M/M/1 Retrial Queueing System with Vacation Interruptions under Pre-Emptive Priority Service

A. Muthu Ganapathi Subramanian
Tagore Arts College, Pondicherry
csamgs1964@gmail.com

G. Ayyappan
Pondicherry Engineering College, Pondicherry
ayyappanpec@hotmail.com

Gopal Sekar
Tagore Arts College, Pondicherry
gopsek28@yahoo.com

Abstract

Consider a single server retrial queueing system with pre-emptive priority service and vacation interruptions in which customers arrive in a Poisson process with arrival rate λ_1 for low priority customers and λ_2 for high priority customers. Further it is assume that the service times follow an exponential distribution with parameters μ_1 and μ_2 for low and high priority customers respectively. The retrial is introduced for low priority customers only. The server goes for vacation after exhaustively completing the service to both types of customers. The vacation rate follows an exponential distribution with parameter α. The concept of vacation interruption is used in this paper that is the server comes from the vacation into normal working condition without completing his vacation period subject to some conditions. Let k be the maximum number of waiting spaces for high priority customers in front of the service station. The high priority customers will be governed by the pre-emptive priority service. We assume that the access from orbit to the service facility is governed by the classical retrial policy. This model is solved by using Matrix geometric Technique. Numerical study have been done for Analysis of Mean number of low priority customers in the orbit (MNCO), Mean number of high priority customers in the queue (MPQL), Truncation level (OCUT), Probability of server free and Probabilities of server busy with low and high priority customers and probability of server in vacation for various values of λ_1, λ_2, μ_1, μ_2, α and σ in elaborate manner and also various particular cases of this model have been discussed.

Keywords: Single server – pre-emptive priority service – matrix geometric method – classical retrial policy-single vacation-exhaustive service – vacation interruptions.

1. Introduction

Queueing systems in which arriving customers who find all servers and waiting positions (if any) occupied may retry for service after a period of time are called Retrial queues [1, 2, 7, 8]. Because of the complexity of the retrial queueing models, analytic results are generally difficult to obtain. There are a great number of numerical and approximations methods are available, in this paper we will place more emphasis on the solutions by Matrix geometric method [4, 6, 11, 12, 13, 14]. Vacation policies have been extensively discussed by many researchers in queueing and retrial queueing models. In this paper we introduce this vacation interruption policy for retrial queueing system with priority service. The concept of

vacation interruption was discussed by **Jihong Li and Naishuo Tian [10]** in the working vacation under classical queueing models who described it as

"*We introduce a new policy: the server can come back from the vacation to the normal working level once some indices of the system, such as the number of customers, achieve a certain value in the vacation period. The server may come back from the vacation without completing the vacation. Such policy is called vacation interruption*."

The above policy is modified and has been introduced for priority services in Retrial queueing system. The vacation interruption for priority service in retrial queueing system is governed by the following principle i.e. if the server is on vacation and if atleast one high priority customer enters the system, then the server will be called back and he should immediately start to serve interrupting his vacation period.

2. Motivation and Examples

This model arises due to many real time examples. We are illustrating some examples here to better understand of this model

a. This model suited for military and defence. Officers may come home for vacation by availing their leave. If there is any complicated situations happen in the border, then they will be call back to join duty immediately by interrupting their vacation.

b. It is suited for General Hospital (GH) also in which doctors will be call back to their service to attend the emergency cases by interrupting theirt vacation

c. It is suited for officials in the Government sector. If there is any natural calamity or declaration of election during this time the officials will be call back to resume their work by interrupting their vacations.

3. Model Description

Consider a single server retrial queueing system with pre-emptive priority service [3, 5, 9], where the server goes for exhaustive service type vacation with vacation interruptions in which two types of customers arrive in a Poisson process with arrival rate λ_1 for low priority customers and λ_2 for high priority customers. These customers are identified as primary customers. Further it is assumed that the service times follow an exponential distribution with parameters μ_1 and μ_2 for low and high priority customers respectively. Vacation rate follows an exponential distribution with parameter α. Retrial is introduced for low priority customers only. Let k be the maximum number of waiting spaces for high priority customers in front of the service station.

