

Research Article

Video Broadcasting Using Queue Proportional Scheduling

Dimitris Toumpakaris and Stavros Kotsopoulos

*Wireless Telecommunications Laboratory, Department of Electrical and Computer Engineering,
University of Patras, 26500 Rio, Greece*

Received 31 May 2007; Accepted 12 September 2007

Recommended by Tasos Dagiuklas

Queue Proportional Scheduling (QPS) has been shown to be throughput optimal for Gaussian Broadcast Channels. This paper examines the use of QPS for Video Broadcasting. First, the behavior of QPS is examined as the scheduling frequency is reduced and a method is proposed that uses statistics on the arrival rates to improve its performance. The reduction of the scheduling frequency simplifies the scheduler and decreases the required operations. Then, the packet delay variation is modeled using a Markov Chain approach leading to a method for approximating the packet delay distribution. Based on the resulting distribution, it is discussed how the video encoding rate can be chosen in order to reduce the expected distortion of streams transmitted through Broadcast Channels.

Copyright © 2007 D. Toumpakaris and S. Kotsopoulos. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

Wireless systems have been experiencing constant growth and increased popularity during the past decade. Cellular telephones are now part of most people's everyday life. Fueled by their success and the increased appetite of customers for new and improved services, next-generation cellular systems are targeting broadband applications such as data transfers and video streaming. The aim is to provide mobile users with high rates and seamless roaming [1–3]. When the users are stationary, even higher rates can be offered [4].

One of the services that is expected to gain popularity over the following years is video broadcasting. Digital Video Broadcasting systems will eventually replace analog transmission. In addition to DVB services, video-on-demand download services will be offered to mobile phone or computer network users. Therefore, a base station that is serving a cell will have to broadcast different video streams to the mobile users of the cell. Cellular system downlinks are typical examples of Broadcast Channels (BCs) [5] where a single transmitter sends data to more than one receivers. A well-known information theoretic result is that the attainable rate vectors in a Gaussian BC form a capacity region that can be achieved using superposition coding at the transmitter and successive interference cancellation (SIC) at each receiver. The performance of practical systems often deviates from the optimal

bound. For example, TDMA systems use time division that is sub-optimal, in general, whereas CDMA systems use superposition, but do not use SIC at the receiver where all other users are treated as noise. In this paper it is assumed that the optimal BC performance is achieved by the transceiver architecture. If this is not the case the loss in performance can be taken into account using a nonzero gap value. It is also assumed that both the transmitter and the receiver have perfect Channel State Information (CSI). This is done because the focus of this study is not on how to achieve the Gaussian BC capacity, but on how to manage the available resources in a BC in order to deliver video to the mobile users.

The capacity region of the Broadcast Channel is the union of the rate vectors that can be achieved assuming that the traffic is regular, that is, that during each time period the number of bits transmitted to a user at the physical layer is the same as the number of bits that are sent for transmission by the link layer. However, in practice, the physical layer of a communications system may be receiving data in bursts. For example, if the wireless link is the last hop of a TCP link, the packets may be arriving at irregular intervals at the transmitter due to delays along the data pipe, different routings, packet losses, and so forth. In this case the system may become unstable and the lengths of the queues of some users may not be bounded even if the average rate of each user lies inside the BC capacity region. A significant amount of

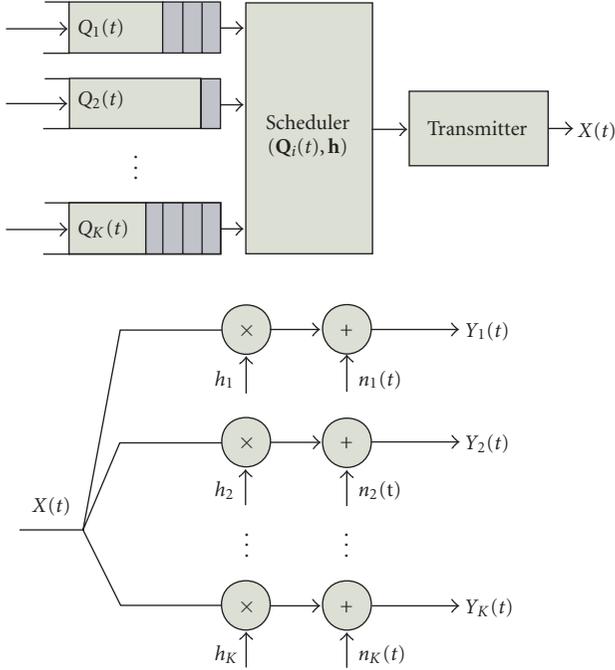


FIGURE 1: System model.

research effort has been devoted to the problem of achieving the capacity region when the incoming traffic is random. Luckily, it turns out that the set of all arrival rate vectors for which it is possible to keep each queue length finite, referred to as the network capacity region, is the same as the BC capacity region. The scheduling policies that achieve the network capacity region are called throughput optimal. Several throughput optimal policies have been proposed for the BC [6–8]. Their common characteristic is that they rely not only on CSI, but also use Queue State Information (QSI). Therefore, they are cross-layer approaches.

Among the throughput optimal cross-layer approaches, Queue Proportional Scheduling (QPS) [7–9] has been shown to have very desirable delay properties. Although its delay optimality for Gaussian Broadcast Channels has not been proved to date, it results in the smallest average packet delay among the known throughput-optimal algorithms, thus making it a good candidate for video transmission where large delays may lead to packet losses, and, consequently, distortion. As will be explained in more detail in Section 2, QPS allocates resources in the BC based on the channel state as well as the queue lengths. In this paper, a simplified version of QPS is proposed that uses Queue State Information less frequently in order to reduce the computational burden. This way the scheduler becomes simpler, since it does not require access to the queue during each scheduling period. It is shown that, under some conditions on the average arrival rate, the modified algorithm is throughput optimal. However, as is expected from the fact that less information is used, it exhibits performance degradation compared to QPS with continuous use of QSI. This is verified using simulation. Then, the packet delay is modeled using Markov

Chains. More specifically, a Markov Chain model is fitted to simulation data and is then used to approximate the probability distribution of the delay of the packets. It is shown that, although the service rate depends on the queue size as well as on the states of the other queues, the approximation is satisfactory. Using information on the expected delay and the corresponding distortion it is possible to choose the video encoder rate in a system employing QPS in order to control the quality of video that is delivered to the users of a BC.

