Obtaining packet response times for nonblocking ATM switches

Xi-Ren Cao∗, Junjie Wang
Department of Electrical and Electronic Engineering, The Hong Kong University of Science and Technology,
Clear Water Bay, Kowloon, Hong Kong

Received 5 December 1999; received in revised form 8 May 2002

Abstract
In this paper, we propose an approach that yields accurate approximations for the packet response times in a generic ATM switch. The approach combines three existing methodologies: power series algorithm for light traffic, saturation analysis for heavy traffic, and the Newton–Padé rational approximation for curve fitting. This approach works especially well for small to medium size switches, for which the traditional assumption of the Poisson arrival to the transmission channels does not apply well. Numerical samples reveal a close agreement between the approximant and the simulation result.

© 2003 Elsevier Science B.V. All rights reserved.

Keywords: Power series algorithm; Rational approximation; Newton–Padé approximation

1. Introduction
In this paper, we analyze the performance of a generic nonblocking ATM switch with input queuing. Such a switch can be simply described as a box with \(N\) input lines and \(N\) output lines that constructs a cross-bar structure. At each cross point there is a gate that can be turned on or off to relay the ATM cells from an input line to the corresponding output line. Each output can accept one cell at a time. The buffers at the inputs are assumed to be infinite in capacity (see Section 2 for a detailed model).

A considerable amount of work has been done in performance evaluation of generic packet switches. Karol et al. [1] derives the maximum throughput and the packet queuing time. In this work, it is assumed that the size of switch is infinite in order to get a closed-form solution; the results for finite-size switches are obtained by simulations. It is also assumed in [1] that the destinations of consecutive packets are independent. Li [2,3] extends the analysis to nonuniform or correlated traffic, the maximum throughput (or bounds) is obtained in these cases. In [4], the effect of the speed-up factor on the packet delay and loss is investigated. All these studies are based on the assumption that the arrival times of packets at various input ports are synchronized and the packets have a fixed length. Hence, the results obtained

∗ Corresponding author. Tel.: +852-2358-1485; fax: +852-2358-7048.
E-mail address: eecao@ust.hk (X.-R. Cao).
in these studies apply to nonblocking ATM switches, where the time axis is slotted with the slot size equal to the transmission time of an ATM cell and the slots at all input lines are aligned. On the other hand, Chen and Stern [5] investigates the throughput and buffer allocation of a generic packet switch, where the packet length is assumed to be variable and packet arrivals are asynchronous. Because of the correlation among parallel input (output) queues in the switch, closed-form solutions to the performance analysis are generally unavailable. It is necessary to resort to some approximations or numerical methods. For example, it is assumed in [1,5] that the size of the switch is infinite such that the superposition of all the arrival processes tends to be a Poisson process, thereby leading to an M/G/1 queuing analysis. This assumption yields good results for large-size switches. Realizing this limitation, Cao [6] derives a closed-form solution to the maximum throughput for ATM switches of arbitrary size in the case of correlated traffic, where the packet has a geometrically distributed length. In addition, Cao and Towsley [7] extends the results in [6] to ATM switches with general packet length distributions, and Iun and Cao [8] extends the results to switches with internal speed-up and studies cell loss probabilities of such switches.

In this paper, we study the packet response time in a generic nonblocking input queuing ATM switch. While previous results apply to a switch with an infinite size [1,5], we are interested in switches with a finite number of inputs and outputs. In particular, the number of inputs and outputs, $N$, may be small, e.g. $N = 4$. The reason for this setup is that when $N$ is relatively large (e.g. $N > 16$), the queuing delay can be approximated without much loss of accuracy by assuming that $N$ goes to infinity. However, this approximation will introduce a substantial discrepancy when the actual switch size is small. In this case, one has to develop simulation models, which are usually time-consuming and inflexible [1,5]. Furthermore, even for large switches the traffic may not be evenly distributed. If only a number of nodes communicate with each other, then the part of the switch which connects these nodes may behave similar to a small-size switch. Therefore, it is worthwhile to develop simpler methods to investigate the performance measures for small switches. One of the major contributions of this paper is to propose a computationally efficient method as an alternative to the simulation-based techniques in switch performance analysis. Our approach, as will be shown, can be applied to switches with an arbitrary number of input lines, thereby making our studies applicable in all cases.

