An Approximate Analysis of Two Class WFQ Systems

G. Horváth, M. Telek
Budapest University of Technology and Economics
Dept. of Telecommunication

Abstract— **The class based weighted fair queueing (WFQ) is applied in a lot of computer and communication systems. It is a popular way to share a common resource. Its efficient analytical solution is an open question for a long time. Different solutions were proposed, using complex analysis or numerical techniques. But all of these methods have their limits in usability. In this paper we present a very simple approach that provides a fast approximation for the queue length and waiting time measures. Although it is simple and looks rough, the comparison with simulation shows that it provides reasonable accuracy with an execution time less than a second.**

Keywords— **Weighted Fair Queueing, Coupled Processor Model**

I. Introduction

Class based weighted fair queueing is a service policy in multiclass systems. Consider a service station that provides its resources for customers belonging to different classes. The customers inside a class are waiting in an FCFS manner for their service. There are weights assigned to each class. The ratio of server capacity available for a class is given by the ratio of weights of the classes that are "active" (there are customers waiting in the queue belonging to that class). So the "importance" of the customers is regulated by the weight assigned to their class.

If the customer arrivals are according to Poisson processes, and service times are exponentially distributed, the system can be modeled by a "two dimensional" Markov chain.

There were many methods proposed to give the solution of this Markov chain. First we list numerical solutions. In [1], [2] the authors consider the same problem, but they call this kind of system *Coupled Processor Model* They express the steady state probabilities of the two dimensional Markov chain as a power series of the load, and give a recursive way to compute the coefficients of the powers. With that approach only a few number of classes can be handled (2-3), and as the load tends to 1 a large number of coefficients have to be computed to reach a given accuracy. An other approach ([3]) approximates the infinite model with a finite one, and uses a kind of Gaussian elimination to solve the finite Markov chain. During the Gaussian elimination the structure of the system is exploited, and reasonable speedup is achieved. But again, if the load is high, the reduced finite Markov chain has two many states and speed decreases fast.

In [4] the authors provide the generating function of the two dimensional Markov chain. The result is a two-variable complex (actually analytical) function, where – to be determined – the one dimensional boundary generating functions have to be expressed. This gives the difficulty of this approach, since it needs Wiener-Hopf factorization (see also [5]).

In this paper we consider a 2 class system, however the approach itself can be extended to more classes as well. Contrary to the solutions mentioned above, the arrival intervals and service times are given with two moments. Using this input two moments of the waiting time are approximated.

II. Concept of the Approximation

The concept is to approximate the 2-queue system as the queues were separated, and construct a service process for each that approximately imitates the behaviour of the original server.

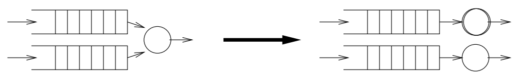

Fig. 1. Queue separation

For example looking at queue 1 the server capacity is changing between the full capacity and decreased capacity (according to the ratio of weights) depending on whether queue 2 is idle or busy. So the idea is simple: let characterize the busy period process of queue 2, and costruct a modulated server process for queue 1 where the modulation of the server is given by the busy period process of queue 2.

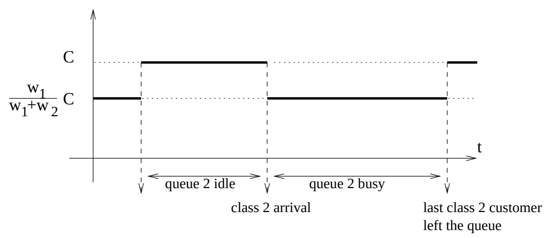

Fig. 2. Server capacity seen by a class 1 customer

As soon as the servers are separated, the queues are modeled by quasi birth-death processes (QBDs), and solved using matrix geometric techniques ([6]). In the Markov chain that models a queue the states are duplicated, corresponding to the idle or busy state of the other queue. In one state the customers can use the full server capacity, in the other they receive reduced service rate according to weights. Figure 3 shows the macro-structure of the Markov chain.

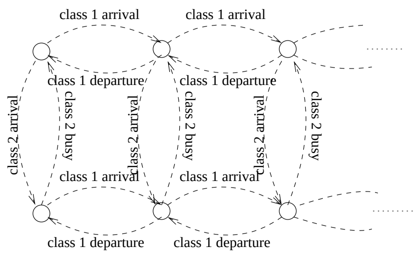

Fig. 3. Structure of the Markov chain

Before the technical details, we summarize the steps of the algorithm to help understanding its structure.

