July 1989

DCIEM No. 89-RR-45

AD-A216 870

QUEUING THEORY APPROACH TO MODELLING THE DYNAMICS OF THE HUMAN-COMPUTER INTERFACE

J.M.E. van de Vegte

Defence and Civil Institute of Environmental Medicine
1133 Sheppard Avenue West
P.O. Box 2000
North York, Ontario
M3M 3B9

DEPARTMENT OF NATIONAL DEFENCE - CANADA

Table of Contents

Abstract

This paper studies a model of the human-computer interface based on the idea of a message passing from one partner (the human) to the other (the computer) by means of an arbitrary number of layers which represent levels of abstraction of the message. The model uses a queuing theory formulation and assumes Poisson arrival and service characteristics to describe the movement of packets through the interface and to develop a criterion for network stability. Simulations are included to illustrate some of the theoretical results obtained.

QUEUING THEORY APPROACH TO MODELLING THE DYNAMICS OF THE HUMAN-COMPUTER INTERFACE

1. Introduction

The human-computer interface may be modelled as a generator, a sender, a transmitter, and a receiver of information-carrying packets called messages, with feedback possible in the case of a message received in error. This set of operators is shown in Figure 1.1, where G is the message generator, E_S is the encoder of the message sender, T_F is the forward transmitter, D_R is the decoder of the message receiver, E_R is the encoder of the message receiver, T_R is the reverse transmitter, D_S is the decoder of the message sender, and O represents the point at which messages exit from the network. A message generated at G is coded in some manner at E_S and then transmitted. Subsequent messages arriving at E_S (or any other node) must wait for the current message to be completely served. Messages that are received at D_R are decoded. If they are understood, they pass to O and leave the system. If there is ambiguity, error messages are fed back to the sender via the feedback path: First, an error message coded at E_R is transmitted to D_S where it is decoded. Then a new message is coded at E_S to resolve the ambiguity, and the cycle repeats itself.

As an example from an actual human-computer interchange, imagine a person sitting at a terminal who wishes to change the computer directory in which he is working. He codes and transmits this intention by typing on his keyboard: *cd filex* (change directory to the directory called *filex*). In fact, this person only has two directories available to him: *mail* and *files*. Therefore, the message received by the computer is erroneous and the user's command is not carried out. To correct the problem, the computer gives feedback by coding and transmitting to the screen the message: *No such directory. Did you mean "cd files"?*. The person responds with *yes* and finally the original message successfully reaches the computer. The working directory is changed to *files*.

In this type of scenario, new messages must wait for service at each service centre (or "node") in Figure 1.1, and the service at each node takes some finite amount of time to perform. In view of these facts, a queuing theory model seems a natural choice. By using such a model it becomes possible to estimate how many messages must wait for service at a given node and how long they must wait. Thus, a queuing theory model of a human-computer interface may to some extent allow prediction of the conditions which can lead to communication breakdown. Queuing theory analysis of networks of queues is difficult, though, and the general network flow problem is of current research interest. However, there are nice results for special cases, one of the most interesting being the work of Jackson (1).

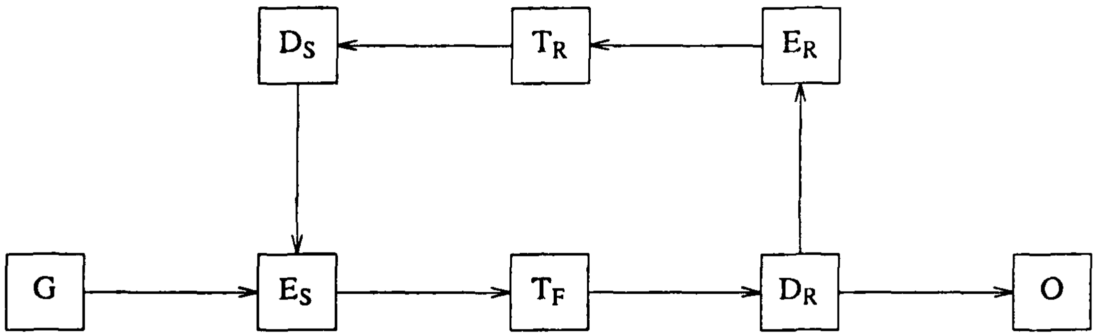

Figure 1.1 Model of the Human-Computer Interface

In Jackson's terminology, a network consists of multiple nodes whose interconnections describe the progress of tasks, packets, messages or other entities through the network. The following description characterizes an "open" Jackson network (paraphrased from Giffin (2)):

1. The network contains more than one service centre (or "node").
2. Each service centre may contain multiple servers, allowing multiple queues, with each server at the centre having an identical exponential service time distribution. A server is assumed to process only one message at a time.
3. Arrivals at any given node may come from outside the system, or from other nodes in the network.
4. Arrivals from outside the network occur in Poisson fashion.
5. When a message completes service at a particular centre, it may leave the system, or be routed to another centre, its path being controlled by a fixed probability distribution associated with the node it is leaving.
6. Queues of unlimited length are allowed at each node.

7. Total arrival rate at each node must be less than the potential service rate at that node.

Note that feedback from a network node to a previous node is permitted. (Closed Jackson networks imply that a message never leaves the network.)

It is possible to formulate the model of the human-computer interface in terms of an open Jackson network, given that the distributions of arrival and service times may reasonably be assumed to be exponential. (This assumption must be tested experimentally for real interfaces.) The message error rate, E, is modelled by requiring that a proportion E of the messages entering the network be fed back from D_R to be corrected, while a proportion (1-E) leaves the network after first arrival at D_R.

When Poisson/exponential streams are assumed, three interesting properties apply (2):

1. Combining independent Poisson streams produces a Poisson stream, with a mean arrival rate equal to the sum of the means of the arrival rates of the input streams. If two Poisson streams are described by

$$p(X_1 = x) = e^{-\lambda_1} \frac{\lambda_1^x}{x!}$$

$$p(X_2 = x) = e^{-\lambda_2} \frac{\lambda_2^x}{x!}$$

then their characteristic functions are (Fisz (3, 108-112)) (the notation E[.] refers to mathematical expectation and is not related to E the message error rate):

$$\phi_1(t) = E\,[e^{itX_1}] = e^{[\lambda_1(e^{it} - 1)]}$$

$$\phi_2(t) = E\,[e^{itX_2}] = e^{[\lambda_2(e^{it} - 1)]} \quad .$$

Since the streams are independent, the characteristic function of $X = X_1 + X_2$ is

$$\phi(t) = E\,[e^{itX}] = E\,[e^{itX_1}]\,E\,[e^{itX_2}]$$

$$= e^{[\lambda_1(e^{it} - 1)]}\, e^{[\lambda_2(e^{it} - 1)]}$$

$$= e^{[(\lambda_1 + \lambda_2)(e^{it} - 1)]}$$

which is the characteristic function of a Poisson process of mean $\lambda_1 + \lambda_2$.

2. Splitting a Poisson stream of mean arrival rate λ (events per unit time) into several streams, each with associated probability r_i, produces output streams with mean rates $r_i\lambda$. This property may be demonstrated using the multinomial distribution (2, 227). Suppose a Poisson stream splits into b paths and the probability of a message taking path i is r_i. Then after m messages have entered the original stream, the probability that m_1 messages have passed to path 1, m_2 messages, to path 2, and so on, is (according to the multinomial distribution) described by

$$p(m_1, \ldots, m_b \mid m) = \frac{m!}{m_1! \ldots m_b!} r_1^{m_1} \ldots r_b^{m_b}$$

where $\sum_{i=1}^{b} m_i = m$ and $\sum_{i=1}^{b} r_i = 1$. The Poisson probability of getting m messages after t seconds if the mean rate of arrival is λ is

$$p(m) = \frac{(\lambda t)^m e^{-\lambda t}}{m!}$$

Therefore,

$$p(m_1, m_b, m) = \frac{m!}{m_1! \ldots m_b!} r_1^{m_1} \ldots r_b^{m_b} \frac{(\lambda t)^m e^{-\lambda t}}{m!}$$

$$= \prod_{i=1}^{b} \frac{r_i^{m_i}}{m_i!} (\lambda t)^{m_i} e^{-r_i \lambda t} \quad .$$

This joint distribution is a product of Poisson distributions, so the b output streams are each Poisson with mean rates $r_i\lambda, i=1, \ldots, b$.

3. An exponential server operating on a Poisson stream produces a departure process which has the same statistical parameters as the input process. To see this, consider a node with a Poisson arrival stream of mean λ and an exponential service distribution of mean μ, with the density function of the arrival stream denoted by a(t) and service stream, by s(t). Then the density function for interarrival times is, by definition,

$$a(t) = \lambda e^{-\lambda t}$$

and, for the service times,

$$s(t) = \mu e^{-\mu t} \quad .$$

Suppose a message leaving the node leaves behind other messages in a queue. Then the time until the next message leaves will be exponentially distributed due to the service time at the node. Therefore, the Laplace transform of the departure time distribution, d(t), given a non-zero queue at the node, is

$$L(d(t) \mid m>0) = \frac{\mu_1}{\mu_1 + s} \quad .$$

If, on the other hand, the original message leaves no subsequent messages in a queue, then the next departure from the node will not occur until an arrival has occurred, and the arrival has been serviced. Thus, in this case, the Laplace transform of the distribution of departure times is

$$L(d(t) \mid m=0) = \left[\frac{\lambda}{\lambda + s}\right]\left[\frac{\mu}{\mu + s}\right] \quad .$$

From the properties of the node (Gross and Harris (4, 58-73)),

$$pr(m>0) = \frac{\lambda}{\mu}$$

and

$$pr(m=0) = 1 - \frac{\lambda}{\mu} \quad .$$

Combining the two cases gives

$$L(d(t)) = L(d(t) \mid m>0)\, pr(m>0) + L(d(t) \mid m=0)\, pr(m=0)$$
$$= \frac{\lambda}{\lambda + s} \quad .$$

Inverting this Laplace transform gives

$$d(t) = \lambda e^{-\lambda t}$$

which is the same interarrival time distribution as at the first node. Exponential service has not altered the Poisson stream.

Using the method of steady-state balance equations, Jackson (1) was able to show that the joint probability distribution of a network of queues could be expressed as a product of marginal M/M/c distributions. (The notation refers to input distribution/service distribution/number of servers per node. In this case, M refers to the exponential distribution, where

M stands for Markovian, and c indicates that there are c servers per node.) This result implies that the network acts as if each node were an independent M/M/c queue, with parameters λ_i (arrival rate at node i) and μ_i (service rate at node i). When the flow is strictly feed-forward, the properties of stream combination and splitting described above apply, and all of the internal flows in the network are in fact Poisson. However, when there is feedback in the network, the internal flows are not Poisson. Jackson's result is interesting because, whether the internal flows are Poisson or not, the network still behaves as if its nodes are independent M/M/1's.

Appendix 1 contains the details of Jackson's equations for queue length and waiting time. The result that is important for this paper is that, when only a single server is available at any node in the network, the expected queue length at node i is

$$L_{q_i} = \frac{k_i^2}{1 - k_i} \tag{1.1}$$

where $k_i = e_i/\mu_i$, e_i is the effective arrival rate at node i from all sources and μ_i is the service rate at node i.

2. Network Instability

When network conditions are such that increasingly large queues form at the service nodes in the network, the chance of instability or blocking increases. Suppose λ is the arrival rate at a given node and μ is the service rate at that node. Then there are in general three cases to consider for stationary problems: $\mu>\lambda$, $\mu=\lambda$, $\mu<\lambda$.

