

Performance Analysis of Buffered Banyan Networks

Thomas H. Theimer, Erwin P. Rathgeb, Manfred N. Huber

University of Stuttgart
Institute of Communications Switching and Data Technics

Abstract

Banyan Networks are used in multiprocessor computer applications as well as in new, high performance packet switch architectures. In this paper we will give a classification of the most common Banyan Networks and outline an analysis approach for the rather general class of Delta-b Networks with multiple buffers. Based on this approach we will discuss the effects of the approximations involved and present a refined analysis algorithm for the special case of a single buffered Delta-2 Network. The results of both algorithms will be compared with simulation results to assess their accuracy.

1 Introduction

In addition to the existing services offered by today's Integrated Services Digital Network (ISDN), future networks should also include broadband services like video communication, graphic applications and high speed data communication. For this Integrated Broadband Communication Network (IBCN) the Asynchronous Transfer Mode (ATM) has been proposed as a well suited switching principle.

The ATM technique enables the integration of various services with different bandwidth requirements into a uniform network since the information transfer is based on packets (cells). Future services with requirements which are unknown today can be added easily. The characteristic features of ATM are high speed links, simplified protocols and high capacity switching nodes. The required throughput can only be achieved by a distributed and highly parallel architecture of the switching nodes.

In several publications Banyan networks have been proposed for the implementation of an ATM switching node. These interconnection networks are well known from multiprocessor computer applications, where they are used for the interconnection of multiple processors and memories.

In this paper we focus on the performance analysis of a special type of Banyan Networks suitable for an ATM switching node. After a general classification of Banyan Networks, the analysis of Delta-b Networks with multiple buffers is briefly described and the approximations which are made to simplify the analysis are discussed. Then a refined analysis of single buffered Delta-2 Networks is developed to eliminate the main approximations. Finally, the accuracy of several analysis methods is compared with simulation results.

2 Classification of Banyan Networks

Banyan networks belong to the class of multistage interconnection networks and have been defined by Goke and Lipovsky [6] using graph theoretical methods. In a simplifying approach we shall consider a Banyan network as a multistage interconnection network which has the property that there is exactly one path between any input port and any output port (single path network). The general class of Banyan networks can be structured into several subclasses:

A L -level Banyan is a Banyan where only adjacent stages are connected by links. This means that each path leads exactly through L stages. The L -level Banyan class can be subdivided into *Regular* and *Irregular Banyans*. Regular Banyans are constructed from a single type of basic elements with F inputs and S outputs, whereas the type of the elements of an Irregular Banyan is varying throughout the network. The Generalized Delta-networks [3] are an example of Irregular Banyans.

For an economical implementation of switching networks Regular Banyans will be preferred because they are composed of identical elements. Two subclasses of Regular Banyans are the *CC-Banyans* and the *SW-Banyans* (see [6]).

We will restrict our further discussions to the SW-Banyan class, because it includes most of the existing implementations.

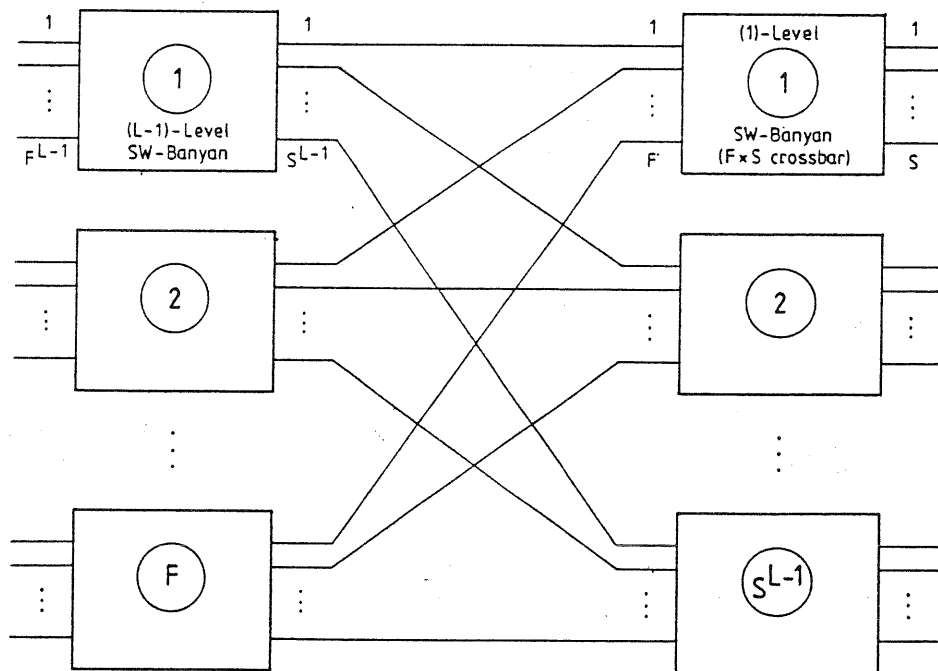


Figure 1: Recursive construction of a (L) -level SW-Banyan

A property of the SW-Banyans is, that these structures can be constructed recursively from a crossbar structure with F inputs and S outputs. This basic $(F \times S)$ crossbar element can be considered as a (1)-level SW-Banyan. To get larger SW-Banyan structures with L levels, several $(L - 1)$ -level SW-Banyans are connected with an extra stage of $(F \times S)$ basic elements as depicted in Figure 1. Each $(L - 1)$ -level SW-Banyan has F^{L-1} input links and S^{L-1} output links. The additional crossbar

switches must be connected to the SW-Banyans in a regular manner, i.e. if a crossbar is connected to output i of SW-Banyan number 1, it must be connected to output i of all other SW-Banyans as well.