- Construct PH representation of the arrival and service processes (Section II-A)
- Compute busy periods (Section II-B). This step will produce the parameters of 4 PH random variables, since the computation detailed in that section has to be performed for queue 1 and queue 2, beside full capacity service and beside reduced capacity service.
- Construct the QBD that models queue 1 (Section II-C) and compute the waiting time parametes. Do the same for queue 2.

A. Arrival and Service

There are two classes. Measures and notations associated to class i are denoted by superscript $^{(i)}$.

The arrival process is characterized by the arrival intensity $\lambda^{(i)}$ and the squared coefficient of variation of the inter arrival times ${c_A^2}^{(i)}$. Based on these two moments a second order acyclic phase type distribution ([6]) is constructed having the same moments (of course, assuming $c_A^2 \geq 0.5$). This PH random variable is described by its transient generator matrix $D^{(i)}$, the vector of absorbing transitions $d^{(i)}$, and initial probability vector $\delta^{(i)}$. It is easy to check that the following PH r.v. has the proper moments:

$$D^{(i)} = \begin{bmatrix} -\frac{\lambda^{(i)}}{{c_A^2}^{(i)}} & \frac{\lambda^{(i)}}{{c_A^2}^{(i)}} \\ 0 & -2\lambda^{(i)} \end{bmatrix} \qquad d^{(i)} = \begin{bmatrix} 0 \\ 2\lambda^{(i)} \end{bmatrix}$$

$$\delta^{(i)} = \begin{bmatrix} \frac{1}{2{c_A^2}^{(i)}} & 1 - \frac{1}{2{c_A^2}^{(i)}} \end{bmatrix}$$

The length of the job brought by the customers is characterized by its mean $m_l^{(i)}$ and squared coefficient of variation ${c_l^2}^{(i)}$. Instead of these measures we compute and use the service rate and its squared coefficient of variation of the queues, beside full capacity (other queue is idle):

$$\mu_f^{(i)} = \frac{C}{m_l^{(i)}} \qquad {c_{Sf}^2}^{(i)} = {c_l^2}^{(i)},$$

and beside reduced capacity (other queue is busy):

$$\mu_r^{(i)} = \frac{w^{(i)}}{\sum_i w^{(i)}} \frac{C}{m_l^{(i)}} \qquad {c_{Sr}^2}^{(i)} = {c_l^2}^{(i)},$$

where C denotes the server capacity and $w^{(i)}$ denotes the weight assigned to class i. Again, as above a PH distribution is constructed from these parameters:

$$S_f^{(i)} = \begin{bmatrix} -\frac{\mu_f^{(i)}}{{c_{Sf}^2}^{(i)}} & \frac{\mu_f^{(i)}}{{c_{Sf}^2}^{(i)}} \\ 0 & -2\mu_f^{(i)} \end{bmatrix} \qquad s_f^{(i)} = \begin{bmatrix} 0 \\ 2\mu_f^{(i)} \end{bmatrix}$$

$$\sigma_f^{(i)} = \begin{bmatrix} \frac{1}{2{c_{Sf}^2}^{(i)}} & 1 - \frac{1}{2{c_{Sf}^2}^{(i)}} \end{bmatrix}$$

For the reduced capacity case $S_r^{(i)}$, $s_r^{(i)}$ and $\sigma_r^{(i)}$ are constructed similarly.

B. Busy Period

Let look at Figure 3 again. The transitions related to arrivals and services are already described, by PH distributions (see the last section). What is still missing is the busy period computation. We will compute two moments of the busy period of the queues as they were isolated (without the impact of the presence of the other queue), beside full and reduced server capacity. We have PH arrival and service process, so the Markov chain has a QBD structure:

$$Q = \begin{bmatrix} A_1' & A_0' & & & \\ A_2' & A_1 & A_0 & & \\ & A_2 & A_1 & A_0 & \\ & & \ddots & \ddots & \ddots \end{bmatrix}$$

The blocks of the generator of queue i with full server capacity are the following:

$$\begin{aligned} {A_0}_f^{(i)} &= d^{(i)}\delta^{(i)} \otimes I_{2\times 2} \\ {A_1}_f^{(i)} &= D^{(i)} \oplus S_f^{(i)} \\ {A_2}_f^{(i)} &= I_{2\times 2} \otimes s_f^{(i)}\sigma_f^{(i)} \\ {A'_0}_f^{(i)} &= d^{(i)}\delta^{(i)} \otimes \sigma_f^{(i)} \\ {A'_1}_f^{(i)} &= D^{(i)} \\ {A'_2}_f^{(i)} &= I_{2\times 2} \otimes s_f^{(i)} \end{aligned}$$

The blocks beside reduced capacity are obtained similarly.

