The State Probabilities of the System $M|M|m|n$ with the Waiting Time Restriction

Artur Vardanyan
Institute for Informatics and Automation Problems of NAS RA
Yerevan, Armenia
e-mail: artvardanyan@asnet.am

Vladimir Sahakyan
Institute for Informatics and Automation Problems of NAS RA
Yerevan, Armenia
e-mail: vladimir.sahakyan@sci.am

ABSTRACT
The queue state in multiprocessor computing systems is an actual problem for their use in the process of tasks scheduling. In this paper, a system of differential equations is obtained describing the probabilities of the system state at time t, which can be solved using numerical methods. However, the probabilities sought-for the system $M|M|m|n$ with a constraint on the residence time can be obtained in terms of the Laplace-Stieltjes transform.

Keywords
Multiprocessor Cluster-Type System, Cluster computing, Queueing theory, Multiprocessor Queueing System, Waiting time restriction.

1. INTRODUCTION
In classical queueing theory it is usually assumed that tasks that cannot get service immediately after arrival either join the queue (and then are served according to some queueing discipline) or leave the system forever. Sometimes tasks arriving for execution may be "impatient", that is, they leave the queue after a certain waiting time [1,2].
This paper addresses the problem of obtaining the state probabilities of the system $M|M|m|n$ for the exponential distribution of the arrival, execution, and service failure tasks when each task has a waiting time restriction.

2. SYSTEM DESCRIPTION
Suppose that a task stream enters a computing system consisting of m processors ($m \geq 1$). Each task is characterized by three random parameters (ν, β, ω), where ν is the number of computational resources(processors, cores, cluster nodes, etc.,) required to perform the task, β is the maximum time required to complete the task and ω is the possible time that the task can wait before assigning to run, after which it leaves the system without service [3].
By using David Kendall's notation(which is widely used to describe elementary queueing systems), the system under consideration can be represented as $M|M|m|n$. So, the system parameters are described:
m - the maximum number of computational resources;
n - the maximum permissible number of tasks in the queue;
α - a random value of the time interval between neighboring entrances, which has the probability distribution:

$$P(\alpha < t) = 1 - e^{-at},$$

where a is the intensity of the incoming stream;
β - a random value of the task execution time, which has the probability distribution:

$$P(\beta < t) = 1 - e^{-bt},$$

where b is the intensity of service;
ω - a random value of the permissible waiting time for a task in the queue, which has the probability distribution:

$$P(\omega < t) = 1 - e^{-wt},$$

where w is the intensity of the failure of service for a task from the queue;
ν - a random value of the number of required computational resources for performing a task, which has the probability distribution:

$$P(\nu \leq k) = \frac{k}{m}, k = 1, 2, ..., m.$$

Tasks will be accepted for service in the order of their entry into the system, i.e., FIFO discipline is used (First-In-First-Out). Those tasks that arrive at the time of full occupation of the queue (there are already n tasks in the queue) receive a denial of service.

3. BASIC NOTATIONS AND EQUATIONS
To analyze our system we need to identify the following basic notation:
$L_{i,j}$ - the state of the system when i tasks are serviced and j tasks are waiting in the queue,
$P_{i,j}(t)$ - the probability that the system is in the $L_{i,j}$ state at the moment of time t.
Due to finite numbers n and m, the number of possible states of the system is finite.
It is known, that the total flow from several elementary flows is also elementary, and the probability that more than one event can occur during a short h time is $o(h)$. Considering this fact, let's list all possible cases related to states of the system at the moment of time t, when during the h time the system goes into the state $L_{i,j}$:

1. at the moment of time t, the state of the system was $L_{i,j}$ and over the next h time there was no change in the system;
2. at the moment of time t, the state of the system was $L_{i,j-1}$ and over the next h time one task arrived and joined the queue;

3. at the moment of time t, the state of the system was $L_{i-k+1,j+k}$, where $k = 1, 2, ..., min(i, n - j)$ and over the next h time one task completed the service and left the system, the first k tasks from the queue were accepted to service;

4. (a) at the moment of time t, the state of the system was $L_{i,j+1}$ and over the next h time one task from the queue, not the first task, left the queue(one's waiting time ran out);

   (b) at the moment of time t, the state of the system was $L_{i-k,j+k+1}$, where $k = 0, 1, ..., i - 1$ and over the next h time the first task of the queue left the queue(its waiting time ran out) and the first k tasks from the queue were accepted to service.