This paper is organized as follows. Section 2 examines the degradation of the performance of QPS as the frequency of using QSI for scheduling decreases and proposes a modification that reduces the performance gap. It is also shown that the modified scheme is throughput optimal under a condition on the average arrival rates. In Section 3 the packet delay is modeled using a Markov Chain model leading to a method that approximates the delay distribution. Section 4 discusses how the distribution of the packet delays can be used to predict the video distortion corresponding to a given encoder rate leading to a discussion on choosing the encoder rate for video streams that are sent to users of a Broadcast Channel. Finally, Section 5 contains concluding remarks.

2. QPS WITH LESS FREQUENT USE OF QUEUE STATE INFORMATION IN GAUSSIAN BROADCAST CHANNELS

Figure 1 depicts the system model that is used in this article. Packets arrive randomly to each queue and are scheduled for transmission. The scheduler allocates the resources of the BC using information on the channel taps h_i (CSI) and the queue states $Q_i(t)$ (QSI). In this article, the scheduler uses Queue State Information only periodically. Moreover, the channel taps are assumed to be constant.

The output signal $X(t)$ is broadcast to the channel, and the signal at each receiver i is equal to

$$Y_i(t) = h_i X(t) + n_i(t), \quad i = 1, \dots, K. \quad (1)$$

This paper assumes a Gaussian BC, that is, the $n_i(t)$ are i.i.d. zero-mean Gaussian random variables with double-sided power spectral density equal to $\mathcal{N}_0/2$. The capacity region of a Gaussian BC in bits/s, assuming, without loss of generality, that $|h_i|^2 \leq |h_j|^2$ for $i < j$, and that the available bandwidth is equal to $2W$ is given by [5]

$$C_{\text{BC}} = \left\{ \mathbf{R}_i : R_i \leq W \log \left(1 + \frac{\alpha_i |h_i|^2 P}{\mathcal{N}_0 W + \sum_{j>i} \alpha_j |h_j|^2 P} \right) \right\}, \quad (2)$$

where P is the average power of $X(t)$ and the $\alpha_i \geq 0$ trace the whole simplex, that is, $\sum_i \alpha_i = 1$. The capacity region is achieved by superposition coding at the transmitter and by successive interference cancellation at each receiver. When the traffic is regular, the transmitter can accommodate any rate vector \mathbf{R} that is inside the capacity region (2). In the following, \mathbf{R} is the number of bits transmitted during the scheduling interval (that is assumed to be equal to 1 for simplicity), so, it is expressed in bits and not in bits/sec.

In this paper it is assumed that traffic arrives irregularly and in packets. The number of packets $A_i(t)$ arriving at queue i during a time period is a Poisson process with rate λ_i , and arrivals at each queue are independent. The packet lengths M_i in bits are assumed to be i.i.d. exponentially distributed with $E[X_i] = \mu_i$ and independent of $A_i(t)$. Therefore, the arrival rate of bits is $\lambda_i \mu_i$. Infinite-capacity queues are considered. At the end of each interval, the scheduler decides on the rate R_i of each queue based on the channel gains \mathbf{h} and the vector of bits (or packets) $\mathbf{Q}(t)$. Other than that, no knowledge of the statistics of the arrival process is required as long as the arrival rate lies inside the capacity region C_{BC} . In this article, $\mathbf{Q}(t)$ may not be used by the scheduler during some scheduling periods as is explained in more detail in this section. The resulting queue state after transmission is

$$Q_i(t + T_s) = (Q_i(t) - R_i(t))^+ + Z_i(t), \quad (3)$$

where $a^+ = \max\{a, 0\}$, and $Z_i(t)$ is the number of bits arriving at queue i during one scheduling period T_s .

Queue Proportional Scheduling (QPS) calculates the rate vector $\mathbf{R}(t) = [R_1(t) \ R_2(t) \ \dots \ R_K(t)]$ according to

$$\begin{aligned} \mathbf{R}(t) &= \max_x \{x\mathbf{Q}(t)\} \\ \text{subject to } &\begin{cases} x\mathbf{Q}(t) \in C_{BC} \\ x \leq 1. \end{cases} \end{aligned} \quad (4)$$

Therefore, $\mathbf{R}(t)$ is a scaled version of $\mathbf{Q}(t)$. If $\mathbf{Q}(t)$ is inside the capacity region, $\mathbf{R}(t) = \mathbf{Q}(t)$, else $\mathbf{R}(t)$ is the intersection of the ray $x\mathbf{Q}(t)$ and the boundary of C_{BC} . In [8] the bit-based QPS is considered, and it is shown that $\mathbf{R}(t)$ is the solution of a Geometric Program and is therefore globally optimal. Note that, since $\mathbf{R}(t) \leq \mathbf{Q}(t)$ when QPS is used, (3) can be rewritten as

$$Q_i(t + T_s) = Q_i(t) - R_i(t) + Z_i(t). \quad (5)$$

In [8] the bandwidth W , the scheduling period T_s and the average packet length for all queues are set to 1. In this article the scheduling period remains equal to 1, but Queue State Information is only used once every L scheduling periods. Naturally, if, during scheduling period t , the information on $\mathbf{Q}(t - L)$ is used, the scheduling will not be done based on the current needs of the user corresponding to each queue. It is expected (and verified by simulation) that this will lead to larger fluctuations of the service rates, and, consequently, larger average queue sizes and packet delays. However, if the scheduler knows the average arrival rate of each queue it can approximate the queue size $\mathbf{Q}(t)$ by $\mathbf{Q}(t - L) + \lambda L - \sum_{l=0}^{L-1} \mathbf{R}(t - l)$. From this point on, λ is the bit arrival rate, that is, the product of the packet arrival rate and the average packet size. This is done for simplicity and for compatibility with the notation used in some of the references. Although not as accurate as the actual $\mathbf{Q}(t)$, this approximation will, on the average, be better than $\mathbf{Q}(t - L)$. As L grows, that is, as use of QSI becomes less frequent, $\mathbf{Q}(t)$ will be close to λ , assuming that $\lambda \in C_{BC}$. Based on the above observations, the following heuristic modification of QPS is proposed in order to reduce the use of QSI for scheduling.