To compute the kth moment of the busy period (${m_{Bf}^k}^{(i)}$) generated by an arrival the following equation has to be solved:

$${m_{Bf}^k}^{(i)} = (-1)^k (\delta^{(i)} \otimes \sigma_f^{(i)}) \frac{d^k}{ds^k} G_f^{(i)}(s)|_{s=0} h,$$

where $G_f^{(i)}(s)$ satisfies the following matrix equation:

$$sG_f^{(i)}(s) = {A_2}_f^{(i)} + {A_1}_f^{(i)} G_f^{(i)}(s) + {A_0}_f^{(i)} \left(G_f^{(i)}(s)\right)^2$$

The 0th derivative of $G_f^{(i)}(s)$ at $s = 0$ leads to the fundamental matrix geometric equation (for matrix G), see for example [6]. For the first derivative we have the following implicit equation:

$$\frac{d}{ds}G_f^{(i)}(s)|_{s=0} = \left(A_1{}_f^{(i)} - A_0{}_f^{(i)}G_f^{(i)}(0)\right)^{-1} \cdot \cdot \left(I - A_0{}_f^{(i)}\frac{d}{ds}G_f^{(i)}(s)|_{s=0}\right)G_f^{(i)}(0)$$

which we solved by a fix point iteration.

In our approximation only two moments are utilized, they are the following (from the definition, after some algebra):

$$m_{Bf}^{1\,(i)} = -\left(A_2{}_f^{(i)} + A_0{}_f^{(i)} + A_0{}_f^{(i)}G_f^{(i)}(0)\right)^{-1} h$$

$$m_{Bf}^{2\,(i)} = 2\left(A_2{}_f^{(i)} + A_0{}_f^{(i)} + A_0{}_f^{(i)}G_f^{(i)}(0)\right)^{-1} \cdot \cdot \left(A_0{}_f^{(i)}\frac{d}{ds}G_f^{(i)}(s)|_{s=0} - I\right) m_{Bf}^{1\,(i)}$$

Having these two moments the same PH fitting is performed like before, with the generator denoted by $B_f^{(i)}$, absorbing transitions denoted by $b_f^{(i)}$, and initial probability vector denoted by $\beta_f^{(i)}$.

To build up the Markov chain on Figure 3, we will also need the phase probability vector of the arrival process at the moment when the busy period finishes. This is denoted by $\alpha_f^{(i)}$ and can be easily computed from matrix $G_f^{(i)}(0)$:

$$\alpha_f^{(i)} = (\delta^{(i)} \otimes \sigma_f^{(i)}) \cdot G_f^{(i)}(0) \cdot (h_2 \otimes I_{2\times 2})$$

C. Queue Model

Now, we construct the block matrices of the QBD modeling queue i. The index of the other queue will be denoted by j (thus, if $i = 1$ then $j = 2$ and vice versa). As seen on Figure 3, the state space is divided into two parts. In the upper part, where the other queue is idle, queue i receives full service capacity. In this part of the state space, one has to keep track (1) the phase of the arrival process of queue i, (2) the phase of service process of queue i, and (3) the phase of arrival process of queue j. In the lower part, which represents the case when queue j is busy, and queue i receives reduced server capacity, the phase of the busy period of queue j has to be kept track instead of the phase of its arrival process. This behaviour is reflected in the definition of the block matrices of the QBD:

$$C_0^{(i)} = \begin{bmatrix} d^{(i)}\delta^{(i)} \otimes I \otimes I & 0 \\ 0 & d^{(i)}\delta^{(i)} \otimes I \otimes I \end{bmatrix}$$

$$C_1^{(i)} = \begin{bmatrix} D^{(i)} \oplus D^{(j)} \oplus S_f^{(i)} & I \otimes d^{(j)}\beta_r^{(j)} \otimes I \\ I \otimes b_r^{(j)}\alpha_r^{(j)} \otimes I & D^{(i)} \oplus B_r^{(j)} \oplus S_r^{(i)} \end{bmatrix}$$