We shall apply three existing methodologies, namely the power series algorithm (PSA), the saturation analysis, and the rational approximation method, to obtain the entire curve for the packet response time under different traffic loads. Specifically, the work is performed in three consecutive steps:

1. We first solve the global balance equation by power series analysis to yield the average response time under light traffic.
2. We then apply the closed queuing network model developed in [6,7] to investigate the asymptotic queuing behavior under heavy traffic; we shall determine the maximum throughput (saturation point) and the rate at which packet response time approaches infinity near saturation.
3. After obtaining the results in steps (1) and (2), we then apply the rational approximation method to generate a Newton–Padé approximant of the packet response times under all traffic loads.

Corresponding to those technical steps, we organize the paper in the following way. In Section 2, we describe the fabric of a generic nonblocking ATM switch and the traffic model. In Section 3, we apply the PSA to solve the global balance equation, thereby yielding the packet delay under light traffic. In Section 4, we model the switch under saturation as a closed queuing network and obtain the maximum throughput of the switch. Furthermore, the pole pattern of the packet delay function is explored in order to determine the approaching rate to the asymptotic line. In Section 5, we apply the results in Sections 3 and 4...
to generate the Newton–Padé approximant of the packet response times. The approximants are compared with simulation results. Finally, we conclude the paper in Section 6.

2. The model

Fig. 1 illustrates the structure of a generic nonblocking ATM switch with input queuing. It can be simply described as a box with $N$ input lines and $N$ output lines that relay the cells arriving at its inputs to their requested outputs. The buffers at input ports are assumed to be infinite in capacity.

The model is similar to that in [7] except that the input queue lengths in [7] are assumed to be infinitely long (under heavy traffic) and those in our model are finite (under low or medium traffic). Thus, the analysis in [7] only gives maximum throughput, while our analysis yields the packet response time. We assume that the packets arrive at the $i$th input port according to an independent Poisson process with rate $\rho \lambda_i$, where the system load $\rho$ will be used as the variable in the power expansion in Section 3. The relative arrival rates $\lambda_i$ are normalized such that $0 \leq \rho < 1$.

Packets are divided into cells, and we assume that the packet length is a multiple of the cell size. Cells belonging to the same packet are destined to the same output. The packet length in the $i$th input queue is geometrically distributed with a mean of $s_i = 1/\mu_i$. We study the case where the destinations of consecutive packets arriving to a particular input line are correlated. Define $\theta_{k+i,j}$ to be the probability that the destination of the packet in the $i$th input queue is $j$ given that the preceding packet is destined to $i$. The subscript $k$ can characterize the traffic class associated with a particular input queue. In a symmetrical case where all input queues are identical, we can remove the subscript $k$ and, thus, the packets to all input queues share the same routing probabilities $\theta_{i,j}$. For uncorrelated traffic, $\theta_{i,j}$ can be written as $\theta_j$, where the destination of any packet does not depend on the destination of its preceding one. For simplicity, we illustrate the main idea by assuming that routing probabilities are identical for all inputs, i.e., $\theta_{k+i,j} = \theta_{i,j}$ for all $k$. 

![Fig. 1. Structure of a generic nonblocking ATM switch.](image-url)
Packets (and cells) in an input line are served in a first-come-first-served fashion. The first packet (or cell) in an input queue is called a HOL (head of line) packet (or cell). The HOL packet of an input queue contends with the HOL packets of the other input queues that have the same destination. The HOL packets (note the HOL cell) destined for output link $j$ form a (logical) request queue for $j$. There may be more than one packet in a request queue. The dashed boxes in Fig. 1 indicate the request queues, and the bullets indicate the HOL packets in these queues. After a HOL packet has been transmitted to the output link, the next packet in the same input queue (if any) becomes the HOL packet of that input queue and hence is placed in the request queue of its destination.

The HOL packet is transmitted cell by cell. A HOL cell is transmitted to its destined output in a time slot if there is no HOL cells from other input queues requesting the same output link. However, during an output contention where more than one cell request a particular output port, only one HOL cell will be selected for transmission by a certain scheduling discipline. There are several disciplines by which the contention is resolved, e.g., FCFS, random selection, round-robin, etc. Since the cell transmission times are synchronized in time slots, the queuing model of an ATM switch is a discrete time model in nature. However, as shown in [7], the behavior of the system approaches that of a system with processor sharing when the length of the cell goes to zero. Furthermore, when the cell length goes to zero, the packet length distribution converges from a geometric distribution to an exponential distribution with mean $1/\mu_i$ (for input queue $i$). Therefore, the steady-state probability flow balance equation for an ATM switch can be approximated by a continuous model with exponentially distributed service times and the processor sharing service discipline. Simulation results in [7] show that the approximation is quite accurate for ATM switches. We will use this approximation throughout this paper. (The proof in [7] is for the saturation case; it is clear that the same conclusion holds when the input queues are not saturated.)

