Chapter 7

Route Division Multiple Access Performance Evaluation

7.1 Introduction

7.1.1 The Problem

This chapter addressed the problem of evaluating the performance of RDMA when compared with more traditional switching techniques: Packet Switching (PS) and Circuit Switching (CS). The study is carried using both analytical and simulation models.

7.1.2 Main Results

Analytical approximations. The time-dependent behavior of RDMA complicates the performance study. Usual analytical performance analysis techniques ignore time-dependent (or transient) behaviors when simplifying models in search for tractable solutions. This chapter presents approximate closed-form formulas for end-to-end delay and loss in RDMA-, RDMA+, and RDMA++. The analytical approximations are very accurate when checked with the simulation results.

Simulation studies. The natural rescue when analytical techniques fail is simulation. Even though completely realizable, simulation studies must be carefully implemented to

avoid extremely long executions and big state-spaces. This chapter presents simulation results on RDMA-, RDMA+, and RDMA++ performance.

Performance study. RDMA is compared favorably with respect to PS and CS for Poisson arrivals and bursty on-off arrivals. For low input loads, PS outperforms RDMA and CS because there is no admission delay waiting for a band or a circuit. When the load increases, frame header processors become saturated and then RDMA outperforms PS. CS always performs worst than RDMA because of the penalty incurred waiting for a circuit that is larger than the one waiting for a band. QoS parameters such as minimal delay and no jitter can be guaranteed with RDMA by using priority bands. Finally, when the propagation delay is large (in wide area communications), the time waiting for a band is negligible when compared with the propagation time. In this scenario, the performance advantages of PS with respect to RDMA become negligible.

7.1.3 Chapter Organization

The reminder of this chapter is organized as follows. Section 7.2 analyzes the performance of RMDA for a single node. Section 7.3 simulates the performance of RDMA for a single node. Section 7.4 simulates the performance of RMDA for a network of nodes. Section 7.5 describes future work. Finally, Section 7.6 summarizes.

7.2 Analysis: Contention for Outgoing Links

This section studies how RDMA resolves contention for an outgoing link. The situation is described in Figure 7.1 where a set of incoming links is competing for an outgoing link. The incoming links are partitioned in classes such that two links are in the same if and

only if they are part of the same routing tree. In the figure, there are T routing trees and tree i , $1 \le i \le T$, contains n_i links. Each link is labeled L[x,y], where x is the respective routing tree and y is the link order within the tree. The outgoing link needs to be time divided among the T trees. The goal of this section is to build a performance understanding of this scenario using analytical techniques.

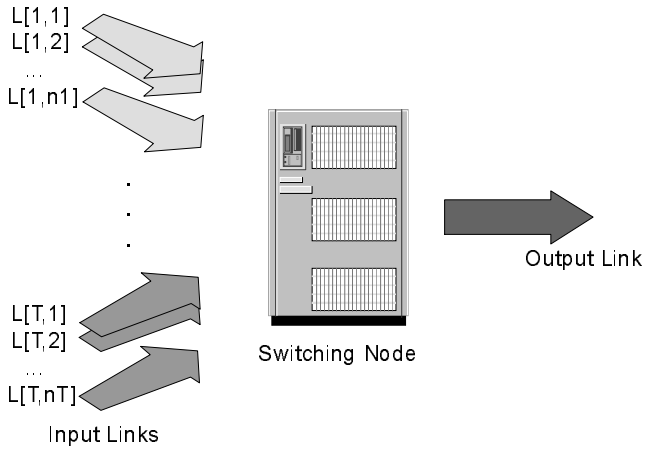

Figure 7.1: Contention for an outgoing link in RDMA

One interesting characteristic of RDMA is that each tree behaves independently of the others. From the perspective of a particular tree, the output link is available to it for a given period U and then becomes unavailable during a period V. The link is effectively a server that can service frames at a given link rate. The server cycle between two states, an operational state of duration U and a vacation state of duration V. The cycle is $C = U + V$. From this point onwards, the study will concentrate on one such independent trees.

It is assumed that the tree under study contains n links. The arrival process is Poisson with mean $\frac{\lambda}{n}$ in each link. It can be thus concluded [Kleinrock 75] that the merged input

stream has a Poisson distribution $A(t)$ with mean λ. The Poisson assumption in each link can be justified by noticing that incoming traffic to a contention band is usually random and does not require QoS guarantees.

The frame sizes are distributed according to a generic distribution $B(x)$ with mean μ. The whole tree can be modeled as an M/G/1 system with input rate λ and service rate μ that cycles between an active period of size U and a vacationing period of size V. The system utilization is $\rho = \frac{\lambda}{\mu}$. Notice that, for the system to be stable, $\rho < \frac{U}{U+V}$.

7.2.1 The RDMA- System

Under RDMA- regimen, it is assumed that sources will generate traffic according to the Poisson distribution during each band. If the traffic sources are not Poisson in nature, a filter can be placed between the sources and the input to the system to decrease the chances of burst losses. The filter will output traffic according to the Poisson distribution from any generic input traffic. The filter is thus a G/M/1 system and can be analyzed accordingly [Kleinrock 75] in order to find its contribution in the overall end-to-end frame delay.

If the band size U is large compared to the mean transmission rate μ, then one can assume that the RDMA- system operates like a Type I Counter described in [Karlin and Taylor 75] and summarized here. The Type I Counter registers instantaneous signals. Signals arrive according to a generic arrival distribution $A(t)$ and take some time to be recorded also generic with distribution $B(x)$. While a signal is being recorded, other signals are lost. The similarity with RDMA- is apparent if one replaces the transmission process

for the recording process. While a packet is being transmitted in RDMA-, other arriving packets are lost.

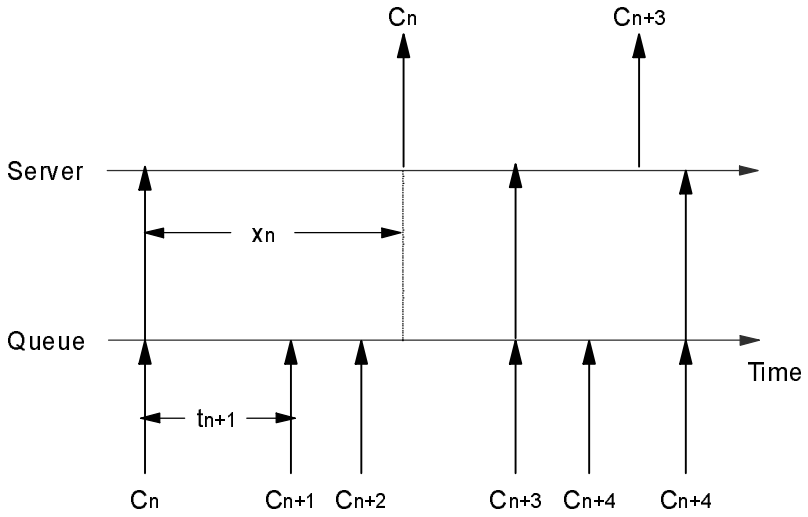

Figure 7.2: RDMA- model

Figure 7.2 illustrates the model for the RDMA- systems. C_n is the n-th frame to arrive to the system, t_{n+1} is the interval between the arrival of frames C_n and C_{n+1}, and x_n is the time to transmit frame n. The figure depicts three stages of the process: frames arriving to the queue, frames leaving the queue and immediately entering the server (that is, starting transmissions), and frames leaving the server. The arrows represent frames and the horizontal lines the queue and the server. If the link is not transmitting any information, the arriving frame is immediately serviced. Otherwise, it is lost. In the figure, only frames C_n, C_{n+3}, and C_{n+4} are not lost.

Following [Karlin and Taylor 75], let z_i denote the interval of time between the i-th and (i+1)-th successful transmissions, γ_t be the residual life of t_i at time t, $S_k = \sum_{i=1}^{k} t_i$ be

the time of the *k*-th arrival, and $N(t) = \sup\{n: S_n \le t\}$ be the number of arrivals to the system by time *t*. The residual life γ_t of t_i is the time remaining for the arrival of the next frame at time *t*. Then z_1 can be computed as:

$$z_1 = x_1 + \gamma_{x_1} = S_{N(x_1)+1} \tag{7.1}$$

That is, the interval between the first and second successful transmissions is equal to the time to transmit the first frame plus the residual time for the arrival of the next frame after the first finished being transmitted.

Example 7.1: Distribution of the inter-departure time when the service has a deterministic distribution.

If the service time is always $\dot{x}$, then $B(x) = \begin{cases} 1 \text{ if } x \ge \dot{x} \\ 0 \text{ if } x < \dot{x} \end{cases}$, where $\frac{1}{\mu} = \dot{x}$.

Then, if $z > \dot{x}$:

$$\Pr\{z_1 \le z\} = \int_0^{z-\dot{x}} \lambda e^{-\lambda x} dx = 1 - e^{-\lambda(z-\dot{x})}$$

and if $z \le \dot{x}$:

$$\Pr\{z_1 \le z\} = 0.$$

This example shows the inter-departure distribution when ATM [De Prycker 93, Le Boudec 92] cells are being transmitted. The time to transmit an ATM cell is $\dot{x}$.