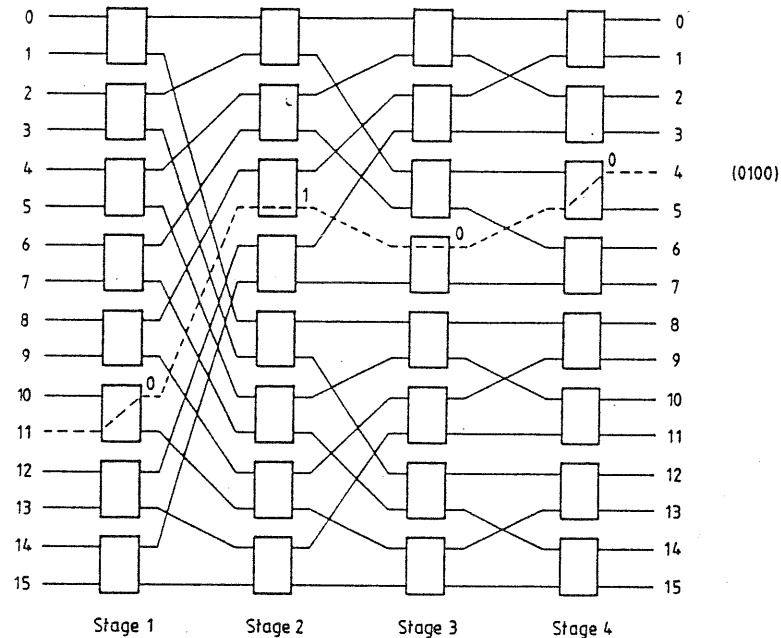


Figure 2: Delta-2 network with 4 stages and 16 input terminals

Delta networks as defined by Patel [13] have the same topological structure as SW-Banyans. They are constructed of n stages corresponding to the L levels of the SW-Banyan. Analogous to the SW-Banyan, each stage consists of several switching elements with a input ports and b output ports, so the network has a^n input terminals and b^n output terminals.

The definition of Delta networks is including the application of a specific routing scheme called "self routing". Using this scheme, the movement of the packets through a Delta network is controlled by a destination address which must be unique for every output terminal of the network. The destination address is a number of base b with n digits corresponding to the n stages of the network. Each digit directly specifies which output port of a switching element in a specific stage will receive the packet.

This *digit controlled routing* (self routing) mechanism has the advantage that it can be implemented in hardware and that a central control is only needed for the initial setup of a virtual connection. The switching functions are distributed over all switching elements and can be done in parallel.

Rectangular Delta networks are constructed from switching elements which have the same number of inputs and outputs ($a = b$). Consequently, the number of switching elements per stage is constant and the number of network inputs is equal to the number of network outputs. A rectangular Delta network composed of b -input switching elements is called *Delta- b network*.

An example of a Delta-2 network with 4 stages is shown in Figure 2. The dashed line indicates the path of a packet from input terminal 11 to output terminal 4, which has the binary destination address '0100'.

Bidelta networks (Bidirectional Delta networks) have a special topological structure; they remain a Delta network even if the network input links are interpreted as network output links and vice versa. All Bidelta networks are topologically equivalent and can be transformed into each other by relabelling the switching elements and the links [3].

Many of the well known interconnection networks including the Baseline, Reverse Baseline, Omega, Flip, Bit-Reversal and Inverse Bit-Reversal networks belong to the class of Bidelta networks. The network depicted in Figure 2 has the topology of a Baseline network [17].

3 Analysis of Delta-b Networks with Multiple Buffers

In this section we give an outline of the basic analysis method for Delta-b networks with an arbitrary number of buffer places at each input of the switching elements, because similar principles are used in the enhanced analysis variants described in the following sections. A detailed description of the analysis is given in [7].

To simplify the analysis of the network, we make the following assumptions:

1. The network is operated synchronously, that means the packets are transmitted only at the beginning of a time slot given by the packet clock and thus the time axis is considered to be discrete.
2. All packets have a fixed length.
3. Packets arrive independently at the network input links and their destination addresses are distributed uniformly (at random) over all output links at stage n .
4. Each input link of the network is offered the same traffic load, which is defined as the probability that a packet is arriving in one clock cycle.
5. The input buffers of a switching element are independent.
6. There is no blocking at the output links of the network. This means that the output links have at least the same speed as the internal links.

As a consequence of the assumptions 3, 4 and 5, the switching elements in a particular stage as well as the various buffers within these elements are considered as statistically indistinguishable. Therefore, the analysis can be done by investigating just one buffer in every stage.

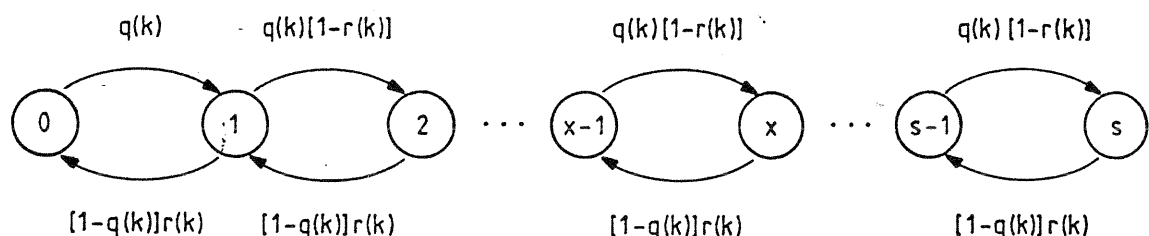


Figure 3: State diagram of the Markov Chain

A buffer at stage k is modeled by the simple Markov Chain shown in Figure 3 where we assume, that packets are arriving at the buffer independently with probability $q(k)$

and leave the buffer with probability $r(k)$. From the Markov Chain we can derive the state probabilities of the buffers as a function of $q(k)$ and $r(k)$.

In the next step, two aspects of the operation of the network have to be considered. The first one is, that we assume a backpressure mechanism which prevents any packet loss within the network. This leads to a steady state throughput (defined as the probability, that a packet is transmitted on a link in one clock cycle) which is equal on any link within the network. The second one is, that the buffer is assumed to support simultaneous read and write access. Thus, even a full buffer is able to accept a packet if another packet is leaving the buffer in the same clock cycle. With these considerations, the probability that a buffer at stage k is ready to accept a packet can be expressed as a function of the state probabilities and of $r(k)$.

Since the buffers of adjacent stages are connected via the $b \times b$ crossbar, the influence of this switching matrix has to be modeled, too. This is done by using the relationship between the probability q_{in} that a packet is offered to an input of the crossbar and the probability q_{out} that it is forwarded to the next stage given in eqn. (1) according to [13,2] :

$$q_{out} = 1 - \left(1 - \frac{q_{in}}{b}\right)^b \quad (1)$$

The probability q_{out} is less than q_{in} due to the blocking effects that occur if more than one packet is destined to a specific output of the crossbar in one clock cycle. After analyzing the Markov Chain depicted in Figure 3 this equation can be used to compute the state probabilities of stage $k - 1$ from the state probabilities of stage k .

With the assumption for the boundary of the network (assumption 6) the analysis can be started at the last stage of the network and the complete network can be analyzed successively without any iterations. Various performance measures like the mean total transfer time of a packet can be derived from the state probabilities.