In this model the status of the server may be idle, busy or in vacation. The server cannot be in vacation if there is atleast one high priority customer present in the system. If the server is free at the time of a primary customer (low/high) arrival, the arriving customer begins to be served immediately by the server and customer leaves the system after service completion. Otherwise, if the server is busy or in vacation then the low priority arriving customer goes to orbit and becomes a source of repeated customers. This pool of source of repeated customers may be viewed as a sort of queue. Every such source produces a Poisson process of repeated customers with intensity σ. If an incoming repeated customer (low) finds the server free, it is served and leaves the system after service, while the source which produced this repeated customer disappears. If the server is busy and there are some waiting spaces then the high priority customer can enter into the queue and waits for service. If there are no waiting spaces then the high priority customers cannot enter into the service station and will be lost for the system.

If the server is in vacation and a high priority customer enters the system then as per vacation interruption policy the server comes back immediately and starts to serve the customer. Otherwise, the system state does not change.

If the server is engaging a low priority customer and if at that time a high priority customer enters, then this high priority customer will immediately get the service and that low priority customer goes to orbit without completing the service. This type of priority service is called the **Pre-emptive priority service**.

Most of the queueing system with repeated attempts assume that each customer in the retrial group seeks service independently of each other after a random time exponentially distributed with rate $\boldsymbol{\sigma}$ so that the probability of repeated attempt during the interval $(t, t + \Delta t)$ given that there were n customers in orbit at time t is $\mathbf{n\sigma\ \Delta t + O(\Delta t)}$. This discipline for access for the server from the retrial group is called **classical retrial rate policy**. The input flow of primary calls (low and high), interval between repetitions and service times are mutually independent.

4. Matrix Geometric Solutions

Let N(t) be the random variable which represents the number of low priority customers in the orbit at time t, H(t) be the random variable which represents the number of high priority customers in the queue (in front of the service station) at time t and S(t) represents the server state at time t.

The random process is described as

$$\{ < N(t) , H(t), S(t) > / \ N(t) = 0, 1, 2, \ldots ; \ H(t) = 0, 1, 2, \ldots, k \ ; \ S(t) = 0, 1, 2, 3 \}$$

S(t) = 0 means the server is idle at time t.

S(t) = 1 means the server is busy with a low priority customer at time t.

S(t) = 2 means the server is busy with a high priority customer at time t.

S(t) = 3 means the server is in vacation at time t.

The possible state space is

$$\{ (u , v , w) / u = 0,1,2,3, \ldots \; ; \; v = 0 \; ; \; w = 0,1,2,3 \} \cup$$
$$\{ (u , v , w)/ \; u = 0,1,2,3, \ldots \; ; \; v =1,2,3, \ldots , k \; ; \; w = 2\}$$

The infinitesimal generator matrix **Q** is given below

$$\mathbf{Q} = \begin{pmatrix} A_{00} & A_0 & O & O & O & \ldots \\ A_{10} & A_{11} & A_0 & O & O & \ldots \\ O & A_{21} & A_{22} & A_0 & O & \ldots \\ O & O & A_{32} & A_{33} & A_0 & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \end{pmatrix}$$

we define

$T_1 = -(\lambda_1+ \lambda_2)$ $T_2 = -(\lambda_1+ \lambda_2+\mu_1)$ $T_3 = -(\lambda_1+ \lambda_2+\mu_2)$ $T_4 = -(\lambda_1+ \lambda_2+\alpha)$

$T_6 = -(\lambda_1+\mu_2)$ $T_8 = -(n\sigma+ \lambda_1+ \lambda_2)$ $T_9 = -(M\sigma+ \lambda_1+ \lambda_2)$ $T_{10} = -(\lambda_2+\mu_1)$

$T_{11} = -(\lambda_2+\mu_2)$ $T_{12} = -(\lambda_2+\alpha)$

Matrices A_{00}, $A_{01,}$ $A_{n\ n\text{-}1}$, $A_{n\ n}$ and $A_{n\ n+1}$ are square matrices of order k+4 for n = 1, 2, 3, . . .

$$\mathbf{A_{00}} = \begin{pmatrix} T_1 & \lambda_1 & \lambda_2 & 0 & 0 & 0 & \ldots & 0 & 0 \\ 0 & T_2 & 0 & \mu_1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & T_3 & \mu_2 & \lambda_2 & 0 & \ldots & 0 & 0 \\ \alpha & 0 & \lambda_2 & T_4 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & \mu_2 & 0 & T_3 & \lambda_2 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & T_3 & \lambda_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & \mu_2 & T_6 \end{pmatrix}$$

$\mathbf{A_{nn-1}} = (a_{ij})$ for n = 1, 2, 3, . . .
where $a_{ij} = n\sigma$ if $i = 1$ and $j = 2$
$= 0$ otherwise.