Since the processes $\{t_n\}$ and $\{x_n\}$ are independent, the law of total probabilities gives:

$$\Pr\{z_1 \le z\} = \int_0^z \Pr\{x + \gamma_x \le z | x_1 = x\} dB(x) \tag{7.2}$$

Defining the distribution of γ_t as $\hat{A}_w(t) = \Pr\{\gamma_t > w\}$ (an explicit formula for it can be found in [Karlin and Taylor 75]):

$$\Pr\{z_1 \le z\} = \int_0^z \{1 - \hat{A}_{z-x}(x)\} dB(x) \tag{7.3}$$

Since the arrival process is Poisson, the memory-less property of the interarrival exponential distribution leads to the conclusion that γ_t follows an exponential distribution. Thus for RDMA-:

$$\Pr\{z_1 \le z\} = \int_0^z B(z-x)\lambda e^{-\lambda x} dx \tag{7.4}$$

and:

$$\Pr\{z_1 > z\} = 1 - \int_0^z B(z-x)\lambda e^{-\lambda x} dx \tag{7.5}$$

But:

$$\int_0^z \lambda e^{-\lambda x} dx = 1 - e^{-\lambda z} \Leftrightarrow \int_0^z \lambda e^{-\lambda x} dx + e^{-\lambda z} = 1 \tag{7.6}$$

Replacing the number 1 in Equation 7.5 by the number 1 in Equation 7.6:

$$\Pr\{z_1 > z\} = \int_0^z \lambda e^{-\lambda x} dx + e^{-\lambda z} - \int_0^z B(z-x)\lambda e^{-\lambda x} dx \tag{7.7}$$

$$\Pr\{z_1 > z\} = e^{-\lambda z} - \int_0^z \{1 - B(z-x)\}\lambda e^{-\lambda x} dx = e^{-\lambda z} - \int_0^z \overline{B}(z-x)\lambda e^{-\lambda x} dx \tag{7.8}$$

Finally, one can conclude that Equation 7.8 is valid for any z_n because $\{z_n\}$ are the inter-arrival times of a renewal process [Karlin and Taylor 75].

Examples 7.1 and 7.2 illustrate how Equation 7.4 can be used to compute the distribution of the inter-departure times of successfully transmitted frames.

Example 7.2: *Distribution of the inter-departure time when the service has a Poisson distribution.*

In this case, $B(x) = 1 - e^{-\mu x}$.

Then:

$$\Pr\{z_1 \le z\} = \int_0^z (1 - e^{-\mu(z-x)})\lambda e^{-\lambda x} dx = \int_0^z \lambda e^{-\lambda x} dx - \lambda e^{-\mu z} \int_0^z e^{(\mu-\lambda)x} dx$$

$$\Pr\{z_1 \le z\} = 1 - e^{-\lambda z} - (\lambda e^{-\mu z} \frac{e^{(\mu-\lambda)z} - 1}{\mu - \lambda}) = 1 - e^{-\lambda z} - \lambda \frac{e^{-\mu z} - e^{-\lambda z}}{\lambda - \mu}$$

$$\Pr\{z_1 \le z\} = 1 - \frac{\lambda e^{-\mu z} - \mu e^{-\lambda z}}{\lambda - \mu}$$

This example shows the inter-departure distribution when ATM cells are being transmitted in batches of random sizes.

The reminder of this section uses Equation 7.8 to derive the mean loss rate in RDMA-. One can observe that frames are being transmitted at rate $\frac{1}{E[z]}$ while they are being received at rate $\frac{1}{E[x]} = \lambda$. Thus, the packet loss rate L is:

$$L = 1 - \frac{1/E[z]}{1/E[x]} = 1 - \frac{1}{\lambda E[z]} \tag{7.9}$$

The expectation can be computed as:

$$E[z] = \int_0^\infty \Pr\{z_1 > z\} dz = \int_0^\infty e^{-\lambda z} dz + \int_0^\infty \int_0^z \overline{B}(z-x)\lambda e^{-\lambda x} dx dz \tag{7.10}$$

But:

$$\int_0^\infty e^{-\lambda z} dz = \frac{1}{\lambda} \tag{7.11}$$

and:

$$\begin{aligned}\int_0^\infty \int_0^z \overline{B}(z-x)\lambda e^{-\lambda x} dx dz &= \int_0^\infty \int_x^\infty \overline{B}(z-x)\lambda e^{-\lambda x} dz dx \\ &= \int_0^\infty \lambda e^{-\lambda x} \int_x^\infty \overline{B}(z-x) dz dx \\ &= \int_0^\infty \lambda e^{-\lambda x} \frac{1}{\mu} dx \\ &= \frac{1}{\mu}\end{aligned} \tag{7.12}$$

Replacing Equations 7.11 and 7.12 in 7.10, one obtains:

$$E[z] = \frac{1}{\lambda} + \frac{1}{\mu} = \frac{\lambda + \mu}{\lambda\mu} \tag{7.13}$$

Finally:

$$L = 1 - \frac{1}{\lambda(\frac{\lambda+\mu}{\lambda\mu})} = \frac{\lambda}{\lambda+\mu} = \frac{\rho}{\rho+1} \tag{7.14}$$

An interesting conclusion that can be derived from Equation 7.14 is that under RDMA- policy, when $\rho \to 1$, $L \to 0.5$. That is, the upper bound on the loss rate for Poisson arrivals and generic service time is 50%.

The delay in the node in RDMA- is equal to the service time of a frame. The mean delay D is thus:

$$D = \frac{1}{\mu} \tag{7.15}$$

7.2.2 The RDMA++ System

RDMA++ is a challenging system to analyze because of its time dependent behavior. From the point of view of an end node, the server is the routing tree. It is active during its band and then goes on vacation during other bands. A complete state description of such a

system must include the current server status and how long it will last. References [Federgruen and Green 86, Ott 87] study systems with deterministic time dependent vacations and can provide only limited insight on the system behavior.

This section presents an approximation for the average delay and loss in the RDMA++ system when the queueing discipline is M/G/1 and the system load is low to medium.

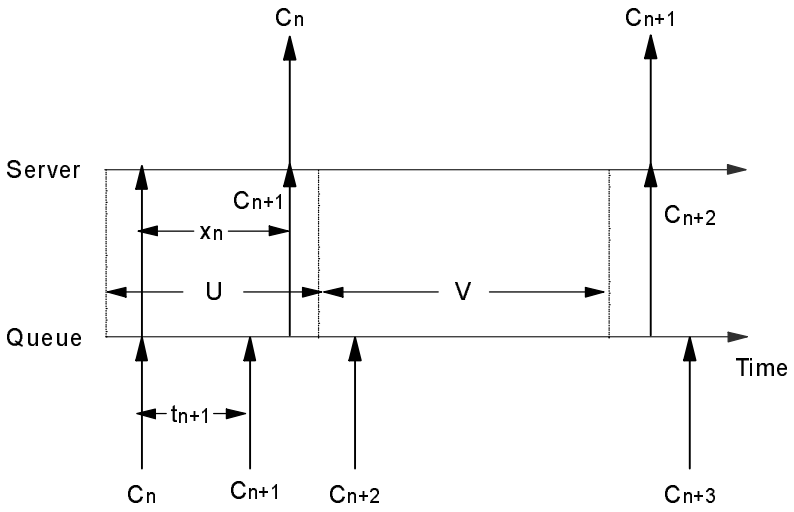

Figure 7.3: RDMA++ model

Figure 7.3 illustrates the model for the RDMA++ system. The terminology is the same as in Figure 7.2. Frame C_n arrives to an empty system and is serviced immediately. C_{n+1} waits until C_n is finished and then begins to be serviced. During its service time, the server goes on vacation and then returns to complete the service. Only then is C_{n+2} admitted into service.

Following [Bertsekas and Gallager 92], let w_i be the waiting time in the queue for the i-th frame, q_i be the number of frames found waiting in the queue by the i-th frame upon arrival, and r_i the residual service time seen by the i-th arriving frame. The residual service

time is the service time remaining for the frame in service upon i's arrival. Similarly, v_i is the residual vacation time seen by frame i upon arrival. Then:

$$w_i = r_i + v_i + \sum_{j=i-q_i}^{i-1} x_j + \left\lfloor \left(r_i + \sum_{j=i-q_i}^{i-1} x_j \right) \Big/ U \right\rfloor V \tag{7.16}$$

That is, the waiting time for frame i upon arrival is equal to the sum of four components: (1) residual service time for the frame in service, (2) the residual vacation time, (3) the time to service all frames in the queue, and (4) the vacations incurred while servicing the frames in front of the arriving one. This last component is vacation period V multiplied by the total number of cycles necessary to service the frame in service upon arrival plus the ones in the queue. The number of cycles in question is the floor of the total time to service the frames in the queue and the one in service upon arrival divided by the total operational time in each cycle.

The next step is to take the expectation of the terms in Equation 7.16. Nonetheless, expectation of the last term of the summation in the right side is difficult to compute. One can recur to the following approximation for low loads. From the point of view of the arriving frame, it is very likely that all work in front of it will be finished during one active period because the load in the system is small. In this case, $\left\lfloor \frac{1}{U}\left(r_i + \sum_{j=i-N_i}^{i-1} x_j \right) \right\rfloor = 0$.

