Probability Density for Number of Customers Served During the Busy Period of a Finite Auto-Correlated MEP/MEP/1 Queue or Probability Density for Services During the Busy Period of a Finite Auto-Correlated MEP/MEP/1 Queue

Chaitanya Garikiparthi, Appie van de Liefvoort, Kenneth Mitchell
University of Missouri - Kansas City
School of Computing and Engineering
5100 Rockhill Rd
Kansas City, MO 64110
{cng8t7, appie, mitchellke}@umkc.edu

Abstract

In this paper we study the sample paths during a busy period of a Finite MEP/MEP/1 system, where both the arrivals and service processes can be serially auto-correlated Matrix Exponential Processes and provide a mechanism to compute the expected number of customers served during a finite queue’s busy period. We then show some numeric results by computing the probabilities of serving exactly \( n \) customers during a busy period for different allowable system (queue) sizes.

I. INTRODUCTION

Consider a server in a certain facility that has finite resources (memory, disk space etc). Most performance measure studies of interest like latencies, system times, waiting times etc study the system from the perspective of an incoming customer with an objective of reducing the delays experienced by the customer as he progresses through the system, which is fine from the customers or the arrivals perspective. But another point of view that can be equally important if one needs to take certain proactive measures, for example to avoid certain breakdowns, is to look at the system from the server’s perspective. For example, how many customers are being served by the server on a continuous basis, i.e., between the servers idle times. Understanding these characteristics could perhaps lead us to avoid certain server breakdowns due to fatigue or overload etc.

Most processes in telecommunications and computer networks exhibit a high degree of variance and are known to be serially correlated. Therefore, in order to develop accurate models to represent these systems, we need to allow for the arrival and the service processes that characterize the system to be both general and correlated.

The busy period for a system is the time interval between any two successive idle periods. It starts when a customer arrives to an empty system and ends when the departing customer leaves the system idle for the first time thereafter. In effect, a simple busy period is equivalent to a first passage from level 1 to level 0. Furthermore, the first passage from a higher level say ‘\( l \)’ to ‘\((l - 1)\)’ is also of interest. Here, if we let \( l - 1 \) denote a threshold, we are interested in the transient behavior around this threshold. Now consider that the system has finite resources which is usually the case. Models that represent these systems naturally lead to finite (non)markovian or finite QBD chains. Due to the restrictions presented by the finite boundaries and the effect of the boundary on the states leading to the boundary, certain queueing studies including the busy period analysis are more intricate for the finite system as compared to their infinite counterparts.
II. Model Description

A. Matrix Exponential Process

We use Linear Algebraic Queueing Theory (LAQT) to study the path taken by a queueing system during a busy period. Here, we briefly review the needed material. A matrix exponential (ME) distribution [13] is defined as a probability distribution whose density can be written as

\[ f(t) = p(0) \exp(-Bt)L e', \quad t \geq 0, \]

where \( p(0) \) is the starting operator for the process, \( B \) is the process rate operator, and \( e' \) is a summing operator, a vector usually consisting of all 1’s. The \( n^{th} \) moment of the matrix exponential distribution is given by \( E[X^n] = n!p(0)V^n e' \), where \( V \) is the inverse of \( B \). The class of matrix exponential distributions is identical to the class of distributions that possess a rational Laplace-Stieltjes transform. As such, it is more general than continuous phase type distributions which have a similar appearance.

The joint density function for the first \( k \)-successive events is described by a Matrix Exponential Process (MEP).

\[ f_k(x_1, \ldots, x_k) = p(0) \exp(-B x_1) L \ldots \exp(-B x_k) L e', \]

where matrix \( L \) is the event generator matrix, \( p \) is the starting state for the process and \( e' \) is a summing operator, a vector usually consisting of all 1’s. If the process is stationary, then the starting vector \( p \) satisfies \( p(0) = p(0)V L \).