3. Packet delay in light traffic

The behavior of multi-queue/multi-server systems such as queues in a packet switch are generally formulated as a multi-dimensional birth–death process. It is, however, difficult to solve for the steady-state probability by analytical methods because of the correlation among those queues. One has to resort to some numerical methods to obtain the performance measure in such cases. The PSA approach attracts much attention for its accuracy and flexibility [9,10]. Compared with simulation, the power series method saves computation and yields more accurate results; indeed, the random error in simulation results may affect the accuracy of the Newton–Padé approximation. The fundamental idea of the PSA is to expand the system steady-state probabilities as a power series with respect to the system load as defined in Section 2, whereas the expansion coefficients can be computed by recursively solving the global balance equation. In principle, PSA can be widely applied to a large class of multiple queueing systems with Markovian arrival process (MAP) and phase-type (PH) service characteristics. However, when the state space is large, the required amount of computation becomes intolerable; hence we only apply this method to the light traffic case.

Following the model established in Section 2, we formulate the queuing behavior as a multi-dimensional Markov process. Let $\mathbf{n} = [n_1, n_2, \ldots, n_N]$ be the vector indicating the input buffer occupancy, where $n_i$ denotes the number of packets in the $i$th buffer. Because of the output contention among the input queues, a single vector $\mathbf{n}$ is insufficient to uniquely describe the system status. Hence, we have to extend the state
presentation to include the destination characterization of the HOL packets in all input queues. Let \( m = [m_1, m_2, \ldots, m_N] \) be the destination vector of the HOL packets of input queues, where \( m_i \in \{1, 2, \ldots, N\} \) denotes the destination of the HOL packet in \( i \)th queue. Clearly, \( m_i \) does not have a meaning if \( n_i = 0 \); for simplicity in notation, we define \( m_i = 0 \) if \( n_i = 0 \). All possible states \((n, m)\) collectively compose the state space \( \mathcal{S} \). Moreover, we define the vector \( \mathbf{n}_i = n_i - e_i = [n_1 - 1, \ldots, n_i - 1, \ldots, n_N] \), and similarly, \( \mathbf{n}_i^+ = n_i + e_i = [n_1 + 1, \ldots, n_i + 1, \ldots, n_N] \), where \( e_i \) is the unit vector with all entries being \( 0 \) except the \( i \)th entry being \( 1 \). Further, the sum of all components of the vector \( \mathbf{n} \) is denoted as \( |\mathbf{n}| \), i.e., \( |\mathbf{n}| = \sum_{i=1}^N n_i \).

Let \( P(n, m) \) denote the probability that the system is in state \((n, m)\). A transition from state \( \mathbf{n} \) to \( \mathbf{n}_i^+ \) occurs when a packet arrives to the \( i \)th input queue. Likewise, the transition from state \( \mathbf{n} \) to \( \mathbf{n}_i \) occurs when the HOL packet in \( i \)th queue completes its transmission. With the processor sharing rule, the departure rate of the \( i \)th HOL packet can be written as \( d_i(n, m) = \mu_i/C_i(n, m) \) for \( i > 0 \), where \( C_i(n, m) \) denotes the number of HOL packets that have the same destination as the \( i \)th HOL packet at state \((n, m)\). Obviously, \( d_i(n, m) = 0 \) if \( n_i = 0 \). With a continuous model and process sharing, the global balance equation is

\[
\rho \sum_{j=1}^N \lambda_j I(n_j \geq 2) P(n_j^+, m) + \rho \sum_{j=1}^N \lambda_j I(n_j = 1) \phi_{n_j} P(n_j^+, m - m_j e_j) \\
+ \sum_{j=1}^N \sum_{l=1}^N d_j(n_j^+, m - m_j e_j + l e_j) I(n_j > 0) \phi_{n_j} P(n_j^+, m - m_j e_j + l e_j) \\
+ \sum_{j=1}^N \sum_{l=1}^N d_j(n_j^+, m + l e_j) I(n_j = 0) P(n_j^+, m + l e_j) \\
\forall (n, m) \in \mathcal{S},
\]

