Journal of NAMP 69, 1 (2025) 165-174

The Nigerian Association of Mathematical Physics

Journal homepage: https://nampjournals.org.ng

MODELING THE EFFICIENCY OF MANPOWER ALLOCATION IN AN ACADEMIC DEPARTMENT.

***L. I. Igbinosun, N. R. Udoenoh and S. Ologun**

Department of Mathematics, University of Uyo, P. M. B. 1017, Uyo, Nigeria

ARTICLE INFO

Article history:
Received xxxxx
Revised xxxxx
Accepted xxxxx
Available online xxxxx

Keywords:
Traffic optimization,
Traffic Intensity,
Markov Chains.

ABSTRACT

This paper presents the application of Markov Chain in modeling the efficiency of manpower in an academic department during students' registration. The system is formulated as a finite-source queuing model where students arrive for registration at a given rate, and staff serve them at an exponential service rate. Key performance metrics such as the expected number of students in the system, busy and idle periods, and waiting times are derived. Simulation shows the system's evolution over time; the queue decreases and staff transition from fully busy to idle as fewer students remain. This analysis helps optimize staffing and resource allocation during peak registration periods.

2010 Mathematics Subject Classification: 60K25, 68M20, 90B06.

1. Introduction

Humans can be taken as machines that perform one form of task or another, the performance of any machining system plays a vital role in human life. Due to machine breakdown, there may not only be a loss of production but also a loss of cost and inconvenience. To avoid this loss, the spare and appropriate repair facility should be considered. In view of such a design the standby units play an important role so that the machining system may keep working to provide the desired grade of service all the time. If an online unit fails, an available standby unit replaces it and the failed machine is sent for immediate repair. A standby unit may be 'cold' stand by type which has zero failure rate whereas 'warm' standby has failure rate non-zero and less than the failure rate of an online unit. Today's machining systems are highly sophisticated and complex. These are comprised of a number of complicated parts. Failure of any part(s) or whole machine directly affects the service system being sought.

*Corresponding author: L. I. IGBINOSUN

E-mail address: luckyigbinosun@uniuyo.edu.ng

https: https://doi.org/10.60787/jnamp.vol69no1.506

Thus, a machining system can become out of order at any stage or can have different reasons or modes for its failure or inactivity. An $M/\mathrm{EK}/1$ machine repair problem having N identical automatic machines maintained by a single non-reliable service station was studied by [9]. Armstrong [1] suggested preventive maintenance through age repair policies for a range of machine repair problems. [3] made the performance prediction of a two-mode failure machine interference model with spare. Performance analysis of state dependent machine repair system with mixed standbys and two modes of failure were done by [4]. [6] studied a manufacturing system tending to failure concerning two machines working in passive redundancy, whose turning out one part experienced two modes of failure and repair. [5] examined a machine repair problem with homogeneous machines and stand-bys under the care of multiple technicians operating a synchronous vacation policy. Any interruption in the operation of machining systems involved in manufacturing/production causes a lot to the concerned organization. To tackle this problem and to maintain the system during machine breakdown, standbys play an essential role. Yuan [10] studied the optimal management problem of an $M/\mathrm{M}/2/\mathrm{K}$ queuing system with controlling arrival and service of a two-removable-server system. In some academic departments for example, students would naturally come at their own pace to register their courses (subjects) if there were no time-lines fixed for it. However, most students rush for courses' registration whenever the examination schedule is announced. The aim of this study, however, is to apply a profitable and comprehensive model by considering a finite - source queuing model to modeling the allocation of staff handling students' registration in an academic department. The transient probabilities are obtained to establish the system performance indices. The remaining sections of the paper are organized as follows: The model description and methodology used are given in section 2. In section 3, the mathematical analysis and some performance measures are derived. Finally, the discussion is presented in section 4.

2. Methodology

This paper discusses a finite - source queuing model applied to modeling the allocation of staff handling student registration in an academic department. This system can be viewed as a Markov Chain, where students arrive, are served, or wait in a queue. We analyze the manpower ow using Markov Chain in the following steps:

- Each state represents the number of students either waiting in the queue or being served at a given time.
- State S_0: No students in the system, and the server (staff) is idle.
- State S_n: n students are either being served or waiting.
- The system follows a birth-death process, where:
- Birth rate λ represents the arrival rate of students.
- Death rate μ represents the service rate at which students are served.

Markov Chain Dynamics:

- The system transitions from state S_n to S_{n+1} as new students arrive at rate λ.
- The system transitions from S_n to S_{n-1} as students are served at rate μ.
- Idle time is when the system remains in state S_0.

