Simulation Studies of the Basic Packet Routing Problem

Author: Elena Sirén 48314u
Supervisor: Pasi Lassila

February 6, 2001

Abstract

In this paper the simulation of a basic packet routing problem using the programming language C++ and the simulation library CNCL is considered. First the features special to the simulation library CNCL will be explained shortly. The aim is to via simulation study different algorithms to route packets in a system consisting of two nodes connected by two links. At first all queues are assumed to be infinite and the objective is to minimize the mean packet delay. Later also finite queues are examined.

Contents

List of Appendices ii

Concepts and acronyms iii

1 Introduction 1

2 CNCL 1

2.1 CNCL Overview 1
2.2 Process Description 2
2.3 Simulation Description 3

3 Optimal Static Routing Schemes 5

3.1 Decision at Packet Arrivals 5
3.2 Decision at Service Completions 6
3.3 Comparison between the Static Routing Schemes 7

4 Optimal Dynamic Routing Schemes 9

4.1 Packet Based Dynamic (PBD) Scheme 9
4.2 Bit Based Dynamic (BBD) Scheme 9
4.3 Comparison between the Dynamic Routing Schemes 10
4.4 Static Routing Schemes versus the Dynamic Ones 12

5 Finite Queues 14

6 Conclusions 18

List of Appendices

1. The source code for static optimal routing scheme with decisions at packet arrivals, 2mm1.c
2. The source code for static optimal routing scheme with decisions at packet arrivals (finite queues), 2mm1lim.c
3. The source code for static optimal routing scheme with decisions at service completions, mm2.c
4. The source code for static optimal routing scheme with decisions at service completions (finite queues), mm2lim.c
5. The source code for dynamic optimal routing scheme with decisions based on number of packets, pbd.c
6. The source code for dynamic optimal routing scheme with decisions based on number of packets (finite queues), pbdlim.c
7. The source code for dynamic optimal routing scheme with decisions based on amount of unfinished work, bbd.c
8. The source code for dynamic optimal routing scheme with decisions based on amount of unfinished work (finite queues), bbdlim.c

Concepts and acronyms

- λ The arrival rate is the intensity of the arriving packets.
- μ The service rate
- ρ The traffic load
- B The length of a packet
- BBD (Bit Based Dynamic) Scheme is (in this work) a dynamic routing scheme with decisions based on the amount of unfinished work.
- c The capacity of the link
- CNCL is a simulation library implemented with the programming language C++. The simulations described in this document are done by using CNCL.
- Dynamic routing means routing that is based on the instantaneous state information in addition to the static traffic and link parameters.
- $E[D]$ The mean packet delay is in this paper the sum of the waiting and serving times.
- PBD (Packet Based Dynamic) Scheme is (in this work) a dynamic routing scheme with decisions based on the number of packets waiting to be served.
- M/M/1 Queuing system with a single server and infinite buffer. Arriving process is Poisson and serving times are exponentially distributed.
- Static routing is routing, that is entirely based on the static system information i.e. link and traffic parameters.

1 Introduction

Simulation is a statistical method used for performance analysis of systems. Compared to mathematical analysis, simulation yields estimates of the considered performance measures. Usually simulation is used when the system is very complicated and mathematical analysis is either impossible or very difficult. In this paper, simulation of models, originally studied analytically in [1], is considered.

The system to be studied consists of two nodes in a connectionless packet-switched network with two parallel links as illustrated in Figure 1. Packets arrive at node 1 according to a Poisson process with intensity λ, and they are destinated to travel from node 1 to node 2. Each packet is routed separately via either of the two links, whose capacities are denoted $c_1 = \mu_1/B$ and $c_2 = \mu_2/B$, where B is the average length of a packet. The packet lengths are assumed to be exponentially distributed. The faster link is indexed by c_1.

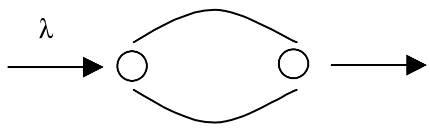

Figure 1: The system

The problem is, how the packets have to be routed to get the best possible result. First systems with infinite queues are considered, and the aim is to minimize the mean packet delay. Later also finite queues are studied. In all cases the arrival rate λ and the mean service rates μ_1 and μ_2 are assumed to be known. In static routing schemes no other information is needed. We start with considering these kinds of schemes. More complicated schemes are based also on the state information of the system. In these systems the state is first described by the number of packets and later by the amount of information still to be transmitted. Both static and dynamic schemes can additionally be classified depending on the moment the routing decision is made. There are two alternatives: either immediately when the packet arrives or at the service completion.

2 CNCL

2.1 CNCL Overview

CNCL is a C++ library created at Communication Networks laboratory, at the Aachen University of Technology in Germany. CNCL provides support for generation of random numbers, collection and analysis of statistics and event driven simulation. The simulation models considered in this paper have been created using the programming language C++ and the simulation library CNCL.

In the following, a brief description of the most important features are described. For a more detailed description, see [2]. In order to understand the function and operational principle of CNCL, we start with studying event driven simulation. There are Events, EventHandlers and an EventScheduler. Events are sent from one EventHandler to the other in order to change the state of the EventHandler. EventHandlers are state machines that receive and process Events. The executed routine depends on the state of the EventHandler and on the incoming Event. The whole simulation run is controlled by an EventScheduler, which sorts the Events according to the time stamps and priority and passes them to the addressed EventHandlers. The corresponding classes, which perform the above discussed tasks are CNRNG, CNRandom, CNStatistics, CNQueue, CNEvent, CNEventHandler, CNEventScheduler and CNSimtime.

Random numbers are important in simulation. CNCL provides a selection of classes that support the generation of random numbers. CNRNG is the abstract base class for all CNCL random number generators. It has several derived classes. In these simulations the Fibonacci random number class, CNFiboG, is used. To create random numbers, that are distributed in some special manner, also the class CNRandom is needed. This abstract base class is for instance inherited by classes CNUniform and CNNegExp. In this work mostly the class CNNEgExp is used in order to create exponentially distributed random numbers.