where \( I(E) \) is the indicator function, i.e., \( I(E) = 1 \) if the condition \( E \) holds; otherwise, \( I(E) = 0 \). In (1), \( \phi_{n_j} \) denotes the conditional probability that the destination of the packet is \( m_j \) given that it finds no preceding packet in buffer \( j \). It cannot be obtained directly from the pre-defined routing probabilities \( \theta_{ij} \) \((i, k > 0)\) because the destination of its preceding packet cannot be found from the state \((n, m)\) with \( n_i = 0 \). A simple method to determine its value is to take the equilibrium point. This is done by solving the following conservation equation combined with the law of total probability \( \sum_{k=0}^N \phi_k = 1 \):

\[
\begin{bmatrix}
\phi_1 \\
\phi_2 \\
\vdots \\
\phi_N
\end{bmatrix} = \begin{bmatrix}
\theta_{1,1} & \theta_{1,2} & \cdots & \theta_{1,N} \\
\theta_{2,1} & \theta_{2,2} & \cdots & \theta_{2,N} \\
\vdots & \vdots & \ddots & \vdots \\
\theta_{N,1} & \theta_{N,2} & \cdots & \theta_{N,N}
\end{bmatrix} \begin{bmatrix}
\phi_1 \\
\phi_2 \\
\vdots \\
\phi_N
\end{bmatrix}
\]

(2)

The law of total probability for the steady-state probabilities gives

\[
\sum_{n_i \geq 0} \sum_{m_1}^\infty \sum_{m_2}^\infty \cdots \sum_{m_N}^\infty P(n, m) = 1.
\]

(3)
In principle, Eq. (1) can be solved numerically by applying the PSA. The details can be found in [9,10]; here we only briefly reproduce the results that are necessary for our calculation. First, it can be proved that the following limits exist for all states \((n, m) \in S\):

\[
\lim_{\rho \to 0} \rho^{-|n|} b(0; n, m) = \lim_{\rho \to 0} \rho^{-|n|} P(n, m),
\]

(4)

(The proof requires an assumption that for any state \((n, m)\) with \(|n| > 0, \sum_{j=1}^{N} d_j(n, m) > 0\). That is, at least one server will work if there is any packet in the queues, which is true in our switch model. For details, see [9].)

Next, we define

\[
Q_0(n, m) = \rho^{-|n|} P(n, m).
\]

(5)

Applying Eq. (5) to the global balance equation, we have

\[
\begin{bmatrix}
\sum_{j=1}^{N} \lambda_j + \sum_{j=1}^{N} d_j(n, m)
\end{bmatrix} Q_0(n, m)

= \sum_{j=1}^{N} \lambda_j I(n_j \geq 2) Q_0(n^*_j, m) + \sum_{j=1}^{N} \lambda_j I(n_j = 1) \phi_{n,j} Q_0(n^*_j, m - m_j e_j)

+ \rho \sum_{j=1}^{N} \sum_{l=1}^{N} d_j(n^*_j, m - m_j e_j + l e_j) I(n_j > 0) \theta_{n,j} Q_0(n^*_j, m - m_j e_j + l e_j)

+ \rho \sum_{j=1}^{N} \sum_{l=1}^{N} d_j(n^*_j, m + l e_j) I(n_j = 0) Q_0(n^*_j, m + l e_j) \quad \forall (n, m) \in S.
\]

(6)

Combining (5) with (6) yields

\[
b(0; 0, 0) = 1.
\]

(7)

Letting \(\rho \to 0\), we obtain that for \(|n| > 0,

\[
\begin{bmatrix}
\sum_{j=1}^{N} d_j(n, m)
\end{bmatrix} b(0; n, m)

= \sum_{j=1}^{N} \lambda_j I(n_j \geq 2) b(0; n^*_j, m) + \sum_{j=1}^{N} \lambda_j I(n_j = 1) \phi_{n,j} b(0; n^*_j, m - m_j e_j).
\]

(8)

Next, we define

\[
Q_k(n, m) = Q_{k-1}(n, m) - \rho^{k-1} b(k - 1; n, m), \quad k = 1, 2, \ldots
\]