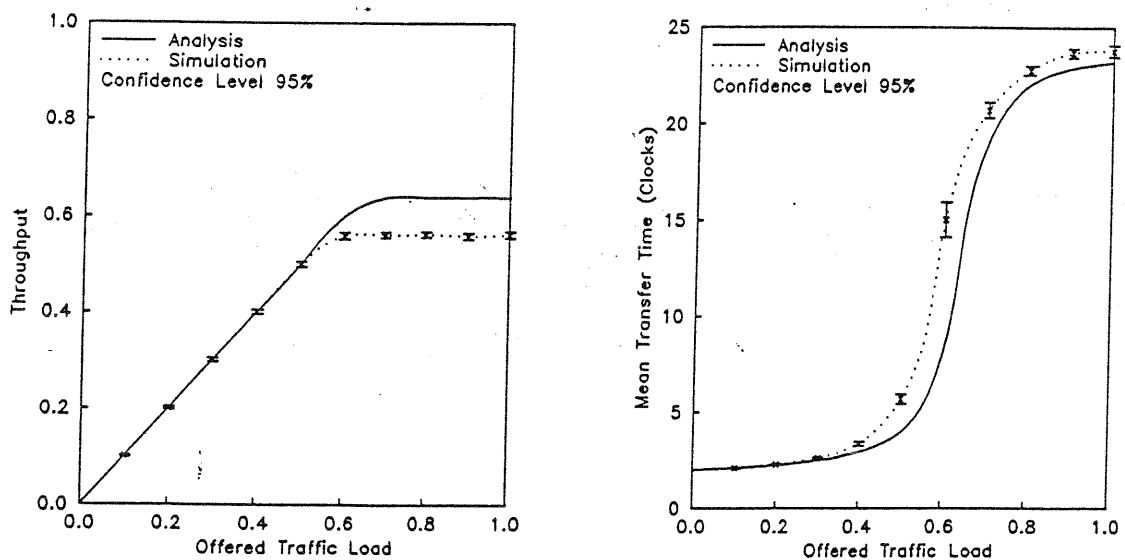


Figure 4: Analysis of a Delta-8 Network with 2 stages and buffersize 10

Figure 4 shows the result of this analysis for a Delta-8 Network with 64 inputs and buffersize 10 in comparison with simulation results. The low accuracy at high loads is resulting from several assumptions that have been made in the analysis. First, it

was assumed that the packets arrive at the buffers independent of each other. In reality, in case of blocking the packets remain in their buffers in the preceding stage and contend for the same output in the next clock cycle, i.e. there are correlations between consecutive clock cycles as well as between the states of the buffers in the adjacent stages. Second, the states of the buffers within a switching element were assumed to be independent of each other, which is clearly an approximation.

The exact solution taking these dependencies into account would lead to a multi-dimensional problem that can not be solved with reasonable expense. In the next sections we will present an enhanced approximation for a special case, which includes a refined modeling of the relationship between the stages as well as among the buffers within the stages.

4 Analysis of Single Buffered Delta-2 Networks

4.1 Refined Model

The analysis of single buffered Delta-2 Networks has been presented by Jenq [8]. He employed a 2-state Markov Chain according to Figure 3 to characterize the buffers of the switching elements (SE). However, the same approximations have been made as outlined in the previous section, resulting in the same insufficient accuracy of the results.

To identify the essential approximations which are reducing the accuracy, we consider a buffer that has received a packet in the previous clock cycle. This packet can be forwarded in the current clock cycle, if the destination buffer is ready to accept it and if there is no conflict with the other buffer of the SE. If the packet is unable to leave the buffer, it has to wait for the next clock cycle to start a new attempt.

As the destination address of the waiting packet does not change, there is a close correlation between the buffered packets in consecutive clock cycles. Furthermore, the probability that a packet is able to leave the buffer is significantly decreasing if the packet is blocked, because either another packet has entered the destination buffer or the destination buffer is blocked itself. In both cases the destination buffer is occupied by a packet.

To take these effects of blocking into account, we introduce an additional state to characterize a buffer that contains a blocked packet. Consequently, each buffer has three possible states :

- state '0' buffer is empty
- state 'n' buffer contains a packet which arrived in the last clock cycle
- state 'b' buffer contains a packet which has been delayed for at least one clock cycle

As a further enhancement the dependencies between both buffers of a SE will be explicitly modeled by a 2-dimensional Markov Chain. The resulting state space of a SE has 9 states (see eqn. 2).

In order to simplify the computation of the state transitions between successive clock intervals, each transition will be subdivided into two logical steps. In the first step

(phase 1), the buffers which contain a packet send a request to the next stage indicating that a packet is available for transmission. The packet is forwarded, if there are no collisions with other packets and if the destination buffer at the next stage is ready to accept a packet.

After the completion of phase 1 each buffer has only two possible states. A buffer containing a packet that could not be forwarded during phase 1 is entering a blocked state and the buffer is empty in all other cases.

In the second step (phase 2), the previous stage offers new packets to the buffers at stage k . These packets are accepted only by those buffers which are empty after phase 1. The probability that a packet is offered to a buffer is assumed to be independent of its state. After the packets have entered the buffers, the current clock period is completed. It should be noted that this separation is only logical, because in the reality phase 1 and phase 2 are performed simultaneously in all SE's.

4.2 State Space of a Switching Element

After the description of the refined model, the state space of a SE will be derived. Since each SE contains two buffers which are not independent, a 2-dimensional state space is required to characterize a SE. In the following, the index (x,y) will be used to identify the state of a SE whose upper buffer is in state x and whose lower buffer is in state y :

$$p_{x,y}(t, k) = P\{ \text{a SE at stage } k \text{ is in state } x \text{ (upper) and } y \text{ (lower buffer) at the beginning of clock cycle } t; x, y \in 0, n, b \}$$

The state of a SE after the completion of phase 1 will be characterized by a tilde (\sim) :

$$\tilde{p}_{x,y}(t, k) = P\{ \text{the buffers of a SE at stage } k \text{ are in state } x \text{ (upper) and } y \text{ (lower buffer) at the beginning of phase 2 of clock cycle } t; x, y \in 0, b \}$$

4.3 State Transitions of a Switching Element

The transition probabilities of phase 1 depend on the state of the SE. Therefore, the following notation will be introduced to describe the particular transitions :

$$r_{x,0}(t, k) = P\{ \text{the packet in the upper buffer of a SE at stage } k \text{ can be forwarded during clock cycle } t \mid \text{the lower buffer is empty and the upper buffer is in state } x; x \in n, b \}$$

$$r_{n,n}^{1,0}(t, k) = P\{ \text{the packet in the upper buffer of a SE at stage } k \text{ can be forwarded during clock cycle } t \mid \text{both buffers are in state } n \}$$

The suffix $(1,0)$ indicates that the packet in the upper buffer can be forwarded ('1'), while the packet in the lower buffer has to wait ('0'). As there are many possible combinations, this notation should serve as an example.

