A scheduler for a wireless system

Problem presented by

José Gil

Motorola

Problem statement

Several users, with different traffic types, share a wireless channel, and a scheduler is used to assign the order in which they are served. The overall capacity of the channel is limited by the total transmission power that is available. One possibility is for the scheduler to order the traffic by increasing values of W_j/C_j where W_j is the normalized throughput for traffic of type j and C_j is a 'credit function' depending on the traffic type j and the corresponding channel quality. The Study Group was asked to consider the performance of this scheduler and how the queuing time and packet loss depend on the capacity of the system and the parameters that occur in defining throughput, credit and channel quality. An explicit version of the model was constructed and analysed, from which it can be seen how to choose the parameters so that the scheduler has the desired behaviour.

Study Group contributors

David Allwright (Smith Institute)
Tristram Armour (Smith Institute)
Vera Hazelwood (Smith Institute)
Robert Leese (Smith Institute)
Steve Noble (Brunel University)
Ida Pu (Goldsmiths College London)

Report prepared by

David Allwright (Smith Institute)
Tristram Armour (Smith Institute)

1 Introduction

This problem deals with the scheduling of data packets for different users who share a common wireless channel of limited capacity and who have different service time requirements. From a system point of view it is important to maintain a high throughput, that is, a high number of packets served per unit of time. However from a user point of view it is important to minimise the service time. This poses a trade-off problem, which is one of the goals we wish to analyse.

The wireless transmission channel alternates between good and bad states over time. While the channel is in a bad state the transmission of a packet fails and the packet needs to be retransmitted. When the state of the channel is good, packets are transmitted successfully and do not require retransmission. The system transmission efficiency or throughput might be defined as the number of data packets that can be transmitted successfully in a given time.

The scheduler we wish to study is designed to consider both the wireless channel conditions and the user's quality of service requirements. For this purpose users are assigned a credit every scheduling frame, which is a function of the wireless channel conditions and the quality of service required, which depends on the traffic class.

The original problem statement given by Motorola (see Appendix A) was further refined by discussion at the Study Group to yield the following agreed model, which is to be considered as operating in successive time frames indexed by $t = 1, 2, 3, \ldots$. The traffic classes are indexed by j (running from 1 to J) and they are also referred to as users, and we think of each user's data packets as being sent over a different wireless channel. The channel quality experienced by user j during time frame t is denoted by $R_j(t)$ and we shall think of this as the packet rate per unit power for user j. We consider the credit function for user j

$$C_j(t) = u_j \left(R_j^{\text{avg}}(t)\right)^{\alpha} \left(\frac{R_j(t)}{R_j^{\text{avg}}(t)}\right)^{\beta}, \tag{1}$$

where u_j is a traffic class weight, α and β are channel condition weights and $R_j^{\text{avg}}(t)$ is the average effective channel quality computed by

$$R_j^{\text{avg}}(t) = (1 - \psi)R_j^{\text{avg}}(t - 1) + \psi R_j(t) \tag{2}$$

for some $0 < \psi < 1$. (For a discussion of the motivation for this see Section 3.) To define our model for the operation of the buffer we introduce notation illustrated in Figure 1:

$$\begin{aligned}
X_j(t) &= \text{number of data packets for user } j \text{ in the buffer} \\
&\qquad \text{at the start of time frame } t, && (3)\\
A_j(t) &= \text{number of data packets offered by user } j \text{ to the buffer} \\
&\qquad \text{during time frame } t, && (4)\\
f_j(t) &= \text{fraction of } A_j(t) \text{ that is admitted to the buffer in time frame } t, && (5)\\
D_j(t) &= \text{number of data packets for user } j \text{ transmitted during time frame } t. && (6)
\end{aligned}$$

The arrivals process is that packets arrive in batches according to a Poisson process with arrival rate $c_j\lambda$. The size of a batch is a random variable with range $\{0, \ldots, K\}$, and we

