Volume: 13 Issue: 05 Pages: 5113-5118(2022) ISSN: 0975-0290

A Study on Queuing Systems and its Deterministic Measures

CS Reddy
Department of Mathematics, Cambridge Institute of Technology – NC, Bangalore, India-561203
Email: abcdef@gmail.com
Krishna Anand S
Department of Artificial Intelligence, Anurag University, Hyderabad-500088
Email:skanand86@gmail.com

ABSTRACT

In this paper we show that queueing theory can accurately model the flow of in-patient in hospital. In which arrival rate, service rate and number of parallel servers are all considered as fuzzy numbers. Further Robust Ranking technique is used to find the expected mean queue length and waiting time in queue. Further numerical illustration is also given to justify the validity of the model. In this model capacity of the system is infinite.

Keywords -**Queuing System, Kendell Notation, system performance measures, arrival, and departure rates.**

Date of Submission: Feb 20, 2022 Date of Acceptance: Apr 26, 2022

1. INTRODUCTION

Queuing models are used to predict the performance of service systems when there is uncertainty in arrival and service times. In this note we will explain queuing terminology and discuss some simple queuing models. This note uses the terminology and conventions of Macros, a spreadsheet add-in to analyze queuing systems.

The simplest possible (single stage) queuing systems have the following components: customers, servers, and a waiting area (queue), see figure 1. An arriving customer is placed in the queue until a server is available. To model such a system, we need to specify:

- the characteristics of the arrival process.
- the characteristics of the service process; and
- how (in what order) waiting customers are dispatched to available servers.

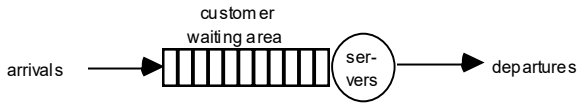

Fig 1. Single stage queuing system.

In this note we will always assume that customers are served in the order in which they arrive in the system (First-Come-First-Served or FCFS). For the characteristics of the arrival and service processes we will make various assumptions, and in general, queuing models are classified according to the specific assumptions made.

In section 2 below we will discuss probability basics, and in section 4 we will go over various performance measures for queuing systems. This material is somewhat technical in nature, but it is necessary for a precise understanding of what queuing models can and cannot do.

In section 3 some basic queuing models will be discussed in detail:

- M/M/s – a multi-server model with Poisson arrivals and Exponential service times.
- G/G/s – a multi-server model with General arrival process and General distribution of service times.
- M/M/s/N – a multi-server model with Poisson arrivals, Exponential service times and a finite facility size so that no more than N customers can be present at any time.
- M/M/s Impatient – a multi-server model with Poisson arrivals, Exponential service times and Impatient customers prone to balking or reneging.

Finally, section 4 gives an overview of basic queuing system parameters and performance measures.

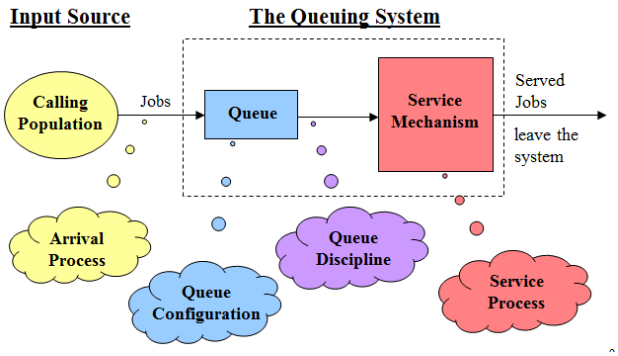

Fig 2.Basic Components of the Queuing system

2. The Exponential Distribution

Clearly, it is impossible to always predict in advance precisely how much service time a customer requires. Hence the service time of a customer is assumed to follow some probability distribution. The exponential distribution is the most frequently used distribution in queuing models. Quite often, we assume that the service time of a customer is independent of the service time of all other customers and follows an exponential distribution. In addition, as we will see in the next subsection, the time between two

Volume: 13 Issue: 05 Pages: 5113-5118(2022) ISSN: 0975-0290

consecutive arrivals to a queuing system is also frequently assumed to follow an exponential distribution.