To simplify the mathematical expressions for the state transitions illustrated in Figure 5, we introduce the matrix $\vec{P}(t, k)$ containing the state probabilities $p_{x,y}(t, k)$

$$\vec{P}(t, k) = \begin{pmatrix} p_{0,0}(t, k) & p_{n,0}(t, k) & p_{b,0}(t, k) \\ p_{0,n}(t, k) & p_{n,n}(t, k) & p_{b,n}(t, k) \\ p_{0,b}(t, k) & p_{n,b}(t, k) & p_{b,b}(t, k) \end{pmatrix} \quad (2)$$

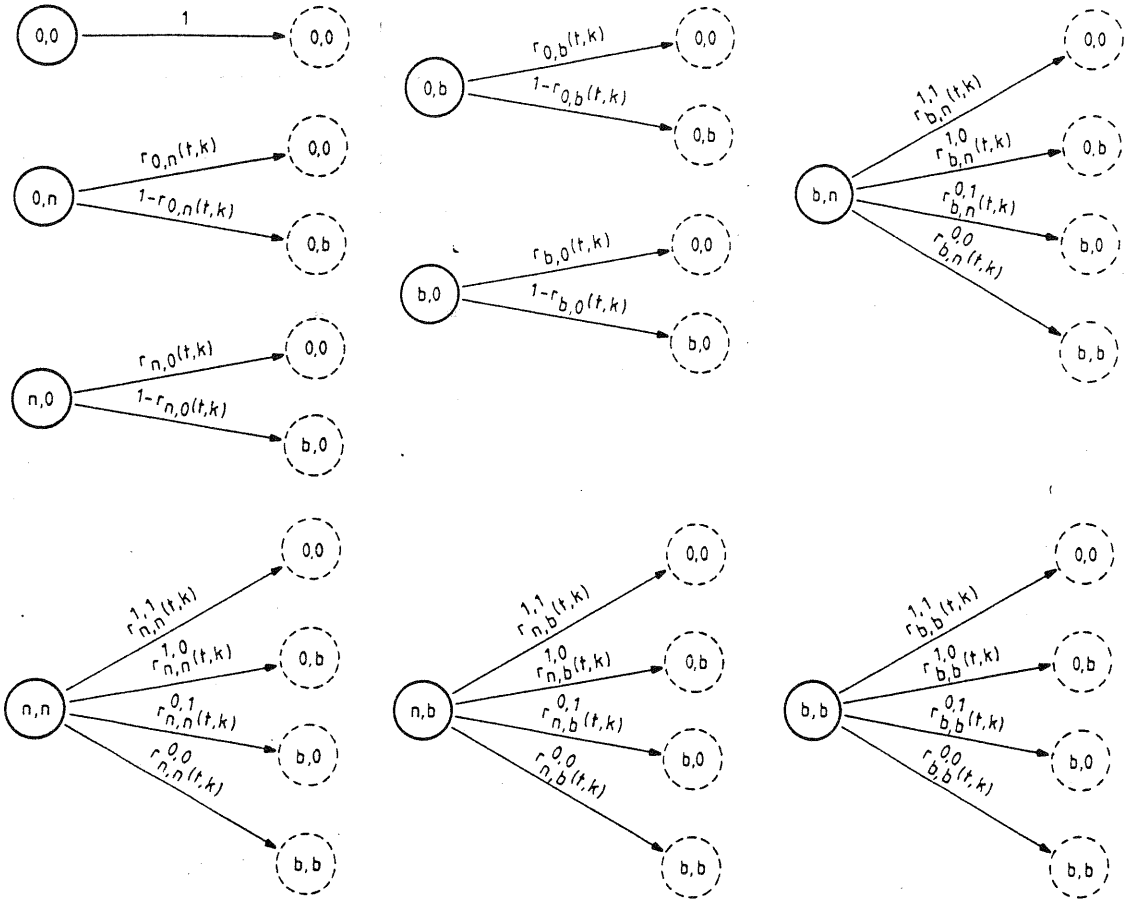


Figure 5: State transitions of a SE during phase 1; the states at clock event t are denoted by bold circles, and the intermediate states are indicated by dashed circles

In the following, the symbol ' \odot ' denotes a multiplication of two matrices analogous to the scalar product of vectors :

$$\begin{pmatrix} a_{00} & \cdots & a_{x0} \\ \vdots & & \vdots \\ a_{0y} & \cdots & a_{xy} \end{pmatrix} \odot \begin{pmatrix} b_{00} & \cdots & b_{x0} \\ \vdots & & \vdots \\ b_{0y} & \cdots & b_{xy} \end{pmatrix} := \sum_{i,j} a_{ij} \cdot b_{ij} \quad (3)$$

Using Fig. 5, the state probabilities at the end of phase 1 are calculated from the above defined transition probabilities and the matrix $\vec{P}(t, k)$:

$$\vec{p}_{0,0}(t, k) = \begin{pmatrix} 1 & r_{n,0}(t, k) & r_{b,0}(t, k) \\ r_{0,n}(t, k) & r_{n,n}^{1,1}(t, k) & r_{b,n}^{1,1}(t, k) \\ r_{0,b}(t, k) & r_{n,b}^{1,1}(t, k) & r_{b,b}^{1,1}(t, k) \end{pmatrix} \odot \vec{P}(t, k) \quad (4)$$

$$\vec{p}_{0,b}(t, k) = \begin{pmatrix} 0 & 0 & 0 \\ 1 - r_{0,n}(t, k) & r_{n,n}^{1,0}(t, k) & r_{b,n}^{1,0}(t, k) \\ 1 - r_{0,b}(t, k) & r_{n,b}^{1,0}(t, k) & r_{b,b}^{1,0}(t, k) \end{pmatrix} \odot \vec{P}(t, k) \quad (5)$$

$$\tilde{p}_{b,0}(t, k) = \begin{pmatrix} 0 & 1 - r_{n,0}(t, k) & 1 - r_{b,0}(t, k) \\ 0 & r_{n,n}^{0,1}(t, k) & r_{b,n}^{0,1}(t, k) \\ 0 & r_{n,b}^{0,1}(t, k) & r_{b,b}^{0,1}(t, k) \end{pmatrix} \odot \vec{P}(t, k) \quad (6)$$

$$\tilde{p}_{b,b}(t, k) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & r_{n,n}^{0,0}(t, k) & r_{b,n}^{0,0}(t, k) \\ 0 & r_{n,b}^{0,0}(t, k) & r_{b,b}^{0,0}(t, k) \end{pmatrix} \odot \vec{P}(t, k) \quad (7)$$