Examples for such processes are a Poisson process \( (B=[\lambda], L=[\lambda]) \), a renewal process \( (L = Be'/p) \), and a Markovian Arrival Process (MAP) \( (B = -D_0, L = D_1) \). Note that \( B \) and \( L \) are not limited to being Markovian rate matrices. So every MAP is an MEP, but not vice versa (see also [7]). By implication, stationary MEP’s are dense in the family of all stationary point processes as well, [2]. For additional details, see [13], [9], [8].

B. \( H \) Operators

Let the arrival and service processes be represented by \( <B_a, L_a> \) and \( <B_s, L_s> \) respectively. The conditional probability that an arrival event occurs before the service event given that the starting vector is \( p(0) \) is given by

\[ \Pr[A < S \mid p(0)] = p(0)(\widehat{B}_a + \widehat{B}_s)^{-1}\widehat{L}_a e'. \]

where \( \widehat{B}_a = B_a \otimes I_s, \widehat{B}_s = I_a \otimes B_s, \widehat{L}_a = L_a \otimes I_s \) and \( \widehat{L}_s = I_a \otimes L_s \), and \( \otimes \) is the Kronecker product operator which embeds the arrival and service processes into system space. Here, \( (\widehat{B}_a + \widehat{B}_s)^{-1} \) represents the average time that both the arrival and service processes are concurrently active, and \( \widehat{L}_a \) represents the arrival event occurring before a service completion. The trailing \( e' \) sums up the probabilities distributed in vector form and is usually a column vector of all one’s of appropriate dimensions.

In an MEP/MEP/1 system, the conditional probability that two successive events are both arrivals is \( p(0) \) is \( p(0)(\widehat{B}_a + \widehat{B}_s)^{-1}\widehat{L}_a \cdot (\widehat{B}_a + \widehat{B}_s)^{-1}\widehat{L}_a e'. \) Define operators \( H_a \) for arrival event happening before the service and \( H_s \) for service event happening before the arrival by unconditioning on the initial state of the system, as follows:

\[ H_a = (\widehat{B}_a + \widehat{B}_s)^{-1}\widehat{L}_a \quad \text{and} \quad H_s = (\widehat{B}_a + \widehat{B}_s)^{-1}\widehat{L}_s. \]

Essentially these \( H \) operators allow us to track the path evolution by embedding at the event transitions in the continuous time Markov chain. At each observed transition point, the appropriate \( H \)-operator is applied (and normalized if needed) to update the internal state of the discrete time Markov chain, thus allowing both the arrival and service processes involved to be non-renewal. We summarize what \( H_a \) and \( H_s \) are for different systems in the table I.

Please note that the \( H \)-operators introduced here differ from the similarly named operators in [13].
\[
\begin{array}{c|c|c}
\text{Type} & \frac{1}{\lambda + \mu} & \frac{1}{\lambda + \mu} \\
\hline
\text{M/M/1} & (\lambda I + B_s)^{-1}\lambda & (\lambda I + B_s)^{-1}B_s e_p \\
\text{M/M/1} & (B_s + \mu I)^{-1}B_s e_p & (B_s + \mu I)^{-1}\mu \\
\text{ME/ME/1} & (\overline{B}_s + B_s)^{-1}\overline{L}_a & (\overline{B}_s + B_s)^{-1}\overline{L}_s \\
\end{array}
\]

\text{TABLE I}

H operators for different systems

C. Summary of Results for Infinite MEP/MEP/1 Queue

In this section we present the summary of results presented in PREVIOUS PAPER as they relate to the derivations for the finite queue case. Consider a system that just had a transition from level \((l - 1)\) to level \(l\) and let \(p(0)\) be the current internal state of the system. Let \(D_{l,l-1}\) represent the first passage process wherein the system transitions from level \(l\) to level \((l - 1)\) ending when level \((l - 1)\) is reached for the first time. A busy period is a special case of this first passage when \(l = 1\). Let \(N_{l,l-1}\) be the discrete random variable for the number of customers served during the first passage \(D_{l,l-1}\). Define \(Y_i\)'s as follows:

\[
\begin{align*}
Y_0 &= I \\
Y_{n-1} &= \sum_{i=0}^{n-2} H_a Y_{n-i-2} H_s Y_i, \quad n > 1.
\end{align*}
\]

where \(I\) is an identity matrix of the dimensions of either the service process or the arrival process whichever is an MEP and it would be in the product space if both of these are MEPS. \(Y_i\) is the operator that transfers the internal state of the system as the system transitions from level \(l\) back to level \(l\) while traversing only states \(l, l+1, l+2, \ldots\) and after having served exactly \(i\) customers. Here, \(Y_i\) is independent of the level \(l\), as all the information that differentiates transitions for different levels is present in the system starting vector on which \(Y_i\) operates, and \(Y_i\) depends only on the number of arrivals and departures. The operator \(Y_i H_s\) represents serving exactly \((i + 1)\) customers while transitioning down by one level.

The probability that exactly \(n\) customers are served during \(D_{l,l-1}\) conditioned on the internal system state being in \(p(0)\) at the transition from level \((l - 1)\) to level \(l\) is given by,

\[
d_{n,l} \triangleq \text{Prob}[N_{l,l-1} = n | p(0)] = p(0)Y_{n-1} H_s e', \quad n \geq 1.
\]

The \(z\)-transform for the number of customers served during this first passage \(D_{l,l-1}\) is

\[
y(z) = \sum_{n=1}^{\infty} \text{Prob}[N_{l,l-1} = n] z^n = b_1 z + b_2 z^2 + \ldots
\]

Since \(Y_{n-1}\) forms the core of \(d_{n,l}\), one can now define the matrix \(z\)-transform

\[
Y(z) = z I + H_a Y(z) H_s Y(z)
\]

(3)

III. Busy Period of a Finite MEP/MEP/1 Queue

Consider a finite queueing situation where the queue size is limited to \(q\). Let \(Y_i^j\) represent all possible paths that represent exactly \(i\) arrivals and departures (exactly \(i\) customers being served) that are of height less than or equal to \(j\), but with possible loops at level \(j\), representing the customers that get dropped when the queue is full (\(j = k\)).

Figure. 1 shows all possible paths wherein exactly three arrivals and three departures occur during a sample path, such that the sample path always stays above the starting level (level 1, in this case) and ends exactly at this level;
i.e., all sample paths that begin at some level and end at the same level never taking any excursions below that level. In the case of an infinite queue there are five such possible paths (the count given by Catalan numbers). Let these paths be represented by $Y_3$.

\[ Y_3 = \begin{cases} I \end{cases} \]

Fig. 1. Paths with exactly three arrivals and three departures

If now, we require that the sample paths do not go above a given height (the freedom of the sample paths to go beyond a certain height is curbed) representing a finite queue, and we allow the arrival process to be active when the queue is full, but that the arrivals to a full queue get dropped. All possible sample paths that represent exactly three arrivals and three departures within a channel of width two are shown in Figure 2 (or alternatively if the channel starts after the first arrival to an empty queue, all possible paths that are of height less than or equal to three are shown). Note that we now have possible loops at the tops of the sample paths when the tops correspond to allowable height and the loop representing the arrivals that are being dropped when the queue is full. The count for the number of paths that represent serving a given number of arrivals and services for this finite queue are not given by the Catalan numbers anymore.