Let the Queue State Information be used once every L times. Also, assume that the modified QPS starts operating at time $t = 0$. Then, the rate vector $\mathbf{R}(t)$ is equal to

$$\begin{aligned} \mathbf{R}(t) &= \max_x \{x\mathbf{Q}(t)\} \\ \text{subject to } &\begin{cases} x\mathbf{Q}(t) \in C_{BC} \\ x \leq 1 \end{cases} \end{aligned} \quad (6)$$

for $(t \bmod L) = 0$, and

$$\begin{aligned} \mathbf{R}(t) &= \text{vec} \left(\min \left\{ \max_x \{ \lambda_i x \}, Q_i(t) \right\} \right) \\ \text{subject to } &\lambda x \in C_{BC} \end{aligned} \quad (7)$$

otherwise, where $\text{vec}(x_i) = [x_1 \ x_2 \ \dots \ x_K]^T$ the vector with elements x_i . Therefore, the optimization is similar to QPS, with the difference that, for times $nL + l$, $l = 1, 2, \dots, L - 1$ the average arrival rate λ is used instead of the actual QSI $\mathbf{Q}(t)$. If the packet arrival rate is constant or changes relatively slowly, that is, if λ can be estimated accurately and does not need to be updated often, $\mathbf{R}(t)$ can be precalculated and stored. Therefore, the computational complexity is reduced roughly by a factor of L . However, a practical system will need to update an estimate of λ , so the reduction in complexity will be less pronounced. Similar to QPS, the rate $\mathbf{R}(t)$ does not exceed $\mathbf{Q}(t)$. This is easily implemented by stopping transmission in a given queue if it becomes empty before the end of a transmission period.

In the following two theorems the throughput optimality of the modified QPS algorithm is established under the condition that the arrival rate λ is constant and satisfies a constraint on its distance from the boundary of the capacity region C_{BC} . The proof is constructed using the same approach as in [9]. First, it is shown that $E[\mathbf{Q}(t)]$ becomes proportional to λ as $t \rightarrow \infty$.

Theorem 1. *Assume that the modified QPS policy is used in a Gaussian BC, and that λ is such that $\alpha\lambda$ is at the boundary of C_{BC} . Then, $\alpha < L/(L - 1)$, and, as $t \rightarrow \infty$, $E[\mathbf{Q}(t)|\mathbf{q}_0] \rightarrow w(t)\lambda$, where \mathbf{q}_0 is any initial state of the queue and $w(t)$ is a function of time.*

Proof. Given in the appendix. \square

Note that, as L increases, the average rate λ should be closer to the boundary of C_{BC} for throughput optimality to be guaranteed by the theorem.

Having proved the convergence of $E[\mathbf{Q}(nL)|\mathbf{q}_0]$ to the direction of λ , throughput optimality is shown along the lines of [9].

Theorem 2. *In a Gaussian BC, the modified QPS policy is throughput optimal, as long as the conditions of Theorem 1 hold.*

Proof. Given in the appendix. \square

For the evaluation of the performance of the modified QPS algorithm, a two-user scenario is chosen, similar to the

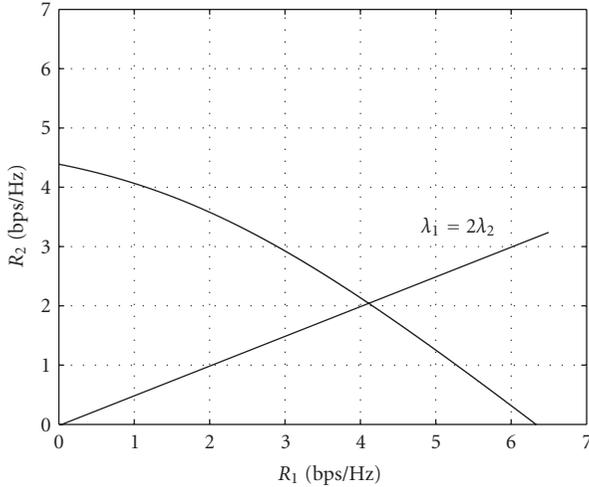


FIGURE 2: C_{BC} for the 2-user scenario.

one in [9]. The SNR of user 1 is equal to 19 dB, whereas the SNR of user 2 is 13 dB. Moreover, $\lambda_1 = 2\lambda_2$. The capacity region of the Gaussian BC for this scenario is shown in Figure 2. What is also shown is the line $\lambda_1 = 2\lambda_2$. During the periods where QSI is not used, the modified QPS chooses a rate vector along the segment formed by the intersection of the line and the capacity region. Therefore, the maximum average bit rate that can be achieved is equal to $\lambda_{\max} = [4.1 \ 2.05]$ bits/s. Figure 3 compares the average packet delay of Queue 1 for different scheduling methods. λ is varied from $[3.7 \ 1.85]$ to $[4 \ 2]$. Due to the nature of QPS, the delays of Queue 2 are similar and their behavior is similar. The dotted lines in Figure 3 depict the degradation of the performance of QPS as the QSI is used less frequently. It is assumed that $\mathbf{Q}(nL)$ is used to compute $\mathbf{R}(nL + l)$, $l = 0, 1, \dots, L - 1$. The dashed lines correspond to the performance of the modified QPS. The modified QPS obtains an estimate of λ by averaging the arrivals during each scheduling period. The performance is evaluated after a sufficient number of iterations of the simulation in order to allow the queues to reach a steady state. Moreover, it is assumed that λ does not change during the simulation. A total of 10^5 scheduling periods proved to be satisfactory for simulation. The queue is allowed to converge during the first 10^4 scheduling periods before delay samples are taken.