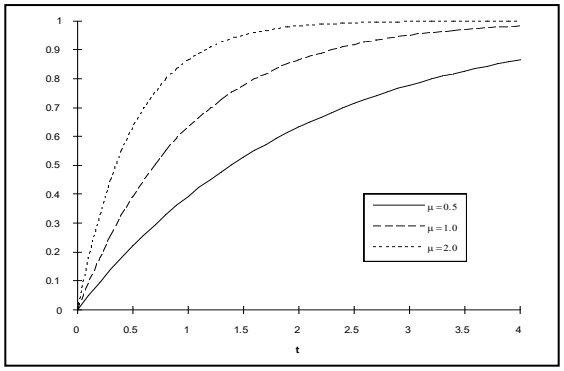

Fig 3. Exponential Cumulative Distribution Functions for arrival rate 0.5, 1.0, 2.0.

Let S be (the random variable describing) the service time of an arbitrary customer. Then S is said to follow an exponential distribution if for all $t \geq 0$:

$$\Pr\{S \leq t\} = 1 - e^{-\mu t}$$

The parameter μ is called the service rate and gives the average number of customers that a single server can process in the long run if she or he never runs out of customers to serve. This distribution is plotted for several values of μ, σin figure 3.

Some useful and/or interesting facts about the exponential distribution are:

- mean service time = E[S] = 1/μ,
- the standard deviation of S = ∑[S] ,
- the coefficient of variation of S =cv[S] =1.

In addition, the exponential distribution is what is called memoryless. This means that the distribution of the remaining service time of a customerwho is currently in service follows the same exponential distribution as the service time of a different customer who starts service now. Formally, for all t, $s \geq 0$ we have

$$\Pr\{S \leq t + s | S > s\} = 1 - e^{-\mu t}$$

Note that the right-hand side of this equation does not depend on s at all.

It is this property that makes the exponential distribution easy to work with in queuing models: it is not necessary to keep track of how long a customer has already been in service since his remaining service time always follows the same distribution. In the commonly used shorthand notation for queuing models, the exponential distribution is represented by an "M" for Memoryless.

3 The Poisson Process

Arrivals to a service system usually occur in a random, unpredictable fashion. Even if a good forecast of the total number of arrivals is available, there is often still considerable uncertainty about the precise timing of arrivals. Consider, e.g., the checkout counter at a large hotel. Management has a good estimate of the number of guests that will check out that day, but there is considerable uncertainty about the number that will check out in the 7-7:30am time interval, and how the checkouts will bunch together during this interval. In a queuing model we therefore need some assumptions about the arrival process. A common assumption in many queuing models is that customers arrive according to a so-called Poisson Process. Formally, this process can be defined as follows.

Let Ti be the arrival time of the i^{th} customer. Arrivals are said to follow a Poisson Process if the successive inter-arrival times $T_1–T_0$, $T_2–T_1$, …, $T_i–T_{i-1}$, …, are independent and all follow the same exponential distribution, i.e., for all $t \geq 0$ and I =1, 2, … we have

$$\Pr\{T_i - T_{i-1} \leq t\} = 1 - e^{-\lambda t}$$

Here the parameter ⊿ is called the arrival rate, and it gives the average or expected number of arrivals per time unit.

An alternative way of looking at the arrival process is to count arrivals. Let N(t) = the number of arrivals up to time t. For every t, N(t) is a random variable that simply counts the number of arrivals. Then the (random!) number of arrivals between time s and time t is given by N(t) – N(s). Of course, $N(t) \geq N(s)$ whenever $t \geq s$, so that N(t) and N(s) are not independent. A typical realization of a Poisson process is depicted in figure 3.