An important part of simulation is gathering the statistics. The abstract base class for this purpose is called CNStatistics, and it provides the functions to store the data. The derived class used in this work is called CNMoments. The mean, variance, deviation and several other statistics concerning the data are obtained simply by using the corresponding function. In the simulations here, the gathered data is the delay of each packet and, in the case of a finite queue, the information if the packet was lost or not. From these the mean packet delay and the blocking probability can be estimated.

2.2 Process Description

The general outline of the simulation process is similar in all simulations considered in this paper. The simulation of the schemes with decisions at the service completion is an exception, which will be considered at the end of this subsection. In the schemes with decision on packet arrival there are four types of EventHandlers:

- **Generator** generates the jobs.
- **Controller** routes the jobs to the right queue.
- **Queue** stores the packets waiting to be served and sends them to the server, when it is free.
- **Server** handles the jobs.

The EventHandlers communicate by sending Events to each other. The four Event types in the schemes with the decision making at packet arrival are:

- **EV_TIMER_G** signals the generator that it is time to generate a new job.
- **EV_JOB** describes a packet in the system.
- **EV_TIMER_S** signals the server that a packet has been transmitted and the transmission of a new packet can be initiated.
- **EV_SERVER_FREE** signals the queue that the server is free to take a new job.

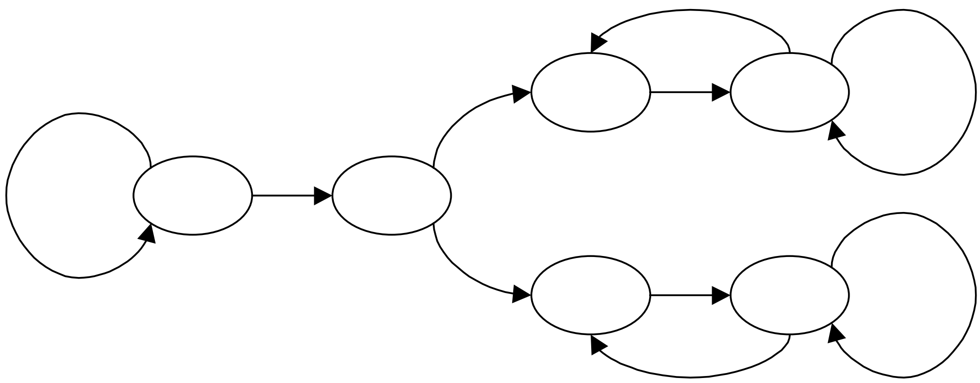

Figure 2: The process, when the routing decision is made at the packet arrival

In Figure 2 the simulation process is described. The generator sends an EV_JOB event to the Controller, which sends the same event immediately further to either queue. Additionally the Generator sends itself an EV_TIMER_G event in order to generate a new job after an exponentially distributed time. The Queue sends an EV_JOB event to the Server, or if the Server is not free, it waits until it receives an EV_SERVER_FREE event. The Server sends itself an EV_TIMER_S event to signal that the job is done and a new packet can be taken from the Queue.

Figure 3 represents the simulation process of the routing scheme with postponed decision making. The main difference to Figure 2 is that there is only one queue. Additionally, the queue functions also as the Controller and routes the packets to the right Server.

2.3 Simulation Description

The simulation method that is used in this work is called process simulation. The quantity of interest is the steady state mean delay of a packet, $E[D]$. As the simulation method is process simulation, two issues affect the estimation of $E[D]$ from the collected statistics:

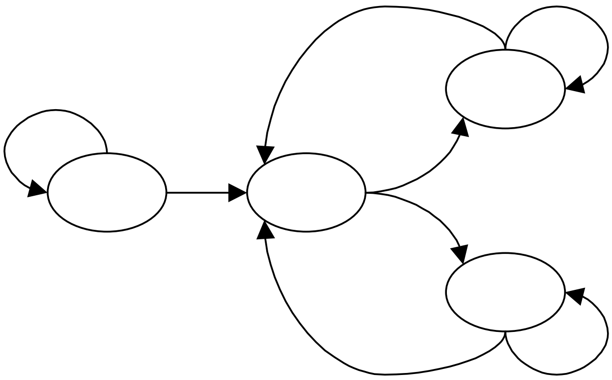

Figure 3: The process, when the decision is made at service completion

- Initial transient: As the simulation is started from a given fixed initial state, it takes some time before the system reaches steady state. In this simulation this has been handled by starting the system from an empty state and then the system is simulated for a time of 100 packet arrivals before the collecting of sample statistics is begun.
- In process simulation the samples are positively correlated. To be able to compute confidence intervals for the estimate of $E[D]$, independent samples are needed. Thus the simulation here has been rerun several times independently.

In the j^{th} simulation run $N = 10000$ samples is generated. Their mean value is given by

$$\hat{D}_j = \frac{1}{N} \sum_{n=1}^{N} D_n, \tag{1}$$

where D_n is the mean delay of one packet. The simulation is then repeated $J = 100$ times and the estimate for mean delay $E[D]$ is obtained from

$$\hat{D} = \frac{1}{J} \sum_{j=1}^{J} \hat{D}_j. \tag{2}$$

Finally, the 95% confidence interval for $\hat{D}$ is given by

$$\hat{D} \pm \frac{1.96 \cdot \sigma(\hat{D}_j)}{\sqrt{J}}, \tag{3}$$

where $\sigma(\hat{D}_j)$ is the standard deviation of $\hat{D}_j$, estimated from

$$\sigma(\hat{D}_j) = \sqrt{\frac{1}{N-1} \left[\sum_{n=1}^{N} D_n^2 - \frac{1}{N} \left(\sum_{n=1}^{N} D_n \right)^2 \right]}. \tag{4}$$

All the simulation models in this paper are ran with several different values of the arrival rate λ. The smallest value of λ is 0.05. The value is increased by 0.05 in every simulation until it gets the value 0.95 in the nineteenth simulation run. The sum of serving rates, $\mu_1+\mu_2$, is kept in the value 1 all the time. Therefore also the traffic load $\rho = \lambda/(\mu_1+\mu_2)$ varies between 0.05 and 0.95. Simulations are performed with five different values of the ratio μ_2/μ_1 (0.2, 0.4, 0.6, 0.8 and 1). The mean packet length B is assumed to equal to 1.

