Mathematical Problems of Computer Science 54, 34-40, 2020.

UDC 519.217

The Steady State Distribution for $M|M|m|n$ Model with the Waiting Time Restriction

Vladimir G. Sahakyan and Artur P. Vardanyan

Institute for Informatics and Automation Problems
e-mail: vladimir.sahakyan@sci.am, artvardanyan@asnet.am

Abstract

The queue state in multiprocessor computing systems is an actual problem for the process of optimal scheduling of tasks. In this paper, a system of equations is obtained describing the distribution of the queue for the system in a steady state. The resulting linear system of equations is solved using conventional numerical methods and can be used in schedulers.

Keywords: 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 distribution 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].

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:

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

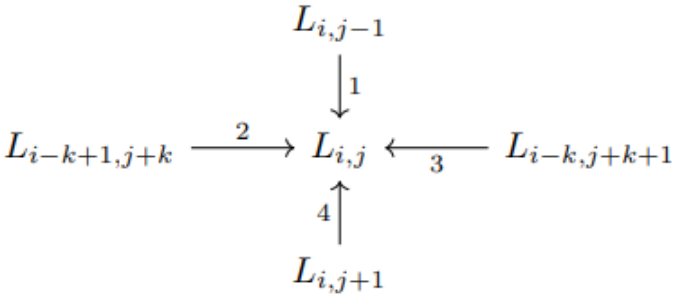
Tasks will be accepted for service in the order of their entry into the system, i.e., FIFO discipline is used (First-In-First-Out). Those tasks that arrive at the time of full occupation of the queue (there are already n tasks in the queue) receive a denial of service.

3. Basic Notations and Equations

Due to the finite number of possible states of the system, the system goes into a steady mode of operation, i.e., in a steady state. 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,
$P_{i,j}$ - the probability that the system is in the $L_{i,j}$ state:

$$P_{i,j} = P(L_{i,j}).$$

Due to finite numbers n and m, the number of possible states of the system is finite. Cases when the system can pass $L_{i,j}$ state from the other state are presented in the following scheme:



1. the state of the system was $L_{i,j-1}$ and one task arrived and joined the queue;
2. the state of the system was $L_{i-k+1,j+k}$, where $k = 1, 2, ..., min(i, n-j)$ and one task completed the service and left the system, the first k tasks from the queue were accepted to service;
3. the state of the system was $L_{i-k,j+k+1}$, where $k = 0, 1, ..., min(i-1, n-j-1)$ and 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;
4. the state of the system was $L_{i,j+1}$ and one task from the queue, not the first task, left the queue(one's waiting time ran out).

Obviously, the probability that the state of the system will be $L_{i,j}$ is the sum of the probabilities of the above cases, it follows that:

$$\begin{aligned} P_{i,j} &= \delta_1(i,j)P_{i,j-1} + \theta_j \sum_{k=0}^{l_1} \xi_{i,k}\delta_2(i,j,k)P_{i-k+1,j+k} + \\ &+ \eta_j \sum_{k=0}^{l_2} \delta_3(i,j,k)P_{i-k,j+k+1} + \eta_j\delta_4(i,j)P_{i,j+1}, \end{aligned} \tag{1}$$

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

$$l_1 = min(i, n-j),$$
$$l_2 = min(i-1, n-j-1),$$
$$\eta_j = \{\, 0 \text{ , for } j = n1, \text{for } 0 \le j < n,$$
$$\theta_j = \{\, 0 \text{ , for } j = 01, \text{for } 0 < j \le n,$$
$$\xi_{i,k} = \{\, 0 \text{ , for } i = m \text{ and } k = 01, \text{for otherwise,}$$

and $\delta_1(i,j)$, $\delta_2(i,j,k)$, $\delta_3(i,j,k)$, $\delta_4(i,j)$ are probabilities for appropriate cases:

$$\delta_1(i,j) = \frac{a}{a+ib+jw},$$

$$\delta_2(i,j,k) = \frac{(i-k+1)b}{a+(i-k+1)b+(i+k)w}\overline{P}(i,j,k),$$

where $k = 0, 1, ..., l_1$ 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),

$$\delta_3(i,j,k) = \frac{w}{a+(i-k)b+(i+k+1)w}\overline{\overline{P}}(i,j,k),$$

where $k = 0, 1, ..., l_2$ and $\overline{\overline{P}}(i,j,k) = 0$ if $i = 0$, but if $0 < i \leq m$, then $\overline{\overline{P}}(i,j,k)$ is the following conditional probability:

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

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),

$$\delta_4(i,j) = \frac{w}{a + ib + (j+1)w}.$$

Calculation formulas for $\overline{P}(i,j,k)$, $\overline{\overline{P}}(i,j,k)$ and some other useful probabilities will be presented in the next section of this article.
Note if $i = 0$ for all $0 < j \leq n$

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

and

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

4. Formulas for Some Useful Probabilities

This section presents the calculation of the values of some probabilistic characteristics. We will use two lemmas proved in the previous article [4].
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 1: *The probability that k processors will be occupied by i tasks, can be obtained in the following way:*

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

Lemma 2: *The probability that i tasks will occupy no more than k processors, can be obtained in the following way:*

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

Lemma 3:

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

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

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

$$\begin{aligned}\overline{P}(i,j,k) &= P\left(\sum_{s=1}^{i+1}\nu_s \le m+\nu_{i-k+1} < \sum_{s=1}^{i+2}\nu_s \Big/ \sum_{s=1}^{i-k+1}\nu_s \le m < \sum_{s=1}^{i-k+2}\nu_s\right)\\ &= \frac{P\left(\sum\limits_{s=1}^{i+1}\nu_s \le m+\nu_{i-k+1} < \sum\limits_{s=1}^{i+2}\nu_s, \sum\limits_{s=1}^{i-k+1}\nu_s \le m < \sum\limits_{s=1}^{i-k+2}\nu_s\right)}{P\left(\sum\limits_{s=1}^{i-k+1}\nu_s \le m < \sum\limits_{s=1}^{i-k+2}\nu_s\right)}.\end{aligned}$$

By using Lemma 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 Q_k, then it is calculated in the following way:

$$Q_k = \sum_{u=i-k}^{m-k} P\left(\sum_{s=1}^{i-k}\nu_s = u\right) q_u,$$

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

$$q_u = P\left(\sum_{s=i-k+2}^{i+1}\nu_s \le m-u < \sum_{s=i-k+2}^{i+2}\nu_s, \nu_{i-k+1} \le m-u < \nu_{i-k+1}+\nu_{i-k+2}\right). \qquad (4)$$

Obviously, in the last probability we deal with independent probabilities and with the help of Lemma 3 for q_u we get the following formula:

$$q_u = \frac{(m-u)(m+u+1)((m+1)k+u)}{2(k+1)m^{k+3}}\binom{m-u}{k}.$$

By using Lemma 1 as a result we get the following formula for Q_k:

$$Q_k = \frac{1}{m^{i-k}}\sum_{u=i-k}^{m-k+1}\binom{u-1}{i-k-1}q_u,$$

where q_u is calculated by the formula (4). So, we get a formula for $\overline{P}(i,j,k)$ probability:

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

Note that we can calculate the probability $\overline{\overline{P}}(i,j,k)$ in the same way as $\overline{P}(i,j,k)$ and we get a formula for $\overline{\overline{P}}(i,j,k)$ probability:

$$\overline{\overline{P}}(i,j,k) = \frac{\sum_{u=i-k}^{m-k} u((m+1)k+u)\binom{u-1}{i-k}\binom{m-u}{k}}{(k+1)(i-k)m^{k+1}\binom{m+1}{i-k+1}}.$$

5. 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 steady mode, equations were obtained: (1), (2) and (3) formulas together, which give probabilistic relations between the states of the system. The resulting system of equations allows us to calculate the probabilities of being in each state of the system, which, in turn, will allow us to find the virtual waiting time for a task. 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] P. P. Bocharov, C. D'Apice, A. V. Pechinkin and S. Salerno, *Queueing Theory*, VSP, Utrecht, Boston, 2004.

[2] A. Vardanyan and V. Sahakyan, "The queue distribution in multiprocessor systems with the waiting time restriction", *Mathematical Problems of Computer Science*, Yerevan, Armenia, vol. 51, pp. 82-89, 2019.

[3] V. Sahakyan and A. Vardanyan, "The state probabilities of the system $M|M|m|n$ with the waiting time restriction", *Proceedings of International Conference Computer Science and Information Technologies*, Yerevan, Armenia, pp. 181-184, 2019.

[4] V. Sahakyan and A. Vardanyan, "The queue state for multiprocessor system with waiting time restriction", *CSIT 2019 Revised Selected Papers*, Publisher IEEE, DOI: 10.1109/CSITechnol.2019.8895093, pp. 116-119, 2019.

Submitted 16.06.2020, accepted 08.10.2020.

Կայուն վիճակի բաշխումը սպասման ժամանակի սահմանափակմամբ $M|M|m|n$ մոդելի համար

Վլադիմիր Գ. Սահակյան և Արթուր Պ. Վարդանյան

ՀՀ ԳԱԱ Ինֆորմատիկայի և ավտոմատացման պրոբլեմների ինստիտուտ
e-mail: vladimir.sahakyan@sci.am, artvardanyan@asnet.am

Ամփոփում

Բազմապրոցեսորային զանգվածային սպասարկման համակարգերում հերթի վիճակը էական նշանակության խնդիր էառաջադրանքների օպտիմալ պլանավորման գործընթացում: Այս հոդվածում ստացվում է հավասարումների համակարգ, որը նկարագրում է համակարգի հերթի բաշխումը կայուն վիճակում: Արդյունքում ստացվող գծային

հավասարումների համակարգը լուծվում է սովորական թվային մեթոդների միջոցով և կարող է օգտագործվել պլանավորման համակարգերում:

Բանալի բառեր՝ զանգվածային սպասարկման տեսություն, բազմապրոցեսորային զանգվածային սպասարկման համակարգ, սպասման ժամանակի սահմանափակում:

Распределение стационарного состояния для модели $M|M|m|n$ с ограничением времени ожидания

Владимир Г. Саакян и Артур П. Варданян

Институт проблем информатики и автоматизации НАН РА
e-mail: vladimir.sahakyan@sci.am, artvardanyan@asnet.am

Аннотация

Состояние очереди в многопроцессорной вычислительной системе является актуальной задачей для процесса оптимального планирования выполнения заданий. В данной работе получена система уравнений, описывающая распределение очереди для системы в установившемся состоянии. Полученная линейная система уравнений решается с помощью обычных численных методов и может быть использована в планировщиках.

Ключевые слова: теория массового обслуживания, многопроцессорная система массового обслуживания, ограничение времени ожидания.