$$C_2^{(i)} = \begin{bmatrix} I \otimes I \otimes s_f^{(i)}\sigma_f^{(i)} & 0 \\ 0 & I \otimes I \otimes s_r^{(i)}\sigma_r^{(i)} \end{bmatrix}$$

$$C_0'^{(i)} = \begin{bmatrix} d^{(i)}\delta^{(i)} \otimes I \otimes \sigma_f^{(i)} & 0 \\ 0 & d^{(i)}\delta^{(i)} \otimes I \otimes \sigma_r^{(i)} \end{bmatrix}$$

$$C_1'^{(i)} = \begin{bmatrix} D^{(i)} \oplus D^{(j)} & I \otimes d^{(j)}\beta_f^{(j)} \\ I \otimes b_f^{(j)}\alpha_f^{(j)} & D^{(i)} \oplus B_f^{(j)} \end{bmatrix}$$

$$C_2^{(i)} = \begin{bmatrix} I \otimes I \otimes s_f^{(i)} & 0 \\ 0 & I \otimes I \otimes s_r^{(i)} \end{bmatrix}$$

The number of phases is 16, so the classical QBD solver algorithms can provide the steady state probabilities and waiting times quickly (see [6]).

III. Numerical Results

Below, we evaluate two examples. In case 1 the moments of the job size of the two classes are similar and in the second case, the job size of customers of class 2 is 10 times longer.

A. Case 1

On the first pair of plots the waiting time moments are depicted as a function of the load of class 1 (Figure 4). As it is expected we obtained that the mean waiting time increases while its squared coefficient of variation decreases with increasing load.

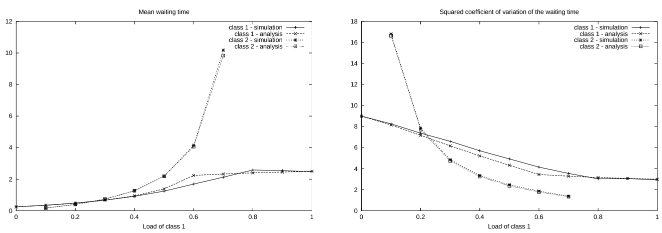

Fig. 4. The effect of the load

The second scenario investigates the influence of job size variance. In the next two plots the squared coefficient of variation of the inter arrival and service times are changed, and again two moments of the waiting times are captured (Figures 5 and 6). According to the results (both simulation and analysis) the increasing variance does only have a very small impact on the waiting time of the other queue.

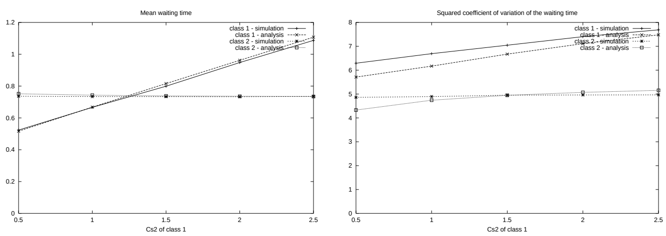

Fig. 5. The effect of the squared coefficient of variation of the service time

Finally, the waiting time is depicted as a function of the weight (Figure 7). Weight 0 means that the other queue has preemptive priority over the corresponding one. The results reflect this behaviour.

B. Case 2

In this case the mean job size of class 2 customers is 10 times larger. The first figure shows the waiting time as a

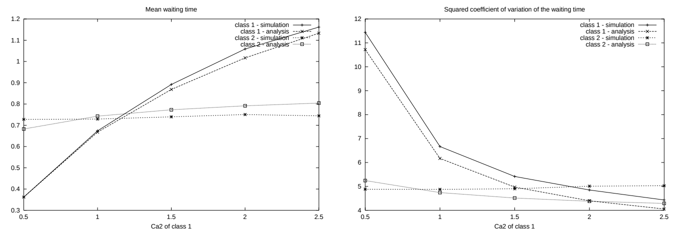

Fig. 6. The effect of the squared coefficient of variation of the inter arrival time

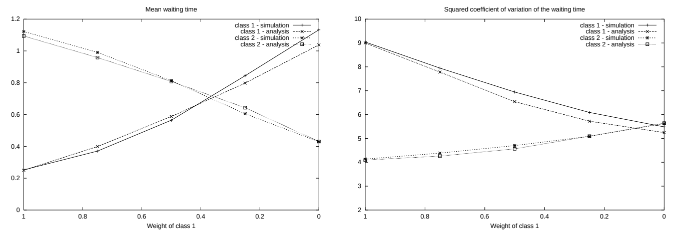

Fig. 7. The effect of the weight

function of the load (Figure 8). Even when the 'large job' (class 2) queue is overloaded, customers in class 1 get their guaranteed service, as it is indicated by its low waiting time.