As can be seen from the figure, as the scheduling based on QSI becomes less frequent, the average packet delay increases for each queue. The modified QPS bridges the gap in performance, especially as L grows. For relatively small values of L use of the modified QPS reduces the average delay by 2 to 3 times compared to the case where $\mathbf{Q}(t - L)$ is used for all L subsequent schedulings. For very infrequent use of QSI ($L = 100$) the improvement is much more pronounced. Note that, for the case of $L = 100$, throughput optimality is not guaranteed by the proofs in this paper, since it only holds for $\lambda_1 > (99/100)4.1 = 4.06$. However, it appears that the modified QPS does not diverge even for average rates less than 4.06.

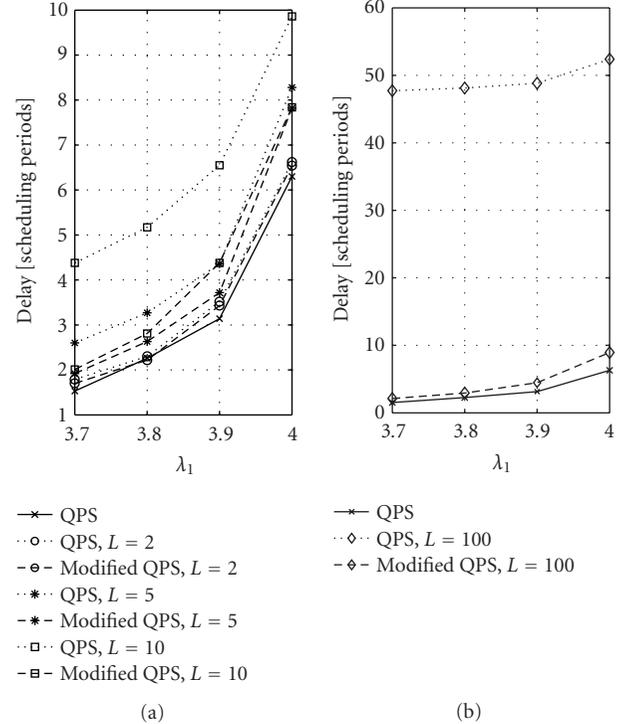


FIGURE 3: Comparison of performance of QPS, QPS with reduced use of QSI and modified QPS.

3. APPROXIMATION OF PACKET DELAY USING MARKOV CHAINS

For video that is transmitted in packet form, what is important is not only the average delay of the packets, but also their delay distribution. More specifically, if a given packet does not arrive within a specific delay window the video decoder may need to decode without using the packet, since either the user cannot tolerate a large delay, or the storage capacity of the receiver buffer will be exceeded. Missing packets result in video quality degradation. Therefore, for the problem of broadcasting examined in this paper, it is useful to be able to obtain the distribution of the packet delay in order to make predictions about the quality of the video stream. A Markov Model is developed in this section whose state denotes the delay of the first (head) packet of the queue in terms of scheduling periods. It is assumed that the only possible transitions are to neighboring states. Again, this is an approximation that is found to work well for QPS.

Clearly, because of queuing, the delays of neighboring packets of a video stream are correlated. During the periods when the queue lengths, and, consequently, the delays become large, it is possible that more than one packet will be delayed. Hence, a model assuming that the delays of neighboring packets of the encoded video stream are independently distributed is not exact unless a sufficient interleaving depth is present. However, in this article it will be assumed that the delays are independent. First, this will provide a lower bound on the video quality that one can expect. Moreover, in order to obtain an accurate estimate of the video

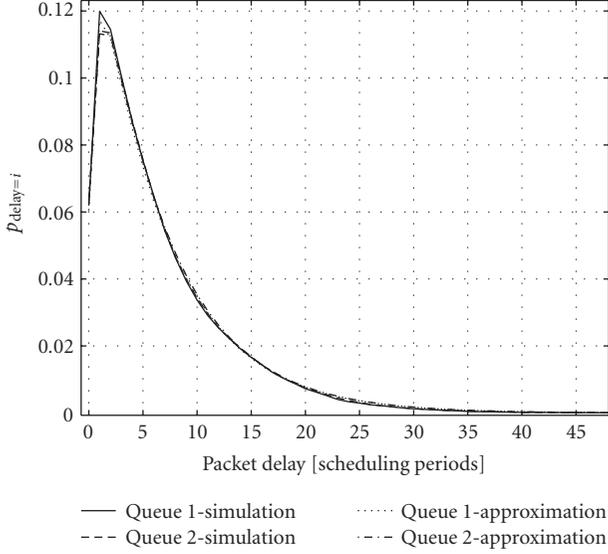


FIGURE 4: Distribution of the packet delay.

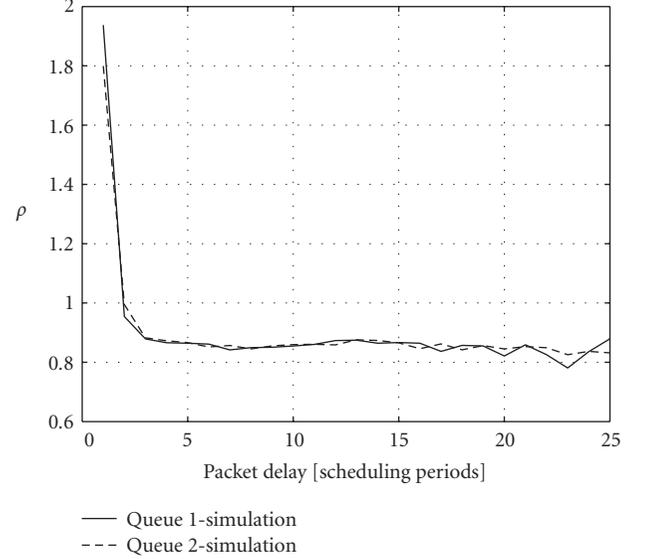
quality, one would need to take into account the particular encoder and decoder that are used, the intra frame ratio, and so forth. One could consider a priority queue where different priorities are given to packets to make sure that enough packets are available for the decoding of a given Group of Blocks (GOB) or frame. Such a scheme will not be accurately described by the Markov Chain presented below, but the approximation may be satisfactory. Another particularity of the system in this article is that the service rate does not depend only on the state of the queue that is being considered, but also on the states of the other queues, since all of them are taken into account for scheduling. However, and despite all the above, simulation results show that the Markov model, albeit simplified, provides a good approximation to the distribution of the delay for a system using a QPS-like scheduler and can therefore be used for the prediction of the video quality.