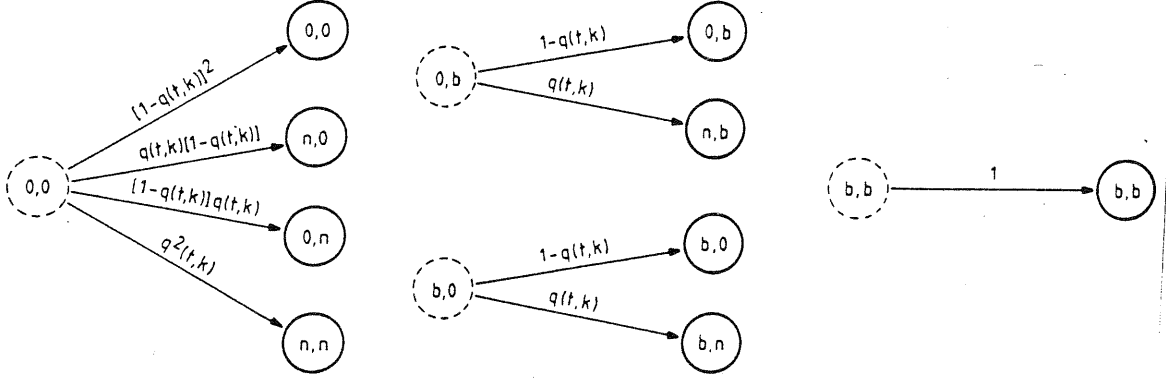


Figure 6: State transitions of a switching element during phase 2; the intermediate states are denoted by dashed circles and the states at clock event $t + 1$ are indicated by bold circles

If phase 1 of clock cycle t is completed, new packets are offered to each buffer of stage k with the probability $q(t, k)$ which is assumed to be independent of the state of the SE. The packets are accepted by those buffers being empty after phase 1, leading to the state transitions illustrated in Figure 6.

The state probabilities at clock event $t + 1$ are easily derived from Figure 6 :

$$p_{0,0}(t + 1, k) = [1 - q(t, k)]^2 \cdot \tilde{p}_{0,0}(t, k) \quad (8)$$

$$p_{0,n}(t + 1, k) = [1 - q(t, k)]q(t, k) \cdot \tilde{p}_{0,0}(t, k) \quad (9)$$

$$p_{n,0}(t + 1, k) = q(t, k)[1 - q(t, k)] \cdot \tilde{p}_{0,0}(t, k) \quad (10)$$

$$p_{n,n}(t + 1, k) = q^2(t, k) \cdot \tilde{p}_{0,0}(t, k) \quad (11)$$

$$p_{0,b}(t + 1, k) = [1 - q(t, k)] \cdot \tilde{p}_{0,b}(t, k) \quad (12)$$

$$p_{b,0}(t + 1, k) = [1 - q(t, k)] \cdot \tilde{p}_{b,0}(t, k) \quad (13)$$

$$p_{n,b}(t + 1, k) = q(t, k) \cdot \tilde{p}_{0,b}(t, k) \quad (14)$$

$$p_{b,n}(t + 1, k) = q(t, k) \cdot \tilde{p}_{b,0}(t, k) \quad (15)$$

$$p_{b,b}(t + 1, k) = \tilde{p}_{b,b}(t, k) \quad (16)$$

If the transition probabilities are known, the state probabilities of the buffers can be computed iteratively. Therefore, the calculation of the transition probabilities of phase 1 and phase 2 will be discussed briefly in the next section. The explicit mathematical expressions are derived in the Appendix.

4.4 Transition Probabilities

The probability that a packet is able to leave the buffer depends on the probability of a collision with a packet from the other buffer and on the probability that the switching element in the next stage is ready to accept the packet. Considering the probability that a packet can be accepted by its destination buffer at stage $k + 1$, there are two cases to distinguish :

- If none of the buffers of a SE at stage k contains a blocked packet, there is no relationship between the states of these buffers and the states of the corresponding destination buffers at stage $k + 1$. These buffers can have any of the three possible states (0, n, b).
- If a packet was blocked at stage k , the destination buffer of this packet at stage $k + 1$ always contains a packet. The destination buffer is in state 'n', if it received a packet from the other buffer of the SE at stage k in the previous clock cycle, otherwise it must be blocked.

It should be noted that a blocked packet also has an influence on a packet which arrives at the other buffer of the same SE. If the new packet is destined for the same output link as the blocked packet (probability 0.5), both packets have the same probability to be forwarded to the next stage.

The probability $q(t, k)$ that a packet is offered to a buffer at stage k can be computed considering the fact that no packets are lost on the internal links of the network due to the backpressure mechanism. Consequently, the probability that a packet is entering a buffer at stage k is equal to the probability that a packet is transmitted on an output link of a SE at stage $k - 1$, which can be calculated from the state probabilities of stage $k - 1$ and from the transition probabilities of phase 1. As a packet can only enter a buffer if the buffer is ready to accept it, the probability $q(t, k)$ is easily derived from the probability that a packet is entering a buffer at stage k .

4.5 Algorithm

Due to the complexity of the state transitions, the analysis of the network is done iteratively. Each step of the iteration starts with the calculation of the transition probabilities of phase 1 and phase 2 (see Appendix). Afterwards, the state probabilities of the intermediate states (eqn. (4) - (7)) and the state probabilities at the next clock cycle (eqn. (8) - (16)) can be computed using the transition probabilities. These steps are repeated until the steady state is reached, i. e. until the throughput on the output links of the network is equal to the throughput on the input links.

In the steady state the mean transfer time $T_f(k)$ of a packet at stage k can be calculated using Little's Law :

$$T_f(k) = \lim_{t \rightarrow \infty} \frac{1 - [p_{0,0}(t, k) + p_{0,n}(t, k) + p_{0,b}(t, k)]}{\rho(t, k)} \quad (17)$$

with the throughput $\rho(t, k)$ as defined in eqn. (56) (see Appendix). The total transfer time of a packet through the network is given by