The approximate expectation of the terms in Equation 7.16 the becomes:

$$E[w_i] = E[r_i] + E[v_i] + E[\sum_{j=i-q_i}^{i-1} E[x_j | q_i]] \tag{7.17}$$

$$E[w_i] = E[r_i] + E[v_i] + \frac{1}{\mu} E[q_i] \tag{7.18}$$

Taking the limit as $i \to \infty$:

$$E[w] = E[r] + E[v] + \frac{1}{\mu} E[q] \tag{7.19}$$

But by Little's Law [Kleinrock 75], $E[q] = \lambda E[w]$ and so:

$$E[w] = E[r] + E[v] + \frac{1}{\mu} \lambda E[w] \tag{7.20}$$

Finally:

$$E[w] = \frac{E[r] + E[v]}{1 - \rho} \tag{7.21}$$

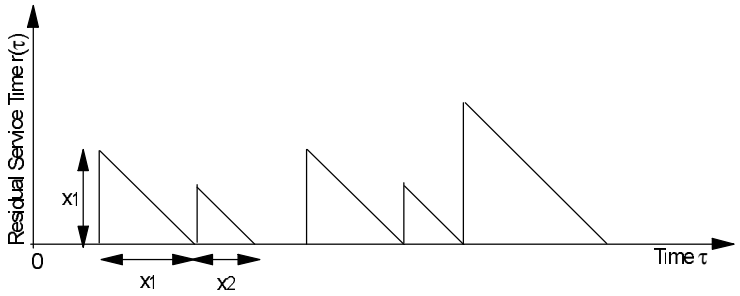

Figure 7.4: Residual service time

The mean residual service and vacation times can be computed using a graphical argument [Bertsekas and Gallager 92]. The residual service time is depicted in Figure 7.4. The residual time at *t*, *r*(*t*) is the time left to complete the frame transmission. When a new frame arrives, *r*(*t*) becomes equal to the service time for the frame. It then decreases linearly with time until it vanishes. The behavior is repeated for the next frame and so on.

The mean residual time at *t* can be computed from the time average of *r*(*t*), given that the number of frames serviced by time *t* is *M*(*t*), as:

$$\frac{1}{t} \int_0^t r(\tau) d\tau = \frac{1}{t} \sum_{i=1}^{M(t)} \frac{1}{2} x_i^2 \tag{7.22}$$

That is, the time average sum of the areas of all triangles in Figure 7.4 by time *t*. Equation 7.22 can also be written as:

$$\frac{1}{t}\int_0^t r(\tau)d\tau = \frac{1}{2}\frac{M(t)}{t}\frac{\sum_{i=1}^{M(t)} x_i^2}{M(t)} \tag{7.23}$$

Taking the limits in Equation 7.23:

$$\lim_{t\to\infty}\frac{1}{t}\int_0^t r(\tau)d\tau = \frac{1}{2}\lim_{t\to\infty}\frac{M(t)}{t}\cdot\lim_{t\to\infty}\frac{\sum_{i=1}^{M(t)} x_i^2}{M(t)} \tag{7.24}$$

The first term in the right hand side multiplication is the one half of the input rate while the second is the expectation of the service time squared. The left hand side is the expectation of the residual service time. Formalizing this observation:

$$E[r] = \frac{1}{2}\lambda E[x^2] = \frac{1}{2}\lambda\overline{x^2} \tag{7.25}$$

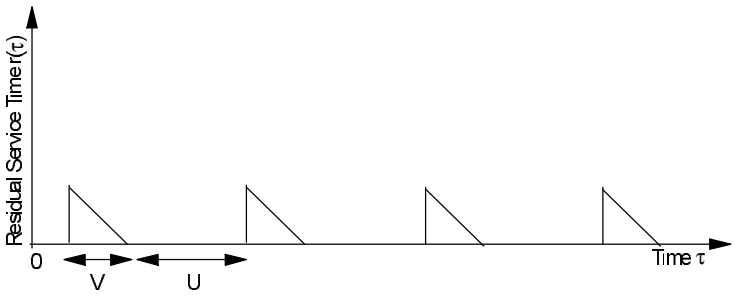

Figure 7.5: Residual vacation time

The mean residual vacation can be computed similarly to the mean residual service time. Figure 7.5 depicts the residual vacation time, where the interpretation is similar to the one for Figure 7.4. The average residual vacation is:

$$\frac{1}{t}\int_0^t v(\tau)d\tau = \frac{1}{t}\cdot\left\lfloor\frac{t}{V+U}\right\rfloor\cdot\frac{1}{2}V^2 \tag{7.26}$$

That is, the time average of the areas of the triangles in Figure 7.5 by time *t*. Taking the limits in Equation 7.28:

$$E[v] = \lim_{t\to\infty} \frac{1}{t}\int_0^t v(\tau)d\tau = \lim_{t\to\infty} \frac{1}{t}\cdot\frac{t}{V+U}\cdot\frac{1}{2}V^2 = \frac{1}{2}\frac{V^2}{V+U} \tag{7.27}$$

Replacing Equation 7.25 and 7.27 in Equation 7.21:

$$E[w] = \frac{1}{2}\frac{\lambda\overline{x^2}}{(1-\rho)} + \frac{1}{2}\frac{V^2}{(U+V)}\frac{1}{(1-\rho)} \tag{7.28}$$

From which it can be concluded that:

$$D = \frac{1}{2}\frac{\lambda\overline{x^2}}{(1-\rho)} + \frac{1}{2}\frac{V^2}{(U+V)}\frac{1}{(1-\rho)} + \frac{1}{\mu} \tag{7.29}$$

Notice that Equation 7.28 can be interpreted as follows. The first component of the summation is the normal waiting time of an M/G/1 system. The second component represents the penalty due to the residual vacation time. It is equal to the mean vacation period ($\frac{V}{2}$) times the probability of being on vacation ($\frac{V}{U+V}$) divided by the mean idle period ($1-\rho$) to obtain the total waiting time due to residual vacations.

Finally, there is no loss in RDMA++ and thus:

$$L = 0 \tag{7.30}$$

An interesting conclusion that can be derived from the RDMA++ analysis is that the same approximation for mean delay can be used in CS. CS operates similarly to RDMA++. The circuit comes periodically within a cycle, in similarity to a band arriving within a cycle. In conclusion, Equations 7.29 and 7.30 are also valid approximations for CS where *U* should be interpreted as the circuit duration.

7.2.3 The RDMA+ System

If the band size U is large compared to the mean transmission rate μ, then one can assume that RDMA+ operates in similarity to a M/G/1 queueing system with periodic losses. Figure 7.6 illustrates the typical behavior of an RDMA+ system. Frames behave like in RDMA++, except when the band ends. At that point, all frames in the system are lost. For example, in the figure, frame C_{n+1} is lost.

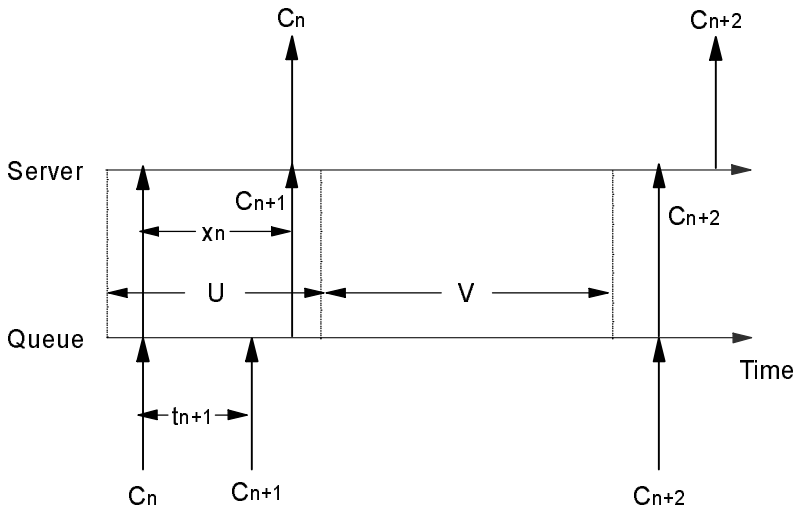

Figure 7.6: RDMA+ model

The model adopted here is just an M/G/1 queueing system. Delays are computed directly using M/G/1 delay formulas [Kleinrock 75]. For example, the mean delay is:

$$D = \frac{\lambda \overline{x^2}}{2(1-\rho)} + \frac{1}{\mu} \tag{7.31}$$

The delay distribution can be computed similarly using the M/G/1 formulas.

The loss probability can be approximated observing the following. If the band is big enough such that the system eventually becomes stable, then the probability that n frames will be lost at the end of the band is equal to the probability that n frames are in the system

at any time. This is the distribution of the number in the system for an M/G/1 system. The average number in the system is:

$$\overline{N} = \frac{\lambda^2 \overline{x^2}}{2(1-\rho)} + \rho \tag{7.32}$$

From which it can be concluded that the average loss in a band of size U is:

$$L = \frac{\frac{\lambda^2 \overline{x^2}}{2(1-\rho)} + \rho}{\lambda U} = \frac{\lambda \overline{x^2}}{2U(1-\rho)} + \frac{1}{\mu U} \tag{7.33}$$

It is interesting to notice in RDMA+ that when $U \to \infty,\ L \to 0$.