3 Optimal Static Routing Schemes

In this chapter two alternatives to route packets are considered. The difference between them is the time when the decision is made. In the first model the decision is made immediately when a packet enters the system. The alternative is to wait until the packet is to be served. In both static routing schemes the arrival rate λ and the mean service rates μ_1 and μ_2 are assumed to be known but no other information is needed.

3.1 Decision at Packet Arrivals

In this scheme the decision is made when the packet enters the system. This means that both nodes have their own queues and each packet is put in either queue, when it arrives to the system. The probability that the packet is routed via link 1 is p and the probability that it is routed via link 2 is $1-p$. Therefore the arrival rates of these two independent M/M/1 queues are $\lambda_1 = p\lambda$ and $\lambda_2 = (1-p)\lambda$. Conditions for the whole system to be stable are $\lambda < \min\{\mu_1/p, \mu_2/(1-p)\}$ and $\lambda < \mu_1+\mu_2$. The probability p that minimizes the mean packet delay $E[D]$, as derived in [1], is given by

$$p = \begin{cases} 1 & , \quad \text{if } \lambda \le \mu_1 - \sqrt{\mu_1\mu_2}, \\ \dfrac{\sqrt{\mu_1}}{\sqrt{\mu_1}+\sqrt{\mu_2}} + \dfrac{\sqrt{\mu_2}}{\sqrt{\mu_1}+\sqrt{\mu_2}} \dfrac{\mu_1-\sqrt{\mu_1\mu_2}}{\lambda} & , \quad \text{if } \lambda > \mu_1 - \sqrt{\mu_1\mu_2}, \end{cases} \tag{5}$$

in which case $E[D]$ equals

$$E[D] = \frac{p}{\mu_1 - p\lambda} + \frac{1-p}{\mu_2-(1-p)\lambda}. \tag{6}$$

In the simulation these equations are used in the following way. The probability is calculated before the simulation begins. Then the simulation is started and when an individual packet arrives, a random number between 0 and 1 is generated. If the number is smaller than or equal to p, the packet is routed via link 1. Otherwise it is routed via link 2. Therefore the proportion of packets, that are routed via link 1, is as great as p, if the number of packets routed is big enough.

Figure 4 illustrates $E[D]$ as a function of the traffic load ρ. The picture on the left describes one of the lines ($\mu_2/\mu_1 = 1$) with 95% confidence intervals compared to the solutions of Eq. (6) (solid line). The five lines on the right represent the mean packet delays when ratio μ_2/μ_1 gets values 0.2, 0.4, 0.6, 0.8 and 1. Each line has a comparable accuracy as the line in the left figure, and thus the confidence intervals have been omitted. The lowest curve corresponds to the value 0.2 and the highest to the value 1. This implies that the more μ_1 differs from μ_2, the smaller mean delay is obtained. However, the differences are fractional compared to the change in $E[D]$ caused by the increase of traffic load ρ.

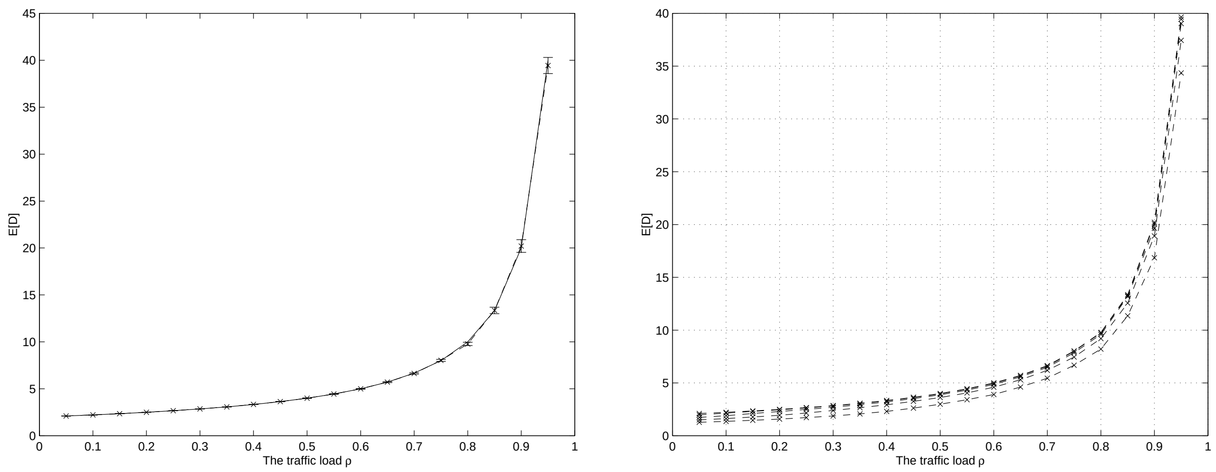

Figure 4: The mean packet delay $E[D]$ of the static optimal routing

3.2 Decision at Service Completions

To get better performance, we can postpone the routing decision. This can be done by having only one common queue. A packet is now routed when either server becomes free. If both are free, the packet is routed via the faster server. This is a very simple way to route and without any other information concerning the packets waiting, there is no other way to make better routing than this. Then, as shown in [1], the mean packet delay $E[D]$ is given by

$$E[D] = \frac{(\lambda + \mu_2)(\mu_1 + \mu_2)^3/(\mu_1 + \mu_2 - \lambda)}{\lambda^2(\mu_1^2 + \mu_2^2) + \lambda\mu_2(\mu_1 + \mu_2)(2\mu_1 + \mu_2) + \mu_1\mu_2(\mu_1 + \mu_2)^2}. \tag{7}$$

In Figure 5 the mean packet delay $E[D]$ is shown as a function of the arrival rate λ (the traffic load ρ). Again, the picture on the left describes one of the lines ($\mu_2/\mu_1 = 1$) with 95% confidence intervals compared to the solutions of Eq. (7) (solid line). The five lines in the figure on the right hand side represent the mean packet delays when the ratio μ_2/μ_1 obtains values 0.2 ,0.4, 0.6, 0.8 and 1. Each line has a comparable accuracy as the line in the left figure, and thus the confidence intervals have been omitted. Just like in the previous case the lowest curve corresponds to the value 0.2 and the highest curve to the value 1.