Similar to (4), we can prove that the limits

\[
b(k; n, m) = \lim_{\rho \to 0} \rho^{k} Q_k(n, m)
\]
exist for all states \((n, m) \in S\). Putting \(Q_1(n, m)\) into (6) and by induction it follows that for \(k > 0\), these limits satisfy

\[
b(k; 0, 0) = - \sum_{1 \leq |n| \leq k} \sum_{m} b(k - |n|; n, m),
\]

and for \(|n| > 0\),

\[
\left[ \sum_{j=1}^{N} d_j(n, m) \right] b(k; n, m) = \sum_{j=1}^{N} \lambda_j I(n_j \geq 2) b(k; n_j^+, m - m_j e_j) + \sum_{j=1}^{N} \lambda_j I(n_j > 0) \phi_{n_j, m - m_j e_j + l e_j}) + \sum_{j=1}^{N} \sum_{l=1}^{N} d_j(n_j^+, m + l e_j) I(n_j = 0) b(k - 1; n_j^+, m + l e_j) \forall (n, m) \in S.
\]

Finally, the steady-state probability can be expanded as a power series with respect to the system load \(\rho\). That is,

\[
P(n, m) = \rho^n \sum_{k=0}^{\infty} \rho^k b(k; n, m)
\]

for any state \((n, m) \in S\).

Of course, it is infeasible to solve for all \(p(n, m)\) given the infinitely many states. However, for the light traffic case, \(\rho \ll 1\), and \(\rho^n\) approaches zero rapidly as \(|n|\) increases. In practice, we can choose an \(\epsilon \ll 1\) and an \(M > 0\), such that \(\rho^M < \epsilon\). Then we assume that \(P(n, m) = 0\) if \(|n| > M\). This certainly reduces computation. Only a reasonable number of coefficients in (11) are to be calculated. These coefficients are obtained recursively according to Eqs. (7)–(10) in such a way: we calculate \(b(0; n, m)\) for all states \((n, m)\) with \(|n| \leq M\), then we compute \(b(1; n, m)\) for all states with \(|n| \leq M - 1\), and so on, until \(b(M; 0, 0)\) is reached. As an example, we shall consider a 4 × 4 switch; under light traffic we set \(M = 5\) and the results are very accurate (see Section 5). For \(\rho < 1\), this means \(\epsilon < 0.1^5 = 0.00001\).

After the steady-state probabilities have been computed by the PSA, we can apply Little’s law to derive the average response time of the packets. This approach can yield very accurate results under light traffic with an error less than 3% compared with simulation results.

4. Asymptotic queuing behavior

The PSA presented in Section 3 can be used to obtain numerically the performance of a switch under light traffic. In this section, we study the behavior of a switch under heavy traffic. As the traffic intensity increases, incoming packets will incur longer delay and eventually, when the arrival rate \(\lambda_i\) is larger
enough (e.g., larger than $\mu_i$), the switch will enter a saturation state. In this section, we investigate the asymptotic queuing behavior near saturation.

The results in this section and the previous section will be used with the rational approximation method discussed in the next section to obtain the entire curve for the response time as a function of the input load. In order to make the rational approximation accurate, we need to calculate both the maximum throughput at which the packet delay goes to infinity as well as the rate at which the delay approaches infinity.

Under saturation, there are infinitely many packets in the input queues and the system state can be represented by the vector $r = [r_1, r_2, \ldots, r_N]$, where $r_i$ is the number of packets in the $i$th request queue. Under saturation, every input queue always has a HOL packet, so we have $\sum_{i=1}^N r_i = N$. Let $P(r)$ be the steady-state probability of $r$. Further, we define $r_0 = [r_1, \ldots, r_i-1, \ldots, r_j+1, \ldots, r_N]$ as a neighboring state of $r$. As described in Section 3, as the cell size goes to zero, the service discipline at each request queue converges to the processor sharing discipline. When a packet in request queue $i$ completes its transmission, the next packet in the same input queue, say input queue $k$, becomes the HOL packet and, hence, is placed in the destination of its request queue, say $j$. This is equivalent to a packet moving from request queue $i$ to request queue $j$, the probability of this is $\theta_{k,i,j}$. Therefore, under saturation the request queues can be modeled as a multi-class closed queuing network with transition probability $\theta_{k,i,j}$. $k$ can be viewed as the class of customer. When $\theta_{k,i,j} = \theta_{i,j}$ for all $k$, there is only one class with transition probability $\theta_{i,j}$.