7.2.4 Network of Nodes

Somewhat surprisingly, the results in the previous sections can be used to approximate the behavior in a network of nodes. This section gives some insight on why the approximations are also valid for a network.

In RDMA, the dynamics in each routing tree are independently of the ones in other routing trees. For this reason, one can take each tree in isolation and analyze it. Additionally, each tree can be seen as a server with multiple incoming links, as in Figure 7.1. There are only a few reasons why both models, the tree model and the node model, are not equivalent. All of them have to do with the number of servers and links that each frame crosses and with the order in which frames are serviced.

The first is that the order of service can vary. In the tree model, sources that are close to the destination are serviced before sources that are distant from the destination. Nevertheless, the order of service is not important to compute both the mean delay and loss, so this difference can be ignored.

The second is that the number of servers that each frame crosses varies. This can be accounted for by analyzing each outgoing link in each tree separately. Then, the results (for delay and loss) may be combined using proper weights (based on the load of each link) to find the end-to-end delay.

The third is that the total propagation delay per frame may also vary according to how many links it crosses. This can be accommodated similarly by computing the average propagation delay taking into account the percentage of frames that will incur the given delay.

Finally, if the service distribution is not Poisson, traffic being input from a node will not present a Poisson distribution when input to the next node. One can approximate the result by assuming that the input to each link is indeed Poisson at each stage of the tree.

One can observe that these are common practices in analyzing network. For example, Reference [Kleinrock 76] makes similar assumptions and can be used to understand how the analysis on a single node can be extended to a network of nodes using the guidelines presented in this section.

7.3 Simulation: Contention for Outgoing Links

The goal of this section is twofold. Firstly, it validates the analysis in Section 7.1 with the simulation of the system depicted in Figure 7.1. Secondly, it compares the behavior of RDMA with PS and CS in a node.

The physical system studied is the one in Figure 7.1 with the following features. There are 10 input links to the node and 1 outgoing link that is shared among the inputs. Each link operates either at 2.4 Gb/s or at 24 Gb/s and transmits ATM cells (each 53 bytes in

size). The arrival process is Poisson. The frame size is either deterministic or distributed according to an exponential distribution. For the RDMA simulations, the cycle period is 125 μs. The band size varies in each case. Finally, the network topology is such that there are 5 bands in each cycle.

7.3.1 RDMA++

The purpose of the study is to show how closely the approximated and simulated RDMA++ mean frame delays match each other. As expected, the approximation works best for low to medium network loads in all experiments conducted.

Figure 7.7 illustrates mean delay when the frame size is fixed at 53 bytes, and the maximum link rate is 2.4 Gb/s. For this figure and the following, the study varies the input load up to 90% of its maximum capacity to avoid unstable behaviors as $\rho \to \frac{U}{U+V}$. The input load axis is measures in thousands of frames per second (Kframes). For example, at 73.584906 Kframes/s, the nominal input rate is 31.2 Mb/s in each input link. Because there are 10 input links, the aggregate input to the outgoing link is 312 Mb/s or 13% of the maximum rate of 2.4 Gb/s. Notice that, because the band size is 20% of the cycle, the maximum aggregate input rate to the outgoing link must be less than 20% or else the system will be unstable.

In Figure 7.8, the maximum link rate is 2.4 Gb/s, and the frame size is exponentially distributed with mean 53 bytes.

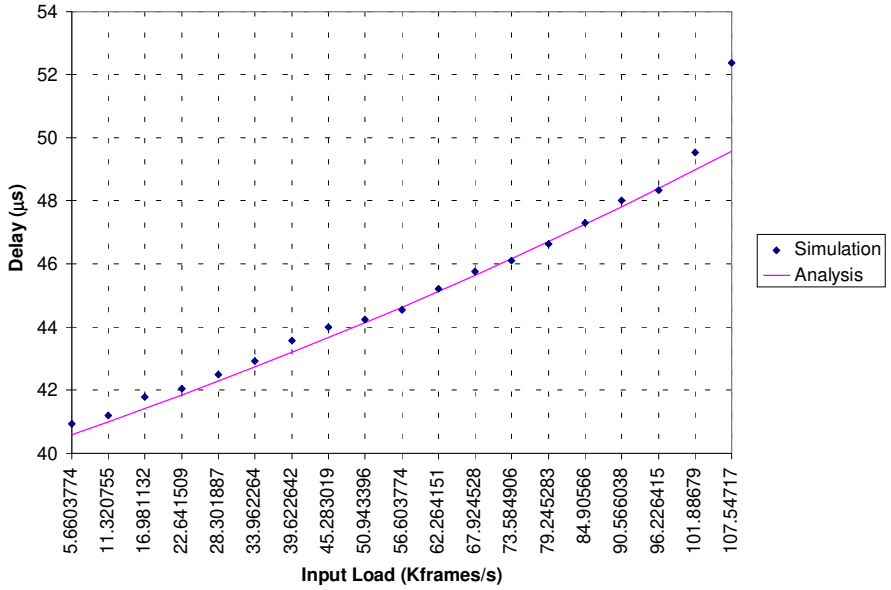

Figure 7.7: Mean frame delay (in µs) for Poisson arrivals and deterministic service under RDMA++ (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 25 µs, the cycle period 125 µs)

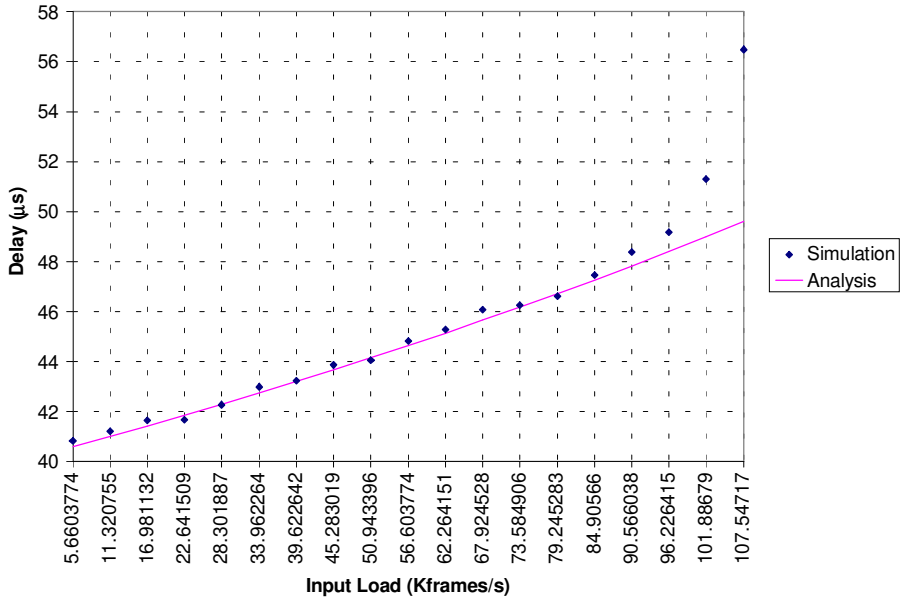

Figure 7.8: Mean frame delay (in µs) for Poisson arrivals and Poisson service under RDMA++ (the mean frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 25 µs, the cycle period 125 µs)

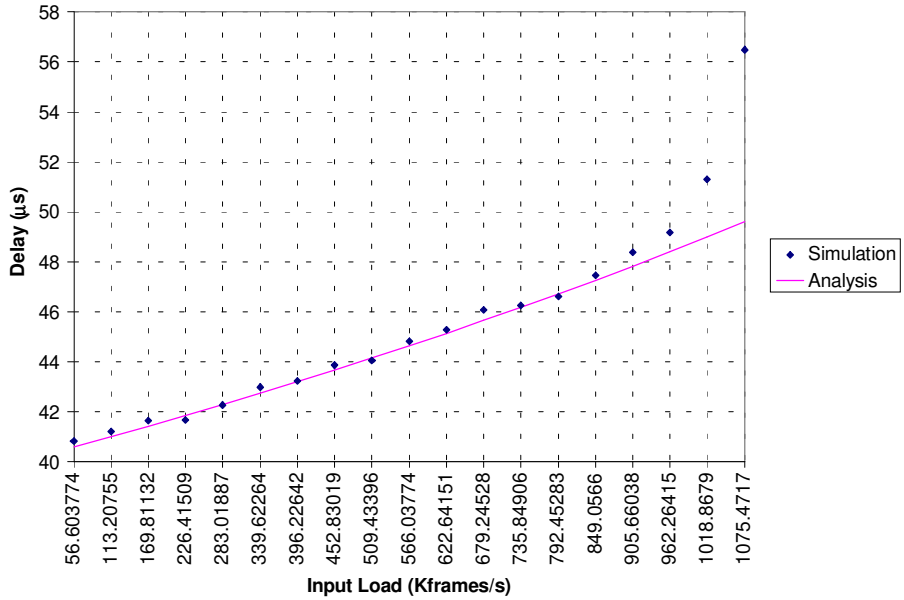

Figure 7.9: Mean frame delay (in µs) for Poisson arrivals and Poisson service under RDMA++ (the mean frame size is 530 bytes, the link speed 24 Gb/s, the band size 25 µs, the cycle period 125 µs)