Transition Probability Matrix:

For a Markov Chain, the transitions between states depend on the current state and follow a Poisson process for arrivals and an exponential service rate for serving students. A transition probability matrix P can be constructed as follows:

$$\begin{pmatrix} 1-\lambda-\mu & \lambda & 0 & 0 & \dots \\ \mu & 1-\lambda-\mu & \lambda & 0 & \dots \\ 0 & \mu & 1-\lambda-\mu & \lambda & \dots \\ 0 & 0 & \mu & 1-\lambda-\mu & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

Here:
P_{ij} represents the probability of transitioning from state i (e.g., i students in the system) to state j. λ is the arrival rate of students, μ is the service rate. This matrix defines the probabilities of transitioning from one state to another, based on student arrivals and service completions.

Steady-State Probabilities:
In the long term, we can calculate the steady-state probabilities π_n, which represent the probability of having n students in the system in state S_n. These probabilities are obtained by solving the balance equations for the Markov Chain. These are calculated using the balance equations:

$$\lambda\pi_n = \mu\pi_{n+1} \text{ for } n \geq 0$$

From which we derive:

$$\pi_{n+1} = \frac{\lambda}{\mu}\pi_n = \rho\pi_n$$

Where $\rho = \frac{\lambda}{\mu}$ is the traffic intensity of the system. At steady state, the sum of all probabilities equals 1:

$$\sum_{n=0}^{S} \pi_n = 1$$

For finite queues, the system can be truncated at a maximum number of students, N, and the steady-state probabilities can be used to assess the long-term distribution of students in the system.
The Key Performance Metrics are:

Expected number of students in the system $E[N]$:

$$E[N] = \sum_{n=0}^{S} n\pi_n$$

Expected number of students in the queue $E[Q]$:

$$E[Q] = \sum_{n=1}^{S} (n-1)\pi_n$$

Expected waiting time $E[W]$:

$$E[W] = \frac{E[N]}{\lambda_{eff}}$$

Expected busy period $E[B]$:

$$E[B] = 1 - (E[I] + E[V])$$

Effective arrival rate λ_{eff}:

$$\lambda_{eff} = \sum_{n=1}^{S} (S-n)\lambda\pi_n$$

3. Application
Assume the following parameters based on the system's capacity, $S = 50$:
$\lambda = 0.0135$ students/second (arrival rate)
$\mu = 0.98$ students/second (service rate)

Using the recursive formula for steady-state probabilities, we can compute the expected number of students, queue length, and other performance metrics for the system at different times. Markov Chains can model the stochastic behavior of students entering, being served, or waiting in the system, allowing us to predict key performance metrics such as queue length, waiting time, and idle periods. This provides valuable insights into optimizing staff allocation and improving registration efficiency in the academic department.

To solve an example, consider a department, say Computer Engineering, with a class size of 50. We will model the system based on the given parameters for the arrival rate (λ), service rate (μ), and the total number of students in the system (S). The system follows a birth-death process, which is a special type of Markov Chain commonly used to model queuing systems.

Using the MATLAB software, the transient and steady-state probabilities is simulated over time to observe how the system evolves. Given an academic department system:

- Initial arrival of students is 50.
- Busy period starts at 100%, indicating staff is fully occupied.
- As time progresses, the number of students decreases, and idle periods begin to increase.

The transitions can be represented in a state-transition diagram, where each state has two possible actions: 1. Arrival of a new student, increasing the state count by 1.

2. Completion of service, decreasing the state count by 1.

To solve for the steady-state probabilities π_n (the probability of having n students in the system), we use the balance equations for each state n:

$$\lambda\pi_n = \mu\pi_{n+1} \text{ for } n \geq 0$$

From this, we derive the recursive formula:

$$\pi_{n+1} = \frac{\lambda}{\mu}\pi_n = \rho\pi_n$$

Where, $\rho = \frac{\lambda}{\mu}$ is the traffic intensity. Substituting the given values:

$$\rho = \frac{0.0135}{0.98} \approx 0.0138$$

Thus, the steady-state probabilities can be computed recursively.

We start with π_0 (the probability of having 0 students in the system) and use the recursive formula to find π_n for $n = 1, 2, \ldots, 50$.

To ensure that the probabilities sum up to 1 (normalization condition):

$$\sum_{n=0}^{S} \pi_n = 1$$

This allows us to calculate π_0 first and then determine π_n for each state. After 25 cycles, the expected number of students in the system drops to 15.52, indicating efficient clearing of the queue.

