ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING

YEAR : 2005 (1353-1364)
VOLUME : 5
NUMBER : 2

A MINI-MAX APPROACH FOR FINDING THRESHOLDS TO CONTROL M/M/2 QUEUES

Malik K. ŞİŞ

Dokuz Eylül University, Engineering Faculty, Computer Engineering Department
Bornova, Izmir, Turkey

E-mail: kemal@cs.deu.edu.tr

ABSTRACT

Within the framework of this paper, we consider an M/M/2 queuing model where a threshold –on the queue size- type control is assumed for using (or not using) the slower server. The optimization policy to control is based upon a mini-max criterion, which minimizes the maximum "first passage time to an idle period" for the servers. We derived the formulas to calculate the exact value of the optimal threshold, and we analyzed some extreme cases for the explicit expressions to connect the optimal threshold values to the queuing parameters.

Keywords: *Congestion, M/M/2 Queues, Optimization, Mini-Max*

1. INTRODUCTION

In the terminology of routing literature, a strategy is a complete decision procedure for directing the traffic in a network, such that the constraints related to the network are satisfied, and a particular performance objective is achieved, meantime. Talking about routing in a network usually recalls "the store-and-forward type" networks, which can be taken as a collection of "nodes" and links [2] that interconnects them. In this effect, a strategy is called local –as opposed to "centralized"- when only node level parameters and node state variables are used to implement the strategy in the individual nodes. Each node implements the local strategy in its own right to achieve its performance objective.[1] A strategy is called "dynamic" –as opposed to "static"- when it is implemented as a function of the state of the node [2]. In this paper, a work toward a dynamic-local routing strategy is evaluated.

Queuing theory is considered to be the most useful mathematical media to deal with the

[1] The true sense of the strategy, however, implies the performance achievement of the overal network.
[2] A static one assigns fixed routes based upon to each node-destination pair based upon the expected traffic behaviour.

Received Date : 10.02.2004
Accepted Date: 15.05.2005

packet-switching type of data communication networks [1]. In this work, also, we will use a particular queue-server connection to represent our node model for mathematical evaluation. A node may represent, in our area of interest, a "packet switching node" or a "router" for that matter. In the simplest non-trivial case, we may consider a switching node with two out-going links and one type of traffic load stored in a single outbound buffer. This may seem too basic for any router application, however, more meaningful applications may also be decomposed into such several basic nodes [3].

In the figure below, we present a network node model (its block diagram), with n bi-directional links and m types of traffic. Classifier sorts the total incoming traffic into m array of buffers, which may correspond to m Queues (Q1-Qm). The accumulated traffic in the buffers is subject to be forwarded by a Nodal (local) routing logic, by means of a switching matrix, to the appropriate outgoing links. Each outgoing link is represented by "a server" whose service rate (μ_i) is proportional with its "capacity" (C_i bits/sec). There are n of them, in the practical cases each bi-directional link will bring one Input and one outgoing link to our model.

So much for the general view of a packet switching node model of ours: We will be interested in a model with one buffer and two outgoing links. That can be a partial node which represents a part of the whole, or a special type of a node that accumulates the traffic from several terminals whose traffic type is all the same, and forwards it through two possible outgoing links to their destination[3]. In this case, there is one queue (Q) representing the buffer that stores the total accumulated traffic in the node. *q(t)* is the size of Q at the time instant *t*. For the two possible outgoing links (or, they may be two possiblr destination servers which may serve to the same purpose (still one type of traffic). Servers (S1 and S2), however, have different service rates (μ_1 and μ_2) in general.

[3] One destinatination may be assumed. That is to say "type of traffic" is according to the destinatination which is a single one.

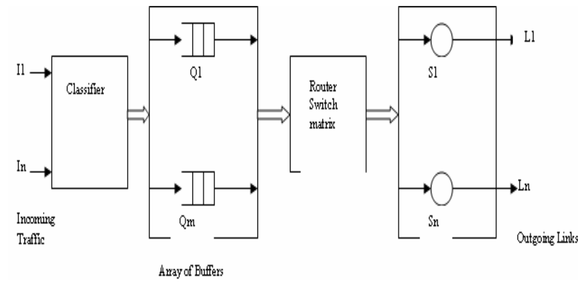

Figure 1: Block diagram of a generalized Node model

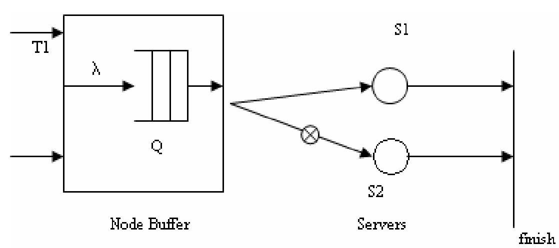

Figure 2: A one Q and two Server model for a store-and-forward type node with threshold on Q.