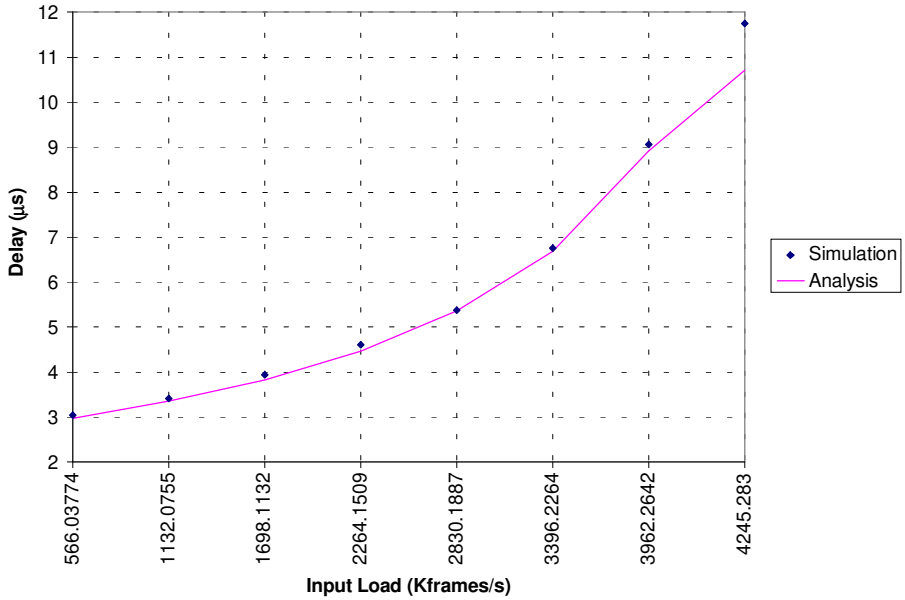

Figure 7.10: Mean frame delay (in µs) for Poisson arrivals and deterministic service under RDMA++ (the frame size is 530 bytes, the link speed 24 Gb/s, the band size 100 µs, the cycle period 125 µs)

Figure 7.9 repeats an experiment similar to the one in Figure 7.8 but with a faster link speed of 24 Gb/s and a mean frame size of 530 bytes. One possible interpretation for this experiment is that arrivals are bursty with mean burst size of 10 ATM cells. The figure displays the mean delay per burst.

Figure 7.10 depicts an experiment similar to the one in Figure 7.9 but with a bigger band size of 100 μs, or 80% of the cycle. Again, one can see the same trend as in the other experiments.

Finally, Figure 7.11 depicts the case of a CS system with fixed size ATM cells and 2.4 Gb/s maximum link rate. Each circuit lasts 2.5 μs and the cycle period is 125 μs. One can see that again the approximation matches the simulation for low to medium loads.

7.3.2 RDMA-

Figure 7.12 depicts the loss rate in RDMA- when the cycle period is 125 μs, the band size 25 μs, the arrival is Poisson within a band, and the frame size fixed at 53 bytes. Similar results can be obtained for exponentially distributed frame sizes. The input load axis indicates the ratio of the total input load in each input link during a band. For example, the ratio 0.8 indicates that the input links are transmitting at 1.92 Gb/s. One can see that the simulation closely matches the analysis and that the maximum loss in the system is 50%, as expected.

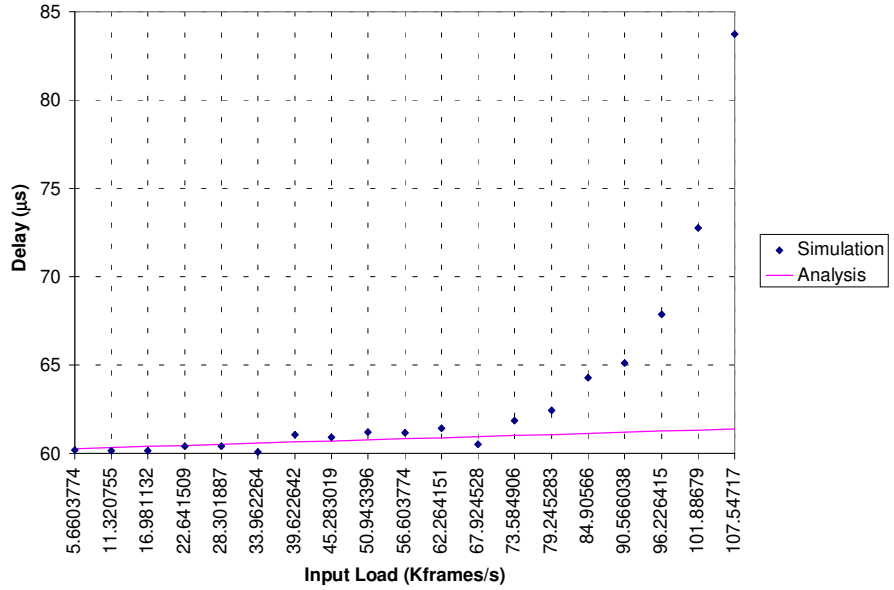

Figure 7.11: Mean frame delay (in µs) for Poisson arrivals and deterministic service under CS (the frame size is 53 bytes, the link speed 2.4 Gb/s, the circuit size 2.5 µs, the cycle period 125 µs)

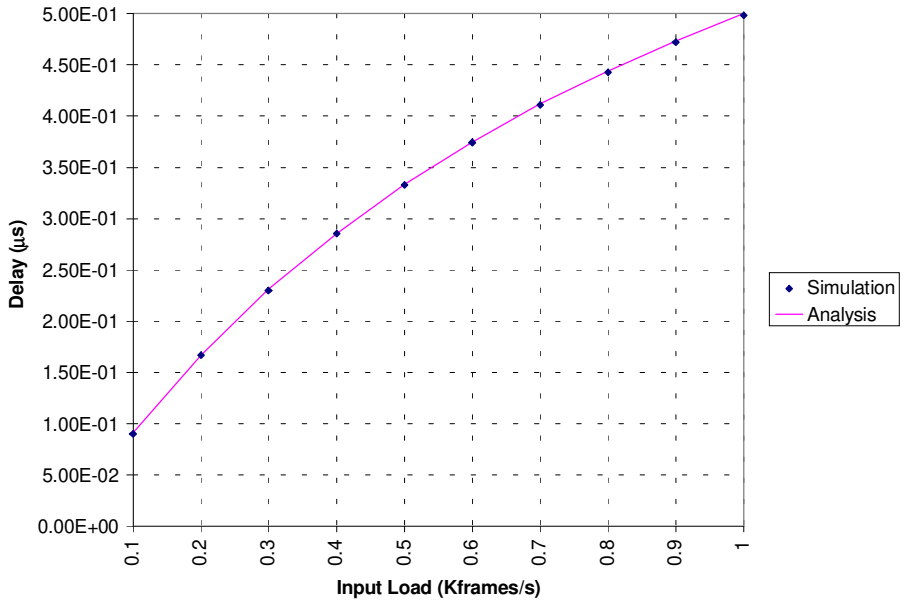

Figure 7.12: Mean loss rate for Poisson arrivals and deterministic service under RDMA- (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 25 µs, the cycle period 125 µs)

7.3.3 RDMA+

Figure 7.13 illustrates the mean delay for RDMA+ with Poisson arrivals within a band. One can see that the analysis is a good approximation of the delay in the system, especially at low loads.

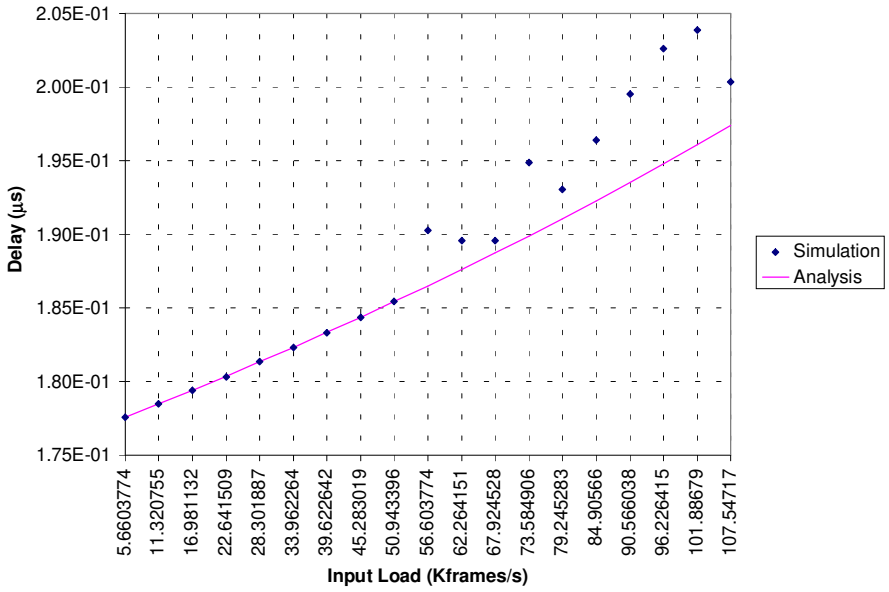

Figure 7.13: Mean frame delay (in µs) for Poisson arrivals and deterministic service under RDMA+ (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 25 µs, the cycle period 125 µs)

7.3.4 Comparison

The goal of this section is to compare RDMA++ performance with PS and CS.