Let $N_{1,0}$ be the discrete random variable for the number of customers served during a busy period when the maximum allowable system size is $s$, and let $d_{n,1}$ represent the probability that $N_{1,0} = n$.

\[ d_{n,1} = \begin{cases} \text{probability} \end{cases} \]

Fig. 2. Paths with exactly three arrivals and three departures within a channel of width two

The problem of computing $d_{n,1}$ can now be visualized as finding all possible paths that fit within a channel of height $q$ ($q = s - 1$), and that have exactly $n - 1$ up transitions and $n - 1$ down transitions (note that the channel being considered starts after the arrival of the first customer, hence the $n - 1$ arrivals and service completions in the channel plus one additional service that completes the transition of the sample paths from level one to level zero at the end, completes the corresponding busy period, resulting in $n$ customers being served). We now have the following set of recursive definitions for various sample paths that serve $i$ customers within a channel of width two, $Y_i^2$:

\[
Y_0^2 = I \\
Y_1^2 = H_a H_s \\
Y_2^2 = H_a Y_0^1 H_s Y_1^2 + H_a Y_1^1 H_s Y_0^2 \\
Y_3^2 = H_a Y_0^1 H_s Y_2^2 + H_a Y_1^1 H_s Y_1^2 + H_a Y_2^1 H_s Y_0^2 \\
\vdots \quad \vdots \\
Y_n^2 = H_a Y_0^1 H_s Y_{n-1}^2 + H_a Y_1^1 H_s Y_{n-2}^2 + \ldots + H_a Y_{n-1}^1 H_s Y_0^2
\]
It is to be noted here that the number of paths that represent serving exactly \( n \) customers is not given by the Catalan number, but once the possible loops at the tops are accounted for, the general structure of the possible paths still resembles the general Catalan recursion \((C_n = \sum_{i=0}^{n-1} C_i C_{n-i-1})\).

For different allowable heights, we have a complete set of \( Y_i^j \)'s. The following matrix gives a better understanding of the relationship between all the different \( Y_i^j \)'s and the \( Y_i \)'s that we see in the case of infinite queues.

\[
\begin{bmatrix}
Y_1^1 & Y_1 & Y_1 & \ldots & \ldots & \ldots \\
Y_2^1 & Y_2^2 & Y_2 & Y_2 & \ldots & \ldots \\
Y_3^1 & Y_3^2 & Y_3^3 & Y_3 & \ldots & \ldots \\
Y_4^1 & Y_4^2 & Y_4^3 & Y_4^4 & Y_4 & \ldots & \ldots \\
Y_5^1 & Y_5^2 & Y_5^3 & Y_5^4 & Y_5 & Y_5 & \ldots & \ldots \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots
\end{bmatrix}
\]

It should be noted from the recursive definitions for \( Y_i^j \)'s and the general matrix structure for different \( Y_i^j \)'s, that once the boundary equations for \( Y_i^j \)'s have been defined and the first column of the matrix is defined, every other element of the matrix can be computed by a Catalan like recursion using the elements that have already been computed, even though the number of paths is not equal to the Catalan number. In-fact, there are an infinite number of paths that correspond to every situation where there is a loop.

\( Y_i^j \)'s can be computed as follows:

\[
Y_0^0 = (I - H_a)^{-1}, \quad Y_1^1 = Y_0^2 = \ldots = I \\
Y_1^1 = H_a(I - H_a)^{-1} H_s, \quad Y_n^1 = (Y_1^1)^n \quad n \geq 2 \\
Y_i^j = \sum_{k=0}^{i-1} H_a Y_k^{j-1} H_s Y_{i-k}^j \quad i \geq 2, 2 \leq j \leq i \\
(4)
\]

In the case of infinite queues, the number of possible paths in which \( n \) customers can be served is given by the \( n - 1 \)st Catalan number; in infinite queues we are dealing with \( Y_i \) without much regard for the ceiling, as the effective ceiling was at infinity. So in-fact \( Y_i \) really represents \( Y_i^\infty \). For finite queues, we now have a full gamut of \( Y_i^j \)'s, and the number of possible paths is not given by the Catalan numbers, but the general structure of Catalan recursion is still preserved.