1.1 Performance Criterion

The following notaions and assumptions are for the parameters of the M/M/2 model which will be discussed in this paper:

λ: Average arrival rate to the queue (Q, buffer)
μ_1: Average service rate for the faster server.
μ_2 : Average service rate for the slower server.
b_1: State variable for the server 1's busy position ($b_1=1$ ➔ Server 1 is busy).
b_2: State variable for the server 2's busy position ($b_2=1$ ➔ Server 2 is busy).
q : State variable for the queue's size.
Thus, we will denote the state of the node $\underline{x}$ with a triplet:
$\underline{x}=(q,b_1,b_2)$
All the state variables ($\underline{x(t)}=(q(t),b(t)_1,b_2\ (t))$are the functions of the time *t*.
We assume here that
$\mu_1 \geq \mu_2$ and $\mu_1+\mu_2 > 1$.
The mathematical formulation of the performance criterion is a model for measuring the actual physical performance of the network. The dynamic local strategy that we consider will be based on a performance criterion to model the

congestion reducing capability of the network at the node level. Furthermore, this performance criterion, we claim, is a good approach to minimizing a delay (source-to-destination) objective in the overall network. If the performance criterion can be optimized by a complete decision procedure –that is a policy $\underset{\sim}{p}$ - then, for the optimum performance criterion, we can write

$$V(x(t)) = \min_{\underset{\sim}{p}} J(\underset{\sim}{p}, \underline{x}(t)) \quad (1)$$

Here, *J(p, x(t))* is the cost function, and for the static strategies, *x(t)* argument would be vanished. How the policy $\underline{p}$ relates to the control of the classifier (null in our case) and routing-decision blocks and to the state of the node is to be given by a list of rules. However, the needed control inputs are also depend on the type of the policy chosen to be implemented.
Based on the node model and the cost function, the form of the optimum policy can be uniquely defined: For example, Kumar [6] showed that the optimal policy, for an M/M/2 node model and for a cost function defined as the mean sojourn time of customers in the system, is of "threshold type".

We are going to look for a policy that will make the expected first passage time to an idle period (FPTIP) for Server 1 and Server 2 equal when a sizable accumulation in the queue exists (congestion!). The sole question to be answered, in order to determine the optimum policy $\underline{p}$, is when to use the slower server to serve to the one at the top of the queue. This question can also be answered by giving an optimum threshold value $\tilde{K}$ on the queue size *q*. That is to say "use Server 2 if, and only if, $q > \tilde{K}$, and leave S2 idle if $q \leq \tilde{K}$". That is the quantitative definition of our optimal policy. Finding optimum thresholds for utilizing of slower servers has been studied by using flow equivalent approach [4]. The exact definitions of *"an idle period, the first passage time to an idle period for Server j (T_j) and the Cost function J($\underline{p}$.$\underline{x}$(t))"* can be found in [1]. The $\underline{p}$ which is optimum in the sense that it minimizes the maximum expected first passage time to an idle period of servers4 has the following form:The optimum policy never leaves Server 1 (the faster server) idle (when there is a customer to be served in the queue), and is a simple threshold policy on the usage of the slower server (Server 2). That is, when Server 2 is idle and there are more than K customers in the queue, a customer is sent to Sever 2 (thus it does not enter to an idle period). Otherwise it is left idle (does enter to an idle period).

Here, K denotes the optimum value of the threshold K (in general).
Property: Under the threshold policy $\not{p}[K]$ (i.e., Policy $\not{p}$ that uses *K* as the threshold value) the state of the node ($\underline{x}(t)=(q,b_1,b_2)$) passes from the state *(K,1,0)* to *(K,1,1)* when the first event (after time *t*) is an arrival.
When $\mu_1 = \mu_2$ then there is no reason for not using Server 2 whenever possible. For this case the optimum threshold value is zero:

$$\mu_1 \approx \mu_2 \Rightarrow \not{K} = 0 \quad (2)$$

By definition we take $\not{K}$ as an integer and $\not{K} \geq 0$.
Definitions and the main theorem
Given that policy *p* uses *K* as threshold value, we define, for the state of the M/M/2 node,

δ states: $\delta_n = (n,1,0)$ for $n=0,..,K$.

γ states: $\gamma_n = (n,1,1)$ for $n=0,..,K,K+1,..$

$$0^* \text{ state: } = (0,0,0) \quad (3)$$

$$\phi^* \text{ state: } = (0,0,1) \quad (4)$$

For the expected FPTIP for Server j in the state $\underline{x}$ given that the threshold *K* is implemented, we denote

$$\overline{T}_j = \overline{T}_j(\underline{x};K) = €\left\{\overline{T}_j(\underset{\sim}{p}) \mid \underline{x}\right\} \quad (5)$$

The following equations define the first passage time to an idle period from specific states [5](note that Server 1 is idle if the state becomes 0^* *or* ϕ, and Server 2 is idle if the state becomes 0^* *or* δ*):*

$$\Delta_q^j(K) = \overline{T}_j(q,1,0;K) \quad (6)$$

$$\Gamma_q^j(K) = \overline{T}_j(q,1,1;K) \quad (7)$$

Thus, $\Delta_n^j(K)$ shows the expected FPTIP for Server *j* given that the initial state is δ_n and the threshold value is set to *K*. Similarly, $\Gamma_n^j(K)$

[4] There can be, say, n servers in general, however we will deal with n=2 only in this paper.
[5] Upper indices will not mean to power of something.

shows the expected FPTIP for Server j given that the initial state is γ_n and the threshold value is set to K. Also let, when $q=K$,

$$Y_K^j=\Delta_k^j(K) \qquad (8)$$

$$Z_K^j=\Gamma_k^j(K). \qquad (9)$$

We will also use simply Z_K and Y_K–to indicate the corresponding FPTIPs.