Therefore, the global balance equation of the request queues of an ATM switch can be approximated as

$$
\left[ \sum_{i=1}^N I(r_i > 0) \mu_i \right] P(r) = \sum_{i=1}^N \sum_{j=1}^N I(r_j > 0) \mu_j \theta_{i,j} P(r_j),
$$

which has the same form as a closed Jackson queuing network, for which a product-form solution exists. The “normalized” maximum throughput (or utilization) $\eta$ can be obtained by solving the above equation and applying the formula $\eta = (1/N) \sum_{i=1}^N p(r_i > 0)$. This result is the same as that derived in [6].

As an example, we list in Table 1 the results for an $N \times N$ switch with symmetric routing (each packet has the same probability of going to other destinations, i.e., $\theta_{i,j} = 1/(N-1)$ for $j \neq i$).

If the arrival rate at the input queues is greater or equal to the maximum throughput, the packet response time goes to infinity. Therefore, the packet response time as a function of the input load has a pole at the maximum throughput $\eta$. In other words, there is a horizontal asymptotic line at $\eta$.

In addition to the saturation throughput, we need to identify the approaching rate of the packet response time to the asymptotic line. That is, we need to determine the pole pattern of the packet response time.

<table>
<thead>
<tr>
<th>Size, $N$</th>
<th>Maximum throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.667</td>
</tr>
<tr>
<td>3</td>
<td>0.600</td>
</tr>
<tr>
<td>4</td>
<td>0.571</td>
</tr>
<tr>
<td>5</td>
<td>0.556</td>
</tr>
<tr>
<td>6</td>
<td>0.545</td>
</tr>
<tr>
<td>7</td>
<td>0.538</td>
</tr>
<tr>
<td>8</td>
<td>0.533</td>
</tr>
</tbody>
</table>
To this end, we consider a single input queue of the switch, which is fed by a Poisson arrival process. When a cell in the input queue reaches the HOL position, if there are no other cells in the request queue of its destination, the cell is transmitted immediately with a transmission time $\tau$. If there are other cells in the corresponding request queue when the cell arrives at the HOL position, then the cell has to compete with other HOL cells. Suppose that the cell has to wait for $w$ other cells to be transmitted first, then the cell will stay in the HOL position $w$ transmission times before it starts its own transmission, and therefore, its service time can be viewed as $(w + 1)\tau$. If the discipline is FCFS, then $w$ is the number of other cells that are already in the request queue when the cell arrives. If the discipline is random selection, then $w$ has a probability distribution. Therefore, the service time of a cell has a general distribution. It is possible that the service times of the consecutive cells are correlated. However, if the cell size is very small, then each packet contains many cells, the effects of correlation will be “averaged out”. Thus, when the cell size is small, the service time of a packet can be viewed approximately as independent to each other. Therefore, the behavior of a single input queue is similar to that of an M/G/1 queue.

It is well known that the queuing time of an M/G/1 queue with an arrival rate of $\lambda$ and a service rate of $\mu$ has the following form [12]:

$$W = \frac{\lambda\bar{x}^2}{2} = \frac{c}{1 - \rho_0},$$

(13)

where $\bar{x}^2$ denotes the second moment of the service time and $\rho_0 = \lambda/\mu$ is the utilization factor.

It is obvious that near saturation the service rate (packet/s) of the single input queue is $\eta\mu$, where $\eta$ is the normalized maximum throughput. Therefore, the behavior of the single queue near saturation is

$$W = \frac{c}{1 - \lambda/\eta\mu} = \frac{c'}{\eta - \rho},$$

(14)

where $\rho = \lambda/\mu$ is the utilization at the input link.

Eq (14) also verifies that $W$ has a pole at $\eta$. We cannot determine the value of $c$ (or $c'$) at this stage, we can leave the specific value to be determined by rational approximation at the next stage, based on the knowledge of response time under light traffic obtained in Section 3.

5. Newton–Padé approximation

One major hurdle in performance analysis is the growing size in practical systems and the resultant prohibitive computation. Fortunately, it is observed that in most cases the performance curves usually have nice shapes, e.g., they are monotonic, convex or concave, etc. Because of these nice properties, researchers have paid a significant attention to interpolation/extrapolation methods; with these methods, the entire performance curve can be predicted based on a limited amount of data (usually obtained with a reasonable amount of computation).