Case 1: $\mu>\lambda$

This is the standard case in which service rate exceeds arrival rate at a given node, and the (finite) equilibrium queue may be computed as described in the previous section (see equation (1.1)).

When queues are fairly long, it is possible to obtain approximate results for cases 1, 2 and 3 by applying a diffusion approximation, described in detail in Newell (5). The diffusion approximation employs a differential equation formulation which allows a description of a system's local behaviour, and which facilitates the treatment of boundary conditions. In particular, applying the approximation to a physical system requires that the system be smooth in the sense that, for a small change in time, the position (or other quantity of interest) changes from x to x+dx, with dx arbitrarily small. In queuing theory applications, x represents queue length, which must be integer. Thus, it is not possible to speak of infinitesimal queue changes. It is possible, however, to postulate a coarse scale of measurement for queue length on which only small changes in queue length occur during small changes in time. Naturally, in order to allow such a scale to be used, queue lengths must be relatively large.

The equation

$$Q(t+\tau) - Q(t) = [A(t+\tau) - A(t)] - [D(t+\tau) - D(t)] \tag{2.1}$$

where Q(t) is the queue length at time t, A(t) is the cumulative number of arrivals at time t, and D(t) is the cumulative number of departures at time t describes the evolution of the queue in time, assuming that arrival and departure processes do not depend on queue length, when the queue is not empty. To allow (2.1) to be formulated as a differential equation, the queue length Q(t) must be measured on a sufficiently coarse scale as described above. That is, τ must be small enough that $Q(t+\tau) - Q(t)$ is very small (on the coarse scale of measurement), yet sufficiently large that this queue length change is large relative to the individual message (integer) scale of measurement. If many individual events do take place between t and $t+\tau$, $Q(t+\tau) - Q(t)$ may be taken to be approximately normally distributed for any reasonable arrival and departure process (5, 103). In fact, as detailed in Newell (5), using the diffusion approximation produces the time-dependent diffusion equation

$$\frac{\partial f(x,t)}{\partial t} = -[\lambda(t) - \mu(t)]\frac{\partial f(x,t)}{\partial x} + \frac{[\lambda(t) + \mu(t)]}{2}\frac{\partial^2 f(x,t)}{\partial x^2} \tag{2.2}$$

where f(x,t) is the probability density of Q(t) at x, $\lambda(t)$ is the arrival process, and $\mu(t)$ is the service process. Manipulations of equation (2.2) under the various conditions of cases 1, 2 and 3 provide the following mean queue lengths (5).

Case 2: $\mu=\lambda$

Use of the diffusion approximation for this case gives (5)

$$L_q(t) = E[Q(t)] = \left[\frac{2t\,(\lambda + \mu)}{\pi}\right]^{1/2} . \tag{2.3}$$

Case 3: $\mu<\lambda$

In this case (5)

$$L_q(t) = E[Q(t)] = Q_o + t + \frac{1}{2}e^{-Q_o} \tag{2.4}$$

where Q_o is the initial queue.

Applying the diffusion approximation to case 1 provides an indication of its merit. Newell finds that for $\mu>\lambda$ the expected queue is

$$L_q = \frac{1+\lambda/\mu}{2(1-\lambda/\mu)} \quad .$$

Letting $k=\lambda/\mu$,

$$L_q = \frac{1+k}{2(1-k)} \quad , \tag{2.5}$$

which is the diffusion approximation corresponding to equation (1.1). As Figure 2.1 shows, the diffusion approximation is excellent for case 1.

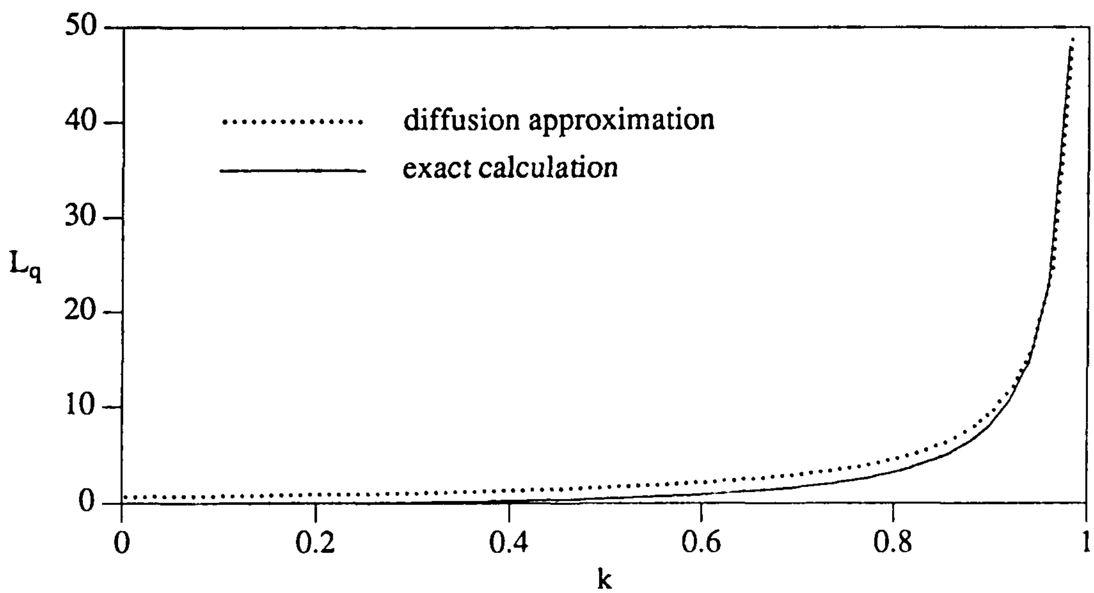

Figure 2.1 Diffusion Approximation of the Steady-State Queue

In cases 2 and 3, where $\mu \le \lambda$, the queues will grow without bound as time increases. In case 2, the growth is proportional to $t^{1/2}$; in case 3, to t. In fact, even in case 1, queue lengths become very large when $k \rightarrow 1$. In a network such as the one depicted in Figure 1.1, the accumulation of a large queue at any one node affects the performance of the entire network. In fact, a complete cessation of communication occurs when messages are blocked at a node because large queues have formed there, effectively severing the communication channel. This circumstance is referred to as instability or blocking of the network, which may occur whenever the arrival rate is greater than or equal to the service rate ($k \geq 1$) at any node in the network.

3. Single Layer Case

Consider a cyclic network like the one in Figure 1.1, but which neglects transmission delays. This network, shown in Figure 3.1, has a message arrival rate λ and an error rate E. The error rate E is the probability that a message received at node 2 will be in error and will require feedback for correction. In this case, the message passes to node 3. Consequently, there is a (1-E) probability that a message will be received correctly at node 2 and will leave the network after only a single transmission. The node numbers represent the following service centres.

Node Number	Node Name
1	Source Encoder
2	Receiver Decoder
3	Receiver Encoder
4	Source Decoder

As indicated in section 2, instability is considered to occur when the steady-state queue becomes infinite, for *any* of the nodes in the network. In Figure 3.1, packet flow rates are given for the network of Figure 1.1. Total flow into a node equals flow out of that node, so, as Figure 3.1 shows, the effective message flow into the network nodes is either $\frac{\lambda}{1-E}$ or $\frac{E\lambda}{1-E}$. Thus instability may occur whenever any of the following inequalities hold (assuming that the service rate at each node exceeds the effective arrival rate):

$$k_1 = \frac{\lambda}{\mu_1(1-E)} \geq 1$$

$$k_2 = \frac{\lambda}{\mu_2(1-E)} \geq 1$$

$$k_3 = \frac{E\lambda}{\mu_3(1-E)} \geq 1$$

$$k_4 = \frac{E\lambda}{\mu_4(1-E)} \geq 1$$

By solving for E, the condition on E for possible network instability may be calculated.

$$E \geq \min\left[1-\frac{\lambda}{\mu_1},\ 1-\frac{\lambda}{\mu_2},\ \frac{\mu_3}{\mu_3+\lambda},\ \frac{\mu_4}{\mu_4+\lambda}\right] \tag{3.1}$$

where μ_i, i=1, 2, 3, 4 are the service rates for each of the nodes, and λ is the external message arrival rate.

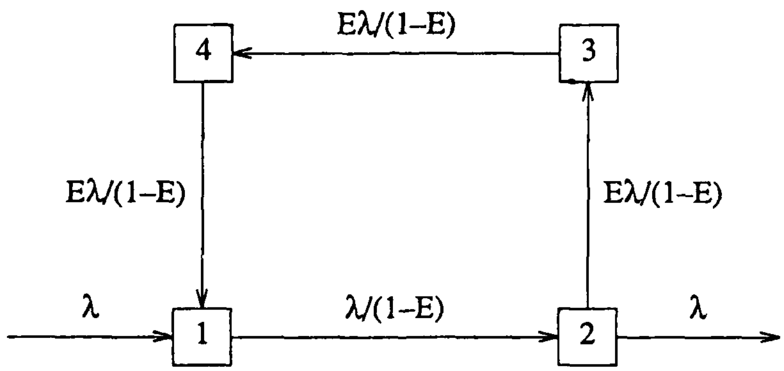

Figure 3.1 Queuing Model of a Single Layer Interface

4. Multi-Layer Case

It is possible to generalize the single layer result. Figure 4.1 shows a multi-layer version in which virtual high level messages are in reality sent via lower level transport processes. For example, sentences are actually passed between partners by means of words, words by phonemes, and phonemes by sound waves. (The more general construction, corresponding to Taylor's Layered Protocols framework (Taylor (6)), will be considered below in section 7.)

Following the same type of derivation as above, and using the same notation, gives the following result, which characterizes the region over which instability can occur at the interface, in terms of the probability of error at the top level receiver decoder.

$$E \geq \min\left[\min_{i \in n_f}\left[1-\frac{\lambda}{\mu_i}\right], \min_{j \in n_r}\left[\frac{\mu_j}{\lambda+\mu_j}\right]\right] \tag{4.1}$$

where n_f is the number of nodes in the forward transmission message path, and n_r is the number of nodes in the reverse (feedback) transmission message path.

