# About the Busy Period of a Multi-server Queueing Model with Simultaneous Service

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

Artur Vardanyan
Institute for Informatics and Automation Problems of NAS RA
Yerevan, Armenia
e-mail: artur.vardanyan@iiap.sci.am

Husik Gevorgyan
Institute for Informatics and Automation Problems of NAS RA
Yerevan, Armenia
e-mail: husikgz@yahoo.com

***Abstract*—This paper investigates the busy period of a multi-server queueing system in which tasks require the simultaneous allocation of a random number of servers. The model considers $m$, ($m \geq 1$) homogeneous servers, Poisson arrivals, and exponentially distributed service times. Each task randomly requests between one and $m$ servers for its execution, and service begins only when the required number of servers is available. By formulating the system dynamics using a recursive relation that captures the evolution of the busy period based on task completion and service initiation events. A general expression for the expected busy period is derived. The results contribute to a deeper understanding of multi-server systems with simultaneous service and offer valuable insights for performance evaluation in modern computing infrastructures.**

***Keywords*—Queueing Theory, multi-server queueing system, multiprocessor queueing system, busy period.**

## I. Introduction

Queueing models play a critical role in the performance analysis of complex service systems, particularly in computing, telecommunications, and logistics. Among the various characteristics of such systems, the *busy period*—the time interval during which the system remains continuously occupied—serves as a key performance metric. It offers insight into system responsiveness, resource utilization, and potential bottlenecks. Precise analysis of busy periods enables the design of efficient scheduling and load balancing mechanisms, especially in systems with shared and limited resources.

Traditional queueing theory has extensively studied busy periods in single-server and simple multi-server models with independent service requirements. Classical results are available for M/M/1 and M/M/c systems, where each task is served independently by a single server. Analytical formulas for the distribution and expected value of the busy period are well-known under these assumptions, with applications in delay prediction, capacity planning, and reliability assessment [1, 2, 3].

However, these models fail to reflect the reality of modern computational systems, where tasks often require the concurrent use of multiple processing units. In particular, many-core processors, distributed computing frameworks, and cloud infrastructures allocate multiple servers or cores to service a single task simultaneously. This has led to the development of multi-server queueing models with *simultaneous service*, where each task requires a random number of servers to be available before service can begin [4, 5].

Despite their relevance, such models have received relatively little analytical attention, mainly due to the complexity introduced by simultaneous resource requirements and the resulting high-dimensional state space. Recent studies have attempted to address this gap through stochastic modeling and numerical methods [6, 7, 8, 9]. However, the behavior of busy periods in these models remains largely unexplored. Understanding the busy period in this context is crucial for accurately estimating system congestion and for designing admission control policies in multi-processor and multi-threaded environments.

This paper focuses on analyzing the busy period of a multi-server queueing system in which each task requires the simultaneous allocation of a random number of servers. Tasks arrive according to a Poisson process, and service times are exponentially distributed. We model the dynamics of such a system using steady-state probabilities and derive expressions related to the probability and expected duration of the busy period.

## II. Model Description

A multi-server queueing system is considered, consisting of $m$ homogeneous servers, where $m \geq 1$. Tasks (also known as customers or jobs) arrive according to a Poisson process with rate $a$.

Each arriving task is characterized by a pair of parameters $(\nu, \beta)$, where:

- $\nu \in \{1, 2, \ldots, m\}$ denotes the number of servers required to process the task simultaneously;
- $\beta > 0$ denotes the service time of the task.

If the required number of servers $\nu$ is not immediately available upon arrival, the task enters a single first-come, first-served (FCFS) queue, where it waits until $\nu$ servers become available. The system assumes an infinite buffer capacity, allowing an unlimited number of tasks to wait in the queue.

Once a task begins service, the requested $\nu$ servers are allocated simultaneously and remain occupied throughout the service duration $\beta$. Upon completion, all $\nu$ servers are released simultaneously. The number of servers required by a task, $\nu$, follows a discrete uniform distribution over the set $\{1, 2, \ldots, m\}$, each assigned an equal probability of $1/m$. The

https://doi.org/10.51408/csit2025_68

service time $\beta$ is exponentially distributed with rate $b > 0$. Figure 1 shows the schema of the described multi-server queueing model. Upon arrival, a task is either accepted for service or, if it cannot be immediately accepted for service, it is queued and serviced in FCFS order. Once the task service begins, it continues uninterrupted until completion.

Thus, these assumptions provide a framework for analyzing the dynamics of task arrival and service completion within the multi-server queueing model in which each task requires a random number of servers simultaneously, and a random service time at all occupied servers.

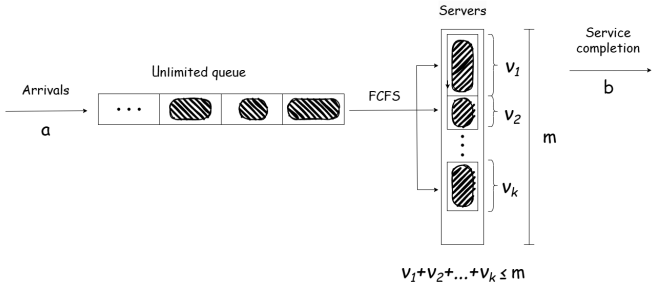


Fig. 1. The schema of the queueing model

## III. Busy Period Analysis

This section presents equations for estimating the expected busy period, i.e., the time until the system becomes idle. By examining the service process and the established notation, the aim is to capture the completion dynamics of tasks in the service and queue.