Figure 4 presents the distribution of the packets delay for the scenario of the previous section, arrival rate $\lambda = [4 \ 2]$ and average packet size $\mu = 1$ for both queues. Again, SNR = 19 and 13 dB, respectively. Scheduling uses QSI during all periods. The reason why the probability distribution has a peak at 1 and not 0 is because the scheduler operates only at the end of a period, so packets that have arrived during an interval may have to wait till the end of that interval in order to be able to leave the queue. This skews the peak of the distribution that would otherwise be at 0. In terms of the Markov Chain the service rate is not constant and depends on the queue state.

From Markov Queue theory, and assuming that the service rate depends on the delay of the first packet of the queue,

$$p_{i+1}\mu_{i+1} = \lambda p_i \Rightarrow \rho_{i+1} = \frac{\lambda}{\mu_{i+1}} = \frac{p_{i+1}}{p_i} \quad i = 0, 1, \dots \quad (8)$$

The values of ρ_i are obtained using the p_i 's that result from the simulation, and are plotted in Figure 5 for both queues. 2

FIGURE 5: Values of ρ obtained by simulation.

million scheduling periods are used for the simulation. Note that they converge as i increases. This is expected since, as the queue grows, the scheduled packet will need to wait for a time longer than a scheduling period in order to leave the system. Hence, in this case, the fact that the scheduling happens in specific instants does not influence the service rate. The oscillations as the delay increases are due to the inaccuracy of the p_i 's due to the fewer number of samples for the less probable states.

Based on Figure 5, the following approximation is used:

$$\begin{aligned} \hat{\rho}_i &= \rho_i & \text{for } 1 \leq i \leq K-1 \\ \hat{\rho}_i &= \rho_K & \text{for } i \geq K. \end{aligned} \quad (9)$$

Then, the delay distribution probabilities are calculated as follows

$$\begin{aligned} \hat{p}_{i+1} &= \hat{p}_{i+1} \hat{\rho}_i, \quad i = 0, 1, \dots, \\ \hat{p}_0 &= 1 - \sum_{i=1}^{\infty} \hat{p}_i = 1 - \sum_{i=1}^{\infty} \left(\prod_{j=1}^i \hat{\rho}_j \right) \hat{p}_0 \\ &= 1 - \sum_{i=1}^{K-2} \left(\prod_{j=1}^i \hat{\rho}_j \right) \hat{p}_0 - \sum_{i=K-1}^{\infty} \left(\prod_{j=1}^i \hat{\rho}_j \right) \hat{p}_0 \\ &= 1 - \sum_{i=1}^{K-2} \left(\prod_{j=1}^i \hat{\rho}_j \right) \hat{p}_0 - \prod_{j=1}^{K-1} \hat{\rho}_j \sum_{m=0}^{\infty} \hat{\rho}_K^m \hat{p}_0 \\ &= 1 - \sum_{i=1}^{K-2} \left(\prod_{j=1}^i \hat{\rho}_j \right) \hat{p}_0 - \frac{\prod_{j=1}^{K-1} \hat{\rho}_j}{1 - \hat{\rho}_K} \hat{p}_0 \Rightarrow \\ \hat{p}_0 &= \frac{1}{1 + \sum_{i=1}^{K-2} \left(\prod_{j=1}^i \hat{\rho}_j \right) + \prod_{j=1}^{K-1} \hat{\rho}_j / (1 - \hat{\rho}_K)}. \end{aligned} \quad (10)$$

The P_i 's are approximated using $K = 6$ and $\hat{\rho}_K = 0.86$ for both queues. The resulting approximation of the packet delay distribution is shown in Figure 4. As can be seen, although the queues for the scenario considered in this paper

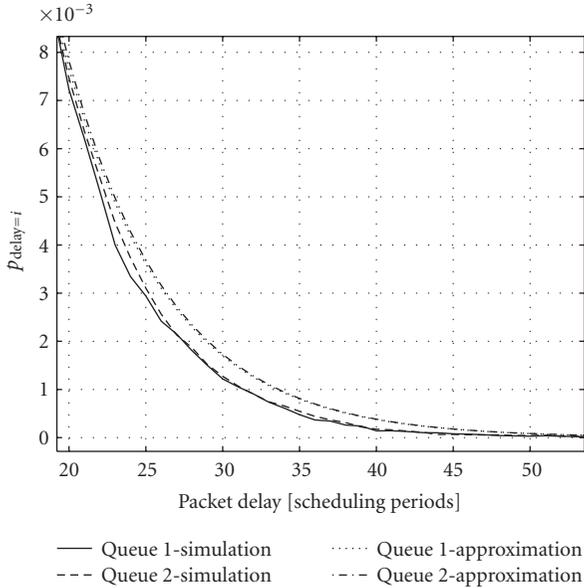


FIGURE 6: Distribution of the packet delay (detail).

are not independent, the approximation is good, and can be used to predict the delay of the packets scheduled by QPS provided that the arriving traffic rates are known. In Figure 6 more detail is shown for the states corresponding to higher delays. The approximation is slightly pessimistic but still very close to the actual values. In the following section, the distribution of the packet delay is used in order to decide on the encoding rate of video streams that are broadcast to different users and are scheduled using QPS.