$$T_f = \sum_{k=1}^n T_f(k) \quad (18)$$

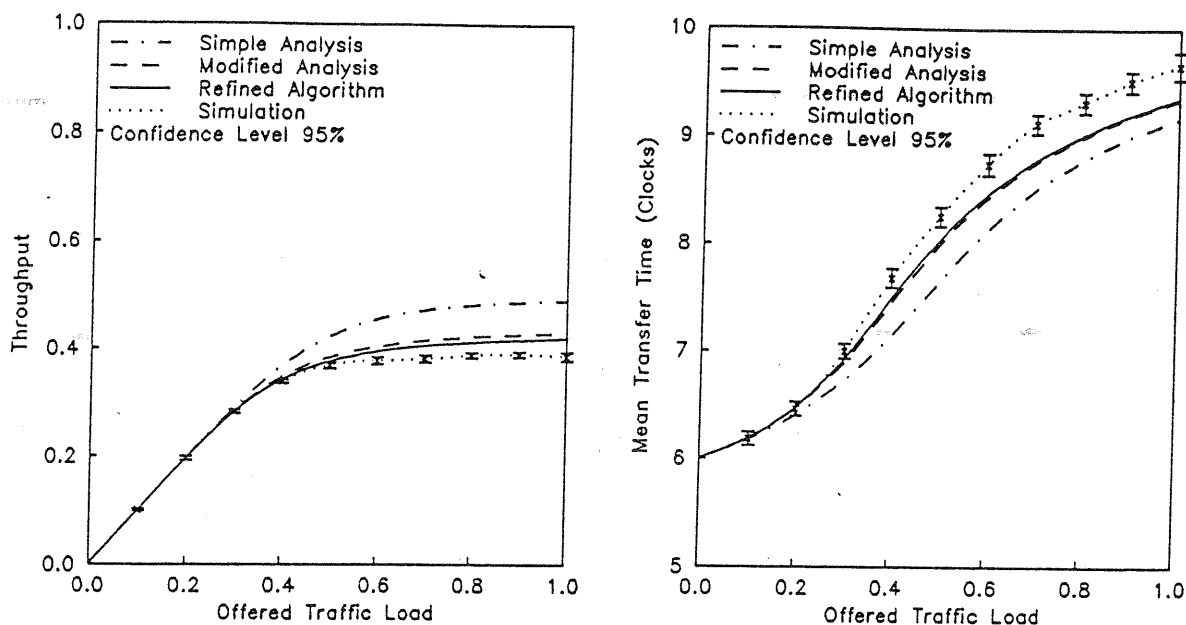


Figure 7: Throughput and delay of a single buffered Delta-2 Network with 6 stages

5 Results

In this section we investigate the accuracy of different analysis methods for single buffered Delta-2 Networks. Therefore, the results of

- the *Simple Analysis* outlined in section 3 which includes the analysis described by Jenq [8]
- a *Modified Analysis* which uses 3 states per buffer, but preserves the assumption that the buffers of a SE are independent
- the *Refined Algorithm* described in section 4

are compared with the results of an event-by-event simulation. The confidence intervals of the simulation results refer to a confidence level of 95%.

Figure 7 shows the normalized throughput and the mean transfer time of a single buffered Delta-2 Network with 6 stages. It is obvious, that the accuracy of all methods is very good as long as the offered traffic load is small. However, if the traffic load approaches the maximum throughput of the network, the accuracy of the *Simple Analysis* method is insufficient. This is due to the fact, that many packets are blocked mainly at the first stages of the network at high traffic rates. Consequently, the detailed modeling of the blocking effects is resulting in a significant improvement of the accuracy. The consideration of the dependencies between the buffers of a SE leads to a further (though small) improvement of the results.

The influence of the size of the network on the performance is depicted in Figure 8. Since every additional stage of the network introduces further collisions, the maximum throughput decreases if the size of the network is extended. Although the *Simple Analysis* method clearly overestimates the performance of large networks, the *Refined Algorithm* presented in this paper is a very good approximation for the performance evaluation of single buffered Delta-2 Networks.

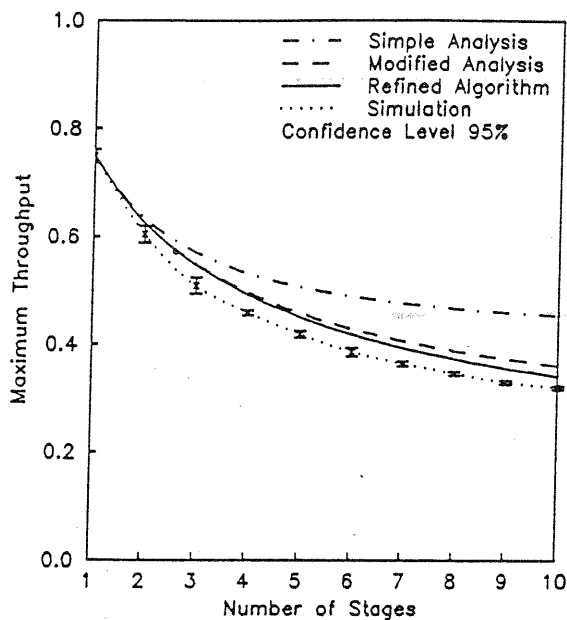


Figure 8: Maximum throughput versus number of stages of a single buffered Delta-2 Network

6 Conclusion

In order to evaluate all the different blocking and correlation effects, we did some detailed simulation studies that showed the internal behaviour of a Banyan Network. The observations made in these studies have been used to model these effects in an accurate way that is still feasible for analysis purposes. The resulting analysis for single buffered Delta-2 Networks shows a significant improvement in the accuracy of the results without requiring considerably more computational effort.

Some more work needs to be done to extend the enhanced analysis algorithm to switching elements with multiple buffers and perhaps to networks built of switching elements with more than two inputs. Another interesting aspect is the consideration of complex traffic patterns, which are more realistic in a connection oriented packet switching environment.

Appendix

The transition probabilities of phase 1 and phase 2 are derived in this Appendix and the explicit mathematical expressions are presented.

A Phase 1

The probability that a packet is able to leave the buffer depends on the probability of a collision with a packet from the other buffer and on the probability that the switching element in the next stage is ready to accept the packet. If two packets are contending for the same output of a SE, one of the packets is randomly selected for transmission and the other packet has to wait for the next clock cycle.