In this paper, we use the Newton–Padé approximation to obtain the packet delay as a function of system load, based on the data in light traffic as obtained in Section 3 and the asymptotical behavior at heavy traffic. The basic idea of Newton–Padé approximation is to fit some given function values at various points by a rational function [15]. The rational function

$$R(x) = \frac{G_p(x)}{F_q(x)} = \frac{a_0 + a_1x + \cdots + a_px^p}{1 + b_1x + \cdots + b_qx^q}$$

(15)
satisfying \( R(x_i) = f(x_i) \) for \( i = 1, \ldots, p+q+1 \) is referred to as a \([p/q]\) Newton–Padé approximant of the function \( f(x) \). Gong et al. [13,14] successively applied this technique to queuing analysis, inventory systems, etc. The application of the \( \epsilon \)-algorithm in [9] is also a way to generate rational approximations.

In previous sections, we have obtained some points of the packet response times under light traffic by the PSA, we have also determined its pole value and pole pattern. We can rewrite

\[
W(\rho) = \frac{1}{\eta - \rho} \left((\eta - \rho)W(\rho)\right),
\]

hence we can finally generate the curve of packet response by applying a Newton–Padé approximation to \( R(\rho) = (\eta - \rho)W(\rho) \), which contains no pole in \( \rho \in [0, \eta) \).

The readers are referred to [13–15] for a detailed discussion about the Newton–Padé approximation. Here we simply give two examples to illustrate its application in our subject. Fig. 2 shows the curve of packet response time of a 4 \( \times \) 4 switch with a correlated traffic. Specifically, we assume \( \lambda_i = \mu_i = 0.1 \) for all \( i \) and a symmetric routing pattern. The maximum throughput is \( \eta = 0.571 \). The study in [13,14] reveals that normally a low-order Padé approximation yields good result. Therefore, we apply a \( [2/5] \) Newton–Padé approximation to \( R(\rho) \). With eight points calculated for very low traffic intensities \( \rho < 0.1 \), we obtain

\[
R(\rho) = \frac{0.5714 - 14.5711\rho + 49.0124\rho^2}{1 - 25.5008\rho + 85.2692\rho^2 + 10.6582\rho^3 + 16.8042\rho^4 - 224.3842\rho^5}
\]

![Fig. 2. Average packet delay of a 4 \( \times \) 4 switch (symmetrical routing).](image-url)
Fig. 3. Average packet delay of a $4 \times 4$ switch (asymmetrical routing).

For comparison, we plot the results obtained by simulation. A close look reveals a high agreement between both curves from simulation and rational approximation. We also constructed a $[3/5]$ and a $[4/5]$ approximation, they are not as good as the $[2/5]$ approximation, but the errors are within 20%. (For a discussion on how sensitive the approximation depends on the order of the polynomials, see [13–15].)

Fig. 3 gives another example for the asymmetric routing case with the routing matrix as listed in Table 2. Again we assume $\lambda_i = \mu_i = 0.1$. The maximum throughput under this example is found to be 0.5621, which is smaller compared with the first example because of the asymmetricity in the routing matrix. The rational approximant for this case is shown in comparison with simulation results.

Finally, we present in Fig. 4 the rational approximant of the packet response times for an uncorrelated traffic, which appears to be a special case for the correlated traffic. In this case, $\lambda_i = \mu_i = 0.25$ and

<table>
<thead>
<tr>
<th>$i$</th>
<th>$j$</th>
<th>$\theta_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0.3</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table 2: Routing probability, $\theta_{ij}$.
\[ \theta_{i,j} = \frac{1}{3} \text{ for all } j \neq i, \quad \theta_{i,i} = 0. \] Again, the approximant is shown to closely agree with the simulation results.

6. Conclusions

In this paper, we propose an approach that yields an accurate approximation for the average packet response time in a generic ATM switch. The approach combines three existing methodologies: PSA for light traffic, saturation analysis for heavy traffic, and the rational approximation for curve fitting. This approach works especially well for small to medium size switches, to which the traditional assumption of the Poisson arrival to the transmission lines does not apply well. Numerical samples reveal a close agreement between the approximant and the simulation result.

It is clear that the assumption that the packet lengths are exponentially distributed can be relaxed by using a Coxian type of distributions. The saturation analysis remains the same; the product-form solution holds for general distributions with process sharing service discipline. However, the computation complexity may increase in the power series analysis, since the state space with Coxian distributions is larger than that with exponential distributions.

Acknowledgements

This work was supported by a grant from Hong Kong RGC.
References