The starting vector for the busy period in this finite queueing case is computed using

\[
p_{bp} Y^q H_s Y_a = p_{bp}
\]

where, \( Y^q = \sum_{i=0}^{\infty} Y_i^q \) represents all possible paths that lie within a channel of width \( q \). The invariance equation for \( p_{bp} \) is similar to that of the infinite queue case except that now the paths that comprise the \( Y \) are limited by the size of the queue, hence replaced by \( Y^q \). The intuition is still valid, that at the start of a random busy period if the starting vector is \( p_{bp} \) then following one of the possible paths \( Y^q \) the busy period ends \((H_s)\), followed by an arrival event occurring causing the start of the next busy period; and in the absence of any more information about the start of the busy period, this busy period is best estimated as a random busy period, hence the invariance of the starting vector.
Alternatively if $X_h$ represents all possible paths within a band of height $h$ with possible loops at the top, then for a given maximum system size $s$, $X_{s-1}$ can be computed using,

$$X_0 = (I - H_a)^{-1}$$
$$X_1 = (I - H_aX_0H_d)^{-1}$$
$$X_2 = (I - H_aX_1H_d)^{-1}$$
$$\vdots$$
$$X_{s-1} = (I - H_aX_{s-2}H_d)^{-1}$$

And the starting vector $p_{bp}$ is computed using

$$p_{bp}X_{s-1}H_sV_aL_a = p_{bp}$$

Now, the probability that exactly $n$ customers are served during a busy period of a finite queue where the maximum height of a sample path or the maximum system size is restricted to $s$ (corresponding to a maximum channel width of size $q = s - 1$ followed by the first arrival), is given by

$$d_{n,1} = \text{Prob}[N_{1,0} = n] = p_{bp}Y_{n-1}H_s\epsilon, \quad n \geq 1$$

A. Mean number served during a finite queue busy period

For a given channel width $q$, define the matrix z-transform $Y^q(z) = Y_0^qz^1 + Y_1^qz^2 + Y_2^qz^3 + \ldots$ for $q > 1$. We can now derive the following recurrence relation for $Y^q(z)$.

$$z^1Y^q_0 = I^1$$
$$z^2Y^q_1 = (H_aY_0^q - 1z^1H_sY_0^q - 1)$$
$$z^3Y^q_2 = (H_aY_1^q - 1z^2H_sY_0^q - 1 + H_aY_0^q - 1z^1H_sY_1^q - 1)$$
$$\vdots$$
$$z^{n+1}Y^q_n = (H_a[Y_{n-1}^q z^n H_sY_0^q - 1 + Y_{n-2}^q z^{n-1} H_sY_1^q - 1 + \ldots + Y_0^q - 1 z^1 H_sY_{n-1}^q]$$

$$Y(z)^q = zI + H_a(Y_0^q z^1 + Y_1^q z^2 + Y_2^q z^3 + \ldots)$$
$$H_s(Y_0^q z^1 + Y_1^q z^2 + Y_2^q z^3 + \ldots)$$

Thus, $Y^q(z)$ satisfies the matrix recurrence equation

$$Y^q(z) = zI + H_aY^q - 1(z)H_sY^q(z) \quad q > 1$$

(5)

Notice the similarity to the matrix quadratic equation in relation to the infinite queue situation. However we now have different $Y^q(z)$’s for different allowable queue sizes, $q$. The boundary equation in the case where $q = 1$ is as follows

$$Y^1(z) = Y_0^1z^1 + Y_1^1z^2 + Y_2^1z^3 + \ldots$$
$$= zI + Y_1^1z^2(I + Y_1^1)z + (Y_1^1)^2z^2 + \ldots$$
$$= zI + Y_1^1z^2(I - Y_1^1)^{-1}$$

At $z = 1$,

$$Y^1(1) = I + Y_1^1(I - Y_1^1)^{-1}$$
$$= (I - H_a(I - H_a)^{-1}H_d)^{-1}$$
Hence for any given allowable height \( q \), \( Y^q(1) \) can be computed using Equation 5. Taking the derivative of Eq. 5 w.r.t \( z \) and evaluating at \( z = 1 \) gives,

\[
Y^{q'}(1) = I + H_a Y^{q-1'}(z)H_s Y^q(1) + H_a Y^{q-1}(1)H_s Y^{q'}(1)
\]  