4. CHOOSING THE ENCODER RATE FOR VIDEO STREAMS SCHEDULED BY QPS

Video streams transmitted in packets are subject to distortion. Distortion results from several sources such as encoder compression, corrupted data and lost or delayed packets. In [10] the authors develop models and derive expressions for the overall distortion of a video stream $D_d = D_e + D_v$. The first term D_e is the distortion because of signal compression at the encoder. It depends on the INTRA frame rate β and the rate R_e at the output of the decoder. R_e may need to be lowered in order to allow for a more redundant channel code, and, therefore, better protection against noise in the channel. D_v is the distortion occurring at the decoder and is related to the lost or corrupted packets that cannot be used by the decoder to reconstruct the transmitted video stream.

In [10], the channel capacity was assumed to be fixed and the reason for varying R_e was in order to leave more (or less) room to the channel code. The stronger the channel code is the smaller the probability of erroneous packets will be, leading to reduced decoder distortion. Therefore, the choice of R_e (and the associated channel code rate) leads to a tradeoff between D_e and D_v . By choosing R_e , the channel code and the

INTRA rate β appropriately, the smallest value of the overall distortion D_d can be found.

In the scenario examined in this paper, a new tradeoff is created between D_e and D_v . In a BC where many users compete for the resources, a larger channel rate also means larger average (and maximum) delays. Hence, allowing the video encoder to send with a faster rate also increases the probability that a packet will not arrive early enough for the decoder to be able to use it. The number of packets with delays that exceed a given threshold adds to the number of packets that are corrupted in the channel and, therefore, the overall number of unusable packets increases. Consequently, this leads to larger distortion.

As explained in [10] the exact value of the distortion depends on many factors such as the particular stream that is being transmitted, the video encoder and the spatial filters of the decoder, all of which are outside the scope of this paper. Therefore, in this article, it is briefly suggested how the effect of the channel delay can be added to the calculation of D_v . Then, the system optimization can proceed along the lines of [10]. From [10], $D_v = \sigma_{u0}^2 P_L \sum_{t=0}^{T-1} ((1-\beta)t)/(1-\gamma t)$, where β is the INTRA rate, γ is the leakage parameter that is determined by the loop filter of the decoder, $T = 1/\beta$ is the INTRA update interval, σ_{u0}^2 describes the sensitivity of the video decoder to an increase in the error rate and P_L is the residual packet error rate. When QPS is used, the proportion $(1 - P_L)$ of packets that are not corrupted in the channel or are lost for other reasons, are subject to delays at the queue of the scheduler. The probability of the delay of a packet exceeding a given threshold T_{del} can be found by forming $P_{late} = \sum_{d=D}^{\infty} p_d = 1 - \sum_{d=0}^{D-1} p_d$, where D is the first value of the delay that exceeds T_{del} . Alternatively, the approximate probabilities \hat{p}_d that were derived using the Markov Chain model can be used. Hence, the new P'_L that should be used for the calculation of D_v is equal to $P'_L = P_L + (1 - P_L)P_{late}$.

5. CONCLUSION

In this paper, Queue Proportional Scheduling was considered with video transmission in mind. First, it was shown that, if an increase in the average packet delay can be tolerated, the use of Queue State Information can become less frequent, therefore simplifying the scheduler. A modified QPS scheduler was proposed that performs better than the approach of simply using outdated QSI for scheduling. The modified scheduler performs better than the simplistic approach without increasing considerably the implementation complexity. Moreover, it was proved that, under certain conditions, the modified QPS is throughput optimal. It was also shown using simulation that, in systems using QPS, the distribution of the packet delay can be approximated satisfactorily by a Markov Chain model. This model makes it easier to obtain an estimate for the tail of the probability distribution, and, consequently calculate the video distortion caused by packets whose delay exceeds the buffer size of the decoder. It was also discussed how the effect of late packet arrivals can be included in the calculation of the distortion.

APPENDIX

Proof of Theorem 1

Let $\mathbf{Q}(t) = \mathbf{q}_t = [q_{t,1} \ q_{t,2} \ \cdots \ q_{t,K}]^T$, where K is the number of users of the BC. Assume, without loss of generality, that $q_{t,1} \neq 0$ and $\lambda_1 \neq 0$. Then, \mathbf{q}_t can be written as $\mathbf{q}_t = w(t)[\lambda_1, \lambda_2 + \Delta\lambda_2, \dots, \lambda_K + \Delta\lambda_K]^T$, where $w(t) = q_{t,1}/\lambda_1$ and $\Delta\lambda_i$ are such that $w(t)(\lambda_i + \Delta\lambda_i) = q_{t,i}$ for $i = 2, \dots, K$. Therefore,

$$E[\mathbf{Q}(t+1) | \mathbf{Q}(t) = \mathbf{q}_t] = \mathbf{q}_t + \boldsymbol{\lambda} - \mathbf{R}(t). \quad (\text{A.1})$$

When the queue state is used for scheduling, $\mathbf{R}(t) = r(t)(\mathbf{q}_t/w(t))$ where $r(t)$ is such that $\mathbf{R}(t)$ is inside the capacity region, and does not exceed \mathbf{q}_t . Hence, $r(t) \leq w(t)$. As is shown in [9],

$$\begin{aligned} E[\mathbf{Q}(t+1) | \mathbf{Q}(t) = \mathbf{q}_t] &= (w(t) - r(t) + 1) \\ &\times [\lambda_1, \lambda_2 + \gamma(t)\Delta\lambda_2, \dots, \lambda_K + \gamma(t)\Delta\lambda_K]^T, \end{aligned} \quad (\text{A.2})$$

where $\gamma(t) = 1 - 1/(w(t) - r(t) + 1)$. If $\mathbf{q}_t \in C_{\text{BC}}$ then $w(t) = r(t)$, $\gamma(t) = 0$ and $E[\mathbf{Q}(t+1) | \mathbf{Q}(t) = \mathbf{q}_t] = \boldsymbol{\lambda}$. Else, $\gamma(t)$ is strictly less than 1.