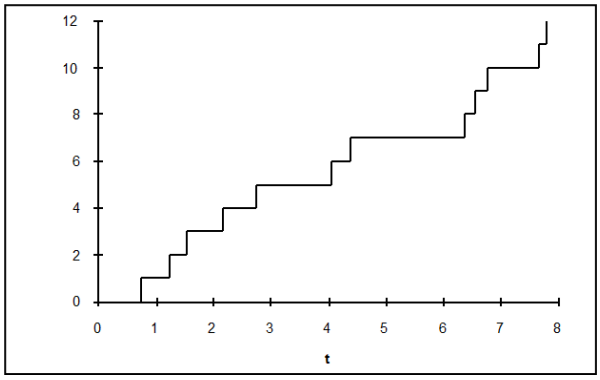

Fig 4. A realization of a Poisson Process.

The name Poisson Process can be explained with the following property: if the arrivals follow a Poisson process, then one can show that for every $s \geq t$, N(t) – N(s) (= the number of arrivals between time s and time t) has a Poisson distribution with mean ⊿(t–s), i.e., for k = 0, 1, …

$$\Pr\{N(t) - N(s) = k\} = \frac{(\lambda(t-s))^k}{k!} e^{\lambda(t-s)}$$

In addition, the numbers of arrivals in any collection of non-overlapping time intervals are independent.

In this section we will define some frequently used parameters and performance measures for queuing systems. Throughout, we will make the assumption that the system is in "steady state", i.e., it has operated for a long time with the same values for all the parameters. Since customers arrive in a random fashion and service times are random, a customer's experience in the system (such as the waiting time experienced by the customer) are also random. Basically, the steady state assumption makes sure that as we take many observations of (say) customer waiting time, the frequency distribution converges to a

Volume: 13 Issue: 05 Pages: 5113-5118(2022) ISSN: 0975-0290

fixed limiting distribution. It is this limiting distribution that the queuing models calculate. We can then interpret a performance measure in two ways: as an expected value for an arbitrarily chosen single customer, or as the average of observations made on a large number of customers (say all those arriving during a given long time interval). It is however wise to realize that each customer will have a different experience in the system.

In the absence of "steady state", the frequency distribution of (say) customer waiting time will not have a fixed limit. This means that the expected waiting time of a customer arriving at 10am may be quite different from the expected waiting time of a customer arriving at 11am. The queuing models that deal with this complication are much more difficult to analyze and are beyond the scope of this note.

4 Warning

Time units play a role in measuring many of the performance measures and parameters. It is important that when you use a queuing model the same time unit is used throughout. If you specify an arrival rate in terms of customers/hr and the mean service time in minutes you will get at best meaningless answers and most likely a lot of trouble!

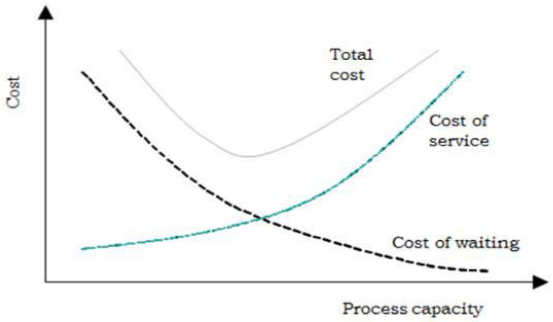

Fig 5. A Cost/Capacity Tradeoff Model

4.1 System Parameters

A summary of parameters is given in table 1.