Let $\pi_{i,j}$ denote the duration of the busy period, given that $i$ tasks are being serviced and $j$ tasks are in the queue ($0 \le i \le m$ and $j \ge 0$).

When considering the service process, the busy period will be expressed in terms of a recursive relation: that is, the busy period when $i$ tasks are being serviced and $j$ tasks are in the queue contains the value of the service time of the task that will complete its execution first among the $i$ tasks, and the busy period of the system after the completion of this task. When the first of the $i$ active tasks completes its execution, the service process transitions in the following manner:

1) There may not be a sufficient number of idle servers to initiate service for any waiting task in the queue.
2) More generally, there may not be enough available servers for $k$ tasks in the queue to start execution simultaneously.

Taking these conditions into account, the busy period can be described by the following recursive relation:

$$\pi_{i,j} = \tilde{\beta}_i + \chi_{i,j}^{(0)}\pi_{i-1,j} + \chi_{i,j}^{(1)}\pi_{i,j-1} + \\ + \chi_{i,j}^{(2)}\pi_{i+1,j-2} + \cdots + \chi_{i,j}^{(k)}\pi_{i+k-1,j-k}, \quad (1)$$

where the parameters satisfy the following conditions: $0 \le i \le m$, $j \ge 0$, $0 \le k \le j$, and $1 \le i + k - 1 \le m$, the variable $\tilde{\beta}_i$ denotes the minimum service time among the $i$ tasks currently in service and the indicator function $\chi_{i,j}^{(k)}$ represents the event in which, immediately after the first of the $i$ active tasks completes execution, exactly $k$ tasks from the queue begin service simultaneously. To ensure a compact and well-defined formulation, we introduce a fictitious task in the queue with a resource requirement $\nu_{i+k+1} = m + 1$, effectively preventing any additional tasks from entering the service when there are no available resources.

Assuming the independence of the random variable $\pi_{i,j}$ from the system's history, and taking the mathematical expectation in equation (1), the expected duration of the busy period can be obtained.

## IV. Conclusion

This study presents a stochastic analysis of the busy period in a multi-server queueing system where each task requires the simultaneous allocation of a random number of servers. By modeling task arrivals as a Poisson process and assuming exponential service times, a recursive relation is derived to characterize the busy period as a function of the system state. The formulation accounts for the dynamics of task execution and the possibility of initiating service for multiple queued tasks after a server release.

The proposed model extends classical queueing theory by addressing the complexities introduced by simultaneous service requirements — a feature common in modern distributed and high-performance computing systems. The derived recursive equation and expected busy period provide a foundation for further analytical or numerical studies on system throughput, resource utilization, and delay.

## References


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

[2] H. Takagi, *Queueing Analysis, Volume 1: Vacation and Priority Systems*, North-Holland, 1991.

[3] D. Gross, C. M. Harris, *Fundamentals of Queueing Theory*, 4th Edition, Wiley, 2008.

[4] M. Tirmazi, A. Barker, N. Deng, M. E. Haque, Z. G. Qin, S. Hand, M. Harchol-Balter, J. Wilkes, "Borg: the next generation", *Proceedings of the fifteenth European conference on computer systems (EuroSys '20)*, pp. 1-14, 2020. DOI: https://doi.org/10.1145/3342195.338751.

[5] NVIDIA Corporation, "Bright Cluster Manager Documentation", Available online: https://docs.nvidia.com/bright-cluster-manager/, Accessed: February 26, 2025.

[6] V. Sahakyan, A. Vardanyan, "The Queue State for Multiprocessor System with Waiting Time Restriction", *IEEE Xplore, Computer Science and Information Technologies 2019, Conference Proceeding*, Yerevan, pp. 116–119, 2019. DOI: https://doi.org/10.1109/CSITechnol.2019.8895093.

[7] V. Sahakyan, A. Vardanyan, "About the possibility of executing tasks with a waiting time restriction in a multiprocessor system", *AIP Conference Proceedings*, 2757.1, pp. 030003, 2023. DOI: https://doi.org/10.1063/5.0135784.

[8] V. Sahakyan, A. Vardanyan, "A Computational Approach for Evaluating Steady-State Probabilities and Virtual Waiting Time of a Multiprocessor Queuing System", *Programming and Computer Software*, Volume 49, pp. S16–S23, 2023. DOI: https://doi.org/10.1134/S0361768823090098.

[9] A. Vardanyan, "Advanced Queueing Model of a Multiprocessor Computing System", *Mathematical Problems of Computer Science*, Volume 62, pp. 43-51, 2024. DOI: http://doi.org/10.51408/1963-0119.