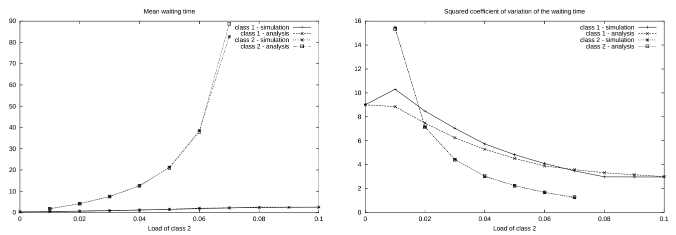

Fig. 8. The effect of the load

Figures 9 and 10 show the effect of the variance on the waiting time. The simulation results show again that the waiting time of queue 2 does not get worse when increasing the variance of the inter arrival or the service time of queue 1. The analysis shows a little correlation, but the difference from simulation remains reasonable.

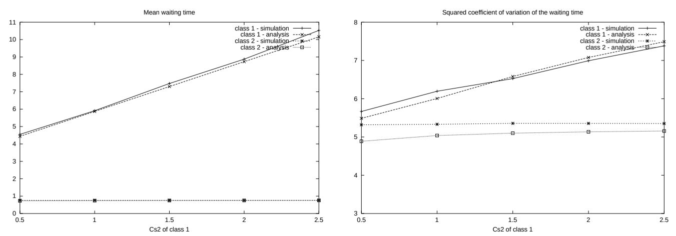

Fig. 9. The effect of the squared coefficient of variation of the service time

On Figure 11 the weight of class 1 is changed. This case provides the worst approximation. We obtain reasonable differences with respect to the mean waiting time, but the squared coefficient of variation difference grows to 20%.

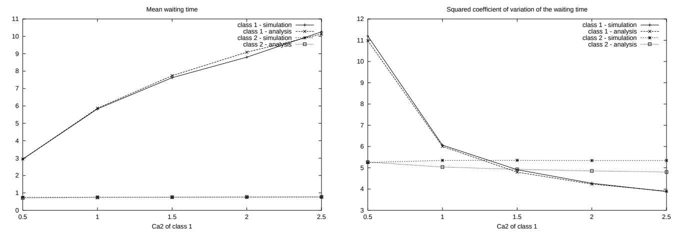

Fig. 10. The effect of the squared coefficient of variation of the inter arrival time

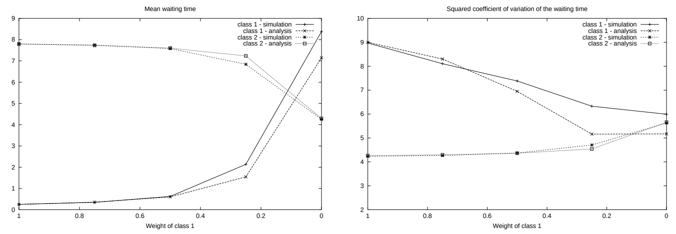

Fig. 11. The effect of the weight

References

[1] G. Koole, "On the power series algorithm," Tech. Rep., Centrum voor Wiskunde en Informatica, 1994.

[2] J.P.C. Blanc, "A numerical study of the coupled processor model," in *Computer Performance and Reliability*, 1988.

[3] Leslie D. Servi, "Algorithmic solutions to two-dimensional birth-death processes with application to capacity planning," *Telecommunication Systems*, vol. 21, no. 2, pp. 205–212, 2002.

[4] Fabrice Guillemin, Ravi Mazumdar, Alain Dupuis, and Jacqueline Boyer, "Analysis of the fluid weighted fair queueing system," *J. Appl. Probab.*, vol. 40, no. 1, pp. 180–199, 2003.

[5] Guy Fayolle, Roudolf Iasnogorodski, and Vadim Malyshev, *Random Walks in the Quarter Plane*, Springer-verlag New York, 1999.

[6] G. Latouche and V. Ramaswami, *Introduction to Matrix Analytic Methods in Stochastic Modeling*, American Statistical Association and the Society for Industrial and Applied Mathematics, 1999.