The CS simulation works as follows. There is a cycle of 125 µs within which end-to-end connections can be allocated. Each connection lasts 2.5 µs. This ensures a fair comparison with RDMA++ where a band of size 25 µs is shared among 10 sources.

CS is expected to perform always worst than RDMA++ because the time waiting for an end-to-end connection to arrive is bigger than the time waiting for a band. In the par-

ticular case being simulated, for instance, each band lasts 25 μs and is repeated every 125 μs while each end-to-end circuit lasts 2.5 μs and is also repeated every 125 μs.

The PS simulation works as follows. There is a processor operating at the same nominal speed of one of the input links. This means, for example, that if 50 instructions are necessary to process each ATM cell, the simulated processor operates at 283 MIPS, which is an optimistic assumption. Additionally, there could be contention for processor use from frames crossing the node to other destinations. Such interference does not happen in RDMA or CS. It is reflected in the simulation by making all input traffic from each link incur some delay to be processed at the node. The number of possible destinations for each input frame is 10, and so the overall input traffic to be processed is 10 times larger than the traffic that will eventually follow the correct output link in Figure 7.1.

It is expected that PS will perform better than RDMA until its processing resources are exhausted at which point RDMA will outperform PS.

The input load axis is actually the load from each input link to the output link. For the PS simulation, the input load to the processor from each input link is 10 times this value. The reason is that the PS simulation generates traffic to all 10 possible destinations and all frames need to cross the processor to be switched.

Figure 7.14 and Figure 7.15 show the mean delay for the three systems. In the first, the link speed is 2.4 Gb/s and the frame size fixed at 53 bytes. In the second, the link speed is 24 Gb/s and the frame size exponentially distributed with mean 530 bytes. As it can be verified, both experiments are in accordance with the expectations.

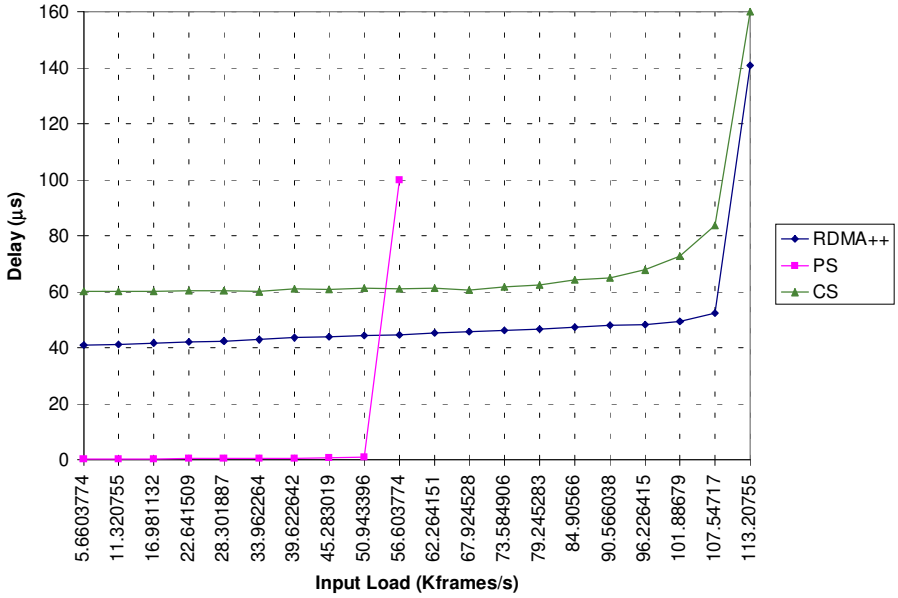

Figure 7.14: Mean frame delay comparison (in µs) for Poisson arrivals and deterministic service (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 25 µs, the cycle period 125 µs)

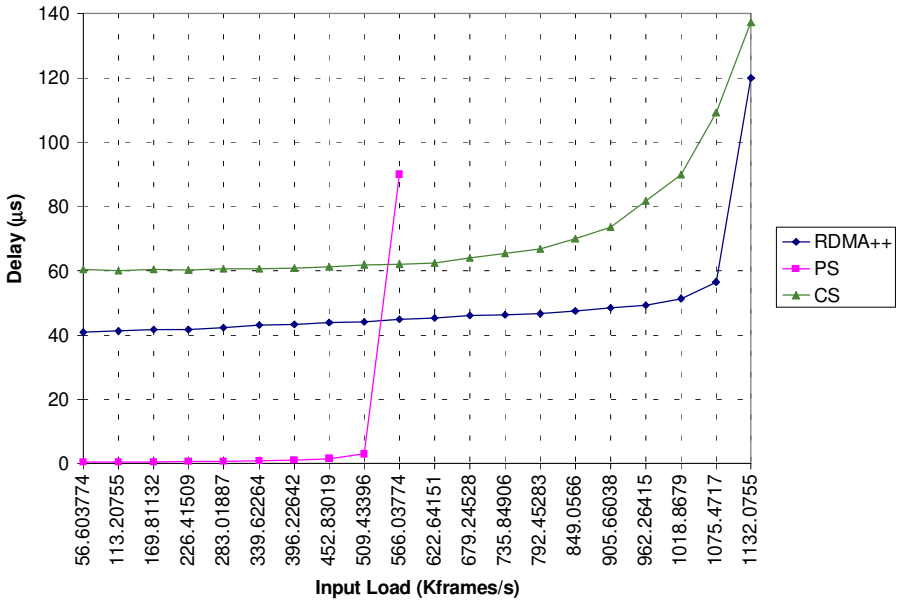

Figure 7.15: Mean frame delay comparison (in µs) for Poisson arrivals and Poisson service (the frame size is 530 bytes, the link speed 24 Gb/s, the band size 25 µs, the cycle period 125 µs)

Notice that PS becomes unstable in both simulations when the input load in each link is about 10% of its maximum capacity. At this point, the traffic to the output link in Figure 7.1 is 566.03774 Kframes/s (as depicted in Figure 7.14 and Figure 7.15) or 1% of the maximum capacity. The remaining 9% are directed to other destinations. The overall input to the processor from the 10 links is 100% of the processor capacity. That is, only 10% of the traffic is directed to the output link, but the remaining 90% still consume processor cycles to be switched. CS and RDMA++ become unstable at 2% of the input link capacity at which point the aggregate traffic for the output link is 20%, which is also the portion of the cycle allocated to the output link.

7.4 Simulation of Networks

This section extends the performance studies of RDMA to a network of nodes. As it will be seen, the results are similar to the ones obtained in the previous section.

The topology studied is depicted in Figure 7.16. It is a symmetric configuration that allows the overlapping of 3 non-interfering trees. An example of 3 non-interfering trees to destinations 1, 6 and 8 is depicted in Figure 7.17. These destinations thus can share a band. Two additional bands are sufficient to serve the 6 trees of the other destination nodes.

The simulation model works as follows. Each node generates ATM cells according to a Poisson process. Destinations are assigned to cells according to a uniform distribution. The link speed is 2.4 Gb/s, that is, each cell needs 177 ns to be transmitted. The cycle period is 125 μs. The propagation delay in each link is negligible (equivalent to 1 cell transmission delay). The bands to all destinations are of the same size. Each cell waits for the

proper destination band at the source nodes and then moves through the network down the respective tree. The goal is to give a comparison of PS, CS, and RDMA++ performance.

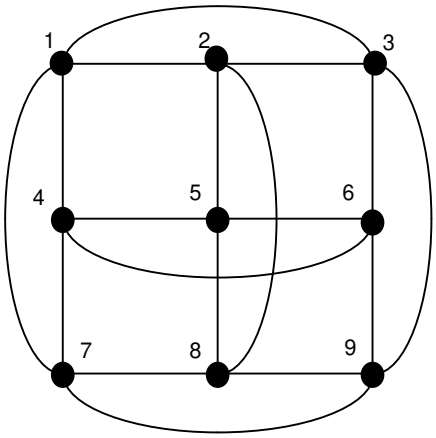

Figure 7.16: Simulated network topology

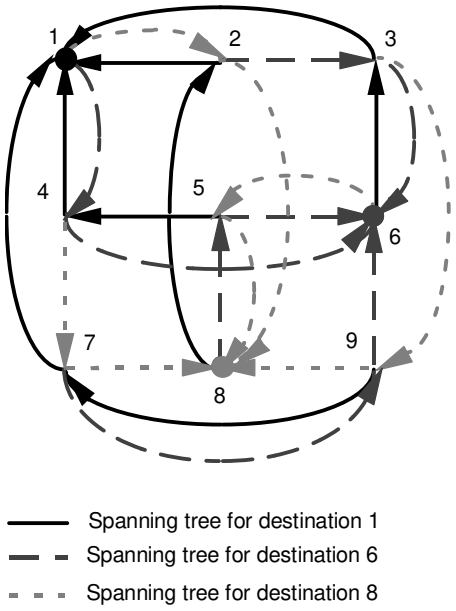

Figure 7.17: Allocation of trees in one band

The PS and CS simulations operate similarly to the ones in Section 7.3, use the same trees allocated for RDMA++ (see Figure 7.16), and cut-through technology [Kermani 78].