Consider now the case of the modified QPS algorithm, and assume that the queue state information gets used once every $L = 2$ transmission periods, that is, for $t, t+2, \dots, t+2n$, information on $\mathbf{Q}(t)$ is used, whereas during the other periods scheduling is based on $\boldsymbol{\lambda}$

$$\begin{aligned} E[\mathbf{Q}(t+2) | \mathbf{Q}(t) = \mathbf{q}_t] &= \mathbf{q}_t + 2\boldsymbol{\lambda} - \mathbf{R}(t) - \mathbf{R}(t+1) \\ &= \mathbf{q}_t + 2\boldsymbol{\lambda} - \frac{r(t)}{w(t)}\mathbf{q}_t - \alpha(t)\boldsymbol{\lambda} \\ &= (w(t) + 2 - r(t) - \alpha(t)) \\ &\times [\lambda_1, \lambda_2 + \gamma(t)\Delta\lambda_2, \dots, \lambda_K + \gamma(t)\Delta\lambda_K]^T, \end{aligned} \quad (\text{A.3})$$

where $\gamma(t) = 1 - (2 - \alpha(t))/(w(t) - r(t) + 2 - \alpha(t))$. If $\mathbf{q}_t \in C_{\text{BC}}$, $w(t) = r(t)$ and $\gamma(t) = 0$. Assuming that $2 - \alpha(t)$ can never become negative, that is, that the average arrival rate is large enough so that it is more than halfway between zero and the boundary of the capacity region, $\gamma(t) < 1$ in all other cases. Therefore, similar to the QPS proof of [9], it can be deduced that the slope of $\mathbf{Q}(t)$ converges to the slope of $\boldsymbol{\lambda}$ in the sense that

$$\theta_\lambda(\mathbf{q}_t) \geq \theta_\lambda(E[\mathbf{Q}(t+2) | \mathbf{Q}(t) = \mathbf{q}_t]), \quad (\text{A.4})$$

where $\theta_\lambda(\mathbf{x}) = \cos^{-1}(\boldsymbol{\lambda}^T \mathbf{x} / \|\boldsymbol{\lambda}\|_2 \|\mathbf{x}\|_2)$, $0 \leq \theta_\lambda(\mathbf{x}) \leq \pi/2$.

For the general case where queue State Information is used every L scheduling periods,

$$\begin{aligned} E[\mathbf{Q}(t+L) | \mathbf{Q}(t) = \mathbf{q}_t] &= \mathbf{q}_t + L\boldsymbol{\lambda} - \mathbf{R}(t) - \sum_{l=1}^{L-1} \mathbf{R}(t+l) \\ &= \mathbf{q}_t + L\boldsymbol{\lambda} - \frac{r(t)}{w(t)}\mathbf{q}_t - \sum_{l=1}^{L-1} \alpha_l(t)\boldsymbol{\lambda} \\ &= \left(w(t) + L - r(t) - \sum_{l=1}^{L-1} \alpha_l(t) \right) \\ &\times [\lambda_1, \lambda_2 + \gamma(t)\Delta\lambda_2, \dots, \lambda_K + \gamma(t)\Delta\lambda_K], \end{aligned} \quad (\text{A.5})$$

where $\gamma(t) = 1 - (L - \sum_{l=1}^{L-1} \alpha_l(t))/(w(t) - r(t) + L - \sum_{l=1}^{L-1} \alpha_l(t))$. Again, $\gamma(t) = 0$ when $\mathbf{R}(t) = \mathbf{q}_t$. Else, $\gamma(t) < 1$ as long as $L - \sum_{l=1}^{L-1} \alpha_l(t) > 0$. This can be guaranteed if, for each l , $\alpha_l(t) < L/(L-1)$ which means that if the boundary of C_{BC} is at $\alpha\boldsymbol{\lambda}$, then $\alpha < L/(L-1)$.

Since the packet arrivals are Poisson, the queue state of each user is a first order Markov process. From the Chapman-Kolmogorov Equations [11]

$$\begin{aligned} E[\mathbf{Q}(t+L) | \mathbf{Q}(0) = \mathbf{q}_0] &= E[E[\mathbf{Q}(t+L) | \mathbf{Q}(t)] | \mathbf{Q}(0) = \mathbf{q}_0] \quad \text{for } t = 0, L, \dots \\ & \quad (\text{A.6}) \end{aligned}$$

From (A.4), $\theta_\lambda(E[\mathbf{Q}(t) | \mathbf{Q}(0) = \mathbf{q}_0]) \geq \theta_\lambda(E[E[\mathbf{Q}(t+L) | \mathbf{Q}(t)] | \mathbf{Q}(0) = \mathbf{q}_0]) \stackrel{(6)}{=} \theta_\lambda(E[\mathbf{Q}(t+L) | \mathbf{Q}(0) = \mathbf{q}_0])$ for $t = 0, L, \dots$. Thus, if $\theta_n \triangleq \theta_\lambda(E[\mathbf{Q}(nL) | \mathbf{Q}(0) = \mathbf{q}_0])$, $n = 1, 2, \dots$, and $\theta_0 \geq 0$, θ_n converges. It can be easily deduced (as, e.g., in [9]), that $\theta_n \rightarrow 0$ as $n \rightarrow \infty$, and, therefore, $E[\mathbf{Q}(nL) | \mathbf{q}_0] \rightarrow w(t)\boldsymbol{\lambda}$.