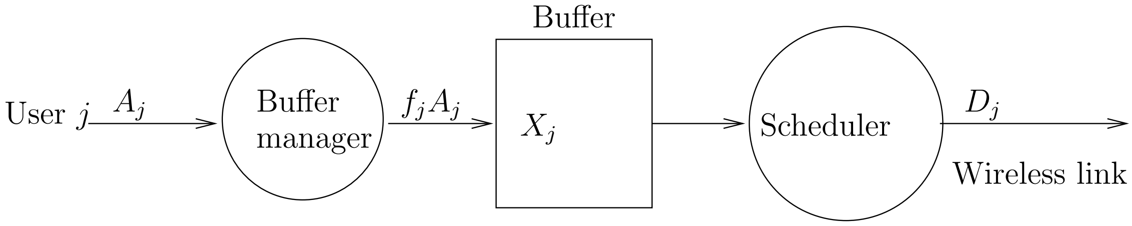

Figure 1: Schematic diagram of user, buffer, scheduler and wireless link. In a given time frame, user j offers A_j data packets to the buffer manager, of which a fraction f_j are admitted. Together with any packets already in the buffer, user j then has X_j packets in the buffer of which D_j are transmitted in the time frame, as determined by the scheduler.

can consider batch sizes to be independent. Thus the A_j are independent, with a certain distribution that depends on j. The buffer operates as a separate queue for each user, of capacity k packets, so in fact

$$f_jA_j = \text{number of packets admitted} = \min(A_j, k - X_j). \tag{7}$$

The update equation for the buffer content size then is

$$X_j(t+1) = X_j(t) - D_j(t) + f_j(t)A_j(t). \tag{8}$$

The averaged or smoothed throughput for user j is defined to be

$$W_j(t) = (1-\phi)W_j(t-1) + \phi D_j(t) \tag{9}$$

for some $0 < \phi < 1$.

At time t the scheduler orders users in increasing value of the sorting metric $S_j = W_j(t-1)/C_j(t-1)$, which determines the order in which user packets are transmitted. In detail, suppose

$$S_{i_1} < S_{i_2} < \ldots \tag{10}$$

so that user i_1 is first in the queue, user i_2 second *etc.* Then if a total power P is available for the transmissions during the time frame, user i_1 transmits using power

$$P_{i_1} = \min\left(\frac{X_{i_1}(t)}{R_{i_1}(t)}, P\right), \tag{11}$$

so he either uses all the available power, or just enough to transmit his data, whichever is the least. The data transmitted by user i_1 therefore is

$$D_{i_1}(t) = R_{i_1}(t)P_{i_1} = \min(X_{i_1}(t), R_{i_1}(t)P). \tag{12}$$

The remaining power $P'_{i_1} = P - P_{i_1}$ is offered to user i_2, and he uses

$$P_{i_2} = \min\left(\frac{X_{i_2}(t)}{R_{i_2}(t)}, P'_{i_1}\right), \tag{13}$$

and transmits

$$D_{i_2} = \min(X_{i_2}, R_{i_2} P'_{i_1}), \tag{14}$$

and so on.

We note various points at which this description and notation clarifies, generalises or supersedes the first statement of the problem by Motorola (as shown in Appendix A):

(1) our credit function (1) generalizes the original one by the inclusion of the factor involving β, and we use u_j rather than w_j to denote the traffic class weights;

(2) we use $1-\psi$ rather than ϕ to define R_j^{avg};

(3) our (9) swops $\phi \leftrightarrow 1-\phi$ compared to the original;

(4) our (9) has just D_j as the driving term rather than D_j/C_j.

Because of the second and third points here, the limit Motorola originally asked about, where ϕ and ψ are close to 1, becomes here the limit where ϕ and ψ are small.

We are interested in the analysis of the following performance measures:

(1) the distribution of system size (queue size including packets in service);

(2) the queueing time distribution for each traffic class;

(3) the packet loss distribution for each traffic class.

We are interested in general solutions that allow the study of these performance measures as a function of:

- the traffic class weights u_j (maximum of four traffic classes);
- the channel condition powers α and β;
- the buffer size k;
- the weights ψ and ϕ (as a starting point it is recommended to start the study with these weights equal to 0).

2 Analysis of simple cases

The process of allocating the available power to the users by the scheduler can have broadly speaking two kinds of outcome:

(1) If $\sum_j (X_j/R_j) \leq P$ then there is enough capacity available to transmit all the data for all the users, so every $D_j = X_j$.

(2) If $\sum_j (X_j/R_j) > P$ then not all the data can be transmitted. In this case there will be some index r such that

- the first $r-1$ users in the queue, $j = i_s$ for $s = 1, 2, \ldots, r-1$, have *all* their data transmitted from the buffer, $D_j = X_j$;
- user r in the queue, $j = i_r$, has only *some* of his data transmitted, $D_j < X_j$;
- users later in the queue, $j = i_s$ for $s > r$, have *no* data transmitted, $D_j = 0$.