Parameter	Description	Models
Arrival Rate	Average number of customers arriving per time unit	All models
Service Rate	Average number of customers that a single server processes per time unit if he or she is never idle	All models
Number of Servers	Number of parallel, identical servers in the system	All models
Number of Positions	Total number of customer positions (waiting plus in service) in the system	M/M/s/N
(Mean) Patience Time	Average amount of time that a customer is willing to wait before service starts	M/M/s Impatient
Coefficient of Variation of the Inter-arrival Times	Measures the variability of the time between consecutive arrivals. $\text{cv}(A) = \frac{\text{std.dev.}(A)}{E[A]}$	G/G/s
Coefficient of Variation of the Service Times	Measures the variability of the service time distribution. $\text{cv}(S) = \frac{\text{std.dev.}(S)}{E[S]}$	G/G/s

4.2 Performance Measures

4.2.1 Load Factor

The system load factor is a measure of the relative load on the system. Formally, it is defined as

$$\text{Load Factor} = \frac{\text{Amount of work arriving at the system per time unit}}{\text{Amount of work that can be processed by the system per time unit}}$$

4.2.2 Fraction Not Served

In some models, some unlucky arrivals never receive service, in the M/M/s/N because of blocking, in the M/M/s Impatient model because of balking or reneging. The probability of not receiving service gives the probability that an arbitrarily chosen arrival will be one of the unfortunate ones. Alternatively, it gives the fraction of all arrivals that never receive service.

4.2.3 Thruput

The thruput is the average number of customers that complete service per time unit. This number is always less than or equal to the lesser of the average number of arrivals per time unit and the total processing capacity of the system in a time unit.

4.2.4 Server Utilization

This is the fraction of time that the average server spends serving customers. If customers are distributed randomly or evenly to the servers, it is also equal to the probability that a server is busy at an arbitrary point in time.

4.2.5 Average Number in System

This gives the time average number of customers in the system (counting both customers waiting for service and customers being served). In the M/M/s/N model, blocked

Volume: 13 Issue: 05 Pages: 5113-5118(2022) ISSN: 0975-0290

customers are not included in this measure. In the M/M/s Impatient model with balking, balking customers are not included in this measure. In the M/M/s Impatient model with reneging, reneging customers are included.

4.2.6 Average Number in Queue

This gives the average number of customers waiting for service (hence customers being served are not included). In the M/M/s/N model, blocked customers are not included in this measure. In the M/M/s Impatient model with balking, balking customers are not included in this measure. In the M/M/s Impatient model with reneging, reneging customers are included.

4.2.7 Average Time in System

This gives the expected amount of time that an arbitrary customer (who ultimately gets served) spends in the system. Alternatively, it gives the average amount of time that those customers who ultimately get served spend in the system (waiting for service and being served). In the M/M/s/N model, blocked customers are assumed not to spend time in the system. In the M/M/s Impatient model with balking, balking customers are assumed not to spend any time in the system. In the M/M/s Impatient model with reneging, reneging customers are assumed to spend their patience time in the queue, and no time in service.

4.2.8 Average Wait in Queue

This gives the expected amount of time that an arbitrary customer spends in the queue before service begins given that the customer is ultimately served. Alternatively, it gives the average amount of time that customers that are ultimately served spend in the queue. In the M/M/s/N model, blocked customers are assumed not to spend time in the system. In the M/M/s Impatient model with balking, balking customers are assumed not to spend any time waiting in the queue. In the M/M/s Impatient model with reneging, reneging customers are assumed to spend their patience time in the queue.

4.2.9 Distribution of the Wait in Queue

The average wait in queue tells only a part of the story. Managers of queuing systems often need to worry about the extremes. In particular, measures of the form Pr{Wait in Queue at t} are of interest. Which value(s) of t are considered important depends on the situation, of course? For the M/M/s/N model and the M/M/s Impatient model with balking, waiting time probabilities are given for served customers only. For the M/M/s Impatient model with reneging, waiting time probabilities are given for all customers.