(6)

The base case when \( q = 1 \), is

\[
Y^1(z) = Iz + Y_1^1z^2 + Y_2^1z^3 + \ldots
\]

Taking the derivative at evaluating at \( z = 1 \),

\[
Y^{1'}(1) = I + 2(Y_1^1)^2 + 3(Y_1^1)^3 + 4(Y_1^1)^4 + \ldots
\]

\[
= (I + Y_1^1 + Y_1^{12} + Y_1^{13} + \ldots) + (Y_1^1 + Y_1^{12} + Y_1^{13} + \ldots) + \ldots
\]

\[
= (I - Y_1^1)^{-1} + Y_1^1(I - Y_1^1)^{-1} + Y_1^2(I - Y_1^1)^{-1} + \ldots
\]

\[
= (I - Y_1^1)^{-1}(I - Y_1^1)^{-1}
\]

\[
= (I - H_a(I - H_a)^{-1}H_d)^{-2}
\]

Hence we can compute \( Y^{q'}(1) \) for a given \( q \). Now the mean number of customers served during a busy period when the allowable system size is \( s \), is given by

\[
E[N_{i,0}^s] = p(0) Y^{s-1'}(1)H_se'.
\]  

(7)

It is to be noted that though the busy period analysis for a finite queue is similar to that of an infinite queue as presented in PREVIOUS PAPER, it is considerably more intricate due to the fact that we now have an upper-bound on the height that a sample path can take. The constructive mechanism presented in the case of the infinite queue however does provide a basic mechanism to study the finite queue.

IV. MM1 CASE

Consider a simple \( M/M/1 \) case where the \( H_a \) and \( H_s \) are scalars, some simplifications are evident. For example, since \( (I - H_a)^{-1}H_s \) is now equal to 1, we have

\[
Y_1^1 = H_a, \quad Y_n^1 = H_a^n \quad n \geq 2.
\]

But no better structure is evident yet for higher level \( Y_i^{j'} \)'s than as defined by Eqs. (4). Consider that the maximum attainable system size is three i.e., the corresponding markov chain does not go beyond three states and any additional arrivals that occur when the system is in state three get dropped. As already noted the count process for the number of possible paths that serve \( n \) customers during a busy period of this system is not given by the catalan numbers. In-fact, when the system is in state three, there can be an infinite number of arrivals that get dropped (represented by \( (I - H_a)^{-1} \)) resulting in an infinite number of possible paths. Hence no closed form expressions for the count process are attempted. We can however for any given fixed queue size compute the probabilities for \( n \) customers served during a busy period using Eq. (4). For an \( M/M/1 \) queue with a utilization of 0.7, the probabilities \( d_{n,1}^s \) are shown in Table II

V. MEP/MEP/1 CASE

Consider an \( MEP/MEP/1 \) system where the arrival process is represented by

\[
p_a = \left[ \begin{array}{cc} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{array} \right], \quad B_a = 1.4 * \left[ \begin{array}{cc} 1 & 0 \\ 0 & 25 \end{array} \right], \quad L_a = 1.4 * \left[ \begin{array}{cc} (1 + \gamma_a) & (1 - \gamma_a) \\ (1 - \gamma_a) & (1 + \gamma_a) \end{array} \right],
\]
where $\gamma_a = 0.9$, is the parameter that controls the correlation decay of the process. Note that the marginal distribution is independent of $\gamma_a$ and has a mean of 0.3714 and a squared coefficient of variation $c^2 = 2.7$. Similarly, consider the service process represented by

$$p_s = \begin{bmatrix} \frac{1}{4} & \frac{3}{4} \end{bmatrix}, \quad B_s = \begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix}, \quad L_s = \begin{bmatrix} 2.775 & 0.225 \\ 0.1 & 3.9 \end{bmatrix}.$$

This system has a utilization of 0.73 and the probabilities of serving $n$ customers in a busy period for different allowable system sizes ($d_{n,1}$) are shown in Table III.