So in this case $\sum_j (D_j/R_j) = P$, and these are the situations where the scheduler is having significant effects on the system. It may be that $r = 1$ in which case only the first user in the queue (user i_1) gets any service during the time frame.

2.1 Simple case with two users

We consider first the situation with just two users, and where the channel qualities R_j are constant in time, but R_1 may differ from R_2. So the credits C_j are constant, and the important variables are the data offered A_j and transmitted D_j. We shall also assume that the buffer is effectively infinite, in other words that we are in a power-constrained case rather than buffer-constrained. Suppose the average data offered in a single time frame by user j is $\overline{A}_j$, so that if $\sum_j \overline{A}_j/R_j \le P$ then there is enough capacity to serve both users, but if this inequality fails then there will be a build-up in the buffer of one or both users. Suppose that *both* users' buffers fill up and that, in the long-term steady state, user j wins the queue (*i.e.* comes first in the queue) a proportion p_j of the time. If one user, j, repeatedly received no service, then W_j would drop steadily, and so S_j would drop steadily, and so eventually user j would win the queue once more. So the p_j will be strictly positive. When user j wins the queue, he obtains transmission of $R_j P$ of the data from his buffer. Thus his long term average throughput is $\overline{W}_j = p_j R_j P$.

If ϕ is small, then the W_j do not vary much with time, and so since both users get a share of the service, their long term average values of S_j must be approximately equal, *i.e.* $\overline{W}_1/C_1 = \overline{W}_2/C_2$. This gives $p_j \propto C_j/R_j$, and so

$$p_1 = \frac{C_1/R_1}{C_1/R_1 + C_2/R_2}, \qquad p_2 = \frac{C_2/R_2}{C_1/R_1 + C_2/R_2}. \tag{15}$$

In a case where $C_j = u_j R_j^\alpha$, this gives $p_j \propto u_j R_j^{\alpha-1}$. So if $\alpha > 1$ and $u_1 = u_2$, the user with the better channel quality is scheduled more of the capacity. These values of p_j give mean throughputs for the users of

$$\overline{W}_1 = \frac{PC_1}{C_1/R_1 + C_2/R_2}, \qquad \overline{W}_2 = \frac{PC_2}{C_1/R_1 + C_2/R_2}. \tag{16}$$

If the actual data rate of user 1, $\overline{A}_1$, is below the limit $\overline{W}_1$ then his buffer will in fact not build up, and he will win the queue almost all the time. Conversely if $\overline{A}_2 < \overline{W}_2$ then user 2 will win the queue all the time and his buffer will remain bounded. Thus we build up a picture of the outcome of the system as shown in Figure 2. In the triangular region OPR

of good service, for points below the diagonal line from $(0,0)$ to $Q = (\overline{W}_1, \overline{W}_2)$ user 2 wins the queue and experiences lower average delays. In that region, as user 1's demand increases and the point moves towards the line QR, user 1 will experience increasing delay. In fact if the queues are treated as conventional Poisson arrivals then the mean delay experienced by user 2 is $\overline{A}_2/R_2/(P - \overline{A}_2/R_2)$ while the mean delay experienced by user 1 is $\overline{A}_1/R_1/(P - \overline{A}_1/R_1 - \overline{A}_2/R_2)$. When the line QR is crossed, the system capacity is exceeded and user 1's buffer will build up. Equally, in the upper part (OPQ) of the "Good service" triangle user 1 wins the queue and the situation is reversed.