5 Mathematical Modelling

Consider a queuing system that can be modeled by a discrete time, two-dimensional Markov process on semi-infinite or finite lattice strip. The process has a Markovian property and the state of system at observation time t can be described by two integer random variables I(t) and J(t). The former one is bounded and referred to as a phase; the latter one may be either unbounded (infinite case) or bounded (finite case) and is referred to as a level of the system. The Markov process is denoted by Z = {I(t), J(t); t ≥ 0} and its state space is ([0, 1,2, . . . , N]×[0, 1,2, . . .]) in the infinite case and ([0, 1,2, . . . , N]x[0, 1,2, . . . L]) in finite case, respectively. If the possible jumps of system's level in transition are only 0, -1 or 1, the corresponding process is known as Quasi Birth-Death (QBD) process.

- Aj − purely lateral transitions −from state (i, j) to state (k, j), (0 ≤ i, k ≤N; i ≠ k ; j = 0,1,2,…..);
- Bj,s − bounded s-step upward transitions − from state (i, j) to state (k, j + s), (0 ≤ i, k ≤ N ; 1 ≤ s ≤ z1; z1 ≥ 1 ; j = 0,1,2,…..);
- Cj,s − bounded s-step downward transitions − from state (i, j) to state (k, j - s), (0 ≤ i, k ≤ N ; 1 ≤ s ≤ z2; z2 ≥ 1 ; j = 1,2,…..);

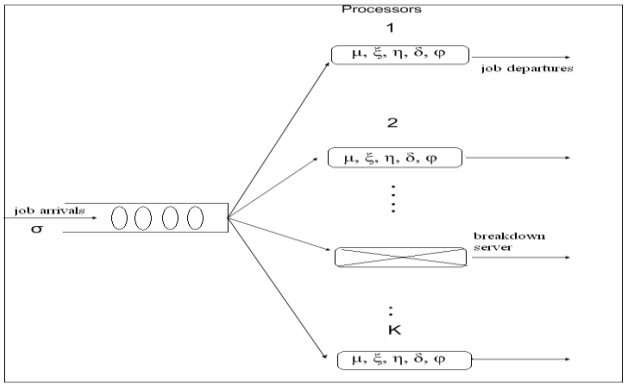

Fig 6. Multiprocessor System with breakdowns and repairs remodeling and rebooting delays

In the homogeneous multiprocessor system with K processors, it is assumed that there is a single repair facility with repair rate. When a processor fails the fault is covered with probability c and is not covered with probability. Subsequent to a covered fault, the system comes up in a degraded mode after a brief remodeling delay, while after an uncovered fault a longer reboot action is required to bring the system up at a degraded mode. Here, degraded mode indicates a state with one less operative processor than the previous state. Remodeling and restarting times are exponentially distributed with meanrespectively.

It is convenient to define row vectors of probabilities corresponding to state with j jobs in the system:

$$v_j=(p_{0,j}, p_{1,j},.....,p_{n,j}) \; ; \quad j=0,1,2,...$$

Then the balance equations for the equilibrium probabilities can be written as

$$V_j[D_j^A+\sum_{s=1}^{z_1} D_{j,s}^B+\sum_{s=1}^{z_2} D_{j,s}^C]=\sum_{s=1}^{z_1} V_{j-s}B_{j-s,s}+V_jA_j+\sum_{s=1}^{z_2} V_{j+s}C_{j+s,s}; \quad j=0,1,2,....,M-1$$

When j is greater than the threshold M, those equations become

$$V_j[D^A+\sum_{s=1}^{z_1} D_s^B+\sum_{s=1}^{z_2} D_s^C]=\sum_{s=1}^{z_1} V_{j-s}B_s+V_jA+\sum_{s=1}^{z_2} V_{j+s}C_s; \quad j\geq M$$

Volume: 13 Issue: 05 Pages: 5113-5118(2022) ISSN: 0975-0290

Figure 4 shows the relationship between the mean queue length and the mean arrival rate ⌟, for different number of servers and is observed that performances increase with proportion to number of servers. Here c is considered as zero. Figure 5 shows the mean queue length as a function of c. It is clearly evident that an increase in c results a decrease in the mean queue length because remodeling delays are shorter than restarting delays.