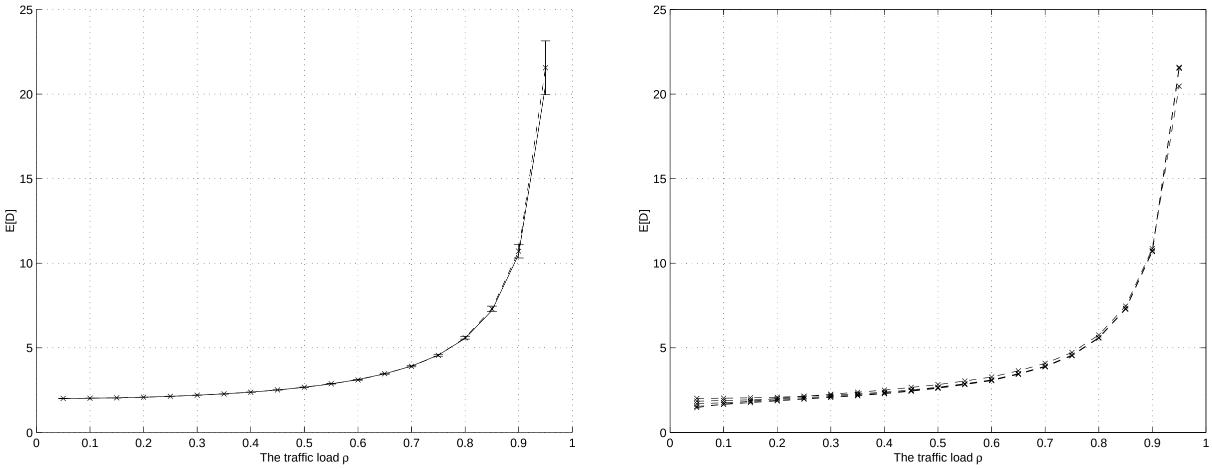

Figure 5: The delay $E[D]$ of the static optimal routing with postponed decision making

3.3 Comparison between the Static Routing Schemes

As illustrated in [1], routing is more efficient when the routing decision is made at service completion. In light traffic there are no essential differences, but in heavy traffic the mean packet delay can be halved by postponing the decisions. Because the same routing decision information is needed for both algorithms, it is natural to choose the more efficient one.

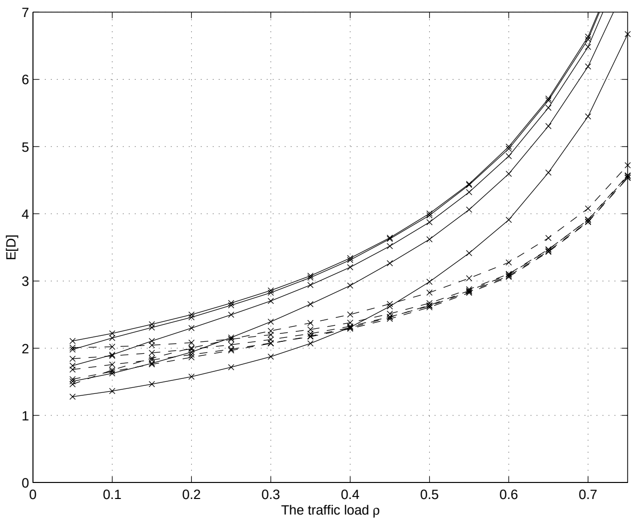

Figure 6: The mean packet delays of static routing schemes

Comparing the mean packet delays in Figure 6, we can also note that when the decisions are postponed, the differences between the delays of different ratios μ_2/μ_1 differ from each other less than when the routing decision is made at packet arrival. The dashed lines represent the results with the postponed decision making and the solid ones represent the results of the routing, in which the decision was made at packet arrival. In the last-mentioned routing scheme, the algorithm was scheduled to calculate the proportion of the packets that are to be directed to particular server, not to route an individual packet. Therefore it gives better results, when almost all packets are to be routed to the same server. That might be one reason for the larger difference between results with different ratios of μ_2/μ_1.

Figure 7 illustrates how the ratio of the results (mean delay with postponed decisions divided by mean delay under arrival time decisions) of different routing schemes varies when ratio μ_2/μ_1 gets values 0.2, 0.4, 0.6, 0.8 and 1. The solid lines are drawn using the equations Eq. (6) and Eq. (7) while the dashed ones describe the results of simulation. However, the simulation results are so close to the analytical solution that the difference can not be seen very clearly. Note that 100 000 samples have been used to obtain the estimates corresponding to the last the points in each line.

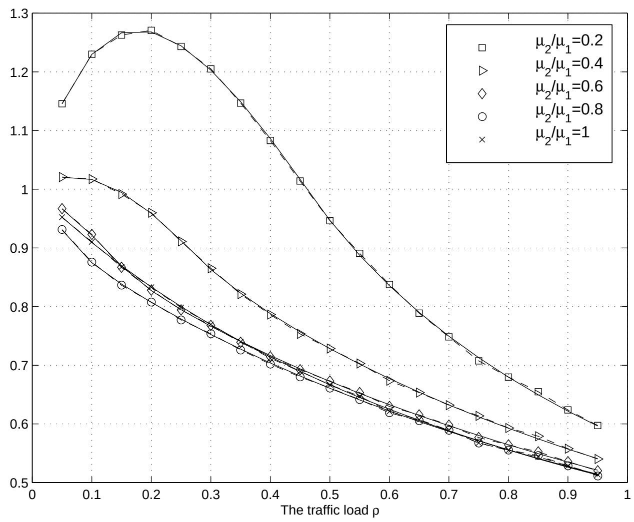

Figure 7: The ratio of delays of static routing schemes

The arrival time routing is more efficient when the ratio μ_1/μ_2 and the traffic load are small. However, the benefit that is obtained in light traffic is fractional and when the traffic is heavy, the mean packet delay can be reduced substantially by postponing the routing decisions. An interesting discovery is that the lowest line is drawn when $\mu_1/\mu_2 = 0.8$.

4 Optimal Dynamic Routing Schemes

The routing schemes that are based on the state information of the system in addition to the traffic and link parameters are called dynamic. Here two such schemes are studied. In both cases, the routing decision is made at packet arrivals. First the state of the system is described by the number of packets waiting. In the other example the state is described by the amount of information that is still to be transmitted. The latter is more complicated but also more precise.