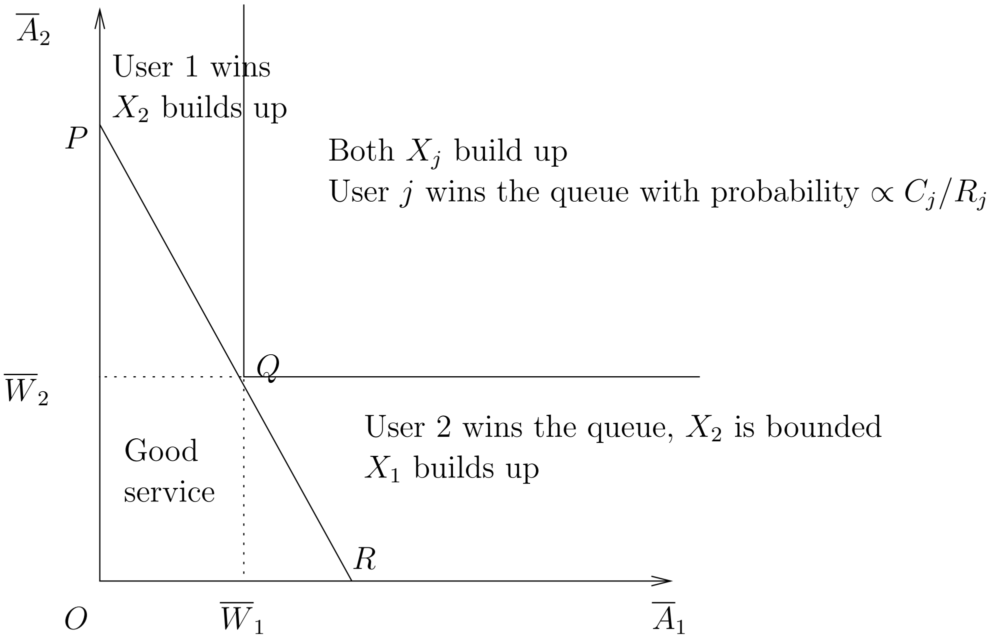

Figure 2: Long term operation of system with two users and ϕ small.

Two illustrations of the behaviour are shown in Figure 3. Here we have taken $R_1 = 1$, $R_2 = 2$ and $C_j = R_j^\alpha$ with $\alpha = 1.4$. In the simulation shown in the upper set of diagrams, there is enough capacity, both users experience good service, and user 2 eventually wins the queue all the time. In the lower diagrams the users are sending more data, and both buffers are building up. So here we expect from (15) the users to be winning the queue with frequency proportional to C_j/R_j, *i.e.* in the proportions about 0.43 and 0.57, *i.e.* user 2 winning the queue about 4 times out of 7. This is just as observed in the simulator.

It is natural to ask how this picture changes when ϕ is *not* small, so that there are significant variations in W and therefore S during the process. We again consider first the case where both buffers build up, so each user always has data ready to transmit. The effect of the scheduler then is that

(1) if $W_1/C_1 < W_2/C_2$ then user 1 wins the queue, $D_1 = R_1 P$, and $D_2 = 0$;

(2) if $W_2/C_2 < W_1/C_1$ then user 2 wins the queue, $D_2 = R_2 P$, and $D_1 = 0$.