The CPU at each switch operates at the same rate of one input link. This means, for example, that if 50 instructions are necessary to process each ATM cell, the simulated processor operates at 283 MIPS, which is an optimistic assumption.

The simulation was run for periods where 10,000 ATM cells per input node were generated. After each of these periods, all statistics were saved and reset. Two RDMA++ experiments were conducted. In the first, all traffic had the same priority (RDMA++<c>). In the second, priority traffic was generated as follows. Each band was equally partitioned into priority sub-bands, one for each input node (RDMA++<p>).

Figure 7.18 depicts the mean packet delay (in μs) for the experiments. The input traffic load is given as a percentage of the 2.4 Gb/s maximum input rate at each node. PS has a steady performance until the input load 50% saturates the CPU capability at the nodes with delays growing unbounded. CS has a similar behavior. The unstable point is 30%. The reason is that this CS simulation allocates all nodes in its path (including the destination). It becomes unstable before RDMA++ because the latter allocates the destination node for the whole tree and not for a single circuit. An alternative CS operation could potentially be improved to allocate only links (and not nodes). In this case, the comparison is the same reported in Figure 7.11.

RDMA++ has a stable performance. Both RDMA++<c> and RDMA++<p> have the same mean packet delay characteristics, as expected from queueing analysis [Kleinrock 75], and thus overlap in the figure. The “Priority” curve plots the mean delay for priority traffic generated for the RDMA++<p> experiment. Priority was assigned randomly to ATM cells according to a uniform distribution. Priority traffic was scheduled to access the

network during its priority band, thus not incurring admission delays. The delay incurred by the priority traffic is only the propagation delay and contention with other cells scheduled at the beginning of the priority band.

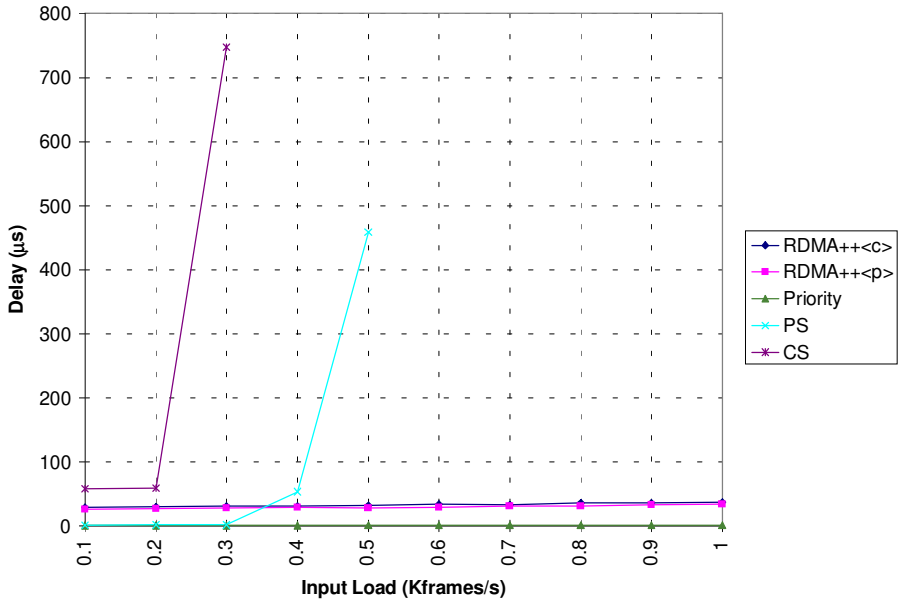

Figure 7.18: Mean frame delay comparison (in µs) for Poisson arrivals and deterministic service (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 41.67 µs, the cycle period 125 µs)

Figure 7.19 shows the network behavior when sources generate bursty traffic according to an on/off model, where the on and off periods are geometrically distributed in the number of cells. The mean on period is 10 ATM cells.

Figure 7.20 displays a multimedia experiment. Source 9 sends motion picture frames to Destination 1. All other sources send normal data traffic generated according to a Poisson process at the load specified in the load axis. The video traffic is scheduled to be generated during the source's priority band, which is of size 10 cells every 125µs cycle. As it can be seen, the network provides high-quality service to the video source while non-QoS

demanding traffic proceeds normally. The isolation of both traffic types can be accomplished by simple band tuning (in microseconds).

Figure 7.21 shows the comparison of analysis and simulation results for the RDMA++ mean packet delay. As it can be seen, the results are in good agreement.

One can observe in Figure 7.17 that each tree has 4 links arriving at the final destination. Additionally, the final destinations in the simulation did not incur transfer delays. For this reason, one of the trees seen as a server has a throughput 4 times larger than the transmission rate at each link. This fact must be input into the formulas of the analysis by making the server 4 times faster than the maximum links rate. Additionally, the maximum load to any of the trees is 25% (and thus low). This is the reason why the analysis (that approximates better the system for low loads) closely matches the simulation results.

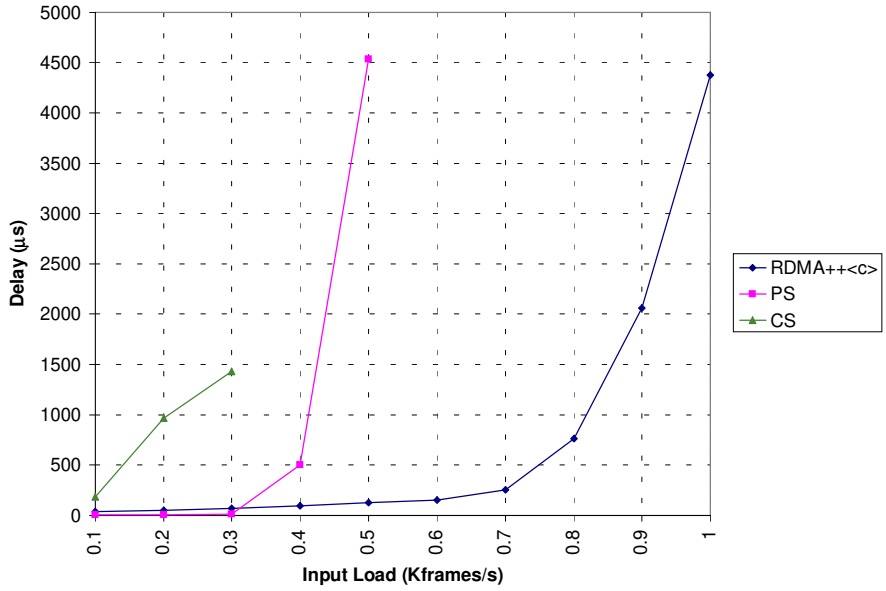

Figure 7.19: Mean frame delay comparison (in µs) for Poisson arrivals and Poisson service (the frame size is 530 bytes, the link speed 2.4 Gb/s, the band size 41.67 µs, the cycle period 125 µs)

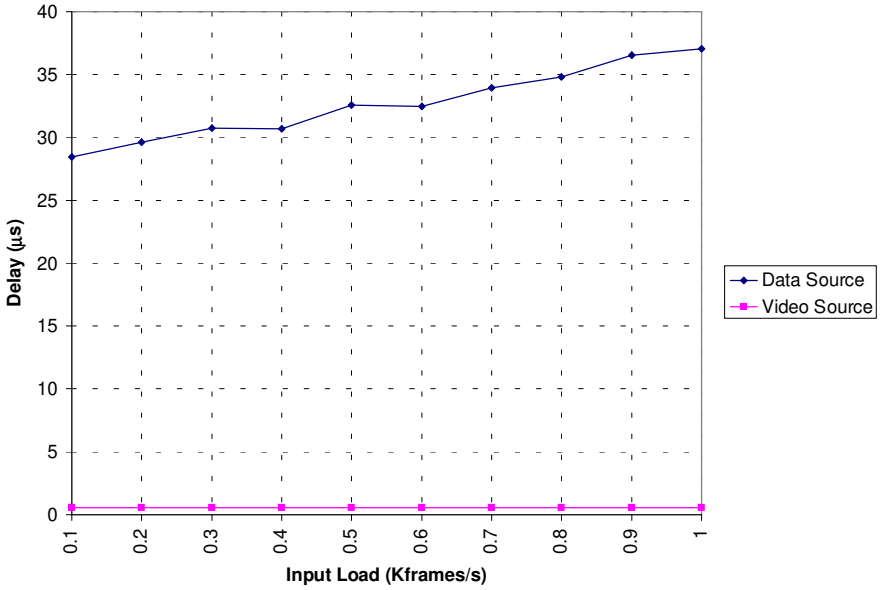

Figure 7.20: Mean frame delay comparison (in µs) for Poisson and deterministic arrivals and deterministic service under RDMA++ (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 41.67 µs, the cycle period 125 µs)