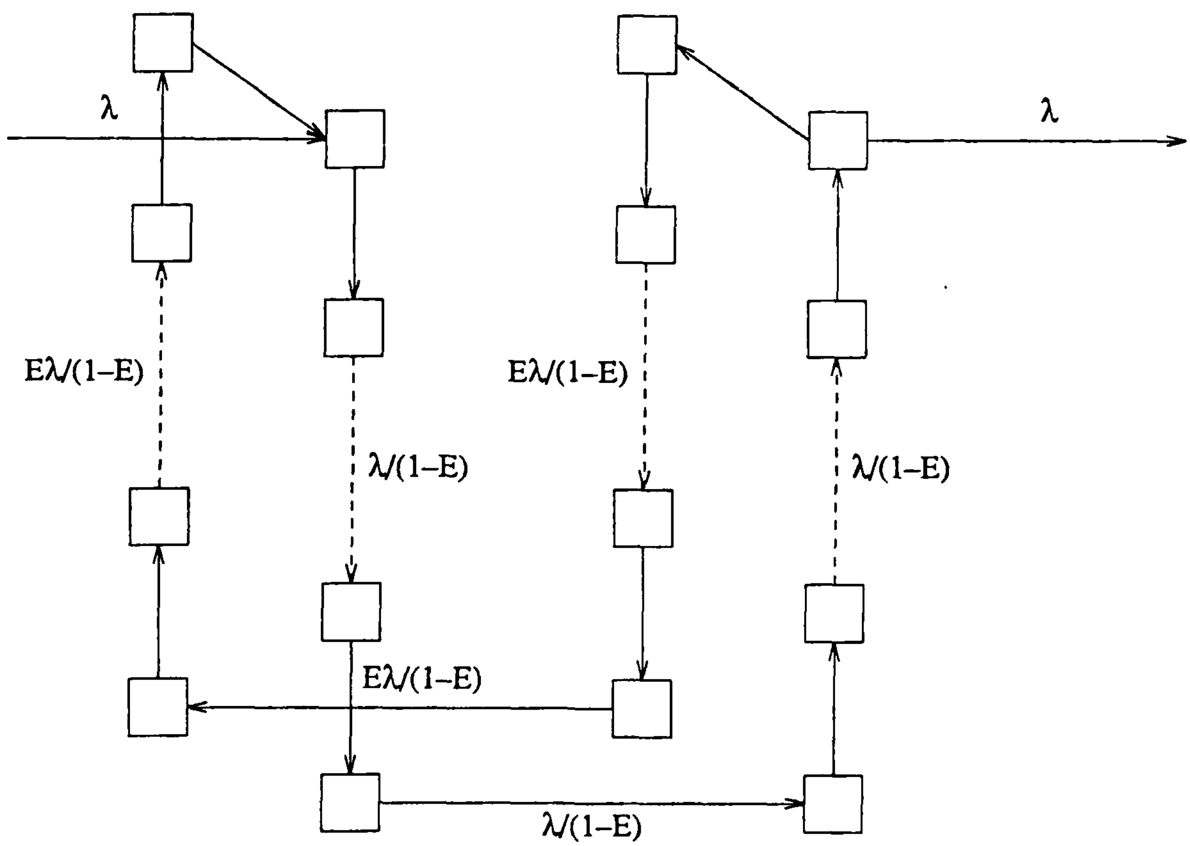

Figure 4.1 Queuing Model of a Multi-Layer Interface

5. Special Case - Nodes with Equal Service Rates

When all the nodes in the network of Figure 4.1 have equal service rates, that is, $\mu_i = \mu, i = 1, \ldots, N$, the condition for instability (equation (4.1)), simplifies to

$$E \geq \min\left[1-\frac{\lambda}{\mu}, \frac{\mu}{\lambda+\mu}\right].$$

The two quantities $1 - \frac{\lambda}{\mu}$ and $\frac{\mu}{\mu+\lambda}$ may be compared directly (by using a common denominator of $\mu(\mu + \lambda)$) to show that $\frac{\mu}{\mu+\lambda} \geq 1 - \frac{\lambda}{\mu}$, so for the special case in which all nodes in the network have the same service rates, the condition for instability is $E \geq 1 - \lambda/\mu$. Note that when $\mu \leq \lambda$, large queues are guaranteed to form, making blocking a certainty. Figure 5.1 shows the regions of stability and instability for this type of network. The surfaces delimiting the region of instability include: a flat surface at E=0 for $\mu \leq \lambda$, and a curved surface for $\mu > \lambda$. Below these limiting surfaces is a region of network stability; above them, instability may occur.

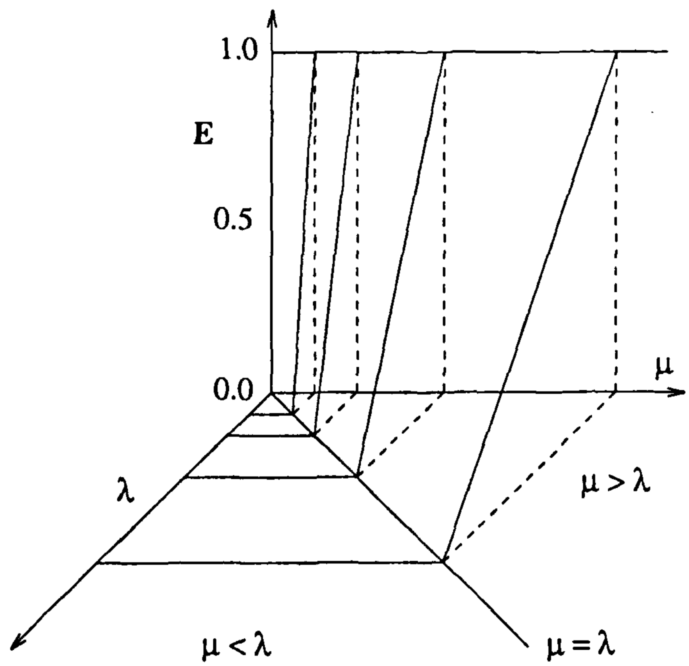

Figure 5.1 Regions of Instability for Networks of Nodes with Equal Service Rates

The shape of the stability regions may be more obvious in Figures 5.2 and 5.3, where cuts at constant service and arrival rates are shown. First, at cross-sections of constant μ (Figure 5.2), the criterion for instability decreases linearly with λ. That is, for a given service rate, the higher the arrival rate, the more easily instability can occur.

Slices of Figure 5.1 at constant λ are shown in Figure 5.3. The curves increase rapidly when μ is only slightly larger than λ, but then increases very slowly for $\mu \gg \lambda$, limiting at E=1 for $\mu \rightarrow \infty$. Thus, for a given arrival rate, the higher the service rate, the less likely instability will occur, but only minor improvements are obtained by raising μ beyond about four to eight times λ.

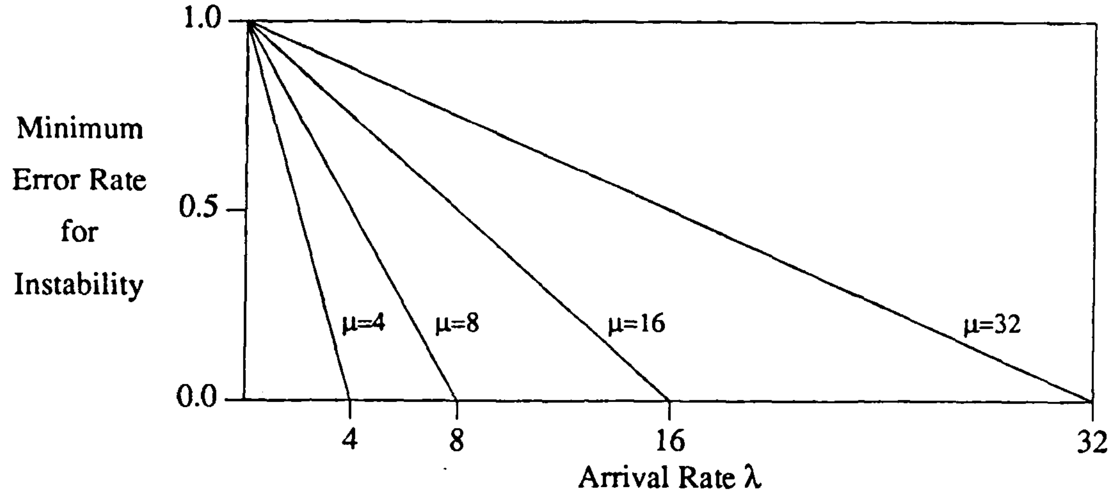

Figure 5.2 Cross-Sections of Figure 5.1 at Constant Service Rates

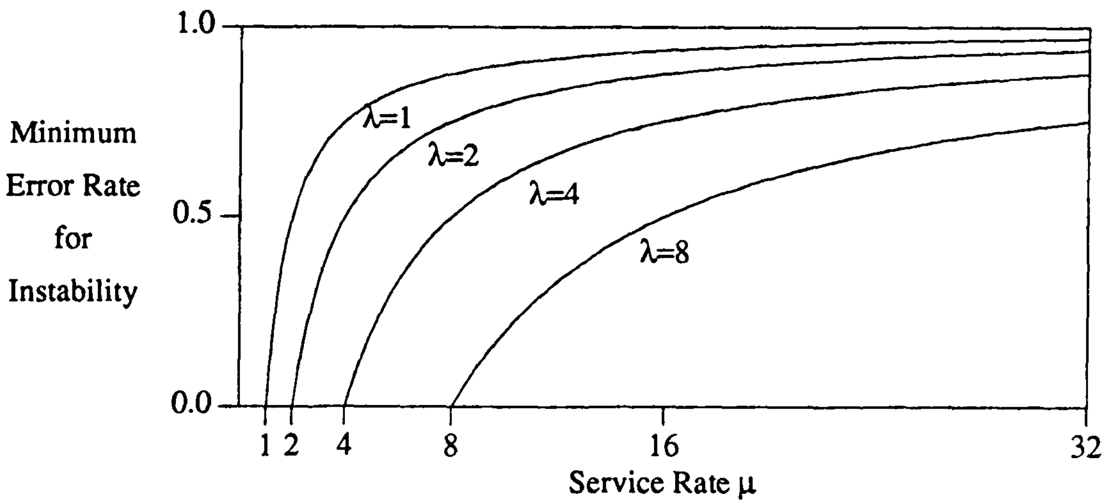

Figure 5.3 Cross-Sections of Figure 5.1 at Constant Arrival Rates

6. Feedback at Lower Levels - Simple Case

In order to avoid problems of instability, it is normally good practice to eliminate all but the simplest and fastest feedback at the lower levels, where messages may be broken down into words, gestures, or letters. For example, if a slow complicated error correction scheme is used at the word level, several new words will be missed each time a word needs to be corrected. This creates new errors and the number of words requiring correction increases dramatically. Ultimately blocking occurs, and no more higher level messages (such as sentences) are able to pass across the interface. The greater the delays at these lower levels, the greater the delays perceived by the operator at the highest levels, and the longer the tails of the exponential distribution describing the overall system delay, the greater the chance for instability. This may be seen from the following expression for the probability distribution of the total waiting time required by a message to pass completely through the network, due to Nelson (7). Nelson gives the results for the multi-channel case. Here, they are simplified to the single channel case, that is, one server per node. The quantity L_j is the probability that there are one or more messages at node j and hence that new arrivals will have to wait. The cumulative probability distribution is

$$P\left[\sum_{j=1}^{N} x_j \le t\right] = 1 - \sum_{j=1}^{N} A_j e^{B_j t} , \tag{6.1}$$

where e_j is taken from Appendix 1,

$$B_j = e_j - \mu_j \tag{6.2}$$

$$L_j = \frac{e_j}{\mu_j} \frac{1}{(1-e_j/\mu_j)^2} \tag{6.3}$$

$$A_j = L_j \left[\prod_{\substack{k=1 \\ k \ne j}}^{N} \frac{1 - L_k B_j}{B_j - B_k}\right], \quad j = 1,\dots, N . \tag{6.4}$$

When service rates are high, $\mu_j \gg e_j$, the B_j become very large negative numbers, shortening the tails of the probability distribution described by equation (6.1). As the service at lower levels becomes poorer, that is, the service rate is not much greater than the arrival rate, the B_j tend towards zero, so the probability that the total waiting time is less than some given time t becomes very small (see equation (6.1)). Thus, the tail of the cumulative probability distribution becomes longer, and the average time spent in the network becomes greater. At the highest levels, longer delays increase the chances of creating queues at the service nodes, and hence the chance of interface instability, because the stability thresholds for the error rates drop.