Figure 6 shows that number of jobs in the queue decreases as number of servers increases with arrival rate 0.66*K and queue length increases as remodeling delay increases. Here again c=0 is considered. Figure 6 shows mean queue length as a function of c, with K = 5. This shows that as c increases mean queue length increases along with arrival rate increases. Whereas figure 7 shows that, for small service values mean queue length decreases as c increases and increases as service rate increases.

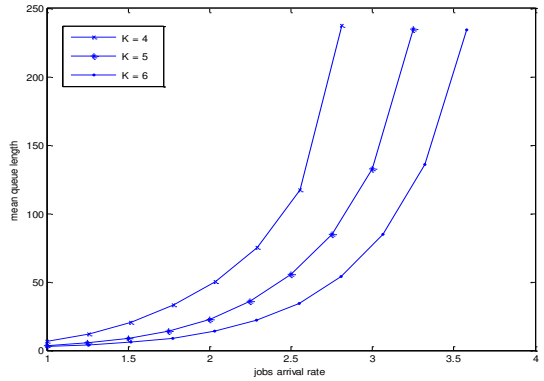

Fig 7. MQL versus me arrival rate

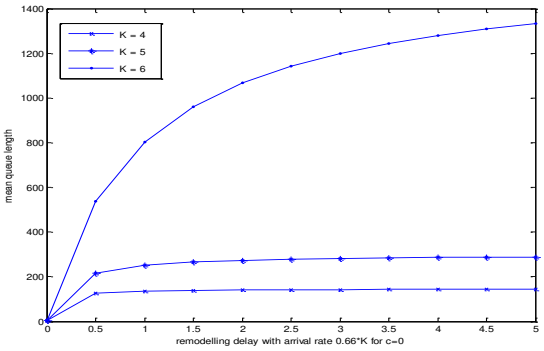

Fig 8. MQL as a function of c and rebooting service for homogeneous multiprocessor systems.

Figure 8 shows mean queue length decreases for constant arrival rate of jobs with respect to number of server increases. It is observed that as c increases, number of jobs in the queue decreases.

REFERENCES

[1] Elliriki, M., Reddy, C. S., Anand, K., & Saritha, S. (2021). Multi server queuing system with crashes and alternative repair strategies. Communications in Statistics-Theory and Methods, 1-13.

[2] Sundararaj, V. (2019). Optimal task assignment in mobile cloud computing by queue based ant-bee algorithm. Wireless Personal Communications, 104(1), 173-197.

[3] Kumar, M. S., Mamatha, E., Reddy, C. S., Mukesh, V., & Reddy, R. D. (2017). Data hiding with dual based reversible image using sudoku technique. In 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 2166-2172). IEEE.

[4] Mamatha, E., Anand, S. K., Devika, B., Prasad, S. T., & Reddy, C. S. (2021). Performance Analysis of Data Packets Service in Queuing Networks System. In 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-4). IEEE.

[5] Kavitha, V., Gupta, M. K., Capdevielle, V., & Haddad, M. (2019). Speed based optimal power control in small cell networks. Computer Communications, 142, 48-61.

[6] Mamatha, E., Reddy, C. S., & Prasad, K. R. (2016). Antialiased Digital Pixel Plotting for Raster Scan Lines Using Area Evaluation. In Emerging Research in Computing, Information, Communication and Applications (pp. 461-468). Springer, Singapore.

[7] Andalib, M. A., Ghaffarzadegan, N., & Larson, R. C. (2018). The postdoc queue: A labour force in waiting. Systems Research and Behavioral Science, 35(6), 675-686.

[8] MMamatha, E., Reddy, C. S., & Prasad, R. (2012). Mathematical Modeling of Markovian Queuing Network with Repairs, Breakdown and fixed Buffer. i-Manager's Journal on Software Engineering, 6(3), 21.

[9] Mamatha, E., Saritha, S., & Reddy, C. S. (2016). Stochastic scheduling algorithm for distributed cloud networks using heuristic approach. International Journal of Advanced Networking and Applications, 8(1), 3009.

