will tend to be very oscillatory. We propose dithering rates during a control period as follows. Assume that a control period consists of \( K \) frame periods. Suppose that the desired rate is \( R \) and denote the smallest supported rate higher than \( R \) and the largest supported rate lower than \( R \) by \( R_h \) and \( R_l \) respectively. Therefore there exists \( 0 \leq \beta \leq 1 \) such that \( R = \beta R_l + (1 - \beta) R_h \). This implies that if we could use rate \( R_l \) for a fraction of \( \beta \) of the \( K \) frames and use \( R_h \) for the remaining frames then on average we obtain the desired rate. Let \( f \) denote the nearest integer to \( \beta K \). The mobile sends a request message containing \( R_l \) and \( f \). If the BS grants the request then the mobile does the following (and the BS anticipates this behavior). Any new frame transmission sent during the first \( f \) transmissions of the control period are sent at rate \( R_l \) while all subsequent new frame transmissions are sent at rate \( R_h \).

C. Transmission Power Limit

A mobile is typically limited by one of three factors, the maximum and minimum supportable rates, its maximum
transmission power (200 mW) and reverse link interference. The last limitation results in the BS not assigning the rate requested by the mobile which means that the mobile may not be able to guarantee the QoS it initially agreed upon. In such a case the reverse link is overloaded.

The mobile becomes power limited when it is in relatively poor radio conditions but its applications have high resource demands. It is not able to transmit at its desired rates because of lack of transmission power. Therefore, when making a request rate decision the mobile should first determine if its available power can support the requested rate. If not then it should request the highest rate that can be supported by its available power. This decision is thus local to the mobile and power information need not be reported to the BS.

D. Buffer Depletion and Overflow

We assume that each mobile has a finite sized buffer allocated to each SI. This buffer may not be large enough to support an application with a large delay threshold running at very high reverse link rates. Furthermore, since the rate granted to a mobile is not exactly what was desired then it is possible that during the subsequent period the buffer may either overflow or underflow. However, it is easy to determine this in advance. Once the mobile determines the rate that will be granted by the BS then it can predict the queue length at the end of the subsequent control period. If this is bigger than the buffer size then the requested rate should be adjusted upward while if the predicted queue length is negative then the requested rate should be adjusted downward. The objective being to reduce the occurrence of buffer overflows and underflows.

V. Simulation Results

In this section we provide simulation results to illustrate the effectiveness of the proposed framework. We focus on the jitter case since it requires the most stringent resource requirements. We simulate the relevant aspects of the reverse link as defined for 1XEV-DV and assume that the BS grants the rate requested by the mobile during each interval. If the BS cannot do this then the system is in congestion. We also focus on a single service instance that requires low jitter around a delay of 1s.

Four H-ARQ channels are simulated with a frame duration of 10ms. For each H-ARQ channel we assume a maximum of two frame re-transmissions. We assume that exactly one RLP frame fits in a physical layer frame. If the third transmission of a physical layer frame fails it is queued for another RLP round of physical layer transmissions in the next available H-ARQ channel (called a quick repeat). The physical layer frame transmission success probabilities are 0.9 for the first transmission, 11/90 for the first re-transmission and 1/11 for the second re-transmission. This results in a residual FER of 1%. We assume an infinite number of RLP re-transmissions. This results in an average of 2 physical layer transmissions per frame.

This implies that \((1 - \rho)\), the transmission efficiency, is 0.5. In practice the efficiency does not vary much and so we fix it at this value in our simulations. We will show that the algorithm adapts to any changes in \(\rho\) and hence is insensitive to this parameter. We use \(\rho = 0\) so that the arrival rate is the ratio of the number of arrivals in the previous period divided by the measurement period. We assume that the measurement period and the control period are both 100 ms. Finally we use \(d_T = 0.9\). With these simplifications the algorithm can be stated as:

At the end of each period \(n\) do { 
\[ b(n) = \text{no. frames served in period} \]
\[ q(n) = \text{present queue size} \]
\[ \tilde{r}(n) = 2(b(n) + 2q(n) - q(n - 1)) \]
\[ r(n) = 2^{\log_2(\tilde{r}(n)) + 0.5} \]
}

Instead of sending the requested rate \(r(n)\) to the BS we assume that two bits are used to send the rate change. These are used to indicate (1) increase the rate by two levels, (2) increase the rate by one level, (3) hold the rate constant and (4) decrease the rate by one level. We assume supported rates are \(2^n\) kbps for \(n = 3\) to 11. We assumed a feedback delay (the time between the determination of a new rate and the use of that rate) of 40ms.

We first investigated how the algorithm reacts to sudden changes in the arrival rate. At time 0 we assume a constant arrival rate of 80 kbps. At time 20s this changes to 40 kbps and at time 40s it once again becomes 80 kbps. In Figure 1 we plot the actual delay of each frame as a function of the arrival time of the frame. Note that the algorithm maintains the delay close to its target value as desired. In Figure 2 we plot the logarithm (to the base 2) of the rate at which a frame is served versus the arrival time of the frame. Because of the overhead incurred due to re-transmissions, the physical layer rate must average 160 kbps when the offered load is 80 kbps. This results in the granted rate oscillating between 128kbps and 256kbps during this period as seen in Figure 2.

Next we consider the reaction of the algorithm to changes in the frame error rate. From 20s to 40s after time 0 the probability of error of the second transmission is changed to 66/90 which results in a residual physical layer FER of 6% and an average of 2.42 transmissions per frame. Figure 3 contains a plot of the frame delay versus its arrival time. Note that the algorithms maintains the delay close to its desired rate even with such a drastic change in frame errors. Figure 4 contains the granted rate versus arrival time. It shows that when the frame error rate increases, the percentage of time that the 256 kbps rate is granted also increases to compensate for the reduced transmission efficiency.
VI. CONCLUSIONS

In this paper we presented a simplified framework for supporting QoS services on the reverse link of a CDMA network. For each QoS attribute we show how a corresponding desired rate can be computed for each service instance. The rate that the mobile requests from the BS is then the sum of these rates over all SIs. This results in very low signaling overhead since two bits of signaling per control period are required. Note that simply passing queue size information to the BS will not provide the same level of control. However, one could instead pass to the BS the number of frames that must be served in the subsequent interval (using the field reserved for buffer reports) and have the BS grant the corresponding rate. However, with such a scheme more than two signaling bits per control period will be required.

REFERENCES