**Table II**

<table>
<thead>
<tr>
<th>$d_{n,1}$</th>
<th>$s = 3$</th>
<th>$s = 5$</th>
<th>$s = 10$</th>
<th>$s = 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.588235</td>
<td>0.588235</td>
<td>0.588235</td>
<td>0.588235</td>
</tr>
<tr>
<td>2</td>
<td>0.142479</td>
<td>0.142479</td>
<td>0.142479</td>
<td>0.142479</td>
</tr>
<tr>
<td>3</td>
<td>0.093178</td>
<td>0.069021</td>
<td>0.069021</td>
<td>0.069021</td>
</tr>
<tr>
<td>4</td>
<td>0.060937</td>
<td>0.041795</td>
<td>0.041795</td>
<td>0.041795</td>
</tr>
<tr>
<td>5</td>
<td>0.039851</td>
<td>0.028345</td>
<td>0.028345</td>
<td>0.028345</td>
</tr>
<tr>
<td>6</td>
<td>0.026062</td>
<td>0.020597</td>
<td>0.020597</td>
<td>0.020597</td>
</tr>
<tr>
<td>7</td>
<td>0.017044</td>
<td>0.015679</td>
<td>0.015679</td>
<td>0.015679</td>
</tr>
<tr>
<td>8</td>
<td>0.011146</td>
<td>0.012343</td>
<td>0.012343</td>
<td>0.012343</td>
</tr>
<tr>
<td>9</td>
<td>0.00729</td>
<td>0.009965</td>
<td>0.009965</td>
<td>0.009965</td>
</tr>
<tr>
<td>10</td>
<td>0.004767</td>
<td>0.008207</td>
<td>0.008207</td>
<td>0.008207</td>
</tr>
</tbody>
</table>

**Table III**

<table>
<thead>
<tr>
<th>$d_{n,1}$</th>
<th>$s = 3$</th>
<th>$s = 5$</th>
<th>$s = 10$</th>
<th>$s = 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.692976</td>
<td>0.692976</td>
<td>0.692976</td>
<td>0.692976</td>
</tr>
<tr>
<td>2</td>
<td>0.130622</td>
<td>0.130622</td>
<td>0.130622</td>
<td>0.130622</td>
</tr>
<tr>
<td>3</td>
<td>0.068845</td>
<td>0.050156</td>
<td>0.050156</td>
<td>0.050156</td>
</tr>
<tr>
<td>4</td>
<td>0.038867</td>
<td>0.024632</td>
<td>0.024632</td>
<td>0.024632</td>
</tr>
<tr>
<td>5</td>
<td>0.023346</td>
<td>0.013933</td>
<td>0.013933</td>
<td>0.013933</td>
</tr>
<tr>
<td>6</td>
<td>0.014738</td>
<td>0.008724</td>
<td>0.008724</td>
<td>0.008724</td>
</tr>
<tr>
<td>7</td>
<td>0.009649</td>
<td>0.005931</td>
<td>0.005931</td>
<td>0.005931</td>
</tr>
<tr>
<td>8</td>
<td>0.006477</td>
<td>0.003348</td>
<td>0.003348</td>
<td>0.003348</td>
</tr>
<tr>
<td>9</td>
<td>0.004418</td>
<td>0.002724</td>
<td>0.002724</td>
<td>0.002724</td>
</tr>
<tr>
<td>10</td>
<td>0.003046</td>
<td>0.002724</td>
<td>0.002724</td>
<td>0.002724</td>
</tr>
</tbody>
</table>

Count for the number of paths
As already noted when the ceiling on the height a sample path can take is defined, the number of possible paths is not given by the catalan number. But for different heights, we get a series of related number sequences, if we consider all the arrivals that get dropped at the end of the queue to be counted as a single possible path. Table III.

**References**