Considering the probability that a buffer is able to accept a packet, there are three cases to distinguish :

- The buffer contains a 'new' packet (state n) :

$$p_a^n(t, k) = P\{\text{the upper buffer of a switching element at stage } k \text{ is ready to accept a packet during clock cycle } t \mid \text{the buffer is in state } n \}$$

- The buffer contains a 'blocked' packet (state b) :

$$p_a^b(t, k) = P\{\text{the upper buffer of a switching element at stage } k \text{ is ready to accept a packet during clock cycle } t \mid \text{the buffer is in state } b \}$$

- The buffer is in any of the three possible states (0, n, b) :

$$p_a(t, k) = P\{\text{the upper buffer of a switching element at stage } k \text{ is ready to accept a packet during clock cycle } t \}$$

Due to the internal symmetry of the traffic, both buffers of a SE are statistically equivalent. Therefore it is sufficient for the calculation of the above defined probabilities to consider only the upper buffer of a SE :

$$p_a^n(t, k) = \frac{\left[\begin{pmatrix} 0 & r_{n,0}(t, k) & 0 \\ 0 & r_{n,n}^{1,0}(t, k) + r_{n,n}^{1,1}(t, k) & 0 \\ 0 & r_{n,b}^{1,0}(t, k) + r_{n,b}^{1,1}(t, k) & 0 \end{pmatrix} \odot \vec{P}(t, k) \right]}{[p_{n,0}(t, k) + p_{n,n}(t, k) + p_{n,b}(t, k)]} \quad (19)$$

$$p_a^b(t, k) = \frac{\left[\begin{pmatrix} 0 & 0 & r_{b,0}(t, k) \\ 0 & 0 & r_{b,n}^{1,0}(t, k) + r_{b,n}^{1,1}(t, k) \\ 0 & 0 & r_{b,b}^{1,0}(t, k) + r_{b,b}^{1,1}(t, k) \end{pmatrix} \odot \vec{P}(t, k) \right]}{[p_{b,0}(t, k) + p_{b,n}(t, k) + p_{b,b}(t, k)]} \quad (20)$$

$$p_a(t, k) = \left(\begin{array}{ccc} 1 & r_{n,0}(t, k) & r_{b,0}(t, k) \\ 1 & r_{n,n}^{1,0}(t, k) + r_{n,n}^{1,1}(t, k) & r_{b,n}^{1,0}(t, k) + r_{b,n}^{1,1}(t, k) \\ 1 & r_{n,b}^{1,0}(t, k) + r_{n,b}^{1,1}(t, k) & r_{b,b}^{1,0}(t, k) + r_{b,b}^{1,1}(t, k) \end{array} \right) \odot \vec{P}(t, k) \quad (21)$$

If none of the buffers of a SE at stage k contains a blocked packet, there is no relationship between the states of these buffers and the states of the corresponding destination buffers at stage $k + 1$. In this case, the probability that the destination buffers at stage $k + 1$ are able to accept a packet is given by $p_a(t, k + 1)$, because these buffers can have any of the three possible states (0, n, b).

With $p_a(t, k + 1)$ resulting from eqn. (21), the transition probabilities referring to those states where none of the buffers is blocked can be computed for the stages $1 \dots n - 1$:

$$r_{0,n}(t, k) = p_a(t, k + 1) \quad (22)$$

$$r_{n,0}(t, k) = p_a(t, k + 1) \quad (23)$$

$$r_{n,n}^{0,0}(t, k) = 0.5[1 - p_a(t, k + 1)]^2 + 0.5[1 - p_a(t, k + 1)] \quad (24)$$

$$r_{n,n}^{0,1}(t, k) = 0.5[1 - p_a(t, k + 1)]p_a(t, k + 1) + 0.25p_a(t, k + 1) \quad (25)$$

$$r_{n,n}^{1,0}(t, k) = 0.5p_a(t, k + 1)[1 - p_a(t, k + 1)] + 0.25p_a(t, k + 1) \quad (26)$$

$$r_{n,n}^{1,1}(t, k) = 0.5[p_a(t, k + 1)]^2 \quad (27)$$

As there is no blocking at the output links of the last stage, we obtain

$$r_{0,n}(t,n) = 1.0 \quad (28) \quad r_{n,n}^{0,1}(t,n) = 0.25 \quad (31)$$

$$r_{n,0}(t,n) = 1.0 \quad (29) \quad r_{n,n}^{1,0}(t,n) = 0.25 \quad (32)$$

$$r_{n,n}^{0,0}(t,n) = 0.0 \quad (30) \quad r_{n,n}^{1,1}(t,n) = 0.5 \quad (33)$$

If a packet was blocked at stage k , the destination buffer of this packet at stage $k+1$ always contains a packet. The destination buffer is in state 'n', if it received a packet from the other buffer of the SE at stage k in the previous clock cycle, otherwise it must be blocked. The probability that the destination buffer received a packet from a specific buffer of the SE at stage k in the previous clock cycle is given by $0.5\rho(t-1, k+1)$, where $\rho(t, k)$ is the throughput on an input link of a SE at stage k as defined in equ. (56).

Consequently, the transition probabilities referring to the case that one buffer is blocked are given by

$$r_{0,b}(t,k) = 0.5\rho(t-1, k+1) \cdot p_a^n(t, k+1) + [1 - 0.5\rho(t-1, k+1)] \cdot p_a^b(t, k+1) \quad (34)$$

$$r_{b,0}(t,k) = 0.5\rho(t-1, k+1) \cdot p_a^n(t, k+1) + [1 - 0.5\rho(t-1, k+1)] \cdot p_a^b(t, k+1) \quad (35)$$

$$r_{n,b}^{0,0}(t,k) = 0.5[1 - r_{n,0}(t,k)] \cdot [1 - r_{0,b}(t,k)] + 0.5[1 - r_{0,b}(t,k)] \quad (36)$$

$$r_{n,b}^{0,1}(t,k) = 0.5[1 - r_{n,0}(t,k)] \cdot r_{0,b}(t,k) + 0.25r_{0,b}(t,k) \quad (37)$$

$$r_{n,b}^{1,0}(t,k) = 0.5r_{n,0}(t,k) \cdot [1 - r_{0,b}(t,k)] + 0.25r_{0,b}(t,k) \quad (38)$$

$$r_{n,b}^{1,1}(t,k) = 0.5r_{n,0}(t,k)r_{0,b}(t,k) \quad (39)$$

$$r_{b,n}^{0,0}(t,k) = 0.5[1 - r_{b,0}(t,k)] \cdot [1 - r_{0,n}(t,k)] + 0.5[1 - r_{b,0}(t,k)] \quad (40)$$

$$r_{b,n}^{0,1}(t,k) = 0.5[1 - r_{b,0}(t,k)] \cdot r_{0,n}(t,k) + 0.25r_{b,0}(t,k) \quad (41)$$

$$r_{b,n}^{1,0}(t,k) = 0.5r_{b,0}(t,k) \cdot [1 - r_{0,n}(t,k)] + 0.25r_{b,0}(t,k) \quad (42)$$

$$r_{b,n}^{1,1}(t,k) = 0.5r_{b,0}(t,k)r_{0,n}(t,k) \quad (43)$$

Again, the transition probabilities of the SE's at the last stage differ from the other stages :

$$r_{n,b}^{0,0}(t,n) = 0.0 \quad (44) \quad r_{b,n}^{0,0}(t,n) = 0.0 \quad (48)$$

$$r_{n,b}^{0,1}(t,n) = 0.25 \quad (45) \quad r_{b,n}^{0,1}(t,n) = 0.25 \quad (49)$$

$$r_{n,b}^{1,0}(t,n) = 0.25 \quad (46) \quad r_{b,n}^{1,0}(t,n) = 0.25 \quad (50)$$

$$r_{n,b}^{1,1}(t,n) = 0.5 \quad (47) \quad r_{b,n}^{1,1}(t,n) = 0.5 \quad (51)$$