Obviously, the probability that the state of the system at the moment of time $t + h$ will be $L_{i,j}$ is the sum of the probabilities of the above cases, it follows that

$$\begin{aligned} P_{i,j}(t+h) = q^{(1)}_{i,j}(t,h)+ \\ + \theta_j q^{(2)}_{i,j}(t,h)+ \\ + \eta_j q^{(3)}_{i,j}(t,h)+ \\ + \eta_j q^{(4)}_{i,j}(t,h) + o(h), \end{aligned} \quad (1)$$

where $0 \le i \le m$, $0 \le j \le n$,

$$\eta_j = \begin{cases} 0, & \text{for } j = n \\ 1, & \text{for } 0 \le j < n \end{cases},$$

$$\theta_j = \begin{cases} 0, & \text{for } j = 0 \\ 1, & \text{for } 0 < j \le n \end{cases}.$$

and $q^{(1)}_{i,j}(t,h)$, $q^{(2)}_{i,j}(t,h)$, $q^{(3)}_{i,j}(t,h)$, $q^{(4)}_{i,j}(t,h)$ are probabilities for appropriate cases:

$$q^{(1)}_{i,j}(t,h) = P_{i,j}(t)e^{-(ib+jw+a)h},$$

$$q^{(2)}_{i,j}(t,h) = ahP_{i,j-1}(t)e^{-(ib+(j-1)w+a)h},$$

$$q^{(3)}_{i,j}(t,h) = \sum_{k=0}^{l_1}\Big(\delta_{i,k}(i-k+1)bhP_{i-k+1,j+k}(t)* \\ *\overline{P}(i,j,k)e^{-((i-k+1)b+(j+k)w+a)h}\Big),$$

where $l_1 = min(i, n - j)$,

$$\delta_{i,k} = \begin{cases} 0, & \text{for } i = m \text{ and } k = 0 \\ 1, & \text{for otherwise} \end{cases},$$

and if $i = 0$ and $0 < j \le n$, then $\overline{P}(i,,j,k) = 0$ and if $i = 0$ and $j = 0$, then $\overline{P}(i,,j,k) = 1$ but for otherwise $\overline{P}(i,j,k)$ is the following conditional probability:

$$\overline{P}(i,j,k) = P\Bigg(\sum_{s=1}^{i-k}\nu_s + \sum_{s=i-k+2}^{i+1}\nu_s \le m < \sum_{s=1}^{i-k}\nu_s + \\ + \sum_{s=i-k+2}^{i+2}\nu_s \Bigg/ \sum_{s=1}^{i-k+1}\nu_s \le m < \sum_{s=1}^{i-k+2}\nu_s\Bigg),$$

here it is assumed that ν_{i-k+1} is the number of required computational resources required to service the task that has left the system(it was serviced over the h time),

$$q^{(4)}_{i,j}(t,h) = jwhP_{i,j+1}(t)e^{-(ib+(j+1)w+a)h}+ \\ + \sum_{k=0}^{l_2}\Big(whP_{i-k,j+k+1}(t)\overline{\overline{P}}(i,k)e^{-((i-k)b+(j+k+1)w+a)h}\Big),$$

where $l_2 = min(i, n - j)$, $\overline{\overline{P}}(i,k) = 0$ if $i = 0$, but if $0 < i \le m$, then $\overline{\overline{P}}(i,k)$ is the following conditional probability:

$$\overline{\overline{P}}(i,k) = P\Bigg(\sum_{s=1}^{i-k}\nu_s + \sum_{s=i-k+2}^{i+1}\nu_s \le m < \sum_{s=1}^{i-k}\nu_s + \\ + \sum_{s=i-k+2}^{i+2}\nu_s \Bigg/ \sum_{s=1}^{i-k}\nu_s \le m < \sum_{s=1}^{i-k+1}\nu_s\Bigg),$$

here it is assumed that ν_{i-k+1} is the number of required computational resources required to service the task that has left the queue(its waiting time ran out over the h time).