Theorem 1: the optimum value of the threshold, K , is the smallest non-negative integer such that $Y_{K+1}^{(1)} > Z_K^{(2)}$.

That is,

$$\not{K} = \min\{K \geq 0 \mid \overline{T}_1\ (K+1,1,0;K+1) > \overline{T}_2(K,1,1;K)\} \qquad (10)$$

Note this mans that the optimal threshold value ~~K~~ is the smallest integer such that the expected FPTIP for the faster server, when K=~~K~~+1, is greater than that of the slower server when K=~~K~~. If there is no such non-negative value then ~~K~~=0. Proof of this theorem is provided in [3][6]. Now what is required is to calculate the values of Z_K^2 and Y_K^1 in terms of K and the node parameters (μ_1, μ_2, and λ). As such we can get the exact expression for the value of ~~K~~. However, as we can see, practically, this is not easy.

2. Z-Transform Analysis for FPTIP Expressions

Throughout this section we will use K to denote the threshold value on the queue size (q), and n will denote the number of customers in the queue. When it is not confusing, we will simply use Γ_n^i instead of $\Gamma_n^i(K)$ *and alike…*

FPTIPs from γ_n states for n>K.

For these states, both servers are employed to reduce the queue size since it is above than the threshold. Therefore, the total average service rate for the system (node) is

$\mu=\mu_1+\mu_2$ and -by definition- $C_e=\mu-\lambda$.

Accordingly, the <u>expected time</u> to reduce the queue size by one is $1/C_e$. The back-ward difference equations for the expected FPTIPs:

$$\Gamma_n^i = \frac{1}{\mu_1+\mu_2+\lambda} + \frac{\lambda}{\mu_1+\mu_2+\lambda}\Gamma_{n+1}^i + \frac{\mu}{\mu_1+\mu_2+\lambda}\Gamma_{n-1}^i$$

for $i=1,2$ and for $n>K$ (11)

Equation (11) can be derived by thinking that [5]

$$\Gamma_n^i = €\{T_i|\underline{x}=\gamma_n\} = \not\tau + \sum_{\underline{\ell}} P(\underline{x}=\underline{\ell} \mid \gamma_n)€\{T_i \mid \underline{x}=\underline{\ell}\} \qquad (12)$$

where $\not\tau$ is the average time between two successive events (a new arrival to the queue,or a departure from a server), $P(\underline{x}=\underline{\ell} \mid \gamma_n)$ denotes the probability of moving from $\underline{x}=\gamma_n$ to a state $\underline{x}=\underline{\ell}$, and note that $\underline{\ell}$ can only be either γ_{n+1} or γ_{n-1}. $€\{T_i \mid \underline{x}=\underline{\ell}\}$ is just the average FPTIP for Server i given that the current state is $\underline{\ell}$. The total average rate of events is $\lambda+\mu_1+\mu_2$. That is, the next event, given that the current state is $(n,1,1)$, can be either a departure from a server or an arrival to the queue. Hence,

$$\not\tau = \frac{1}{\mu_1+\mu_2+\lambda}.$$

Probability of having an arrival to the queue, and thus moving to the state $\gamma_{n+1}=(n+1,1,1)$, is

$$P(\underline{x}=\gamma_{n+1} \mid \gamma_n) = \frac{\lambda}{\mu_1+\mu_2+\lambda}.$$

When the state moves to γ_{n+1}, FPTIP for Server *i, by definition,* becomes $€\{T_i|\underline{x}=\gamma_{n+1}\}=\Gamma_{n+1}^{\ i}$.

When a departure from a server is the next event, we have $\underline{x}\rightarrow\gamma_{n-1}=(n-1,1,1)$. For this case, we can write

$$P(\underline{x}=\gamma_{n-1} \mid \gamma_n) = \frac{\mu_1+\mu_2}{\mu_1+\mu_2+\lambda}.$$

When the state moves to γ_{n+1}, FPTIP for Server *i, by definition,* becomes $€\{T_i|\underline{x}=\gamma_{n-1}\}=\Gamma_{n-1}^{\ i}$. With these equalities, Equation (12) and (11) becomes equivalent. Now, let us define two new parameters from the original node parameters:

$$a_1 \triangleq \frac{\mu_1}{\lambda}, \qquad a_2 \triangleq \frac{\mu_2}{\lambda} \qquad (13)$$

With these, Equation (11) becomes

$$(1+a_1+a_2)\Gamma_n^i=\Gamma_{n+1}^{\ i}+(a_1+a_2)\Gamma_{n-1}^{\ i}+\lambda^{-1} \qquad (14)$$

---

[6] For the interested reader, I can provide the theorem's proof via e-mail.

The equation has the same form for either server, and can easily be solved in terms of $Z_K=\Gamma_K(K)$ s:

$$\Gamma_n^i(K) = \frac{n-K}{C_e} + Z_K^i \quad \text{for } i=1,2 \tag{15}$$

The last equality also holds for $n=K$ as it does for $n>K$. Note that since Z_K^1 and Z_K^2 are, in general, different, so are the solution of Equation (15) for Server 1 and 2.