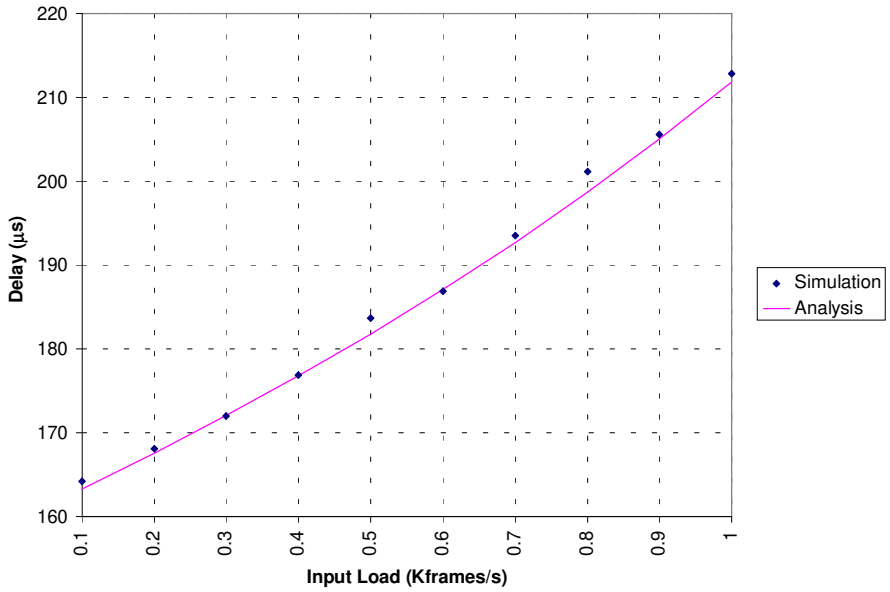

Figure 7.21: Mean frame delay (in µs) for Poisson arrivals and deterministic service under RDMA++ (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 41.67 µs, the cycle period 125 µs)

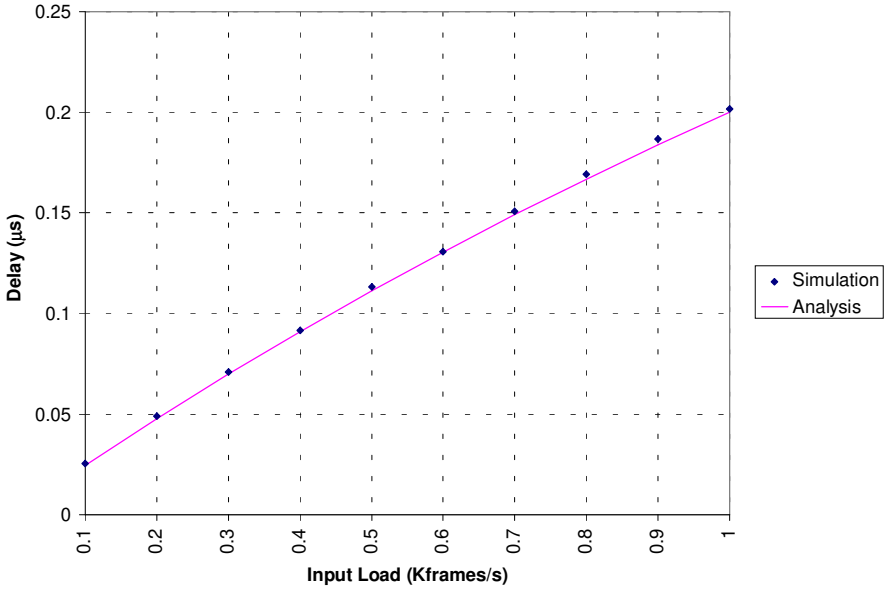

Figure 7.22: Mean loss rate for Poisson arrivals and deterministic service under RDMA- (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 41.67 µs, the cycle period 125 µs)

One can observe in Figure 7.17 that each tree has 4 links arriving at the final destination. Additionally, the final destinations in the simulation did not incur transfer delays. For this reason, one of the trees seen as a server has a throughput 4 times larger than the transmission rate at each link. This fact must be input into the formulas of the analysis by making the server 4 times faster than the maximum links rate. Additionally, the maximum load to any of the trees is 25% (and thus low). This is the reason why the analysis (that approximates better the system for low loads) closely matches the simulation results.

Figure 7.22 compares the simulation and analysis results for the RDMA- mean packet loss rate. Each tree is active only 1/3 of the cycle. Thus the input load in the analysis formulas varies between 0 and 1/3 . This is why the loss rate never reaches the 50% loss upper bound.

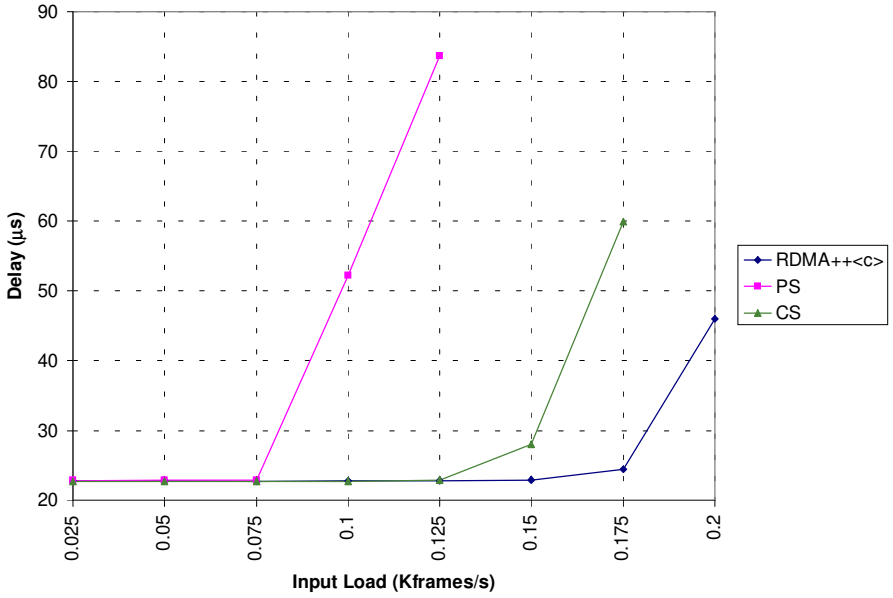

Figure 7.23: Mean frame delay comparison (in µs) for Poisson arrivals and deterministic service (the frame size is 53 bytes, the link speed 2.4 Gb/s, the band size 17.8571 µs, the cycle period 125 µs)

Finally, Figure 7.23 depicts the mean delay when the NSF T3 backbone network operates under RDMA++. The propagation delays are approximated since the exact measures are not available. The link speeds were upgraded to 2.4 Gb/s. The CPU speed was 100 MIPs for the PS simulation and some 50 instructions were necessary to process each ATM cell. It is important to notice that the topology of the NSF backbone is not particularly suitable for RDMA since only two trees can coexist in each band. Nevertheless, the performance advantage of RDMA is clear in the figure.

An important observation that can be derived from the experiment in Figure 7.23 is that, when applied to wide area networks, the time incurred waiting for a particular band is negligible when compared to the propagation delays. For instance, the waiting time for the band in the simulated NSF backbone is at most 125µs (a complete cycle), but the cross-country propagation delay is of the order of 30 ms (240 times larger). Thus, the time

waiting for a band in RDMA++ is a negligible component of the total frame delay, which in this case is very close to the PS frame delay for low loads.

7.5 Future Work

The initial performance assessment of RDMA in this chapter concentrated on understanding basic RDMA performance and on comparing it with PS and CS. The next step is to study policies to allocate RDMA bands and to study their performance. One possible outcome of such study, given specific demands, is the best allocation policy and what is the performance that it can deliver.

The analysis in this chapter had to settle for approximations due to the difficulties in finding closed-form solutions. A natural extension for such analysis is to use mechanisms that rely on numerical algorithms to resolve expressions numerically. Potentially, the same mechanisms could be used to analyze higher moments of the delay and the loss in the network.

7.6 Summary

This chapter provided initial analysis of mean RDMA delay and loss. The goal of the analysis was to understand RDMA performance and compare it with PS and CS.

The initial step was to provide an analytical understanding of RDMA behavior with Poisson arrivals on a single node. A closed form solution was found for RDMA- with generic frame size distribution. It was shown that the loss rate will never exceed 50%. The mean delay for RDMA++ was approximated for low to medium input loads. The mean

delay for RDMA+ could be approximated as well. The loss rate in RDMA+ was shown to be negligible when the band size is large compared to the mean frame size.

All the analytical results were validated through simulations. The simulations were also used to compare RDMA performance with PS and CS in two network topologies. It was found that PS has better performance for low loads. When the load in the system saturates the frame header processors, RDMA outperforms PS. It is important to notice that the PS simulation did not incur admission control delays that may deteriorate PS performance. Additionally, it was shown that, if the propagation delay in the network is large, the time waiting for bands in RDMA is negligible. In such scenario, the final end-to-end delay is comparable to PS even for low system loads. RDMA always outperforms CS because the mean time waiting for a band is smaller than the mean time waiting for a circuit. Additionally, the destination is allocated to a tree and not a circuit, which has the potential of further increasing RDMA performance when compared to CS.