The following derivation treats the case of a two level network, with feedback at both layers. Errors are now given a subscript corresponding to the number of the layer with which they are associated. The error parameters represent the chance that an error will occur at that level, or equivalently, the chance that a detected error will be dealt with at that layer's level of description, for example, words or letters. Thus, at the top level (layer 1), E_1 is the probability that a message will be found to be in error at node 2 and will be fed back via node 3. At the lower level (layer 2), E_2 is the probability that a message received in error at node 6 will be corrected by feedback at layer 2 and will be passed directly to node 7. Figure 6.1 depicts the construction of this system. The nodes shown in the figure are defined as follows:

Node Number	Layer	Node Name
1	1	Source Encoder
2	1	Receiver Decoder
3	1	Receiver Encoder
4	1	Source Decoder
5	2	Source Encoder
6	2	Receiver Decoder
7	2	Receiver Encoder
8	2	Source Decoder

The internal flows are:

From Node	To Node	Flow
1	5	$\frac{\lambda}{1-E_1}$
5	6	$\frac{\lambda}{(1-E_1)(1-E_2)}$
6	2	$\frac{\lambda}{1-E_1}$
2	3	$\frac{E_1\lambda}{1-E_1}$
3	7	$\frac{E_1\lambda}{1-E_1}$
7	8	$\frac{E_1\lambda}{1-E_1} + \frac{E_2\lambda}{(1-E_1)(1-E_2)}$
8	4	$\frac{E_1\lambda}{1-E_1}$
4	1	$\frac{E_1\lambda}{1-E_1}$
8	5	$\frac{E_2\lambda}{(1-E_1)(1-E_2)}$
6	7	$\frac{E_2\lambda}{(1-E_1)(1-E_2)}$

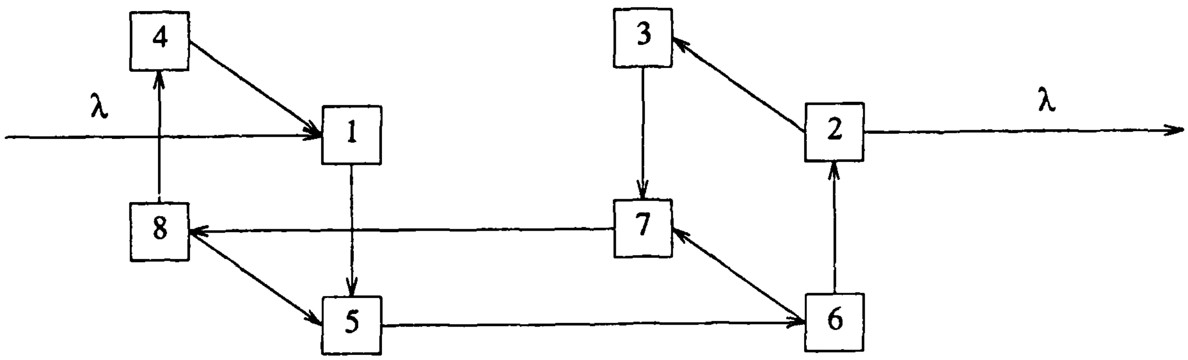

Figure 6.1 Queuing Model of a Two-Layer Interface with Feedback at Each Level

As indicated in section 2, the conditions for instability are determined by using the fact that flow into a node equals flow out of the node to compute the ratio of effective arrival rate to service rate for each node. Instability may occur when $k_i = e_i/\mu_i \geq 1$ for any node i. Note that conditions on both E_1 and E_2 must be met in this case. The first four nodes, at the top level, involve only conditions on E_1, while the last four nodes, on the bottom level, set restrictions on both E_1 and E_2. In other words, the performance criteria for the messages of interest (at the top level) impose restrictions on the behaviour of the supporting channels.

Node 1:

$$k_1 = \frac{e_1}{\mu_1} = \frac{\lambda}{\mu_1(1-E_1)} \geq 1$$

$$\frac{\lambda}{\mu_1} \geq 1-E_1$$

$$\boxed{E_1 \geq 1-\frac{\lambda}{\mu_1}}$$

Node 2:

$$\boxed{E_1 \geq 1-\frac{\lambda}{\mu_2}}$$

Node 3:

$$\frac{E_1\lambda}{\mu_3(1-E_1)} \geq 1$$

$$E_1\lambda \geq \mu_3-\mu_3E_1$$

$$E_1(\lambda+\mu_3) \geq \mu_3$$

$$\boxed{E_1 \geq \frac{\mu_3}{\mu_3+\lambda}}$$

Node 4:

$$\boxed{E_1 \geq \frac{\mu_4}{\mu_4+\lambda}}$$

Node 5:

$$\frac{\lambda}{\mu_5(1-E_1)(1-E_2)} \geq 1$$

$$\frac{\lambda}{\mu_5} \geq 1-E_1-E_2(1-E_1)$$

$$E_1+E_2(1-E_1) \geq 1-\frac{\lambda}{\mu_5}$$

$$E_2(1-E_1) \geq 1-\frac{\lambda}{\mu_5}-E_1$$

$$E_2 \geq \frac{1}{1-E_1}(1-\frac{\lambda}{\mu_5}-E_1)$$

$$\boxed{E_2 \geq 1-\frac{\lambda}{\mu_5(1-E_1)}}$$

Node 6:

$$\boxed{E_2 \geq 1-\frac{\lambda}{\mu_6(1-E_1)}}$$

Node 7:

$$\frac{E_2\lambda}{\mu_7(1-E_1)(1-E_2)}+\frac{E_1\lambda}{\mu_7(1-E_1)} \geq 1$$

$$\frac{E_2\lambda+E_1\lambda(1-E_2)}{\mu_7(1-E_1)(1-E_2)} \geq 1$$

$$\frac{\lambda(E_1+E_2(1-E_1))}{\mu_7(1-E_1)(1-E_2)} \geq 1$$

$$\frac{E_1+E_2(1-E_1)}{(1-E_1)(1-E_2)} \geq \frac{\mu_7}{\lambda}$$

$$E_2(1-E_1)+E_1 \geq \frac{\mu_7(1-E_1)}{\lambda}-\frac{\mu_7(1-E_1)E_2}{\lambda}$$

$$E_2\left[1-E_1+\frac{\mu_7(1-E_1)}{\lambda}\right] \geq \frac{\mu_7(1-E_1)}{\lambda}-E_1$$

$$\boxed{E_2 \geq \frac{\mu_7(1-E_1)-\lambda E_1}{\mu_7(1-E_1)+\lambda(1-E_1)}}$$

Node 8:

$$\boxed{E_2 \geq \frac{\mu_8(1-E_1)-\lambda E_1}{\mu_8(1-E_1)+\lambda(1-E_1)}}$$

The stability of the network in Figure 6.1 may be tested as follows: If the condition

$$E_1 \geq \min\left[1-\frac{\lambda}{\mu_1},\ 1-\frac{\lambda}{\mu_2},\ \frac{\mu_3}{\mu_3+\lambda},\ \frac{\mu_4}{\mu_4+\lambda}\right]$$

holds, then the network is unstable. Otherwise, test the following condition on E_2:

$$E_2 \geq \min\left[1-\frac{\lambda}{\mu_5(1-E_1)},\ 1-\frac{\lambda}{\mu_6(1-E_1)},\ \frac{\mu_7(1-E_1)-\lambda E_1}{\mu_7(1-E_1)+\lambda(1-E_1)},\ \frac{\mu_8(1-E_1)-\lambda E_1}{\mu_8(1-E_1)+\lambda(1-E_1)}\right]$$

If the condition is true, then the network is unstable. If both of the above conditions are false, the network of queues is stable.

It is interesting to observe how the restrictions on E_2 depend on those on E_1 (for nodes 5 through 8). For nodes 5 and 6, the form of the restriction on E_2 is

$$E_2 \geq 1 - \frac{\lambda}{\mu(1-E_1)} .$$

For nodes 7 and 8, the form is

$$E_2 \geq \frac{\mu(1-E_1)-\lambda E_1}{\mu(1-E_1)+\lambda(1-E_1)} \quad \text{or} \quad E_2 \geq \frac{1 - \dfrac{\lambda}{\mu(1/E_1-1)}}{1+\lambda/\mu} .$$

For these nodes, as E_1 increases, the limit on E_2 decreases, and, as E_1 decreases, the limit on E_2 increases. Thus, given that E_1 does not itself cause network instabilities, the higher the error rate at the top level, the more restricted the range of E_2 over which stability may be maintained. That is, the higher E_1, the more easily instability may occur in the network. Conversely, the lower E_1 is, the less stringent the restrictions on E_2 to allow stability to be maintained. Intuitively, there exists at the highest level a need to pass (virtual) messages at a certain rate and with a certain error probability. To support this performance requirement, the lower level channels must obey restrictions which depend on the arrival, service, and error rates.

Note that in real interfaces, large error rates at lower level channels would be expected to cause large error rates at the higher level channels. For example, if many of the characters being typed at a keyboard are received incorrectly by a computer, it is likely that very few commands (which are made up of these characters) will be correctly interpreted.

7. Feedback at Lower Levels - General Case

Figure 7.1 shows the general case for networks of queues involving feedback at lower levels. For each level i, the error rate E_i represents the probability that a message is received in error and is corrected by feedback at that level. Table 7.1 gives the general expressions for the network flows, $k = 1, ..., n$, where n is the number of levels. At the lowest level, the flow from source side to receiver side follows case III, and the flow from receiver side to source side, case II.

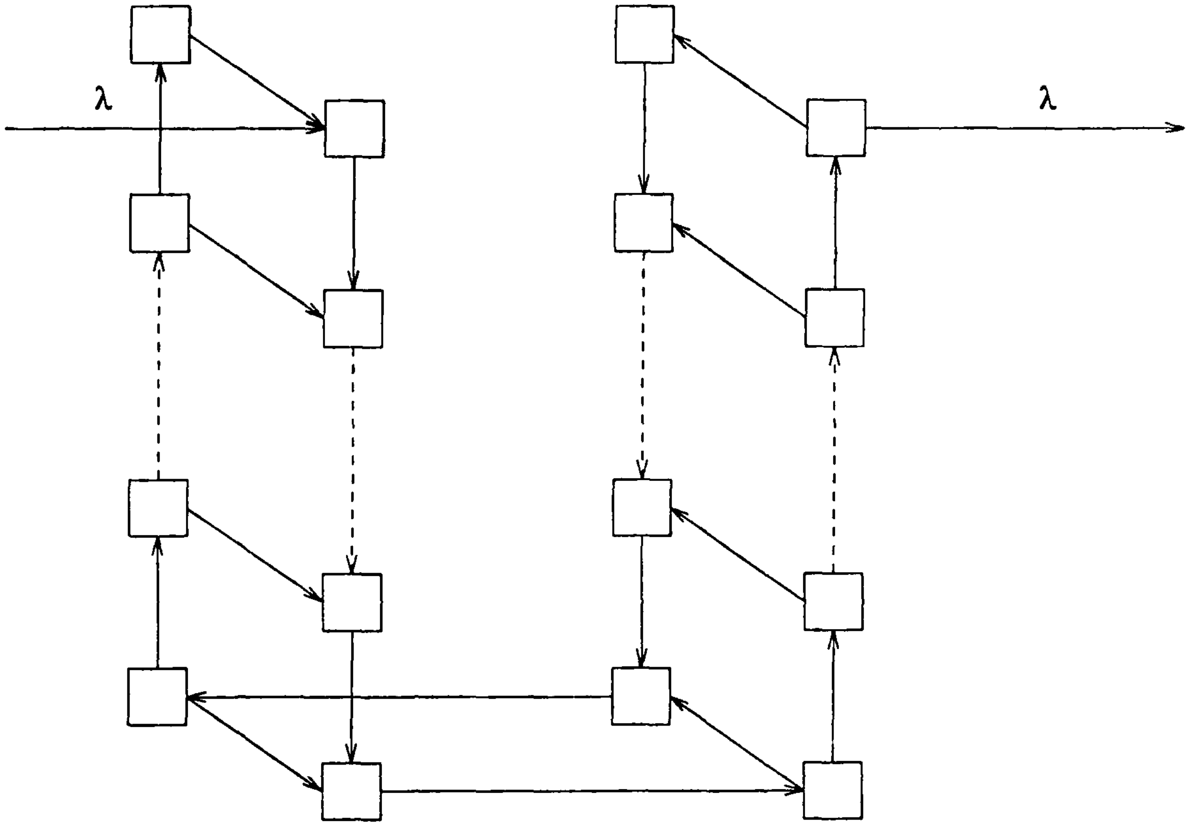

Figure 7.1 Queuing Model of a Multi-Layer Interface with Possible Feedback at Every Level

Case	Source Side	Receiver Side	Flow
I	Across level k	Across level k	$\frac{E_k \lambda}{(1-E_1)...(1-E_k)}$
II	Up into level k	Down out of level k	$\lambda\left[\frac{E_1}{1-E_1}+...+\frac{E_k}{(1-E_1)...(1-E_k)}\right]$
III	Down out of level k	Up into level k	$\frac{\lambda}{(1-E_1)...(1-E_k)}$

Table 7.1 Network Flows for Figure 8.1

With k_i equal to e_i/μ_i as usual, the network may be tested for potential instability by checking for the cases in which $k_i \geq 1$. As usual, the most stringent restriction on the error rates will be the one which characterizes the stability boundary for the network.