- The waiting time for students reduces significantly over time, showing that the system can handle student registration without significant delays.

Figure 1 shows the expected number of students in the system at each cycle. Initially, the number of students is at its maximum (50), but as time progresses and students are served, the expected number decreases. The system clears out students efficiently over time, resulting in a gradual decline in the expected number of students. The second graph illustrates the periods when the server (department staff) is busy or idle over time. Initially, the busy period is at its maximum (100%), indicating that the server is fully occupied with student registrations. Over time, as fewer students remain in the system, the busy period decreases, and the idle period starts to increase, reflecting that the staff has less work as the queue diminishes.

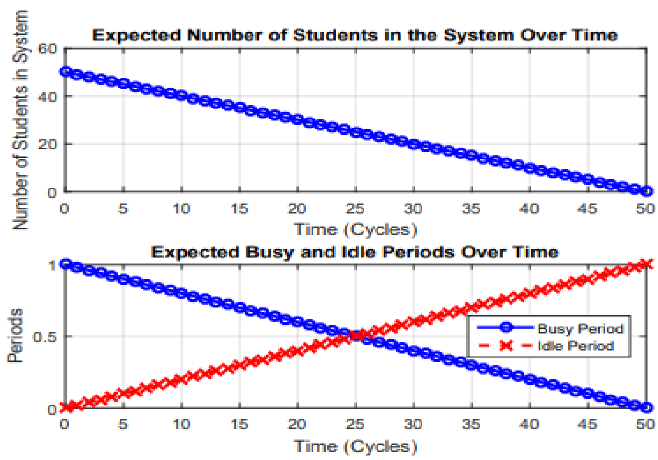


Figure 1: Expected Number of Students Over Time

The graphs represent the dynamic behavior of an academic department's registration system as students queue up to be served. The system uses a Markov Chain to model the arrival of students and the availability of staff for service. The key performance measures we focus on are the number of students in the system, the busy periods, and the idle periods of the staff. At time 0 (Cycle 0), The system is fully loaded with 50 students, meaning the server (staff) is completely busy serving them. As time progresses, the queue clears as the students are processed. The number of students reduces, indicating that the staff is efficiently handling the registrations. At the end of the time Period, the expected number of students drops to near 0, meaning the system has processed all the students, and the queue is empty. Imagine a scenario where 50 students arrive for course registration at an academic department, and they are served one by one by a staff member. As time passes, more students are processed, and fewer students remain in the queue. By the end of the registration period, the system reaches a state where the busy period is minimal, and the staff is idle for much of the time.

Initially, the staff is continuously engaged with registering students, and there is no idle time. As the queue thins out (students are served), the staff gradually becomes less busy, and eventually, they have more idle periods because there are fewer students left to be served, see table below.

Table 1: Steady-State Metrics for the Queuing System

State (n)	(EOI)	(EOB)	(EFM)
0	0.000	50.000	0.00
1	0.010	49.482	0.48
2	0.026	48.702	0.40
⋮	⋮	⋮	⋮

State (n)	(π_n)	($E[N]$)	($E[Q]$)
0	0.986263	0.000000	0.000000
1	0.013593	0.013593	0.000000
2	0.000187	0.000374	0.000187
3	$2.5781e-06$	$7.7343e-06$	$2.7343e-06$
⋮	⋮	⋮	⋮
10	$2.4269e-19$	$2.4269e-18$	$2.4269e-18$
⋮	⋮	⋮	⋮
50	$8.8994e-94$	$4.4497e-92$	$4.4497e-92$

If the traffic intensity $\rho = \frac{\lambda}{\mu}$ increases, it indicates that the arrival rate λ of students is getting closer to the service rate μ, or even surpasses it. This has significant implications on the performance metrics of the queuing system. Let's explore what happens when ρ increases.

Key Impacts of Increasing Traffic Intensity (ρ)

1. Higher Steady-State Probabilities in Upper States: As ρ increases, the steady-state probabilities π_n for higher values of n (more students in the system) increase. This means that there's a greater likelihood of having a larger number of students in the system at any given time, leading to longer queues and more congestion. In the extreme case where $\rho \geq 1$, the 8 system becomes unstable, meaning students arrive faster than they can be served, leading to an infinite build-up in the queue.

2. Increase in Expected Number of Students in the System $E(N)$: As ρ increases, the expected number of students in the system also increases. This is because more students are likely to be waiting or being served at any given time.

$$E(N) = \sum_{n=0}^{S} n\pi_n$$

For example, if the traffic intensity increases from $\rho = 0.0138$ (a very low value) to a higher value like $\rho = 0.8$, the expected number of students in the system will increase significantly.