FPTIPs for the Faster Server from a δ-state

The basic difference equation for $\Delta_n^1(K)$ can be derived from Equation (12) in a similar way which is demonstrated for Equation (11). For this, we have to determine the state transition probabilities and event rates for a $\delta_n=(n,1,0)$ state. For this states, Server 2 is idle and thus a departure from it is not a possible event. A departure from server 1 or an arrival to the queue can hapen, therefore the total rate of events is just

$$\frac{1}{\mu_1+\lambda}.$$

Accordingly, steady-state transition probabilities are

$$P(\underline{x}=\delta_{n+1} \mid \delta_n) = \frac{\lambda}{\mu_1+\lambda},$$

$$P(\underline{x}=\delta_{n-1} \mid \delta_n) = \frac{\mu_1}{\mu_1+\lambda}.$$

With this and definitions in Equation (13), Equation (12) yields to

$$(1+a_1)\Delta_n^1 = \Delta_{n+1}^1 + a_1\Delta_{n-1}^1 + \lambda^{-1} \tag{16}$$

In order to put this into a homogeneous form, let us define

$$d_n = \frac{\Delta_n^1 - \dfrac{n+1}{\mu_1-\lambda}}{\Delta_0^1 - \dfrac{1}{\mu_1-\lambda}} \tag{17}$$

With this, Equation (16) transforms into

$$\left[\Delta_0^1 - \frac{1}{\mu_1-\lambda}\right]((a_1+1)d_n - d_{n+1} - a_1 d_{n-1}) = \frac{1}{\lambda} - \frac{(a_1+1)(n+1)}{\mu_1-\lambda} + \frac{n+2}{\mu_1-\lambda} + \frac{a_1 n}{\mu_1-\lambda} \tag{18}$$

By considering the definition of a_1, one can easily show that the last equality can be simplified as

$$(a_1+1)d_n = d_{n+1} + a_1 d_{n-1} \quad \text{for } n=0,1,..K \tag{19}$$

so long as $\Delta_0^1 \neq 1/\mu_1-\lambda$ and, by definition, $d_m=0$ for $m<0$ and $d_0=1$. Here, we will use $\mathcal{Z}$ transform method to solve the difference equation (19). Let us apply $\mathcal{Z}$–transform to it:

$$(a_1+1)D(z)=zD(z)-z+a_1z^{-1}D(z). \tag{20}$$

This gives the solution, in the z-domain,

$$D(z) = \frac{z^2}{z^2-(a_1+1)z+a_1}. \tag{21}$$

If we decompose the right-hand side to simple fractions, and takr the inverse transform, we arrive to

$$d_n = \frac{a_1^{n+1}-1}{a_1-1}. \tag{22}$$

Accordingly, for $a_1\neq 1$, we arrive to

$$\Delta_n^1 = \frac{n+1}{\mu_1-\lambda} + \frac{a_1^{n+1}-1}{a_1-1}\left(\Delta_0^1 - \frac{1}{\mu_1-\lambda}\right) \quad \text{for } n=0,..,K,K+1. \tag{23}$$

For $a_1=1$, by taking the limit, corresponding value is given by the equation

$$\Delta_n^1 = -\frac{n(n+1)}{2\lambda} + (n+1)\Delta_0^1 \quad \text{for } n=0,..,K,K+1. \tag{24}$$

The values in the last two equations, for n=K+1, also satisfy the following continuity eqution:

$$\Delta_{K+1}^1(K) = \Gamma_K^1(K) \tag{25}$$

The quantity $\Delta_0^1(K)$ remains to be determined and the last eqution is one of the " boundary conditions" which is to be used to find the value of $\Delta_0^1(K)$.

FPTIPs for the Faster Server from a γ-state

Here too, we can say that the basic difference equation for $\Gamma_n^1(K)$ can be derived from

Equation (12) in a similar way which is demonstrated for Equation (11). For this, we have to determine the state transition probabilities and event rates for a $\gamma_n=(n,1,1)$ state where $K \geq n$. For these states, Server 2 is busy but a departure from it leads to a transition to the state δ_n. A departure from server 1 or 2, or an arrival to the queue can happen, therefore the total rate of events is again $\mu_1+\mu_2+\lambda$. Hence, instead of Equation (14), departing from Eqaution (11) once again, and also using the Equation (13) definitions, we can derive the following eqution for $\Gamma_n^1(K)$ for $K \geq n$:

$$(1+a_1+a_2)\Gamma_n^1 = \Gamma_{n+1}^1 + a_1\Gamma_{n-1}^1 + a_2\Delta_n^1 + \lambda^{-1} \quad \text{for } n=0,..,K \tag{26}$$

Note that for $n=0$, on the right-hand side, we have the term Γ_{-1}^1 ! When the current state is $(0,1,1)$ and a departure from Server 1 occurs, the next state becomes $\phi=(0,0,1)$ where Server 2 is busy, but Server 1 is idle. Hence $\Gamma_{-1}^1(K) \equiv 0$. The solution of the last difference equation system can be extended to $\Gamma_{K+1}^1(K)$.Actually Equation (26) can be considered to be valid for all positive integer n values if we define

$$\Delta_{n+1}^1(K) \equiv \Gamma_n^1(K) \quad \text{for } n \geq K. \tag{27}$$

In order to put this into a homogeneous form, let us define

$$g_n = \Gamma_{n+1}^1 - \Delta_n^1 \tag{28}$$

By remembering the orginal difference equation system for Δ_n^1's (Equation (16)), it can be shown that Equation (26) transforms into

$$(a_1 + a_2 + 1)g_n = g_{n+1} + a_1 g_{n-1} \quad \text{for } n=0,..,K \tag{29}$$

Note that $g_{-1} \equiv 0$ because $\Gamma_{-1}^1(K) \equiv 0$ and $\Delta_{-1}^1(K)=0$.