To test for possible instability, start at the highest level, and evaluate k_1. The conditions in this instance depend only on E_1. If stability is indicated at this level, continue to level two. At this level, the condition on E_2 appears as a function of E_1. If the conditions for instability are not met, continue to the subsequent layers, where the new tests will be functions of the preceding E_i. The network is stable if none of the E_i, i=1,..., n, satisfy the conditions for instability.

8. Supporting Simulations

When network conditions are such that equilibrium flows and network stability exist, steady-state queues may be calculated using equation (1.1). When the arrival rate at some node becomes equal to the service rate, equation (2.3) should apply, and when the arrival rate exceeds the service rate, equation (2.4) should be used. As noted, however, equations (2.3) and (2.4) may be used only when very large queues are being considered, which is not typically the case for the movement of packet messages through an interface. Thus, during the simulation of this application, average maximum queues are compared to average final queues. When the maximum queues occur on average at the end of the simulations, the network is deemed to be unstable, because queues at network nodes are growing with time. Illustrative example simulations, produced using the IBM PC simulation package Micro-Saint, are given in the following sections, in which averages were taken over 100 trials of 100 seconds each.

8.1 Example of Single Layer Interface

As an example for section 3, the case of the single layer interface, the following arrival and service rates were used: $\lambda = 2.0$, $\mu_1 = 4.0$, $\mu_2 = 3.5$, $\mu_3 = 4.0$, and $\mu_4 = 0.5$ packets/second. According to the results of section 3, instability should occur for

$$E \geq \min\left[\frac{1}{2}, \frac{3}{7}, \frac{2}{3}, \frac{1}{5}\right]$$

or, $E \geq 0.2$.

To test the theory, simulations were run for E = 0.1, 0.2, 0.3, 0.5, and 0.8. Figure 8.1.1 shows the results. The "1" in Figure 3.1 of section 3 corresponds to the Source Encoder; the "2", to the Receiver Decoder; the "3", to the Receiver Encoder; and the "4", to the Source Decoder.

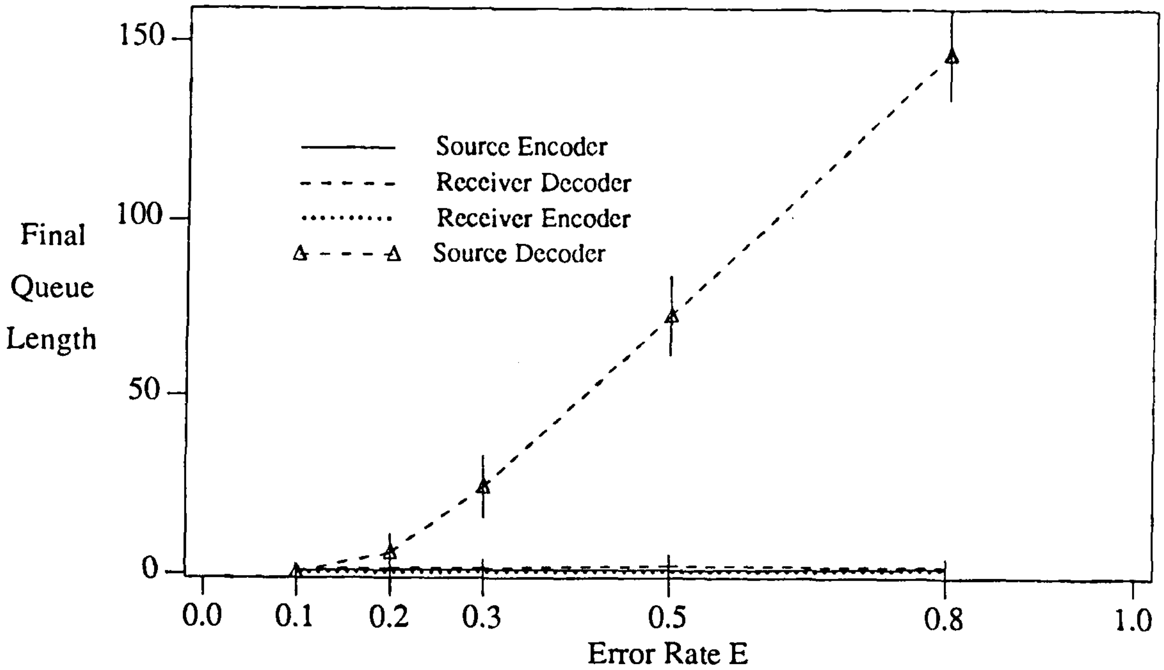

Figure 8.1.1 Average Final Queue Lengths for the Section 3 Example

It is clear from Figure 8.1.1 that the largest queue forms at the source decoder, a fact that is perhaps not surprising because the service rate is so low at that node. The queue is created because the error rate is such that equilibrium is no longer possible, and the queue therefore grows without bound. Under these conditions, steady-state flows between other nodes in the network can no longer be said to exist, so the pile-up of messages at the source decoder blocks messages from cycling through the rest of the network. Thus, large queues form at the source decoder, but nowhere else.

The network is described as unstable when any of the queues associated with its constituent nodes grow with time, even after transients have died out. In this example, then, it is apparent that the queue at the source decoder governs the stability of the network. As mentioned above, the simulation data can show the network leaving equilibrium when the average final queue length approaches the average maximum queue. This approach occurs when there is reasonable overlap between the error bars of the two curves. The greater the overlap, or the closer the final queue length is to the maximum queue length, the more unstable the network.

Figure 8.1.2 compares the average maximum queue length to the average final queue length for the source decoder. Although it is not clear on the figure, the data for the two curves at E = 0.1 are 0.86±1.74 for the final queue and 6.93±2.17 for the maximum queue, so there is a very small probability that the final queue length indeed represents the maximum at this small error rate. For this case, then, a steady-state solution may be computed using equation (1.1) ($L_q = 0.36$). However, there is considerable error bar overlap for all $E \geq 0.2$, tending to indicate that no equilibrium solution is possible for these cases. That is, as the error rate becomes larger, it is increasingly likely that the simulations will give average maximum queue lengths which coincide with average final queue lengths. The fact that the curves approach one another more and more closely as E increases indicates that the network is becoming increasingly unstable (blocked).

The first evidence of instability is indeed seen at an error rate of 0.2 for this network, supporting the theoretical predictions.

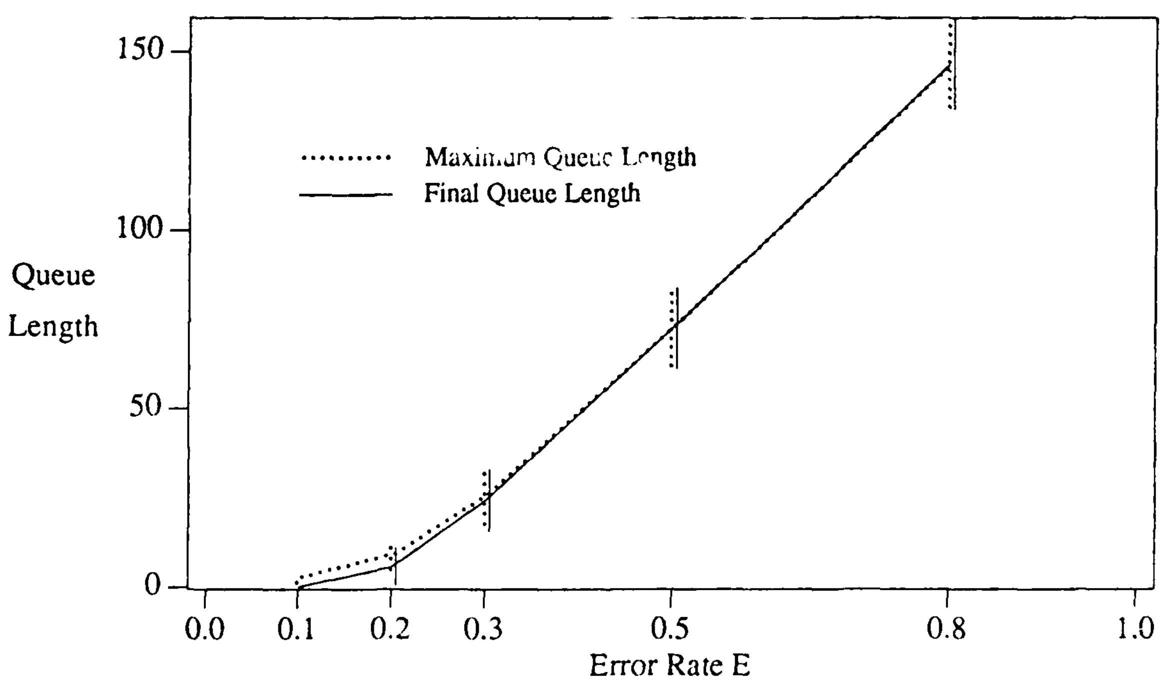

Figure 8.1.2 Average Maximum and Final Queue Lengths for Section 3 Example (Source Decoder)

8.2 Example of Nodes with Equal Service Rates

In section 5, networks of nodes with equal service rates were examined. Figures 8.2.1 to 8.2.4 show average final queue lengths for a single layer example with all service rates equal to 2.0 packets/second, and arrival rates of 0.5, 1.0, 1.5, and 2.0 packets/second.

Clearly, the source encoder governs the stability of the interface since the longest queues form at that node. Therefore, the effect of error rate is investigated for the source encoder in Figures 8.2.5 to 8.2.8 using the same criteria as in section 8.1. Theory predicts that instability will occur for

$$E \geq 1 - \lambda/\mu \ ,$$

or,

$$E \geq 1 - \lambda/2 \ .$$

Table 8.2.1 summarizes the predicted minimum error rates for instability, for various arrival rates.