To calculate $\overline{P}(i,j,k)$, $\overline{\overline{P}}(i,k)$ and some other useful probabilities, we present the formulas in the next section of this article.

Note if $i = 0$ for all $0 < j \le n$

$$P_{0,j}(t) = 0, \quad (2)$$

and

$$\sum_{i=0}^{m}\sum_{j=0}^{n} P_{i,j}(t) = 1. \quad (3)$$

4. LEMMAS AND FORMULAS FOR SOME USEFUL PROBABILITIES

This section presents some useful lemmas and the calculation of the values of some probabilistic characteristics. By $P(i,k)$ is denoted the probability that k processors will be occupied by i tasks:

$$P(i,k) = P\left(\sum_{j=1}^{i}\nu_j = k\right).$$

Lemma 4.1. *The probability that k processors will be occupied by i tasks, can be obtained in the following way:*

$$P(i,k) = \frac{1}{m^i}\binom{k-1}{i-1}, 1 \le i \le k \le m.$$

Proof. To prove the lemma we use the mathematical induction technique. The method of induction requires two cases to be proved. The first case, called the base case, proves that the property holds for $i = 1$:

$$P(1,k) = \frac{1}{m}\binom{k-1}{0} = \frac{1}{m}.$$

The statement is true because if $i = 1$, then

$$P(1,k) = P(\nu = k) = \frac{1}{m}.$$

The second case, called the induction step, proves that if the property holds for number $i - 1$, then it holds for the next natural number i:

$$P(i,k)=\sum_{j=i-1}^{k-1}P(i-1,j)P(1,k-j)= \\ =\frac{1}{m}\sum_{j=i-1}^{k-1}\frac{1}{m^{i-1}}\binom{j-1}{i-2}= \\ =\frac{1}{m^i}\sum_{j=i-1}^{k-1}\binom{j-1}{i-2}. \quad (4)$$

From Combinatorics we know this equality:

$$\binom{i}{i}+\binom{i+1}{i}+...+\binom{i+k-1}{i}=\binom{i+k}{i+1}. \quad (5)$$

Considering the equality (5) to count (4), we get the formula, which was mentioned in Lemma 2.1.:

$$P(i,k)=\frac{1}{m^i}\binom{k-1}{i-1}.$$

Lemma 4.2. *The probability that i tasks will occupy no more than k processors, can be obtained in the following way:*

$$P\left(\sum_{j=1}^{i}\nu_j\le k\right)=\frac{1}{m^i}\binom{k}{i}, 1\le i\le k\le m.$$

Proof. To prove the lemma we use the formula, which we got in Lemma 2.1.

$$P\left(\sum_{j=1}^{i}\nu_j\le k\right)=\sum_{j=i}^{k}P(i,j)=\sum_{j=i}^{k}\frac{1}{m^i}\binom{i-1}{j-1}= \\ =\frac{1}{m^i}\sum_{j=i}^{k}\binom{i-1}{j-1}.$$

To calculate the last sum, we again use the equality (5) and as a result we get that

$$P\left(\sum_{j=1}^{i}\nu_j\le k\right)=\frac{1}{m^i}\binom{k}{i}.$$

Lemma 4.3.

$$P\left(\sum_{i=1}^{k}\nu_i\le s<\sum_{i=1}^{k+1}\nu_i\right)=\frac{1}{m^{k+1}}\left(m-\frac{s-k}{k+1}\right)\binom{s}{k},$$

where $1\le k\le s\le m$.