3. Longer Queues $E(Q)$:

The expected number of students in the queue also increases as ρ increases. This is because the server is busier for longer periods, causing students to accumulate in the queue.

$$E(Q) = \sum_{n=1}^{S} (n-1)\pi_n$$

At higher traffic intensities, the queue grows longer as the system struggles to process students as fast as they arrive.

4. Increased Waiting Time $E(W)$: Waiting time depends on the number of students already in the system when a new student arrives. As ρ increases, more students are in the system on average, causing each new arrival to experience longer waits before being served.

$$E[W] = \frac{E[N]}{\lambda_{eff}}$$

For example, at a higher ρ, the expected waiting time increases dramatically, as more students are ahead in the queue, causing delays.

5. Busy Period $E(B)$: As the traffic intensity increases, the server remains busy for longer periods. In fact, if ρ approaches 1, the server is essentially always busy. The busy period $E(B)$, which was already 1 (indicating a fully busy server) in the previous scenario, will still remain at 1, but this now reflects an even greater load on the system.

Quantitative Example with Increased Traffic Intensity

Let's assume the traffic intensity increases from ρ = 0.0138 to $\rho = 0.8$. Below is a summary of how the metrics change:

Behavior at Different ρ Values:

Table 2: Steady-State Metrics for the Queuing System

Metric	Low $\rho = 0.0138$	High $\rho = 0.8$
$E(N)$	0.014	Much higher ($\approx$ 40+)
$E(Q)$	0.00019	Significantly higher
$E(W)$	1.53 seconds	Much longer (could be minutes or hours)
$E(B)$	1 (fully busy)	1 (fully busy, more pressure)

1. When ρ is low (e.g., 0.1): The system operates efficiently, with only a few students in the system at any given time. The queues are short, and students are served quickly with minimal waiting times. The server may not always be fully busy, leading to some idle time.

2. When ρ is moderate (e.g., $0.5 - 0.8$): The system becomes more congested as the arrival rate gets closer to the service rate. The queue starts to build up, and waiting times increase noticeably. The server is almost always busy, with very little idle time.

3. When $\rho \geq 1$: The system becomes unstable because students arrive faster than they can be served. This leads to an infinite queue and unbounded waiting times. The server is perpetually busy, but it cannot keep up with the demand, leading to continuous system overload.

Graphical Representation of Changes:

If we plot the steady-state probabilities for different values of ρ, we will observe the following;
*Low ρ: The steady-state probabilities πn are concentrated around the lower states, meaning the system spends most of its time with a small number of students.
*High ρ: The steady-state probabilities πn shift toward the higher states, indicating a higher likelihood of the system having many students in the system or in the queue. When the traffic intensity ρ increases, the system performance deteriorates: More students are present in the system on average, leading to longer queues and increased waiting times. The server is fully busy, but it may struggle to keep up with the arrival rate, especially as ρ approaches or exceeds 1. It becomes crucial to either increase the service rate μ (e.g., by adding more staff) or reduce the arrival rate λ to prevent the system from becoming overwhelmed. This highlights the importance of keeping ρ below 1 to ensure stable and efficient system performance.