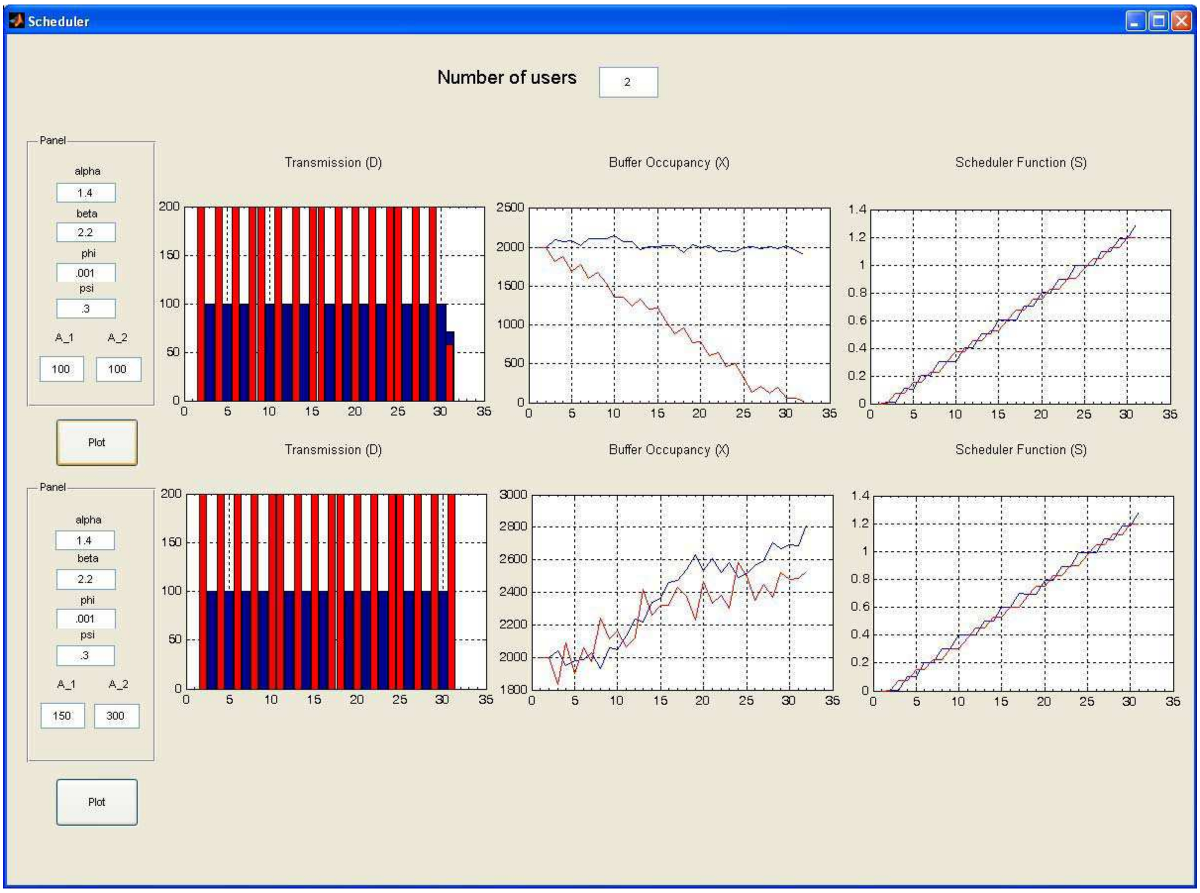

Figure 3: Simulation of two emerging behaviours. In the simulation represented by the upper diagrams, user 2 (red [grey]) has better channel quality than user 1 (blue [black]) but both users get adequate service. In the lower diagrams, the system is overloaded and user 2 wins the queue approximately 0.57 of the time (4 times out of 7).

Since the transmissions obey $\sum_j (D_j/R_j) = P$, we see from (9) that $\sum_j (W_j/R_j)$ will tend to P as $t \to \infty$, and so if we ignore a starting transient we can assume $\sum_j (W_j/R_j) = P$, and we therefore write

$$x_j(t) = \frac{W_j(t)}{R_j P}, \qquad x_1 + x_2 = 1. \tag{17}$$

Then x_1 obeys the nonlinear recurrence equation

$$x_1(t+1) = f(x_1(t), a, \phi) = \begin{cases} \phi + (1-\phi)x_1 & \text{in } x_1 < a, \\ (1-\phi)x_1 & \text{in } x_1 > a, \end{cases} \tag{18}$$

where the parameter a is equal to $(C_1/R_1)/(C_1/R_1+C_2/R_2)$. This function f is piecewise linear, with a discontinuity at $x = a$, as illustrated in Figure 4. After initial transients, this process will lie in the region

$$(1-\phi)a < x_1 < \phi + (1-\phi)a, \tag{19}$$

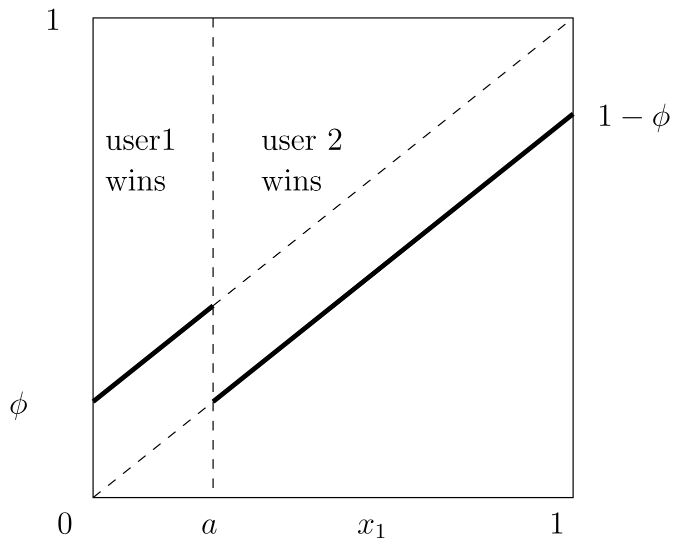

Figure 4: The graph of the discontinuous piecewise linear function $f(x_1)$ is the solid lines.