If we take the $\mathcal{Z}$-transforms of the Equation (29), we get

$$(1+a_1+a_2)G(z) = zG(z) - zg_0 + z^{-1} a_1 G(z) \tag{30}$$

Here $g_0=\Gamma_0^1-\Delta_0^1$ and its value remains to be determined. The solution in the $\mathcal{Z}$-domain is

$$G(z) = \frac{z^2 g_0}{z^2 - (a_1 + a_2 + 1)z + a_1}. \tag{31}$$

If we calculate the poles of $G(z)$ which are

$$z_{1,2} = \frac{1}{2}\left(1 + a_1 + a_2 \pm \sqrt{(a_1 + a_2 + 1)^2 - 4a_1}\right) \tag{32}$$

in terms of these we can decompose $G(z)$ in its fractional terms, and from there we can take the inverse $\mathcal{Z}$-transform to get

$$g_n = g_0 \frac{z_1^{n+1} - z_2^{n+1}}{z_1 - z_2}. \tag{33}$$

Accordingly

$$\Gamma_n^1 = g_0 \frac{z_1^{n+1} - z_2^{n+1}}{z_1 - z_2} + \frac{a_1^{n+1} - 1}{a_1 - 1}\left(\Delta_0^1 - \frac{1}{\mu_1 - \lambda}\right) + \frac{n+1}{\mu_1 - \lambda} \tag{34}$$

Now we can proceed to the task of determining the solutions for g_0 and Δ_0^1.

For this, we have the following boundary conditions:

$$\Delta_{K+1}^1 = \Gamma_K^1 \tag{35}$$

$$\Gamma_{K+1}^1 = \Gamma_K^1 + \frac{1}{C_e} \tag{36}$$

If we use Equation (34) in (35) to find a g_{K+1} value and evaluate Equation (33) for $n=K+1$

To find another expression for g_{K+1}, and finally from this two equations for g_{K+1}, we can solve for g_0 and use this in Equation (33) again, to get

$$g_n = \frac{1}{C_e} \frac{z_1^{n+1} - z_2^{n+1}}{z_1^{K+1} - z_2^{K+1}}. \tag{37}$$

On the other hand, from Equation (27) evaluated for $n=K$, and equation (34) we get

$$\Delta_{K+1}^1 - \Delta_K^1 = g_K. \tag{38}$$

If we avaluate Equations (37) and (23) for $n=K$ and insert them in the last equation, we can get the following solution for $\Delta_0^1(K)$:

$$\Delta_0^1(K) = \frac{1}{a_1^{K+1}}\left[\frac{1}{C_e} \frac{z_1^{K+1} - z_2^{K+1}}{z_1^{K+2} - z_2^{K+2}} - \frac{1}{\mu_1 - \lambda}\right] + \frac{1}{\mu_1 - \lambda} \tag{39}$$

This, of course, is not valid for $\mu_1=\lambda$. If so, then taking limit as $a_1 \rightarrow 1$, we get

$$\Delta_0^1(K) = \frac{1}{C_e} \frac{z_1^{K+1} - z_2^{K+1}}{z_1^{K+2} - z_2^{K+2}} + \frac{K+1}{\lambda} \tag{40}$$

By using Equation (39) in (23) (or, (40) in (24)), we can get the solution for $\Delta_n^1(K)$. By evaluating its value for, n=K, we can find $Y_K^1=\Delta_K^1(K)$ solutions:

$$Y_K^1=\frac{1-a^{-K-1}}{a_1-1}\left[\frac{1}{C_e}\frac{z_1^{K+1}-z_2^{K+1}}{z_1^{K+2}-z_2^{K+2}}-\frac{1}{\mu_1-\lambda}\right]+\frac{K+1}{\mu_1-\lambda}$$

for $\mu_1\neq\lambda$ (41)

$$Y_K^1=\frac{K+1}{C_e}\frac{z_1^{K+1}-z_2^{K+1}}{z_1^{K+2}-z_2^{K+2}}+\frac{(K+1)(K+2)}{2\lambda}$$

for $\mu_1=\lambda$ (42)

FPTIP for the Slower Server from a γ-state

Here the original back-ward type difference equation becomes (in terms of a_i s)

$$(1+a_1+a_2)\,\Gamma_n^2=\Gamma_{n+1}^2+a_1\Gamma_{n-1}^2+\lambda^{-1}$$

for $n=0,..,K$ (43)

The last equation is valid for n=K also, but to evaluate it at this point, let us define

$\Gamma_{-1}^2(K)=\Phi(K)=\epsilon\{T_2|\underline{x}=\phi\}$

Using this definition, (43)s equivalent equation for $n=-1$ (backward equation evaluated at $\underline{x}=\phi$ where Server 1 is idle and departure from Server 1 is not a possible event), we get

$(1+a_2)\Phi=\Gamma_0^2+\lambda^{-1}$. (44)