If both buffers of a SE at stage k are blocked, the corresponding destination buffers at stage $k+1$ are blocked, too. In this case, we obtain the following transition probabilities for the stages $1 \dots n-1$:

$$r_{b,b}^{0,0}(t,k) = 0.5[1 - p_a^b(t, k+1)]^2 + 0.5[1 - p_a^b(t, k+1)] \quad (52)$$

$$r_{b,b}^{0,1}(t, k) = 0.5[1 - p_a^b(t, k + 1)]p_a^b(t, k + 1) + 0.25p_a^b(t, k + 1) \quad (53)$$

$$r_{b,b}^{1,0}(t, k) = 0.5p_a^b(t, k + 1)[1 - p_a^b(t, k + 1)] + 0.25p_a^b(t, k + 1) \quad (54)$$

$$r_{b,b}^{1,1}(t, k) = 0.5[p_a^b(t, k + 1)]^2 \quad (55)$$

Due to the assumption that there is no blocking at the output links of the switch fabric, it is impossible that both buffers of a SE at the last stage are in a blocked state.

B Phase 2

During phase 2, packets from stage $k - 1$ are entering the buffers at stage k . The throughput on an input link of a SE at stage k is defined as the probability that a packet is received on that link and is given by

$$\rho(t, k) = q(t, k) \cdot p_a(t, k) \quad (56)$$

Due to the backpressure mechanism no packets are lost on the internal links of the switching network. Therefore, the probability that a packet is transmitted on an output link of a SE at stage k is equal to the throughput $\rho(t, k + 1)$ on an input link of stage $k + 1$:

$$\rho(t, k + 1) = \begin{pmatrix} 0 & 0.5r_{n,0}(t, k) & 0.5r_{b,0}(t, k) \\ 0.5r_{0,n}(t, k) & r_{n,n}^{1,0}(t, k) + r_{n,n}^{1,1}(t, k) & r_{b,n}^{1,0}(t, k) + r_{b,n}^{1,1}(t, k) \\ 0.5r_{0,b}(t, k) & r_{n,b}^{1,0}(t, k) + r_{n,b}^{1,1}(t, k) & r_{b,b}^{1,0}(t, k) + r_{b,b}^{1,1}(t, k) \end{pmatrix} \odot \vec{P}(t, k) \quad (57)$$

Note that eqn. (57) has been simplified using the symmetry of the internal traffic flow. $\rho(t, k)$ can be computed from eqn. (57) at the end of phase 1. Thus, from eqn. (56) we obtain the probability $q(t, k)$ that a packet is offered to an input buffer of stage k :

$$q(t, k) = \frac{\rho(t, k)}{p_a(t, k)}, \quad k = 2 \dots n \quad (58)$$

with $\rho(t, k)$ from eqn. (57) and $p_a(t, k)$ from eqn. (21).

$q(t, 1)$ is equal to the offered traffic load at the network inputs and must be specified as an input parameter for the analysis.

References

- [1] BATCHER K. E., "The Flip Network in STARAN", *Int. Conf. on Parallel Processing*, 1976, pp. 65-71.
- [2] DIAS D.M., JUMP J.R., "Analysis and Simulation of Buffered Delta Networks", *IEEE Transactions on Computers*, Vol. C-30, No. 4, April 1981, pp. 273-282.
- [3] DIAS D.M., KUMAR M., "Packet Switching in $n \log n$ Multistage Networks", *IEEE GLOBECOM'84*, Atlanta, 1984, pp. 114-120.

- [4] FENG T., "A Survey of Interconnection Networks", *IEEE Computer*, Vol.14, No. 12, Dec. 1981, pp. 12-27.
- [5] GIORCELLI S., DEMICHELIS C., GIANDONATO G., MELEN R., "Experimenting with Fast Packet Switching Techniques in First Generation ISDN Environment", *ISS'87*, Phoenix, 1987, pp. 388-394.
- [6] GOKE L.R., LIPOVSKI G.J., "Banyan Networks for Partitioning Multiprocessor Systems", *First Annual Symposium on Computer Architecture*, 1973, pp. 21-28.
- [7] HUBER M. N., RATHGEB E. P., THEIMER T. H., "Banyan Networks in an ATM-Environment", *ICCC'88*, Tel Aviv, 1988, Paper A 4-2.
- [8] JENQ Y.-C., "Performance Analysis of a Packet Switch Based on Single-Buffered Banyan Network", *IEEE Journal on Selected Areas in Communications*, Vol. SAC-1, No. 6, Dec. 1983, pp. 1014-1021.
- [9] KULZER J.J., MONTGOMERY W.A., "Statistical Switching Architectures for Future Services", *ISS'84*, Florence, 1984, Paper 43 A 1.
- [10] LAWRIE D. H., "Access and Alignment of Data in an Array Processor", *IEEE Transactions on Computers*, Vol. C-24, No. 12, December 1975, pp. 1145-1155.
- [11] LUDERER G. W. R., MANSELL J. J., MESSERLI E. J., STAEHLER R. E., VAIDYA A. K., "Wideband Packet Technology for Switching Systems", *ISS'87*, Phoenix, 1987, pp. 448-454.
- [12] MCMILLEN R.J., "A Survey of Interconnection Networks", *GLOBECOM'84*, Atlanta, 1984, pp. 105-113.
- [13] PATEL J.H., "Performance of Processor-Memory Interconnections for Multiprocessors", *IEEE Transactions on Computers*, Vol. C-30, No. 10, October 1981, pp. 771-780.
- [14] PEASE M. C., "The Indirect Binary n-Cube Microprocessor Array", *IEEE Transactions on Computers*, Vol. C-26, No. 5, May 1977, pp. 458-473.
- [15] TURNER J.S., "Design of an Integrated Services Packet Network", *Ninth Data Communications Symposium*, ACM Sigcomm Computer Communication Review, Vol. 15, No. 4, September 1985, pp. 124-133.
- [16] TURNER J.S., "New Directions in Communications", *1986 Int. Zurich Seminar on Digital Communications*, Zurich, 1986, pp. 25-32.
- [17] WU C-L., FENG T-Y., "On a Class of Multistage Interconnection Networks", *IEEE Transactions on Computers*, Vol. C-29, No. 8, August 1980, pp. 694-702.