and so for ϕ small x_1 remains close to a. This is the situation discussed earlier, where the users share the time frames proportionally to C_j/R_j. However, this model gives the detailed behaviour when ϕ is larger. In fact, in simulations of the recurrence equation we find that the system always settles to a periodic behaviour, but we do not know a proof of this. However, since $f' = 1 - \phi < 1$ at all points where f' is defined, the action of f is measure-reducing, so certainly there is a set of measure zero to which all trajectories converge. If ϕ is large, in fact if

$$\phi > \phi_0 = 1 - \frac{\min(C_1/R_1, C_2/R_2)}{\max(C_1/R_1, C_2/R_2)}, \tag{20}$$

then the users win the queue alternately, and so in that case the long term proportions are $p_1 = p_2 = \frac{1}{2}$. For values of ϕ that are less than ϕ_0 there will be intermediate behaviour, with the p_j between the values given by (15) and $\frac{1}{2}$.

Illustrations of this using the simulator are shown in Figure 5. In the upper diagrams, ϕ is small and the parameter values are as before and the users win the queue in proportions roughly 0.43 and 0.57 as before, in accordance with (15). But in the lower diagrams, ϕ is large, and the usage alternates between users 1 and 2 as described here.

2.2 More than two users

When there are more than two users, much of the discussion above will still apply. In the case where all buffers contain data to transmit, we shall still have $\sum_j (W_j/R_j) = P$, and we can set $x_j = W_j/(R_j P)$ so that $\sum x_j = 1$ as before. Then the time-update system is

$$x_j(t+1) = \begin{cases} (1-\phi)x_j(t) + \phi & \text{if } x_j(t)/a_j = \min_i(x_i(t)/a_i), \\ (1-\phi)x_j(t) & \text{otherwise,} \end{cases} \tag{21}$$