Once again, to get rid off the constant term λ^{-1}, we define the following transformation:

$$f_n=\frac{\Gamma_{n-1}^2-\dfrac{1}{\mu_2}}{\Phi-\dfrac{1}{\mu_2}}.$$

Note that by definition $f_0=1$. By using the above definition, we transform Equations (43) and (44):

$(1+a_1+a_2)f_{n+1}=f_{n+2}+a_1f_n$ for $n=0,..,K$ (45)

$1+a_2=f_1$ for $\Phi\neq 1/\mu_2$

By applying Z-transformation to (45) we can solve for $F(z)$ as

$$F(z)=\frac{z^2-a_1z}{z^2-(a_1+a_2+1)z+a_1}$$

The poles of $F(z)$ are the same z_1 *and* z_2 as in the previous sub-sections. After separating F into its partial fractions we can apply inverse transform to simply get:

$$f_n=\frac{z_1^{n+1}-z_2^{n+1}}{z_1-z_2}-a_1\frac{z_1^n-z_2^n}{z_1-z_2}$$

Note that the solution given above may be expanded to $n=K+1$ and as boundary condition we also have $\Gamma_{K+1}^2=\Gamma_K^2+1/C_e$. After evaluating (43) for $n\rightarrow n+1$ and using in the boundary condition, we can get a solution for $\Phi(K)$:

$$\Phi(K)=\frac{1}{\mu_2}+\frac{1}{C_e}\frac{1}{f_{K+2}-f_{K+1}} \quad (46)$$

Now, we can calculate, first $\Gamma_n^2(K)$ and then, finally, Z_K^2:

$$Z_K^2=\frac{1}{\mu_2}+\frac{1}{C_e}\frac{f_{K+1}}{f_{K+2}-f_{K+1}} \quad (47)$$

Finding the values of $\not{K}$

By using the $Z_K^{(2)}$ given in (47) (and, definition of f_n, of course) and $Y_K^{(1)}$ given in (41) (or in (42) if $a_1=1$) in the inequality set

$$Y_{K+1}^{(1)}>Z_K^{(2)}>Y_K^{(1)} \quad (48)$$

we can have an equation to solve for a non-negative integer K: let $\not{K}$ denote solution of Equation (48) for K. Since $Z_K^{(2)}$ is a monotone decreasing function of K and $Y_K^{(1)}$ is a monotone increasing function of K [3], for a $\underline{K}$ value we get $Y_{\underline{K}}^{(1)}=Z_{\underline{K}}^{(2)}$ and $K=\lfloor\underline{K}\rfloor$[7]. The inequality $Y_{K+1}^{(1)}>Z_K^{(2)}$, according to our Theorem 1 gives, for the smallest integer that proves it, the optimal threshold value $\not{K}$. Hence

$$\not{K}=\lfloor\underline{K}\rfloor \quad \text{where} \quad Y_{\underline{K}}^{(1)}=Z_{\underline{K}}^{(2)} \quad (49)$$

By solving Equation (49), as suggested above, for K, it is possible to produce exact numerical values of $\underline{K}$. However, in general, explicit formulae for $\underline{K}$ may not easily be found. Numerical solution for $\underline{K}$ is always possible, for some limit cases, $\underline{K}$ may also be approximated with good symbolic expressions in terms of node parameters.

$\mu_1>>\lambda$ Case

[7] Floor of $\not{K}$: the largest integer which is less than $\not{K}$.

Here, total excess capacity of the system is large enough that average queue size is small in the steady-state. The case is characterized by the inequality $a_1 >> 1$. We can talk about two sub-cases characterized by the relative value of a_2:

$a_2 << a_1$ **Case**

Server 1 will single handedly carry most of the node traffic. Server 2 is much slower than Server 1, thus, we expect that actual value of threshold ($\not{K}$) will be large:

$$\not{K} >> 1 \Rightarrow a_1^{-K-2} \cong 0$$

Therefore, Expression for $Y_{K+1}^{(1)}$ given in Equation (41) may be re-written as

$$Y_{K+1}^{(1)} \approx \frac{1}{a_1 - 1}\left[\frac{1}{z_2 C_e}\frac{z^{K+2}-1}{z^{K+3}-1} - \frac{1}{\mu_1 - \lambda}\right] + \frac{K+2}{\mu_1 - \lambda}$$

Here, in the last (approximately) equality, we have used $z \stackrel{\Delta}{=} z_1 / z_2$, and whose value happens to be as large as a_1's, for this case. Hence, value of the ratio involving z's powers, in the last equation, turns out to be approximately $1/a_1$. By using this approximate value, we can rewrite the last equality as

$$Y_{K+1}^{(1)} \approx \frac{1}{a_1 - 1}\left[\frac{1}{z_2 C_e}\frac{1}{a_1} - \frac{1}{\mu_1 - \lambda}\right] + \frac{K+2}{\mu_1 - \lambda} =$$

$$\frac{1}{\mu_1 - \lambda}\left[\frac{1}{z_2 a_1 (a_1 + a_2 - 1)} - \frac{1}{a_1 - 1}\right] + \frac{K+2}{\mu_1 - \lambda}$$

Further simplifications can be made for the approximation in the value of $Y_{K+1}^{(1)}$ by using, for this case, $a_1 + a_2 - 1 \cong a_1 - 1$ approximation:

$$Y_{K+1}^{(1)} \approx \frac{-1}{(\mu_1 - \lambda)(a_1 - 1)}\left[1 - \frac{1}{z_2 a_1}\right] + \frac{K+2}{\mu_1 - \lambda} = \frac{K+2-c}{\mu_1 - \lambda}$$

$$\text{where} \quad c \stackrel{\Delta}{=} \frac{1}{a_1 - 1}\left(1 - \frac{1}{z_2 a_1}\right) \cong \frac{1}{a_1}.$$

Here c, in a way, represents, approximately, the number of customers to whom the slower server helped to the faster server to be cleared out of the system. In the flow equivalent approach $c=0$ is the value turn out to be approximated (the flow equivalent means $c=0$!). This is justified by $a_1 >> 1$ [8].