λ	Minimum Error Rate for Instability
0.5	0.75
1.0	0.5
1.5	0.25
2.0	0.0

Table 8.2.1 Predicted Minimum Error Rates for Instability

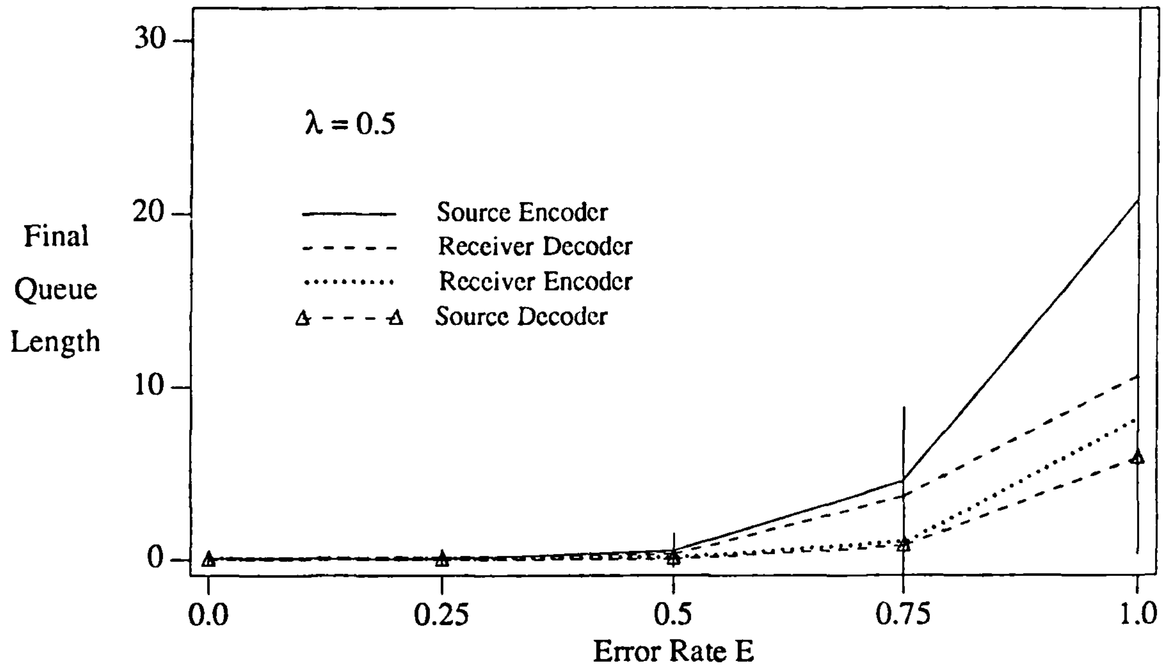

Figure 8.2.1 Average Final Queue Lengths for λ = 0.5 (Section 5)

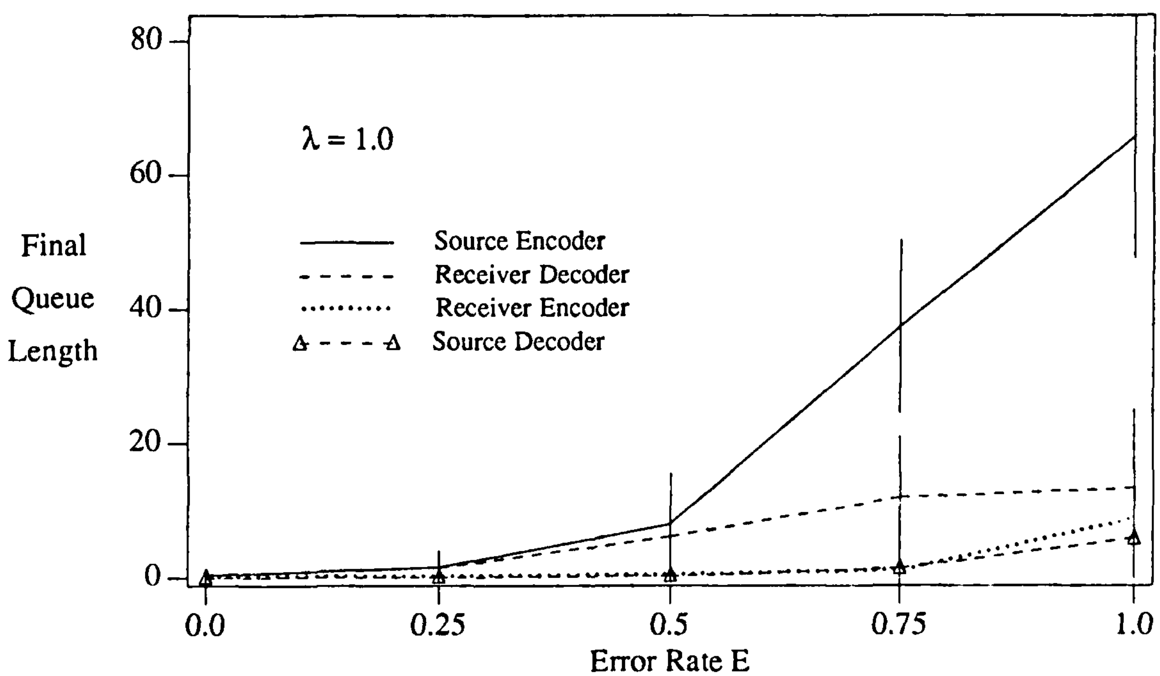

Figure 8.2.2 Average Final Queue Lengths for λ = 1.0 (Section 5)

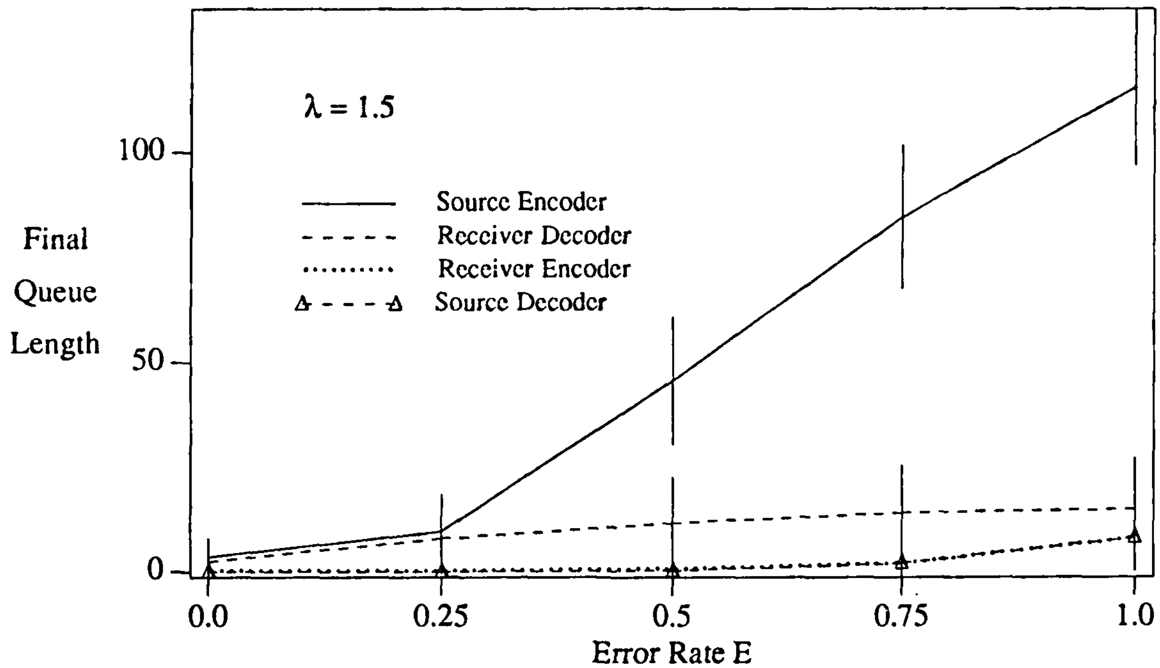

Figure 8.2.3 Average Final Queue Lengths for λ = 1.5 (Section 5)

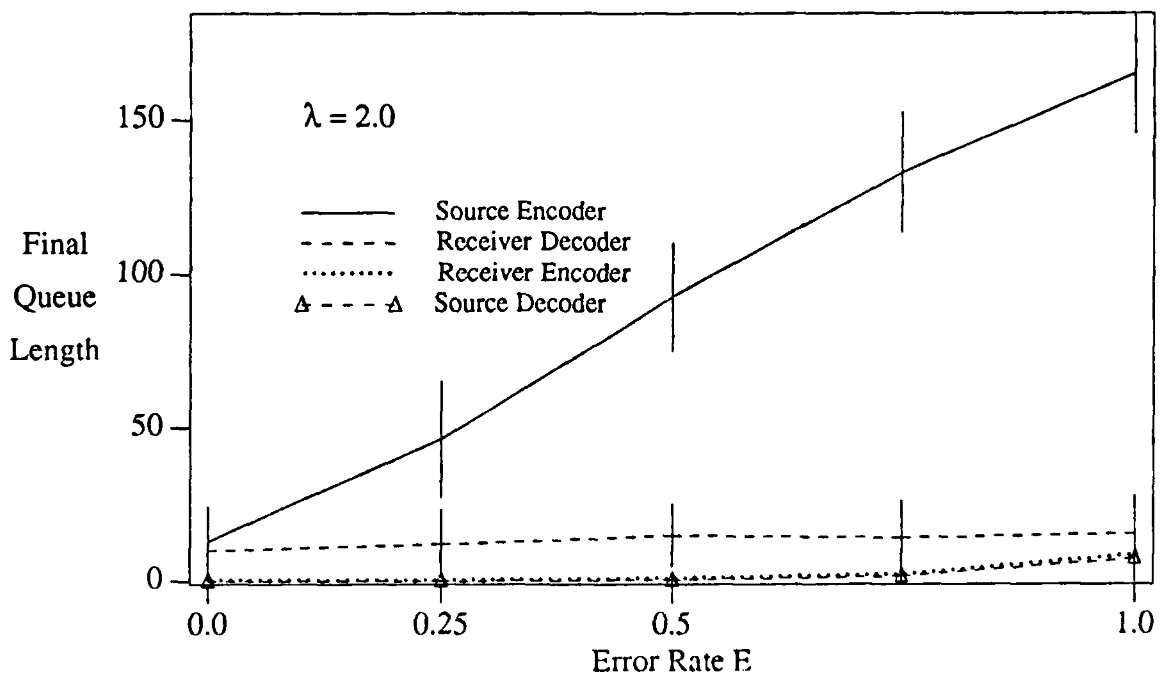

Figure 8.2.4 Average Final Queue Lengths for λ = 2.0 (Section 5)

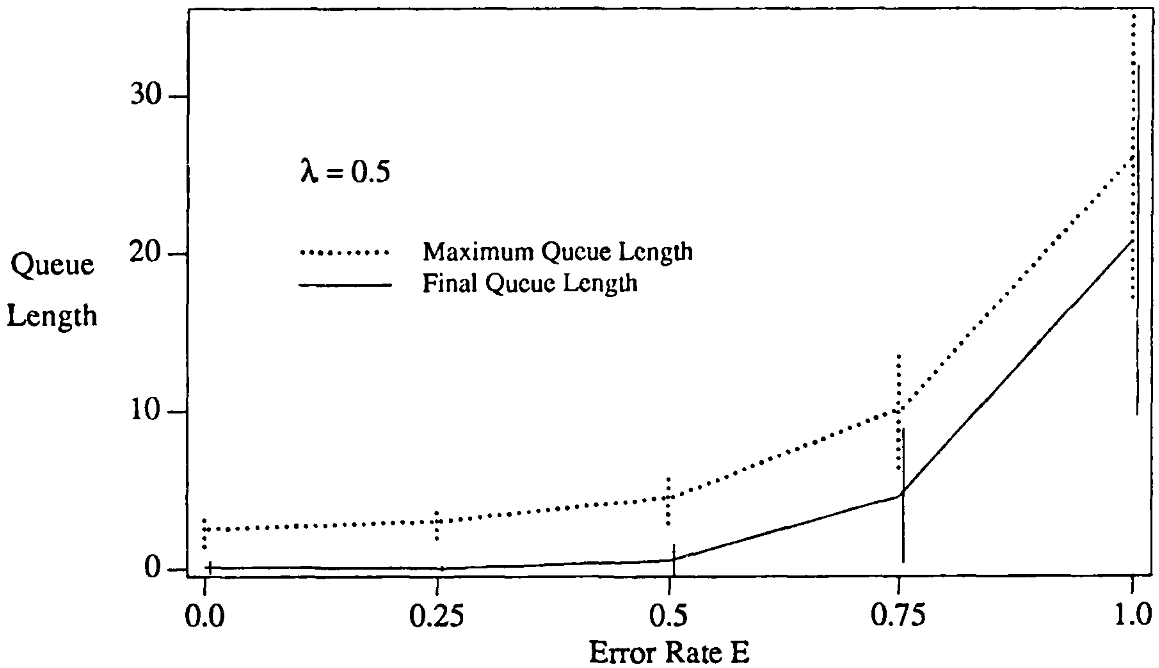

Figure 8.2.5 Average Maximum and Final Queue Lengths for λ = 0.5 (Section 5) (Source Encoder)