4.1 Packet Based Dynamic (PBD) Scheme

Here the decision is based on queue lengths, that is, the number of packets in the queues. This policy is here called the PBD scheme. In addition to the number of packets, traffic intensity (λ) and link parameters (μ_1 and μ_2) are used in obtaining the decision rule. To make a decision, the function $t(x_1, x_2)$ has to be computed, which is given by

$$t(x_1, x_2) = \frac{x_1 + 1}{\mu_1 - p\lambda} - \frac{x_2 + 1}{\mu_2 - (1-p)\lambda}, \tag{8}$$

where p is given by Eq. (5) and x_1 and x_2 denote the queue lengths in terms of the number of packets in queues 1 and 2, respectively. If $t(x_1, x_2) \leq 0$, the packet is routed via link 1 and otherwise via link 2.

Figure 8 illustrates the mean packet delay $E[D]$ as a function of the traffic load ρ. The picture on the left describes one of the lines ($\mu_2/\mu_1 = 1$) with 95% confidence intervals. The five lines in the picture on the right hand side represent again the mean packet delays when the ratio μ_2/μ_1 gets values 0.2, 0.4, 0.6, 0.8 and 1. Each line has a comparable accuracy as the line in the left figure, and thus the confidence intervals have been omitted. The lowest line corresponds to the value 0.2 and the highest to the value 1.

The mean packet delay is different for the different ratios of μ_2/μ_1 when the traffic load is small. Under heavy traffic load, however, the effect of the ratio decreases, and the delay is almost the same for all the values of μ_2/μ_1.

4.2 Bit Based Dynamic (BBD) Scheme

Sometimes the length of the packets varies considerably. In a situation of this kind, the number of the packets in the queues is not very informative. It would be more useful to know the waiting time of an arriving packet, which consists of the remaining service times of the preceding packets. In this example the routing decision is based on the virtual time the packet has to wait before it is to be served. This scheme is called here the Bit Based Dynamic (BBD) scheme.

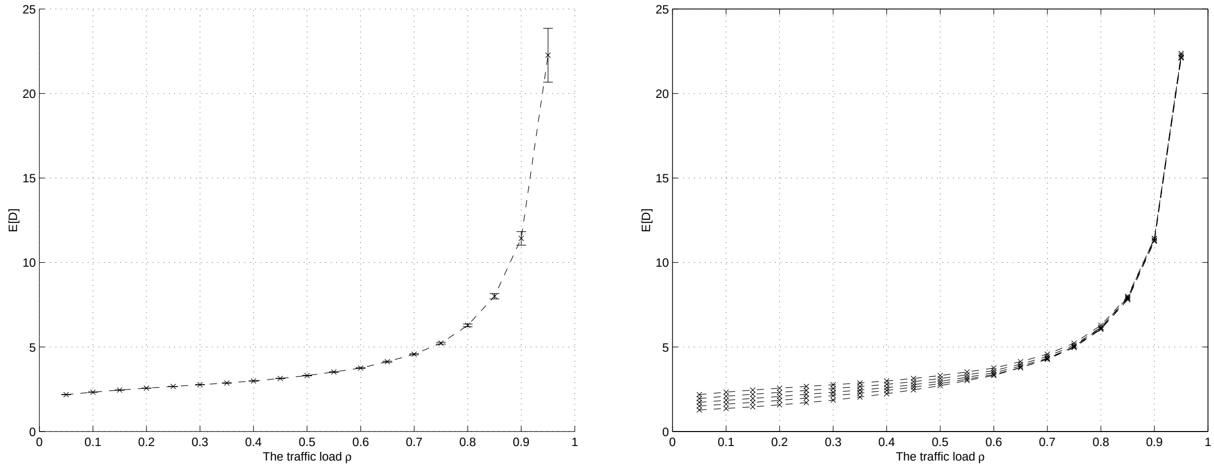

Figure 8: The mean packet delay $E[D]$ of the PBD policy

The decision is made by calculating the value of $t(x_1, x_2; y_1, y_2)$, given by

$$t(x_1, x_2; y_1, y_2) = x_1 + y_1 + \frac{p\lambda\mu_1 y_1(x_1 + \frac{1}{2}y_1)}{\mu_1 - p\lambda} - x_2 - y_2 - \frac{(1-p)\lambda\mu_2 y_2(x_2 + \frac{1}{2}y_2)}{\mu_2 - (1-p)\lambda}, \quad (9)$$

where x_1 and x_2 are amounts of unfinished work in the queues. Parameters y_1 and y_2 denote the service time of an arriving customer routed via the link in question. If $t(x_1, x_2; y_1, y_2) \leq 0$, the packet is routed via link 1 and via link 2 otherwise.

In Figure 9 the mean packet delay $E[D]$ is described as a function of the arrival rate λ (the traffic load ρ). On the left is the line with the ratio $\mu_2/\mu_1 = 1$ with the 95% confidence intervals. On the right there are five lines representing the mean packet delays, when the ratio μ_2/μ_1 gets values 0.2, 0.4, 0.6, 0.8 and 1. Each line has a comparable accuracy as the line in the left figure, and thus the confidence intervals have been omitted. The lowest line corresponds to the value 0.2 and the highest to the value 1.

4.3 Comparison between the Dynamic Routing Schemes

The both dynamic routing schemes are based on the queue lengths. The difference is that in the first model the length of a queue is thought to be the number of the packets, while in the second model the length is the amount of unfinished work measured in time. The second is more accurate since it takes into account the length of an individual packet. It is intuitively clear that routing based only on number of packets is more inaccurate when the variance of the packet lengths increases, but is efficient, if the variance is small.