Now, we should carry out a similar approach to find out the approximate value of $Z_K^{(2)}$ relevant to this case. Let us start with Equation (47) rewritten in the following form:

$$Z_K^{(2)} = \frac{1}{\mu_2} + \frac{c'}{C_e} \qquad \text{where } c' = \frac{f_{K+1}}{f_{K+2} - f_{K+1}}$$

by definition, which we have just made.

For the current case, i.e., $a_2 << 1 < a_1$, it turns out to be that

$$c' \approx \frac{1}{a_1 - 1}.$$

Once again, in flow equivalent analysis, this last value turns to be taken as 0. Now let us construct Equation (49) by using the above produced approximate values, we end up with:

$$Y_{\underline{K}}^{(1)} \approx \frac{\underline{K} + 1 - c}{\mu_1 - \lambda} = \frac{1}{\mu_2} + \frac{c'}{C_e} \approx Z_{\underline{K}}^{(2)}$$

By using the fact that, for this case, $C_e \approx \mu_1 - \lambda$ and the approximated values of c and c', we can solve for $\underline{K}$:

$$\underline{K} = \frac{\mu_1 - \lambda}{\mu_2} - 1 + c + c' \approx \frac{\mu_1 - \lambda}{\mu_2} - 1 + \frac{1}{a_1} + \frac{1}{a_1 - 1} \qquad (50)$$

This threshold value is greater than what the flow equivalent analysis approximate (where $c=c'=0$ is taken). For really large a_1 values, however, our results are the same as what the Flow equivalent produces for $\not{K}$. Then again, for not so large a_1 values, the difference between our approximation and the flow equivalent becomes non-trivial.

---

[8] For this case $z_2 \approx 1$ can be used. This is used in the last approximate value for c.

NUMERICAL EXAMPLE 1

Let us take this case: $\lambda=1$, $\mu_1=2$ and $\mu_2=0.3$, that is $C_e=1.3$, $a_1=2$, $a_2=0.3$. Our approximation produces

$$\underline{K} \approx \frac{\mu_1-\lambda}{\mu_2} - 1 + \frac{1}{a_1} + \frac{1}{a_1-1} = \frac{2-1}{0{,}3} - 1 + \frac{1}{2} + \frac{1}{2-1} = 3{,}833..$$

and $\not{K} = \lfloor \underline{K} \rfloor = 3$.

Flow equivalent analysis approximates these values as

$$\underline{K} \approx \frac{\mu_1-\lambda}{\mu_2} - 1 = \frac{2-1}{0{,}3} - 1 = 2{,}333...$$ and

$\not{K} = \lfloor \underline{K} \rfloor = 2$.

The exact numerical calculations give

$$Y_3^{(1)} = 3.29 < Z_3^{(2)} = 3.86 < Y_4^{(1)} = 4.27,$$

and thus $\not{K} = 3$.

$a_2 \approx a_1$ Case

Server 1 and 2 will together carry the node traffic. Server 2 is almost as fast as Server 1, thus, we expect that actual value of threshold ($\not{K}$) will be small:

$$\underline{K} \approx 0 ➔$$

$$c' = \frac{f_{K+1}}{f_{K+2} - f_{K+1}} \cong \frac{1}{a_1 + a_2 - 1 + \frac{a_2 - a_1 + 1}{2}} = \frac{2}{3a_2 + a_1 - 1}$$

where $Z_K^{(2)} = \frac{1}{\mu_2} + \frac{c'}{C_e}$

as in the previous case. Approximation is an exact equality in $K=0$ case. However $\not{K}=0$ is trivial: When $\not{K}>0$, it turns out to be that

$$c' \cong \frac{1}{a_1 + a_2 - 1} = \frac{\lambda}{C_e}$$

approximation becomes more fitting with the numerical exact solutions. With this last one, $Z^{(2)}_K$ approximation becomes

$$Z_K^{(2)} \cong \frac{1}{\mu_2} + \frac{\lambda}{C_e^2}$$

When it comes to approximate the value of $Y_{K+1}^{(1)}$, as in the previous case, we may start with

$$Y_{K+1}^{(1)} = \frac{K+2-c}{\mu_1 - \lambda}$$

formulation. Here, c corresponds to the number customers served by Server2 to help Server 1. For this case ($\mu_1 \approx \mu_2$), obviously, $c \approx 1$. Thus

$$Y_{\underline{K}}^{(1)} = Z_{\underline{K}}^{(2)} ➔ \quad \frac{K}{\mu_1 - \lambda} = \frac{1}{\mu_2} + \frac{\lambda}{C_e^2} ➔$$

$$\not{K} = \left\lfloor \frac{\mu_1 - \lambda}{\mu_2} + \frac{\lambda(\mu_2 - \lambda)}{C_e^2} \right\rfloor$$

This last equation predicts a bit higher threshold value than the flow equivalent does.