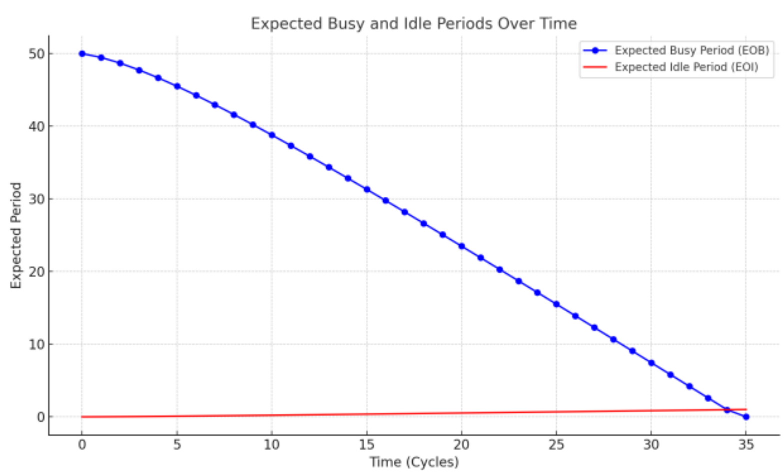
Discussion and Conclusion

Imagine a scenario where 50 students arrive for course registration at an academic department, and they are served one after the other by a staff member. As time passes, more students are processed, and fewer students remain in the queue. By the end of the registration period, the system reaches a state where the busy period is minimal, and the staff is idle for much of the time. Initially, the staff is continuously engaged with registering students, and there is no idle time. As the queue thins out (students are served), the staff gradually becomes less busy, and eventually, they have more idle periods because there are fewer students left to be served. This reflects the process where, at peak registration times, the staff is fully occupied, but as the workload decreases, they experience more breaks and idle times. The efficient performance of the registration system ensures that students are served in a timely manner, clearing the queue within a few cycles. Initially, when the system is fully loaded with students, the staff is busy all the time (100%). As the queue reduces, the staff experiences idle time. The shift from busy to idle periods shows that the staff is able to handle the load efficiently, and as fewer students remain, their workload decreases, allowing for rest periods.

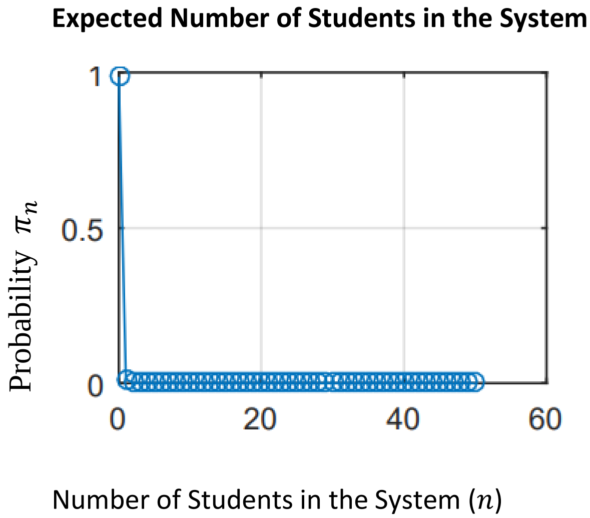
Practical Implications of the results:
The service rate of the staff (0.98) is high enough to efficiently clear the student queue. The arrival rate of students (0.0135) is manageable, preventing significant bottlenecks. The system clears out the queue relatively quickly, reducing the wait time for students significantly after the first few cycles.

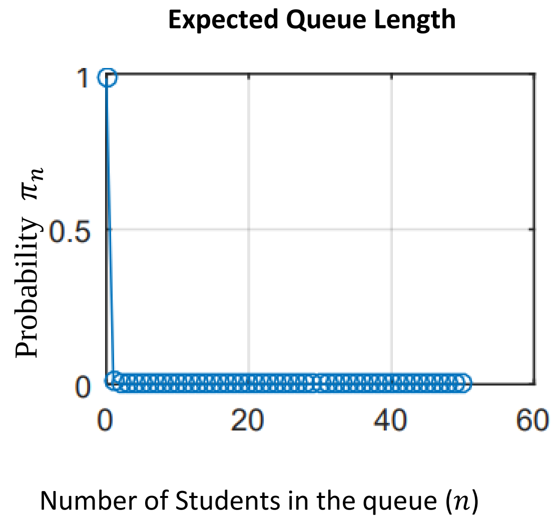
On efficient Staffing, the graphs indicate the importance of efficient staff allocation. The staff is most busy at the beginning of the registration period, but as time progresses, the staff becomes more idle, allowing for optimal staffing decisions. The ability to clear the queue (decreasing number of students) quickly is an indicator of an efficient system. The longer the queue persists, the more the staff will be kept busy, resulting in longer waiting times for students. The increase in idle time over time reflects that once the peak registration period is over, the staff can rest or be assigned to other tasks, optimizing their productivity. This queuing model using Markov Chains helps determine how many staff members are needed to handle peak registration times and how to optimize the allocation of resources to ensure students are served efficiently.

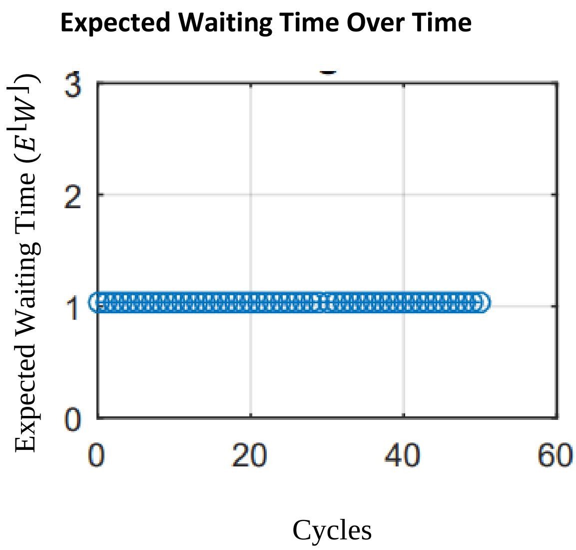
Figure 2 is the graph showing the Expected Busy Period (EOB) and Expected Idle Period (EOI) over time (in cycles).