The Figures 8 and 9 are quite similar to each other. In both figures the mean packet delays are different for the different ratios of μ_2/μ_1, when the traffic load is small. Under heavy traffic load, however, the effect of the ratio decreases, and the delay is almost the same on all the values of μ_2/μ_1. The only difference between the

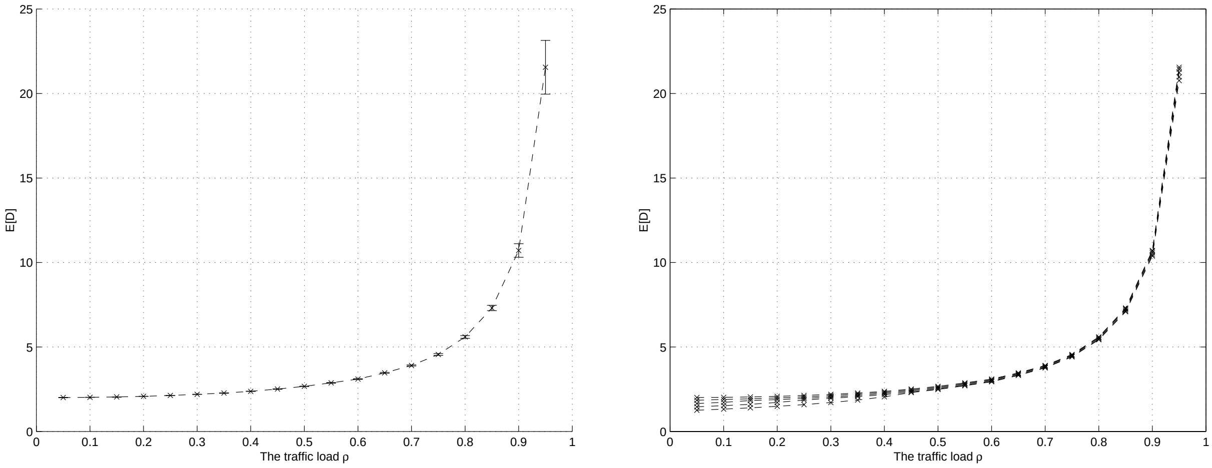

Figure 9: The mean packet delay $E[D]$ of the BBD policy

figures is that the routing based on unfinished work is more efficient and the mean delays are a little lower.

Figure 10 represents the ratio of the results of the dynamic routing policies (mean delay under BBD scheme divided by the mean delay under the PBD scheme) as a function of the traffic load ρ. The routing based on unfinished work is more efficient with all the values of the ratio μ_2/μ_1. The difference is greatest and the line is lowest, when the serving rates of server 1 and 2 are equal ($\mu_2/\mu_1 = 1$) and smallest with the ratio $\mu_2/\mu_1 = 0.2$. The lines are at their minimum, when the traffic load ρ is between 0.3 and 0.6.

Observe that, e.g. the highest line has a discontinuity at 0.46. The reason for this is the discontinuous function p. As can be noted from Eq. (5) the value of p is 1 if $\lambda < \mu_1 - \sqrt{\mu_1\mu_2}$. The conditions for highest line, corresponding to ratio 0.2, are $\mu_2/\mu_1 = 0.2$ and $\mu_2 + \mu_1 = 1$. Then $\mu_1 = 5/6$ and $\mu_2 = 1/6$ and the discontinuity point is at $\lambda = \rho = 0.46$.

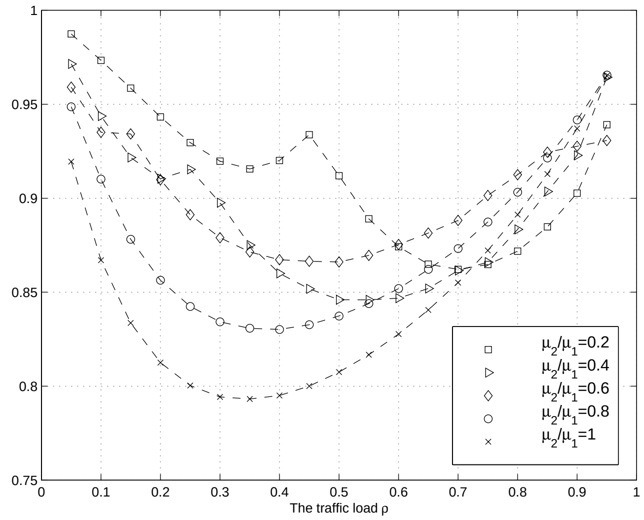

Figure 10: The ratio of delays of dynamic routing policies

4.4 Static Routing Schemes versus the Dynamic Ones

Static routing schemes are simple and demanding only a few parameters to be known, as compared to the dynamic schemes, which need more information concerning the queues. In the following figures the benefit obtained by using the dynamic schemes is seen. Compared to the static routing with arrival time routing decisions, the dynamic policies are able to, at their best, halve the mean delay of the packets. By postponing the decision making a comparable result is also obtained with the static routing policy. In the Figures 11 and 12 the ratios PBD/(static arrival time) and BBD/(static arrival time) are shown. The corresponding results for static routing with decisions at service completions are illustrated in Figure 7.

In Figure 11 the line, whose values decrease most rapidly corresponds to the simulation with the ratio $\mu_2/\mu_1 = 1$. Although under heavy traffic the delay is almost halved compared to static routing, under light traffic the delay ratio obtains values greater than 1. The uppermost line corresponds to the ratio $\mu_2/\mu_1 = 0.2$.

Figure 12 describes the ratio of delays of BBD and static arrival time routing schemes. The uppermost line corresponds to the ratio $\mu_2/\mu_1 = 0.2$ and the lowest lines to the values 0.6, 0.8 and 1. All the values are smaller than 1, which is a benefit compared to the static routing with postponed decision making. Under heavy traffic the results of these two routing schemes coincide but under light traffic the dynamic routing is better.