NUMERICAL EXAMPLE 2

Let us take this case: $\lambda=1$, $\mu_1=3$ and $\mu_2=2$, that is $C_e=4>>\lambda$ and, $a_1=3$, $a_2=2$. Assuming that $a_1 \approx a_2$ approximation is holding, our last approach
produces

$$\underline{K} \approx \frac{\mu_1-\lambda}{\mu_2} + \frac{\lambda(\mu_1-\lambda)}{C_e^2} = \frac{3-1}{2} + \frac{3-1}{16} = 1 + \frac{1}{8}$$

thus $\not{K} = \lfloor \underline{K} \rfloor = 1$.

Flow equivalent analysis approximates these values as

$$\underline{K} \approx \frac{\mu_1-\lambda}{\mu_2} - 1 = 0$$ and thus $\not{K} = \lfloor \underline{K} \rfloor = 0$

.

The exact numerical calculations give

$$Y_1^{(1)} = 0{,}551 < Z_1^{(2)} = 0{,}557 < Y_2^{(1)} = 1{,}034$$

, and thus $\not{K}=1$ which is the same result given in our approximation and 1 higher than what the flow equivalent approximation would predict.

4. CONCLUSION

In this article, we have presented a particular mini-max criterion to measure the performance of a traffic network at the node level. Policy for using an alternate route (link) for the packets which are destined to a certain network may be implemented as a threshold-checking on the size of this flows dedicated queue's size. Also "Random Early Detection" (RED) [7] and its derivatives use threshold on average queue size in order to decide whether to drop or accept an arriving packet to the node. Our analysis will be extended to find new policies in that extent. The cost function on which our criterion is based penalizes unbalanced usage of the link capacities at the node. By minimizing the cost function, we seek to clear all the queues in the node model to their destinations at the same time. Hence, by definition, this performance criterion is a good measure to model local congestion reduction. In other words, optimizing the controls which direct the traffic at this node to outgoing links produces a locally optimal congestion reducing policy. Reducing the accumulation of customers at the node in the most effective way also reduces the expected number of customers at the node. Thus, it reduces the average delay at this node (though, we do not claim our criterion is optimal in that sense). Finding the exact threshold values may be important when extremely un-balanced parameters are at hand. By using Z-transform analysis we have derived the analytical expressions that tie the optimum threshold value to the node-parameters.

In Section 1, we introduced the basic theorems and definitions to apply this optimization problem to M/M/2 queues. Without supplying the proof, we stated that the optimal policy is of the threshold type. We showed the approximate results of this optimization for a certain range of node parameters. The resulting threshold value is either identical to that produced by the minimum average delay (MAD) criterion or is a very close lower bound for it. However, the essential noteworthy feature of this application is the simplicity of its derivation. Thus, we have an approach to obtaining MAD criterion results that does not need the steady state probabilities and is easier to implement. Generalizing the results from a one-queue 2-server model to an N-queue 2-server model, and thus to 2-link nodes in an N-destination network, can be considered as the extension of this article. For M-link nodes in an N-destination network case (the most general local problem), a method to manipulate the result of this work needs to be developed. A purely, even mostly, mathematical method looks almost impossible in the context of an engineering work. From a mathematical point of view, on the other hand, results will probably be too cumbersome to apply to practical cases. However, for limited cases and for practical tests, our method is mathematically justified. Further expansion of this work towards M-link N-destination networks employing an a-priori set of rules looks feasible.

REFERENCES

[1] L. Kleinrock , *Queuing Systems, Vol. I: Theory,* John Wiley *fa* Sons,Inc. 1975.

[2] L. Kleinrock , *Queuing Systems, Vol. II: Theory,* John Wiley & Sons,Inc. 1976.

[3] M. K. Sis *A Dynamic Local Congestion Reducing Strategy Based on a Mini-Max Criterion,* Ph.D. Dissertation Thesis, Polytechnic University 1994.

[4] P.E. Sarachik, "Congestion Reducing Dynamic Routing Strategies for Multidesti-nation Traffic Networks", *IEEE Trans on Commun.,* Vol. COM-24, No. 4, April 1984.

[5] S.Ross, "Applied Probability Models with Optimization Applications", *(Chapter 6) Markov Decision Processes,* Holden -Day, San Francisco, 1970.

[6] W. Lin and P.R, Kumar, "Optimal Control of a Queuing System with Two Heterogeneous Servers", *IEEE Trans on Automat. Contr.* Vol. AC-29, No. 8, August 1984.

[7] Floyd S. and Jacobson V, "Random Early detection Gateways in congestion avoidance", *IEEE/ACM Transaction on Networking.* Vol. 1, No. 3, 1993

SHORT BIOGRAPHY

Dr. Sis has received his BS and MSc degrees in Electrical Engineering in 1978 and in 1980, respectively, both from Istanbul Technical University. In 1994, he has received his Ph.D. in EE from Polytechnic University in New York. After gradution he has been employed as Networking design consultant for several private companies. Since 2001 he has been working for Dokuz Eylül University in Computer Engineering department as a member of faculty. His major research interests are Netwok control, Performance analysis, Digital Audio Processing and Quantum Computation.