The blue line represents the Expected Busy Period (EOB), which starts high (around 50) and gradually decreases as the system clears out students. The red line shows the Expected Idle Period (EOI), which starts at 0 (fully occupied at the beginning) and gradually increases as fewer students remain in the system. This graph reflects the gradual clearing of the queue, with the system becoming less busy and slightly more idle over time.







Sensitivity Analysis on Idle Time: The results show that the server, taking at most two minutes of rest, yields the optimal performance of the server. This means that slight pauses or breaks in the system (likely due to idle time between arrivals) do not significantly impact the system's ability to handle incoming student registrations efficiently.

At the beginning of the registration process, the system is heavily loaded with students (up to 50), and the server is fully occupied ($EOB \approx 1$). The idle period is almost zero ($EOI = 0$). As more students get registered, the busy period remains dominant, but it starts to reduce slightly because the system has fewer students left to serve. The idle period gradually increases but remains low. Toward the end, as the number of students in the queue decreases significantly, the server begins to have more idle time (EOI increases), and the busy period reduces. However, the server is still relatively engaged for most of the time until the very last cycles.

The busy period remains high initially and decreases gradually, showing that the staff is working efficiently and is mostly engaged with student registrations. The idle period: Starts at zero and increases over time, reflecting the system's effective clearing of the queue. As fewer students are left to be registered, idle times increase, indicating that the workload lightens toward the end of the process. This graph provides an important insight into staff utilization, showing that staff are busy during peak periods but eventually have more idle time as the workload decreases, ensuring the registration process is completed efficiently. The insights obtained from this analysis can help improve staffing decisions, ensuring minimal idle time for the staff while optimizing student service during peak registration cycles.

References

[1] Armstrong, M.J. (2002): Age repair policies for the machine repair problem, Euro. J. Oper. Res., Vol.138, No. 1, pp. 127-141.

[2] Jain, M., Baghel, K.P.S. and Jadown, M. (2004): Performance prediction of machine interference model with spare and two modes of failure, Operations Research, Information Technology and Industry (eds, M. Jain and G.C. Sharma), S.R.S. Pub., Agra, pp. 197-208.

[3] Jain, M., Rakhee and Maheshwari, S. (2004): Reliability Analysis of redundant repairable system with degraded failure, Operations research, Information Technology and Industry: (eds, M. Jain and G.C. Sharma), 14 S.R.S. pub., Agra, pp. 215-229.

[4] Jain, M., Sharma, G.C. and Sharma R.: (2008): Performance modelling of state dependent system with mixed standbys and two modes of failure, Appl. Math. Model. Vol. 32, No. 5, pp. 712-724.

[5] Ke, J.C. and Wu, C.H. (2012): Multi-server machine repair model with standbys and synchronous multiple vacation, Comp. Ind. Eng. Vol. 62, pp. 296-305.

[6] Mehrgani, B.E., Nadeau, S. and Kenne, J.P. (2011): Lockout/tag out and operational risks in the production control of manufacturing systems with passive redundancy, Int. J. Prod. Econ., Vol. 132, No.2, pp. 165-173.

[7] Sharma, G.C., Jain, M. and Pundhir, R.S. (2005): Loss and delay multiserver queueing model with discouragement and additional server, J. Raj. Acad. Phy. Sci., Vol. 4, No. 2, pp. 115-120.

[8] Wang, K.H. and Chiu, L.W. (2006): Cost benefits analysis of availability systems with warm standby units and imperfect coverage, Appl. Math. Comp., Vol. 172, No. 2, pp. 1239-1256.

[9] Wang, K.H. and Kuo, M.Y. (1997): Profit analysis of the M/E k /1 machine repair problem with anon-reliable service station, Comp. Ind. Eng., Vol.32, No. 3, pp. 587-594.

[10] Yuan, L. and Meng, X.Y. (2011): Reliability analysis of a warm standby repairable system with priority in use. Appl. Math. Model. Vol. 35, No. 9, pp.4295-4303.