[10] Rathod, D., & Chowdhary, G. (2019). Scalability of M/M/c queue based cloud-fog distributed Internet of Things middleware. International Journal of Advanced Networking and Applications, 11(1), 4162-4170.

[11] Mamatha, E., Saritha, S., Reddy, C. S., &Rajadurai, P. (2020). Mathematical modelling and performance analysis of single server queuing system-eigenspectrum. International Journal of Mathematics in Operational Research, 16(4), 455-468.

[12] Mamatha, E., Sasritha, S., & Reddy, C. S. (2017). Expert system and heuristics algorithm for cloud resource scheduling. Romanian Statistical Review, 65(1), 3-18.

[13] Saritha, S., et al. "A model for overflow queuing network with two-station heterogeneous system." International Journal of Process Management and Benchmarking 12.2 (2022): 147-158.

[14] 14. Hamdi, M. M., Rashid, S. A., Ismail, M., Altahrawi, M. A., Mansor, M. F., &AbuFoul, M. K. (2018, November). Performance evaluation of active queue management algorithms in large network. In 2018 IEEE 4th International Symposium on

Volume: 13 Issue: 05 Pages: 5113-5118(2022) ISSN: 0975-0290

Telecommunication Technologies (ISTT) (pp. 1-6). IEEE.

[15] Reddy, C. C. S., Prasad, K. R., Mamatha, E., & Reddy, B. S. (2009). Performance Evaluation of Heterogeneous Parallel Processor System with Alternative Repair Strategies. i-Manager's Journal on Software Engineering, 4(1), 18.

[16] Elliriki, M., Reddy, C. C. S., & Anand, K. (2019). An efficient line clipping algorithm in 2D space. Int. Arab J. Inf. Technol., 16(5), 798-807.

[17] Qureshi, I. (2014). Cpu scheduling algorithms: A survey. International Journal of Advanced Networking and Applications, 5(4), 1968.

[18] Do, H. T., Shunko, M., Lucas, M. T., & Novak, D. C. (2018). Impact of behavioral factors on performance of multi-server queueing systems. Production and Operations Management, 27(8), 1553-1573.

[19] Reddy, C. S. (2020). Multiprocessor Stochastic Model for Elastic Traffic with Different Service Capacity. International Journal of Advanced Networking and Applications, 12(3), 4601-4605.

[20] Jain, S., & Jain, S. (2016). Probability-Based Analysis to Determine the Performance of Multilevel Feedback Queue Scheduling. International Journal of Advanced Networking and Applications, 8(3), 3044.

[21] Reddy, C. S., Janani, B., Narayanan, S., & Mamatha, E. (2016). Obtaining Description for Simple Images using Surface Realization Techniques and Natural Language Processing. Indian Journal of Science and Technology, 9(22), 1-7.

[22] Bouchentouf, A. A., Cherfaoui, M., &Boualem, M. (2021). Analysis and performance evaluation of Markovian feedback multi-server queueing model with vacation and impatience. American Journal of Mathematical and Management Sciences, 40(3), 261-282.

[23] Saritha, S., Mamatha, E., & Reddy, C. S. (2019). Performance measures of online warehouse service system with replenishment policy. Journal Europeen Des Systemes Automatises, 52(6), 631-38.

[24] Bandi, C., Bertsimas, D., & Youssef, N. (2018). Robust transient analysis of multi-server queueing systems and feed-forward networks. Queueing Systems, 89(3), 351-413.

[25] Saritha, S., Mamatha, E., Reddy, C. S., & Anand, K. (2019). A model for compound poisson process queuing system with batch arrivals and services. Journal Europeen des Systemes Automatises, 53(1), 81-86

[26] Jain, S., & Jain, S. (2016). Analysis of multi level feedback queue scheduling using markov chain model with data model approach. International Journal of Advanced Networking and Applications, 7(6), 2915.