Proof. It's obvious that:

$$P\left(\sum_{i=1}^{k}\nu_i\le s<\sum_{i=1}^{k+1}\nu_i\right)= \\ =\sum_{j=k}^{s}P\left(\sum_{i=1}^{k}\nu_i=j\right)P\left(\nu_{k+1}>s-j\right).$$

Primarily, we use the obvious fact that

$$P\left(\nu_{k+1}>s-j\right)=\frac{m-s+j}{m},$$

and then we use the formula, which we got in Lemma 2.1. for the first probability in sum, as a result we get:

$$P\left(\sum_{i=1}^{k}\nu_i\le s<\sum_{i=1}^{k+1}\nu_i\right)= \\ =\frac{1}{m^{k+1}}\left(\sum_{j=k}^{s}(m-s)\binom{j-1}{k-1}+\sum_{j=k}^{s}j\binom{j-1}{k-1}\right)= \\ =\frac{1}{m^{k+1}}\left((m-s)\binom{s}{k}+k\binom{s+1}{k+1}\right)= \\ =\frac{1}{m^{k+1}}\left(m-\frac{s-k}{k+1}\right)\binom{s}{k}.$$

To calculate $\overline{P}(i,j,k)$ probability, we first perform a simple transformation, then use the conditional probability formula:

$$\overline{P}(i,j,k)=P\left(\sum_{s=1}^{i+1}\nu_s\le m+\nu_{i-k+1}<\sum_{s=1}^{i+2}\nu_s \middle/ \sum_{s=1}^{i-k+1}\nu_s\le m<\sum_{s=1}^{i-k+2}\nu_s\right)= \\ =P\left(\sum_{s=1}^{i+1}\nu_s\le m+\nu_{i-k+1}<\sum_{s=1}^{i+2}\nu_s,\ \sum_{s=1}^{i-k+1}\nu_s\le m< \\ <\sum_{s=1}^{i-k+2}\nu_s\right)\Big/ P\left(\sum_{s=1}^{i-k+1}\nu_s\le m<\sum_{s=1}^{i-k+2}\nu_s\right)$$

By using Lemma 2.3. we can calculate the probability, which is in the denominator of the last fraction:

$$P\left(\sum_{s=1}^{i-k+1}\nu_s\le m<\sum_{s=1}^{i-k+2}\nu_s\right)=\frac{i-k+1}{m^{i-k+2}}\binom{m+1}{i-k+2}.$$

Before the calculation of the probability, which is in the numerator of the fraction, it is denoted by δ_k, then it is calculated in the following way:

$$\delta_k=\sum_{u=i-k}^{m-k+1}P\left(\sum_{s=1}^{i-k}\nu_s=u\right)\tilde{P}_u,$$

where $k=1,2,...,min(i,n-j)$ and

$$\tilde{P}_u=P\left(\sum_{s=i-k+2}^{i+1}\nu_s\le m-u<\sum_{s=i-k+2}^{i+2}\nu_s, \\ \nu_{i-k+1}\le m-u<\nu_{i-k+1}+\nu_{i-k+2}\right).$$

Obviously, in the last probability we deal with independent probabilities and with the help of Lemma 2.3. for $\tilde{P}_u$ we get the following formula:

$$\tilde{P}_u=\frac{(m-u)(m+u+1)\big((m+1)k+u\big)}{2(k+1)m^{k+3}}\binom{m-u}{k}. \quad (6)$$

By using Lemma 2.1. as a result we get the following formula for δ_k:

$$\delta_k=\frac{1}{m^{i-k}}\sum_{u=i-k}^{m-k+1}\binom{u-1}{i-k-1}\tilde{P}_u, \quad (7)$$

where $\tilde{P}_u$ is calculated by the formula (6). So, we get a formula for $\overline{P}(i,k)$ probability:

$$\overline{P}(i,j,k) = \frac{m^{i-k+2}}{(i-k+1)\binom{m+1}{i-k+2}}\delta_i. \quad (8)$$

Note that we can calculate the probability $\overline{\overline{P}}(i,k)$ in the same way as $\overline{P}(i,j,k)$[4].

5. DIFFERENTIAL EQUATIONS AND THEIR L-S TRANSFORM

In this part, the derivation of a differential equations and their transformation are presented using the Laplace-Stieltjes(L-S) transform [5].