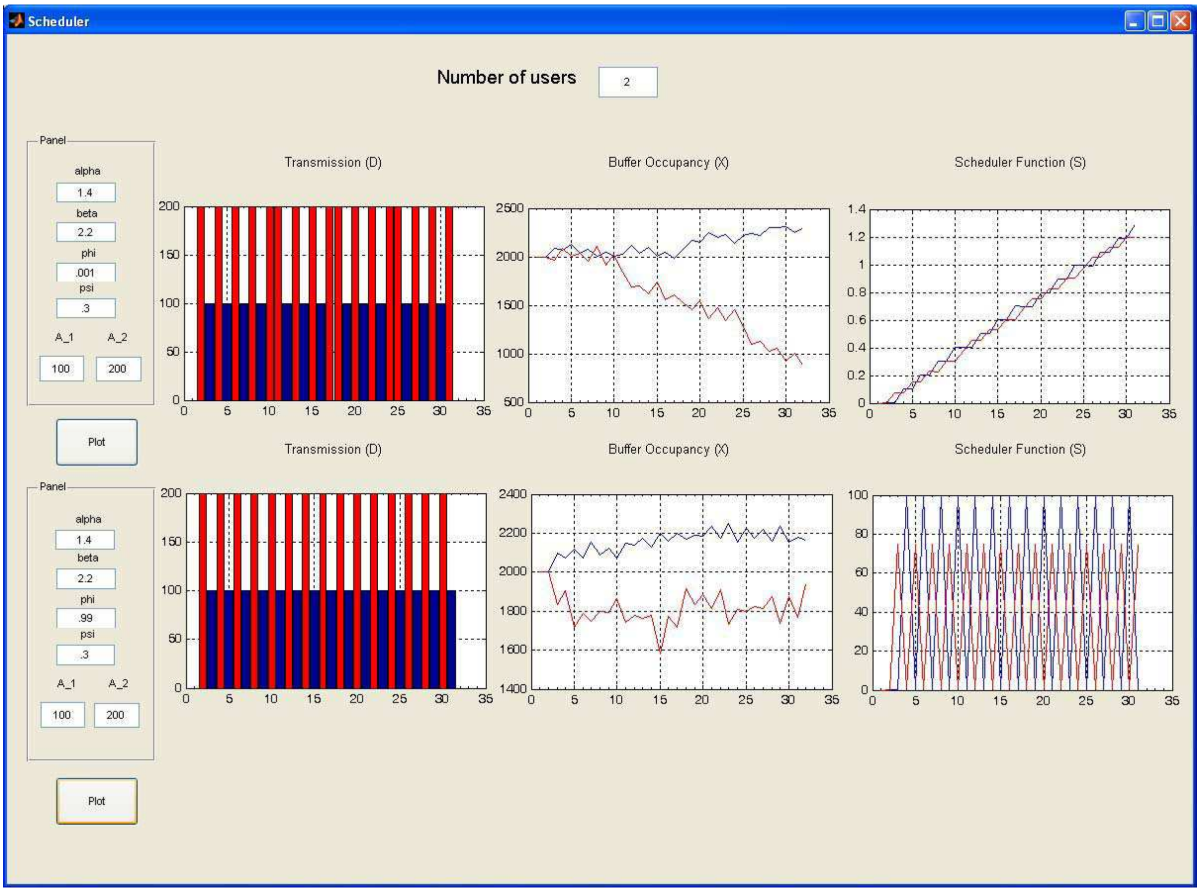

Figure 5: Effects of varying ϕ. In the upper simulation, with ϕ small, users 1 and 2 win the queue in proportion to C_j/R_j. But in the lower simulation, with ϕ large, they win alternately.

where $a_j = C_j/R_j$. (The first case is where user j wins the queue.) We can show that if ϕ is small then the long-term behaviour is close to $x_j = a_j/\Sigma$ where $\Sigma = \sum_i a_i$. Certainly, for some user(s) j it is the case that $x_j \leq a_j/\Sigma$. The user with the least value of $\frac{x_j}{a_j}$ wins the queue, so if user j wins the queue then $x_j \leq a_j/\Sigma$. So

$$\limsup_{t\to\infty}\big(x_j(t)\big) \leq \frac{(1-\phi)a_j}{\Sigma} + \phi. \tag{22}$$

Summing these over all other j and subtracting from 1 we see

$$\liminf_{t\to\infty}\big(x_j(t)\big) \geq \frac{(1-\phi)a_j}{\Sigma} - (J-2)\phi. \tag{23}$$

These are the generalizations to J users of the bounds in (19). For ϕ small, the right-hand sides of (22) and (23) each tend to

$$p_j = \frac{a_j}{\Sigma} = \frac{C_j/R_j}{\sum_i C_i/R_i} \tag{24}$$

and so the users win the queue in these proportions, exactly analogous to the case of 2 users. The behaviour for larger ϕ is governed by the recurrence (21) above, but we have not studied it in detail.

3 Variable channel quality

When the channel quality $R_j(t)$ varies with time, as it generally will in wireless applications, an averaged quantity $R_j^{\rm avg}(t)$ is defined by (2), and channel quality by (1). The idea of this is to allow user j to be able to take advantage of any short-term improvements in his channel quality R_j. If $R_j(t)$ varies with time as shown in Figure 6

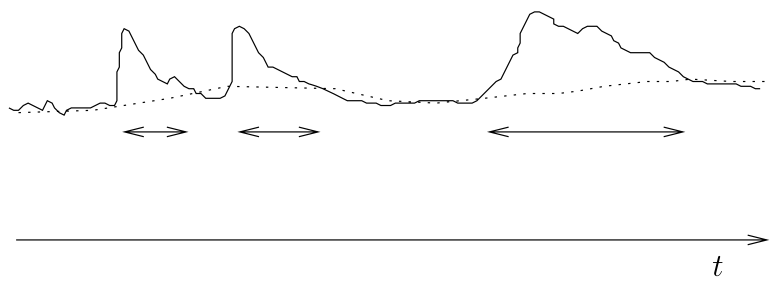

Figure 6: Channel quality $R_j(t)$ (solid curve) has occasional rises above its averaged value $R_j^{\rm avg}(t)$ (dotted curve).

then $C_j(t)$ has large values during the indicated intervals of good channel quality, and so user j gets increased priority during those times. In order to achieve this, it is necessary that ψ is small compared with $1/T$ where T is the typical length of the indicated intervals of good quality. This is because the averaging process (2) has the effect that

$$R_j^{\rm avg}(t) = \psi R_j(t)+\psi(1-\psi)R_j(t-1)+\psi(1-\psi)^2R_j(t-2)+\psi(1-\psi)^3R_j(t-3)+\ldots, \quad (25)$$

and so $R_j^{\rm avg}(t)$ is a weighted average of past values of $R_j(t)$ looking back a mean distance in the past which is

$$\sum_{k=0}^{\infty}\psi(1-\psi)^k k = \frac{1-\psi}{\psi}. \quad (26)$$

So for ψ small, $R_j^{\rm avg}(T)$ is an average of past values of $R_j(t)$ over times of order $1/\psi$ in the past, and if we want $R_j^{\rm avg}(T)$ not to rise significantly during an interval of good channel quality of length T then we need ψ small compared to $1/T$.