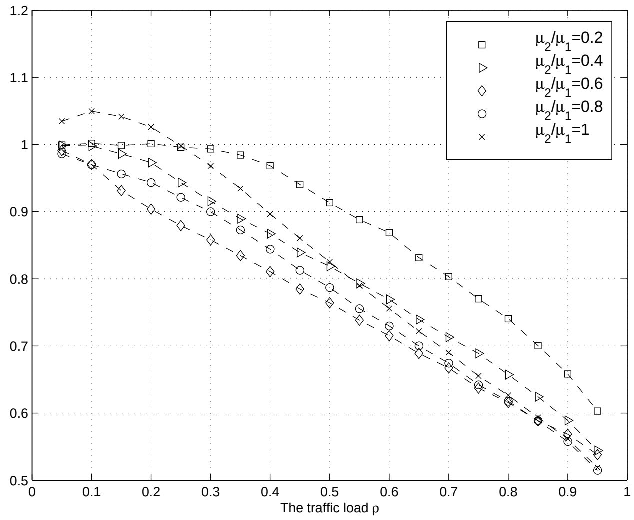

Figure 11: The ratio of delays of PBD and static optimal routing

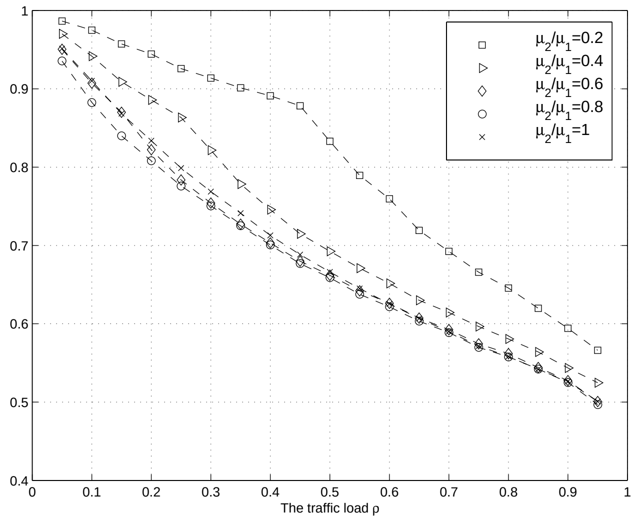

Figure 12: The ratio of delays, BBD and static optimal routing

5 Finite Queues

The routing schemes considered in the previous sections were all derived based on the assumption of an infinite queue size. In reality all queues are finite. Thus, in this section, we examine the effect of the queue finiteness on the delay and loss performance when using the previously described routing schemes. Furthermore, we only consider rather heavily loaded queues, as it is only then that the queue finiteness affects the performance.

We start with considering a situation with a short queue. Simulations with static and dynamic routing schemes were performed with a queue length 4+1, that is, both queues had a queue length 5 if server is included. In the case of static routing with postponed decision making, one queue with the queue length 9+1 is used. Tables 1 and 2 contain the delays and loss probabilities for routing schemes discussed in the earlier sections: static routing with arrival time decision making, static routing with postponed decision making and dynamic routing schemes based on number of packets or the amount of unfinished work.

ρ	μ_2/μ_1	Arrival Time	Postponed	PBD	BBD
0.8	0.2	4.54 ± 0.03	4.62 ± 0.03	4.25 ± 0.02	4.02 ± 0.03
0.9	0.2	5.31 ± 0.04	5.47 ± 0.04	4.73 ± 0.03	4.84 ± 0.04
0.95	0.2	5.67 ± 0.03	5.93 ± 0.04	4.97 ± 0.03	5.25 ± 0.04
0.99	0.2	5.94 ± 0.04	6.32 ± 0.04	5.16 ± 0.03	5.50 ± 0.04
0.8	0.6	5.08 ± 0.03	4.45 ± 0.03	4.48 ± 0.03	4.01 ± 0.03
0.9	0.6	5.57 ± 0.03	5.34 ± 0.05	5.04 ± 0.03	4.64 ± 0.03
0.95	0.6	5.80 ± 0.03	5.80 ± 0.04	5.34 ± 0.03	4.99 ± 0.03
0.99	0.6	5.97 ± 0.03	6.21 ± 0.04	5.57 ± 0.03	5.25 ± 0.03
0.8	1	5.14 ± 0.03	4.48 ± 0.03	4.86 ± 0.03	4.03 ± 0.03
0.9	1	5.60 ± 0.03	5.35 ± 0.05	5.51 ± 0.04	4.61 ± 0.03
0.95	1	5.80 ± 0.03	5.82 ± 0.05	5.85 ± 0.04	4.93 ± 0.03
0.99	1	5.97 ± 0.03	6.23 ± 0.04	6.14 ± 0.03	5.19 ± 0.03

Table 1: The mean packet delays (queue length 10)

The loss probability is smallest in the system with one queue and greatest in the static system with arrival time decision making. Additionally, the loss probability seems to be smaller if the serving rates are equal, because traffic is routed more evenly between the two queues. Also, for the same reason, the blocking probability is greater in the faster queue, when the difference in the serving rates is greater.

ρ	μ_2/μ_1	Arrival Time	Postponed	PBD	BBD
0.8	0.2	0.093 ± 0.002	0.022 ± 0.001	0.050 ± 0.001	0.053 ± 0.001
0.9	0.2	0.127 ± 0.002	0.048 ± 0.002	0.085 ± 0.002	0.086 ± 0.002
0.95	0.2	0.146 ± 0.002	0.066 ± 0.002	0.106 ± 0.002	0.108 ± 0.002
0.99	0.2	0.163 ± 0.002	0.084 ± 0.002	0.124 ± 0.002	0.126 ± 0.002
0.8	0.6	0.090 ± 0.002	0.021 ± 0.001	0.038 ± 0.001	0.044 ± 0.001
0.9	0.6	0.127 ± 0.002	0.046 ± 0.002	0.070 ± 0.002	0.076 ± 0.002
0.95	0.6	0.146 ± 0.002	0.065 ± 0.002	0.090 ± 0.002	0.097 ± 0.002
0.99	0.6	0.163 ± 0.002	0.083 ± 0.002	0.108 ± 0.002	0.116 ± 0.002
0.8	1	0.089 ± 0.001	0.021 ± 0.001	0.031 ± 0.001	0.043 ± 0.001
0.9	1	0.126 ± 0.002	0.047 ± 0.002	0.061 ± 0.002	0.076 ± 0.002
0.95	1	0.146 ± 0.002	0.066 ± 0.002	0.081 ± 0.002	0.097 ± 0.002
0.99	1	0.163 ± 0.002	0.083 ± 0.002	0.098 ± 0.002	0.115 ± 0.002

Table 2: The packet loss probability (queue length 10)

Tables 3 and 4 contain the mean delays and the proportions of lost packets, when the queue length is 49+1 (99+1 for model with one queue). The traffic and links parameters are same as in the previous example. Finally, Table 5 contains the corresponding results for infinite queues.