After performing some simple transformations and calculating the limit of equation (1) for t, when h approaches 0, the following differential equation is obtained:

$$\begin{aligned}\frac{dP_{i,j}(t)}{dt} &= -(ib+jw+a)P_{i,j}(t)+\\ &+\theta_j a P_{i,j-1}(t)+\\ &+\eta_j b\sum_{k=0}^{l_1}\delta_{i,k}(i-k+1)P_{i-k+1,j+k}(t)\overline{P}(i,j,k)+\\ &+\eta_j wjP_{i,j+1}(t)+\\ &+\eta_j w\sum_{k=0}^{l_2}P_{i-k,j+k+1}(t)\overline{\overline{P}}(i,k),\end{aligned} \quad (9)$$

where $0 \le i \le m$, $0 \le j \le n$.

The function

$$p(s,i,j) = \int_0^\infty e^{-st}P_{i,j}(t)\,\mathrm{d}t \quad (10)$$

is called the Laplace-Stieltjes transform of the function $P_{i,j}(t)$. Using the formula of partial integration in the notation (10), for the Laplace-Stieltjes transform, the following relationship will be obtained:

$$p(s,i,j) = \int_0^\infty e^{-st}P_{i,j}(t)\,\mathrm{d}t = \frac{1}{s}\int_0^\infty e^{-st}\,\mathrm{d}P_{i,j}(t) \quad (11)$$

After multiplying both sides of the differential equation (9) by e^{-st}, then integrating both sides with respect to t, from 0 to ∞ and applying (10) and (11), the following system of linear equations is obtained:

$$\begin{aligned}&(s+ib+jw+a)p(s,i,j) = \theta_j ap(s,i,j-1)+\\ &+\eta_j b\sum_{k=0}^{l_1}\delta_{i,k}(i-k+1)p(s,i-k+1,j+k)\overline{P}(i,j,k)+\\ &+\eta_j wjp(s,i,j+1)+\\ &+\eta_j w\sum_{k=0}^{l_2}p(s,i-k,j+k+1)\overline{\overline{P}}(i,k),\end{aligned} \quad (12)$$

where $0 \le i \le m$, $0 \le j \le n$. Note that after applying (10) and (11), the equalities (2) and (3) will have the following form:

$$p(s,0,j) = 0,\ 0 < j \le n$$

and

$$\sum_{i=0}^{m}\sum_{j=0}^{n} sp(s,i,j)) = 1.$$

6. CONCLUSION

In this paper, we presented a multiprocessor queueing system $M|M|m|n$ with waiting time restrictions of tasks. Considering the state of the system at time $t+h$ and changes in the state of the system over the previous time h, where h is a short time, equations were obtained, and then differential equations, which give probabilistic relations between the states of the system. After using the Laplace-Stieltjes transformation for that differential equations, the system of linear equations was obtained. The obtained system of differential equations (9) and their L-S transform system (12) are solvable for given parameters. Such a model of a queuing system can play an important role in multiprocessor systems, and the results obtained can be applied to the development of various scheduling algorithms and schedulers.

REFERENCES

[1] L. Kleinrock, "Queueing Systems: *Volume 1 - Theory*", Wiley Interscience, New York, 1975.

[2] P. P. Bocharov, C. D'Apice, A. V. Pechinkin, S. Salerno, "Queueing Theory", VSP, Utrecht, Boston, 2004, pp.94-98.

[3] V. Sahakyan, Y. Shoukourian, H. Astsatryan, "About Some Queueing Models for Computational Grid Systems", *Mathematical Problems of Computer Science*, Yerevan, Armenia, 2016, vol. 46, pp.55-58.

[4] A. Vardanyan, V. Sahakyan, "The Queue Distribution in Multiprocessor Systems with the Waiting Time Restriction", *Mathematical Problems of Computer Science*, Yerevan, Armenia, 2019, vol. 51, pp.82-89.

[5] W. Arendt, C. J.K. Batty, M. Hieber, F. Neubrander, "Vector-valued Laplace Transforms and Cauchy Problems: *Second Edition*", Birkhuser, Basel, 2011, pp.55-58.