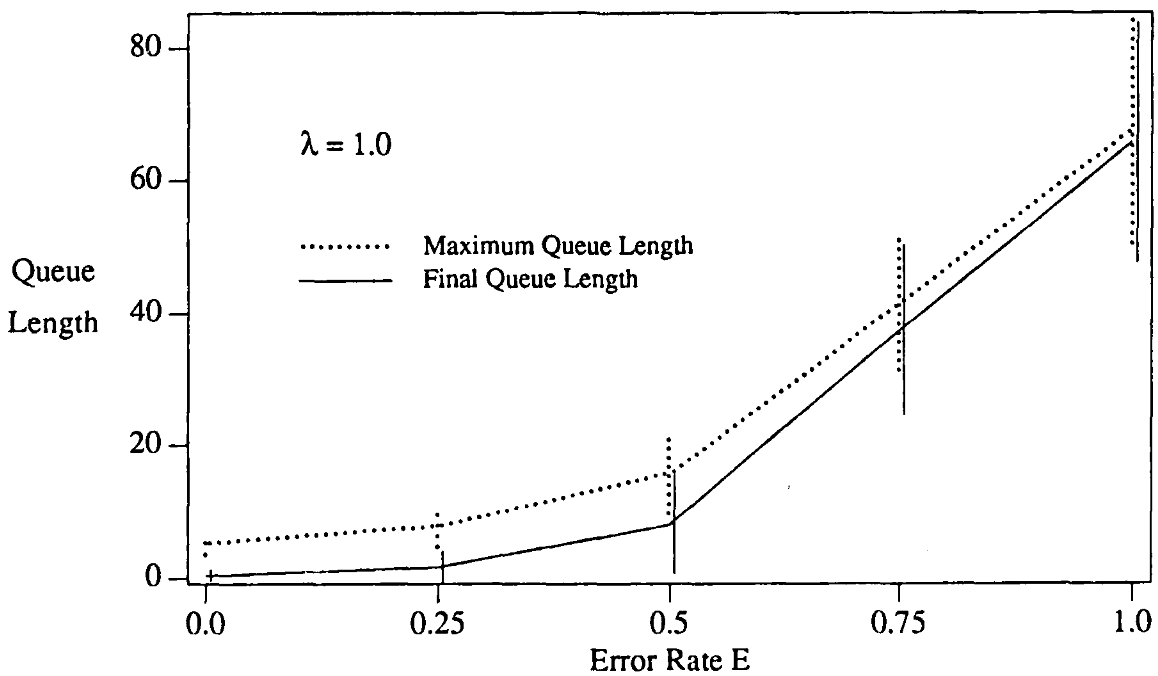

Figure 8.2.6 Average Maximum and Final Queue Lengths for λ = 1.0 (Section 5) (Source Encoder)

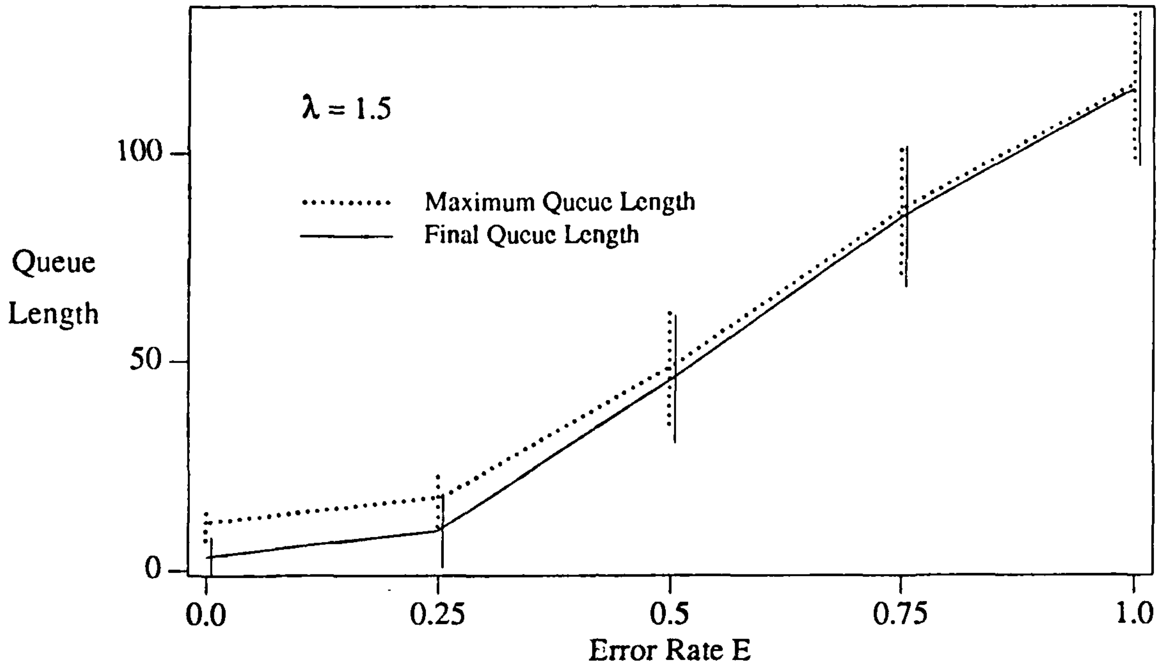

Figure 8.2.7 Average Maximum and Final Queue Lengths for $\lambda = 1.5$ (Section 5) (Source Encoder)

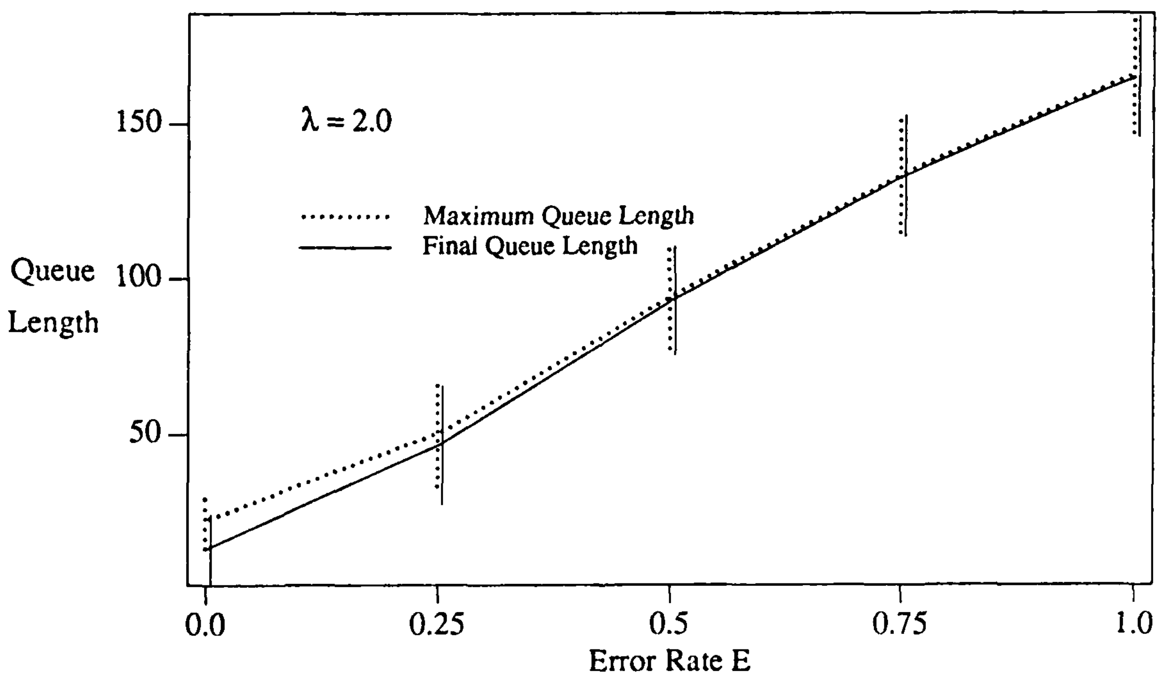

Figure 8.2.8 Average Maximum and Final Queue Lengths for $\lambda = 2.0$ (Section 5) (Source Encoder)

The graphs of the simulation results show that, in general, evidence of instability initially becomes apparent at the error rates predicted by theory. Also, the average final queue length approaches the average maximum queue length more and more closely as error rate increases, as in the previous section.

8.3 Example of Two-Layer Interface with Feedback at Each Level (Section 6)

To illustrate the results of section 6, a two-layer network with feedback was simulated using the following data (where the subscripts correspond to those defined in section 6):

$$\lambda = 4 \tag{8.3.1}$$

$$\mu_1 = 8,\ \mu_2 = 6,\ \mu_3 = 4,\ \mu_4 = 5$$

$$\mu_5 = 6,\ \mu_6 = 8,\ \mu_7 = 12,\ \mu_8 = 9$$

$$E_1 = 0.1,\ E_2 = 0.1,\ 0.3,\ 0.5,\ 0.8$$

Node Number	Layer	Node Name	Arrival Rate	Service Rate
1	1	Source Encoder	4.44	8.00
2	1	Receiver Decoder	4.44	6.00
3	1	Receiver Encoder	0.444	4.00
4	1	Source Decoder	0.444	5.00
5	2	Source Encoder	4.93, 6.34, 8.88, 22.20	6.00
6	2	Receiver Decoder	4.93, 6.34, 8.88, 22.20	8.00
7	2	Receiver Encoder	0.932, 2.34, 4.88, 18.20	12.00
8	2	Source Decoder	0.932, 2.34, 4.88, 18.20	9.00

The first step is to test the conditions on E_1, the error rate at the higher level. Using the data in (8.3.1),

$$E_1 \geq \min\left[\frac{1}{2}, \frac{1}{3}, \frac{1}{2}, \frac{5}{9}\right]$$

or, $E_1 \geq 1/3$. Since $E_1 = 0.1$, no blocking problems are expected due to this level of error.

The second step is to test the conditions on E_2:

$$E_2 \geq \min [\ 0.259, 0.444, 0.722, 0.658\ ]$$

or, $E_2 \geq 0.259$. According to the theory, then, instability, or blocking, would be expected for any $E_2 \geq 0.259$.

In this simulation, when blocking occurs, the queues which block the network form at the source encoder of the lower level (layer 2) (see Figure 8.3.1). This is expected because it is at this node (node 5) that the arrival rate first exceeds service rate (for $E_2 = 0.3$) and where the difference between service and arrival rates is greatest as the error rate goes up. Figure 8.3.2 compares the final and maximum queue lengths for this node. As anticipated, the first symptoms of blocking are observed for $E_1 = 0.1$ and $E_2 = 0.3$, and the network becomes increasingly unstable as the error rate increases.