4 Conclusions

We have formulated a specific realisation of the scheduler that includes its important features and we have analysed its behaviour in some regimes. When the smoothing parameter ϕ is small and all users have data to transmit, they share the capacity in proportion to $C_j/R_j = u_j R_j^{\alpha-1}$. So for $\alpha > 1$ the users with better channel quality R_j are scheduled more of the capacity, and we can also see the influence of the traffic class weights u_j on how this capacity is shared. The detailed behaviour of the system when ϕ is not small is governed by the deterministic discontinuous dynamical system (21) when all users have data to transmit. When demand drops, there will be a transition to a more conventional underloaded queuing system. In that regime, the users with lower $\overline{A}_j/R_j$ will win the queue more often and experience lower delays.

In order for the β factor in the credit function to have the desired effect, R_j^{avg} must be slowly varying compared with channel quality fluctuations, and this requires ψ to be small compared with $1/T$ where T is the time scale of those fluctuations.

At a higher level, this scheduler may be one element in a coupled system where users modify their transmission rates or burst sizes according to the delays and quality of service they experience. This feedback would need to be the subject of a separate study: the model and analysis here is treating the arrival process as given, independent of the service or delay that the users experience. In a system with feedback, there could well be the possibility of instability if users respond to the quality of service too severely or on an inappropriate time scale.

A Full problem statement from Study Group

A SCHEDULER FOR A WIRELESS SYSTEM

This problem deals with the scheduling of data packets for different users that share a common wireless channel and that demand different quality of service in terms of service time.

Generally speaking the scheduling function is necessary because the capacity of the wireless channel is limited and because each user demands different service times. From a system point of view it is important to maintain a high throughput, that is, a high number of packets served per unit of time. However from a user point of view it is important to minimise the service time. This poses a trade-off problem, which is one of the goals we wish to analyse in this problem.

The wireless transmission channel alternates between good and bad state over time. While the channel is in bad state the transmission of a packet fails and the packet needs to be retransmitted. When the state of the channel is good packets are transmitted successfully and do not require retransmission. The system transmission efficiency or throughput might be defined as the time it takes to transmit an arbitrary amount of data packets successfully.

The scheduler we wish to study is designed to consider both the wireless channel conditions and the user's quality of service requirements. For this purpose users are assigned a credit every scheduling frame, which is a function of the wireless channel conditions and the quality of service (traffic class).

We consider the following credit function for a user j:

$$C_j(t) = w_j(R_j^{\text{avg}}(t))^{\alpha}$$

Where:

$$\begin{aligned} R_j^{\text{avg}}(t) &= \text{Average effective data rate} \\ w_j &= \text{traffic class weight.} \end{aligned} \tag{27}$$

The average effective data rate is computed as follows:

$$R_j^{\text{avg}}(t+1) = \varphi R_j^{\text{avg}}(t) + (1-\varphi)R_j(t)$$

Where:

$$R_j(t) = \text{Effective data rate at time } t.$$

We define the following normalised throughput for user j:

$$W_j(t+1) = \phi.W_j(t+1) + (1-\phi).\frac{D_j(t)}{C_j(t)}$$

Where $D_j(t)$ is the amount of data to be sent for user j at time t.

At time $t+1$ the scheduler orders users in increasing value of $\frac{W_j}{C_j}$, which determines the order in which user packets are transmitted.

We consider one queue for every traffic class. The maximum capacity of a queue is k packets. Packets arrive in Poisson batches with mean arrival rate $c_j\lambda$ (j identifies the traffic class). The size of the batches is a random variable with range $\{0, \ldots, K\}$. We consider that batch sizes are independent. We also consider an exponential service time with mean $1/\mu$.

We are interested in the analysis of the following performance measures:

(1) The system size (queue size including packet in service) distribution.

(2) The queueing time distribution for each traffic class.

(3) The packet loss distribution for each traffic class.

We are interested in general solutions that allow the study of these performance measures as a function of:

- the traffic class weights w_j (maximum of four traffic classes),
- the channel condition weight α,
- the capacity of the system K
- the weights ψ and φ (as starting point it is recommended to start the study with these weights equal to 1)