Proof of Theorem 2

It will be shown that, for any $\boldsymbol{\lambda} \in C_{\text{BC}}$ such that $\alpha\boldsymbol{\lambda}$ is at the boundary of C_{BC} and $1 \leq \alpha \leq L/(L-1)$, the queue lengths of all users can be kept finite. The following Lyapunov function is chosen: $L(\mathbf{Q}(t)) = \sum_{i=1}^K Q_i(t)$. Then, assuming that t is a scheduling period when the Queue State Information is used, $L(\mathbf{Q}(t+1)) = \sum_{i=1}^K Q_i(t+1) = \sum_{i=1}^K \{(Q_i(t) - R_i(t))^+ + Z_i(t)\}$. Since $R(t) \leq Q(t)$, $L(\mathbf{Q}(t+1)) = \sum_{i=1}^K (Q_i(t) - R_i(t) + Z_i(t))$. $L(\mathbf{Q}(t+2)) = \sum_{i=1}^K Q_i(t+2) = \sum_{i=1}^K [(Q_i(t) - R_i(t) + Z_i(t) - R_i(t+1))^+ + Z_i(t+1)]$. Again, since the queue lengths cannot become negative, $L(\mathbf{Q}(t+2)) = \sum_{i=1}^K Q_i(t+2) = \sum_{i=1}^K (Q_i(t) - R_i(t) + Z_i(t) - R_i(t+1) + Z_i(t+1))$. Hence, $L(\mathbf{Q}(t+L)) = \sum_{i=1}^K \{Q_i(t) - \sum_{l=0}^{L-1} R_i(t+l) + \sum_{l=0}^{L-1} Z_i(t+l)\}$. Assume that $\mathbf{Q}(0) = \mathbf{q}_0$, where $\max\{q_{i,0}\}$ is sufficiently small. Then, the expected drift of the Lyapunov function conditioned on \mathbf{q}_t is equal to

$$\begin{aligned} E[L(\mathbf{Q}(t+L)) - L(\mathbf{Q}(t)) | \mathbf{Q}(t) = \mathbf{q}_t] &= \sum_{k=1}^K \left\{ L\lambda_k - (R_k(t) | \mathbf{Q}(t) = \mathbf{q}_t) \right. \\ &\quad \left. - \sum_{l=1}^{L-1} (R_k(t+l) | \mathbf{Q}(t) = \mathbf{q}_t) \right\}. \end{aligned} \quad (\text{A.7})$$

It will be shown that, as $\|\mathbf{q}_t\|_\infty = \max\{q_{i,t}\} \rightarrow \infty$, the Lyapunov drift (A.7) becomes strictly negative. $\|\mathbf{q}_t\|_\infty \rightarrow \infty$ also implies that $t \rightarrow \infty$ since a queue cannot grow to infinity during a finite time interval. As was shown in Theorem 1, if a condition on λ holds, $\mathbf{Q}(nL)$ converges to $w(t)\lambda$ as $t \rightarrow \infty$. Hence, at time $t \rightarrow \infty$, QPS will use the value of $\mathbf{Q}(t)$ and the rate $\mathbf{R}(t)$ will be equal to $r(t)\mathbf{Q}(t) \rightarrow r(t)w(t)\lambda = W(t)\lambda$. Regarding the rates $\mathbf{R}(t+l)$, $1 \leq l \leq L-1$, these are, by definition of the modified QPS, equal to $\alpha(t+l)\lambda$. Therefore, for $t \rightarrow \infty$,

$$E[L(\mathbf{Q}(t+L)) - L(\mathbf{Q}(t)) \mid \mathbf{Q}(t) = \mathbf{q}_t] \rightarrow \sum_{k=1}^K \left\{ L\lambda_k - W(t)\lambda_k - \sum_{l=1}^{L-1} \alpha(t+l)\lambda_k \right\}. \quad (\text{A.8})$$

Since λ is in the interior of C_{BC} , $W(t)$ and the $\alpha(t+l)$ will be strictly larger than 1 because the modified QPS algorithm chooses the longest vector along the direction of λ that belongs to the BC Capacity region. When $\|\mathbf{q}_t\|_\infty \rightarrow \infty$ this vector reaches the boundary of the capacity region, and is, therefore, longer than λ . Hence, the Lyapunov drift is strictly negative for any λ satisfying the condition that $\alpha\lambda \in C_{\text{BC}}$, $1 < \alpha \leq L/(L-1)$.

REFERENCES

- [1] IEEE Std 802.16-2004, "IEEE standard for local and metropolitan area networks, Part 16: air interface for fixed broadband wireless access systems," October 2004.
- [2] IEEE Std 802.16e-2005, "IEEE standard for local and metropolitan area networks, Part 16: air interface for fixed broadband wireless access systems, amendment 2: physical and medium access control layers for combined fixed and mobile operation in licensed bands," February 2006.
- [3] Long Term Evolution of the 3GPP radio technology, <http://www.3gpp.org/Highlights/LTE/LTE.htm>.
- [4] IEEE Std 802.11n TGn draft 2.0, "Amendment to standard for information technology-telecommunications and information exchange between systems-local and metropolitan networks- specific requirements- Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: enhancements for higher throughput".
- [5] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley-Interscience, New York, NY, USA, 1991.
- [6] N. McKeown, A. Mekkitikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE Transactions on Communications*, vol. 47, no. 8, pp. 1260–1267, 1999.
- [7] A. Eryilmaz, R. Srikant, and J. R. Perkins, "Throughput-optimal scheduling for broadcast channels," in *Modeling and Design of Wireless Networks*, vol. 4531 of *Proceedings of SPIE*, pp. 70–78, Denver, Colo, USA, August 2001.
- [8] K. Seong, R. Narasimhan, and J. M. Cioffi, "Queue proportional scheduling via geometric programming in fading broadcast channels," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1593–1602, 2006.
- [9] K. Seong, R. Narasimhan, and J. M. Cioffi, "Queue proportional scheduling in Gaussian broadcast channels," in *Proceedings of the IEEE International Conference on Communications (ICC '06)*, pp. 1647–1652, Istanbul, Turkey, June 2006.
- [10] K. Stuhlmler, N. Färber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1012–1032, 2000.
- [11] S. Ross, *Stochastic Processes*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1996.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