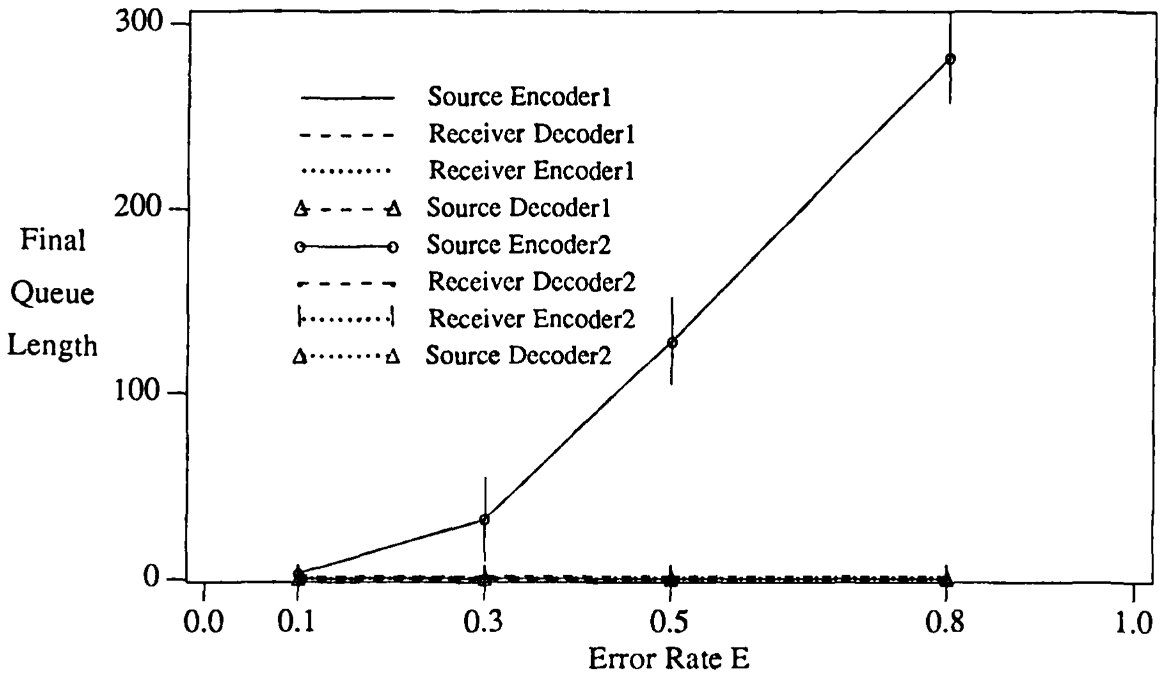

Figure 8.3.1 Average Final Queue Lengths for the Section 6 Example

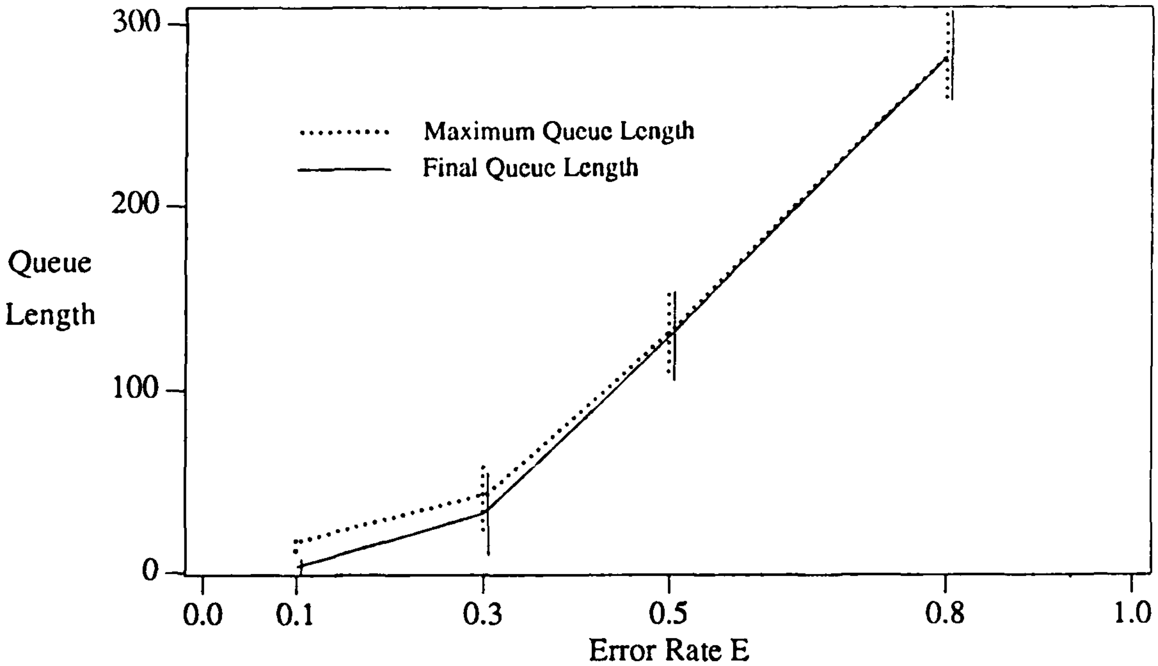

Figure 8.3.2 Average Maximum and Final Queue Lengths for Section 6 Example (Source Encoder 2)

9. Conclusions

Using Jackson's results for networks of nodes having Poisson arrival and service distributions, a model was developed to describe the passage of messages through the human-computer interface. Messages are coded, transmitted and decoded, and error messages are fed back when there is some ambiguity in the understanding of the message. It is necessary to test, for real interfaces, the assumption that the arrival and service of messages have Poisson characteristics. Nevertheless, the stability criterion developed in this paper should provide a guide for interface analysis even when arrival rates are not strictly Poisson.

Furthermore, the model described can be used to judge the effect of allowing feedback of different kinds at various levels of the interaction. The nature of the feedback at some level will have an impact on the service rate at that level, which will affect the stability of the network as a whole. Thus, the model allows quantification of such phrases as "eliminate all but the simplest and fastest feedback at the lower levels" by setting limits on the arrival, service and error rates in order to allow stability to be maintained.

There are problems associated with the precise measurement of interface dynamics. First, the timings may be difficult or impossible to measure, and, second, it may not be possible to determine the error rates associated with the layers which have been identified in the interface. This paper, while it does make assumptions about the arrival and service of the packet messages, provides a theoretical technique for estimating the global stability of a network. Moreover, the simulations which were used to confirm the theoretical results showed that stability can be judged to some extent from the simulation results alone.

10. References

1. JACKSON, J.R. "Networks of Waiting Lines." *Operations Research*, 5: 518-521, 1957.

2. GIFFIN, W.C. Queuing: Basic Theory And Applications. Grid, Inc., Columbus, Ohio, 1978.

3. FISZ, M. Probability Theory and Mathematical Statistics. John Wiley & Sons, Inc., New York, 1963.

4. GROSS, D. and C.M. HARRIS. Fundamentals of Queueing Theory, Second Edition. John Wiley & Sons, Inc., New York, 1985.

5. NEWELL, G.F. Applications of Queueing Theory, G.F. Newell, Great Britain, 1971.

6. TAYLOR, M.M. "Layered Protocols for computer-human dialogue. I: Principles." *Int. J. Man-Machine Studies*, 28: 175-218, 1988.

7. NELSON, R.T. "Waiting Time Distributions for Application to a Series of Service Centers." *Operations Research*, 6: 856-862, 1958.

Appendix 1 - Jackson's Results

This appendix presents Jackson's basic results (see, for example, Giffin (2)). The following notation is used:

$p(m_1, m_2, \ldots, m_N\,;\,t)$ = probability of m_i messages at node i at time t (including all network nodes)

$p(m_1, m_2, \ldots, m_N)$ = probability of m_i messages at node i (including all network nodes)

$p_i(m_i)$ = probability of m_i messages at node i

m_i = number of messages in queue at node i

λ_i = arrival rate at node i due to externally generated demand

e_i = effective arrival rate at node i from all sources

μ_i = service rate per server at node i

c_i = number of servers at node i

r_{ij} = probability that upon completing service at node i, messages will go node j, $\sum_j r_{ij} = 1$

N = number of nodes in the network

The steady state flow into node i is given as

$$e_i = \lambda_i + \sum_{k=1}^{N} r_{ki}\, e_k \quad , \; i = 1, \ldots, N$$

which represents the flow rates from external sources feeding node i plus the internal flow rates from other nodes in the network which feed node i. The steady-state solution shows that the final equation can be factored into a product of marginal probabilities:

$$p\,(m_i, m_2, \ldots, m_N) = p_1(m_1)\, p_2(m_2) \ldots p_N(m_N) \tag{A.1}$$

where

$$p_i(m) = \begin{cases} \dfrac{p_i(0)[e_i/\mu_i]^m}{m!}, & m = 0, 1, \ldots, c_i \\ \dfrac{p_i(0)[e_i/\mu_i]^m}{c_i![c_i]^{m-c_i}}, & m \geq c_i \end{cases}$$

$$p_i(0) = \left[\sum_{m=0}^{c_i-1} \frac{[e_i/\mu_i]^m}{m!} + \frac{[e_i/\mu_i]^{c_i}}{c_i![1 - e_i/c_i\mu_i]} \right]^{-1} .$$

In other words, the lengths of the queues at the different nodes are independent of each other. At steady-state, the mean number of messages in the input queue at node i, the mean number of messages waiting for or being served at node i, the expected queue time at node i, and the expected total queuing and serving time may be computed as follows:

$$L_{q_i} = \frac{p_i(0)[e_i/\mu_i]^{c_i}[e_i/c_i\mu_i]}{c_i![1 - e_i/c_i\mu_i]^2} \tag{A.2}$$

$$L_i = L_{q_i} + e_i/\mu_i$$

$$W_{q_i} = L_{q_i}/e_i$$

$$W_i = W_{q_i} + 1/\mu_i$$

For the case of a single channel per node, the equation L_{q_i} may be simplified to give

$$L_{q_i} = \frac{p_i(0)\,[e_i/\mu_i]^2}{[1-e_i/\mu_i]^2} \quad . \tag{A.3}$$

Furthermore, the expression for $p_i(0)$ becomes $[1 - e_i/\mu_i]$. Let $k_i = \dfrac{e_i}{\mu_i}$. Then, for the single server case,

$$L_{q_i} = \frac{k_i^2}{1 - k_i} \quad . \tag{A.4}$$

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.) Defence and Civil Institute of Environmental Medicine, 1133 Sheppard Ave W North York, Ont, M3M 3B9	2. SECURITY CLASSIFICATION (overall security classification of the document, including special warning terms if applicable) Unclas

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C,R or U) in parentheses after the title.)

Queuing Theory Approach To Modelling The Dynamics of the Human-Computer Interface (U)

4. AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)

Joyce M.E. van de Vegte

5. DATE OF PUBLICATION (month and year of publication of document)	6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)	6b. NO. OF REFS (total cited in document)

6. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

final technical report

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

DCIEM (see 1)

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)	9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)
10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)	10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availabilty (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is high desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both offical languages unless the text is bilingual).

Abstract

This paper studies a model of the human-computer interface based on the idea of a message passing from one partner (the human) to the other (the computer) by means of an arbitrary number of layers which represent levels of abstraction of the message. The model uses a queuing theory formulation and assumes Poisson arrival and service characteristics to describe the movement of packets through the interface and to develop a criterion for network stability. Simulations are included to illustrate some of the theoretical results obtained.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)