ρ	μ_2/μ_1	Arrival Time	Postponed	PBD	BBD
0.8	0.2	8.45 ± 0.16	5.77 ± 0.09	6.16 ± 0.09	5.45 ± 0.08
0.9	0.2	16.43 ± 0.44	10.60 ± 0.36	11.13 ± 0.36	10.10 ± 0.32
0.95	0.2	26.59 ± 0.94	20.35 ± 1.33	19.53 ± 0.97	18.10 ± 0.92
0.99	0.2	42.82 ± 1.47	40.63 ± 2.20	32.72 ± 1.21	36.81 ± 1.96
0.8	0.6	9.86 ± 0.19	5.58 ± 0.09	6.10 ± 0.09	5.53 ± 0.09
0.9	0.6	19.12 ± 0.58	10.42 ± 0.36	11.00 ± 0.36	10.29 ± 0.34
0.95	0.6	30.42 ± 0.99	20.20 ± 1.35	20.27 ± 1.19	19.19 ± 1.14
0.99	0.6	45.22 ± 1.27	40.53 ± 2.22	37.50 ± 1.75	38.33 ± 1.97
0.8	1	10.09 ± 0.19	5.61 ± 0.09	6.30 ± 0.09	5.62 ± 0.082
0.9	1	19.46 ± 0.55	10.43 ± 0.36	11.14 ± 0.36	10.43 ± 0.36
0.95	1	31.09 ± 0.93	20.20 ± 1.35	20.87 ± 1.32	19.89 ± 1.25
0.99	1	45.67 ± 1.29	40.54 ± 2.21	40.76 ± 2.16	38.10 ± 1.87

Table 3: The mean packet delays (queue length 100)

ρ	μ_2/μ_1	Arrival Time	Postponed	PBD	BBD
0.8	0.2	-	-	-	-
0.9	0.2	0.001 ± 0.001	-	-	0.001 ± 0.001
0.95	0.2	0.005 ± 0.001	0.001 ± 0.001	0.002 ± 0.001	0.002 ± 0.001
0.99	0.2	0.015 ± 0.002	0.004 ± 0.001	0.008 ± 0.002	0.007 ± 0.001
0.8	0.6	-	-	-	-
0.9	0.6	0.001 ± 0.001	-	-	-
0.95	0.6	0.003 ± 0.001	0.001 ± 0.001	0.001 ± 0.001	0.001 ± 0.001
0.99	0.6	0.014 ± 0.002	0.005 ± 0.001	0.006 ± 0.001	0.006 ± 0.001
0.8	1	-	-	-	-
0.9	1	-	-	-	-
0.95	1	0.004 ± 0.001	0.001 ± 0.001	0.001 ± 0.001	0.001 ± 0.001
0.99	1	0.014 ± 0.003	0.005 ± 0.001	0.005 ± 0.001	0.006 ± 0.001

Table 4: The packet loss probabilities (queue length 100) "-" means that a reliable estimate was not obtained with in the simulation.)

ρ	μ_2/μ_1	Arrival time	Postponed	JSQ(packets)	JSQ(time)
0.8	0.2	8.21 ± 0.14	5.76 ± 0.081	6.21 ± 0.03	5.42 ± 0.03
0.9	0.2	16.86 ± 0.60	10.86 ± 0.40	11.29 ± 0.13	10.19 ± 0.12
0.95	0.2	34.36 ± 0.78	20.46 ± 0.39	20.91 ± 0.44	19.64 ± 0.35
0.99	0.2	142.82 ± 9.87	89.99 ± 6.83	90.80 ± 6.83	86.82 ± 6.33
0.8	0.6	9.60 ± 0.17	5.58 ± 0.09	6.03 ± 0.03	5.50 ± 0.03
0.9	0.6	19.64 ± 0.64	10.70 ± 0.41	11.18 ± 0.14	10.37 ± 0.12
0.95	0.6	39.03 ± 0.91	21.54 ± 1.59	21.26 ± 0.46	19.79 ± 0.43
0.99	0.6	159.52 ± 9.74	89.86 ± 6.84	90.46 ± 6.81	88.63 ± 6.64
0.8	1	9.78 ± 0.18	5.60 ± 0.09	6.26 ± 0.02	5.58 ± 0.03
0.9	1	20.21 ± 0.68	10.71 ± 0.41	11.21 ± 0.13	10.51 ± 0.13
0.95	1	39.43 ± 0.86	21.55 ± 2.11	20.78 ± 0.44	20.06 ± 0.44
0.99	1	156.84 ± 9.06	89.88 ± 6.84	90.63 ± 6.83	89.88 ± 6.84

Table 5: The mean packet delays (queue length infinite)

6 Conclusions

In this paper the simulation of a system consisting of two nodes with two parallel links was studied. The task was to compare the four different routing schemes by simulating them using the simulation library CNCL. Two of these schemes were static: one with arrival time decision making and the other with the routing decisions made at the service completion. The two dynamic schemes were both based on the queue length. The PBD scheme was based on the number of packets in the queues and the BBD scheme on the amount of data waiting to be served.

While comparing the static schemes, we noticed that by postponing the routing decisions, better performance was obtained. An exception to this was routing with small values of the ratio μ_2/μ_1 and traffic load ρ. If they were small enough, the smaller delay was obtained by using the routing scheme with arrival time decision making. However, this benefit obtained under light traffic was minor. With $\mu_2/\mu_1 = 1$ and traffic load $\rho = 0.95$ the mean packet delay could be halved by postponing the decisions. Another discovery concerning the static schemes was that in the scheme with postponed decision making the mean packet delay was almost independent of the value of μ_2/μ_1, while in the scheme with arrival time decision making the difference between delays with different ratios μ_2/μ_1 was greater.

Comparison between the dynamic routing schemes showed that, the BBD scheme was always more efficient than the PBD scheme. The difference was greatest for ratios μ_2/μ_1 close to 1 and light traffic load. Compared to the static schemes, the dynamic schemes were more efficient than the one with arrival time decision making, but approximately equally efficient compared to the scheme with postponed decision making.

In the simulations of this work packets arrive according to a Poisson process and the service times are exponentially distributed. An interesting subject to continue this work with could be simulations using different arrival processes and service time distributions.

References

[1] Samuli Aalto and Jorma Virtamo, "Basic packet routing problem", in Proceedings of NTS-13, Trondheim, August 20-22, 1996, pp. 85-97.

[2] "CNCL Reference Manual", 1998, available from http://www.comnets.rwth-aachen.de/doc/cncl/cncl_toc.html