$$\mathbf{A_{nn}} = \begin{pmatrix} T_8 & \lambda_1 & \lambda_2 & 0 & 0 & 0 & \ldots & 0 & 0 \\ \mu_1 & T_2 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 \\ \mu_2 & 0 & T_3 & 0 & \lambda_2 & 0 & \ldots & 0 & 0 \\ \alpha & 0 & \lambda_2 & T_4 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & \mu_2 & 0 & T_3 & \lambda_2 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & T_3 & \lambda_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & T_6 \end{pmatrix}$$

$\mathbf{A_{nn+1}} = A_0 = (b_{ij})$ for $n = 0, 1, 2, 3, \ldots$

where $b_{ii} = \lambda_1$ for $i = 2, 3, 4, \ldots, k+4$

$= \lambda_2$ for $i = 2$ and $j = 3$

$= 0$ otherwise.

If the capacity of the orbit is finite say M then

$$\mathbf{A_{MM}} = \begin{pmatrix} T_9 & \lambda_1 & \lambda_2 & 0 & 0 & 0 & \ldots & 0 & 0 \\ \mu_1 & T_{10} & 0 & 0 & 0 & 0 & \ldots & 0 & 0 \\ \mu_2 & 0 & T_{11} & 0 & \lambda_2 & 0 & \ldots & 0 & 0 \\ \alpha & 0 & \lambda_2 & T_{12} & 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & \mu_2 & 0 & T_{11} & \lambda_2 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & T_{10} & \lambda_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & -\mu_2 \end{pmatrix}$$

Let **X** be a steady-state probability vector of **Q**, partitioned as

$\mathbf{X} = (x(0), x(1), x(2), \ldots)$ where **X** satisfies

$$\mathbf{XQ} = \mathbf{0} \quad \text{and} \quad \mathbf{Xe} = 1 \tag{1}$$

$x(i) = (P_{i00}, P_{i01}, P_{i02}, P_{i03}, P_{i12}, P_{i22}, \ldots, P_{ik2})$; $i = 0, 1, 2, \ldots$

5. Computational Methods

Retrial queueing models can be solved computationally by the following techniques.

(a) Direct Truncation Method

(b) Generalized Truncation Method

(c) Truncation Method using level dependent quasi birth- and –death Process (LDQBD)

(d) Matrix Geometric Approximation.

5.1 Direct Truncation Method

In this method one can truncate the system of equations for sufficiently large value of the number of customers in the orbit, say M. That is, the orbit size is restricted to M such that any arriving customer finding the orbit full is considered lost. The value of M can be chosen so that the loss probability is small. Due to the intrinsic nature of the system, the only choice available for studying M is through algorithmic methods. While a number of approaches are available for determining the cut-off point, M, the one that seems to perform well is to increase M until the largest individual change in the elements of **X** for successive values is less than ε a predetermined infinitesimal value.

6. Analysis of Steady State Probabilities

Applying the Direct Truncation Method described in section 4.1, we find the steady state probability vector **X**. If M denotes the cut-off point or Truncation level, then the steady state probability vector $\mathbf{X}^{(\mathbf{M})}$ is partitioned as $\mathbf{X}^{(\mathbf{M})} = (x(0), x(1), x(2), \ldots, x(M))$, where $\mathbf{X}^{(\mathbf{M})}$ satisfies $\mathbf{X}^{(\mathbf{M})}\,\mathbf{Q} = \mathbf{0}$ and $\mathbf{X}^{(\mathbf{M})}\,\mathbf{e} = 1$

where $x(i) = (P_{i00}, P_{i01}, P_{i02}, P_{i03}, P_{i12}, P_{i22}, \ldots, P_{ik2})$; $i = 0, 1, 2, 3, \ldots, M$.

The above system of equations is solved exploiting the special structure of the co-efficient matrix. It is solved by Numerical methods. Since there is no clear cut choice for M, we may start the iterative process by taking, say $M = 1$ and increase it until the individual elements of **X** do not change significantly. That is, if M^* denotes the truncation point then

$$\| \mathbf{x}^{\mathbf{M}^*}(\mathbf{i}) - \mathbf{x}^{\mathbf{M}^*-1}(\mathbf{i}) \|_\infty < \boldsymbol{\varepsilon} \quad \text{where } \boldsymbol{\varepsilon} \text{ is an infinitesimal quantity.}$$

7. Stability Condition

Theorem

The inequality $\left(\frac{F\lambda_1}{\mu_1}\right) < 1$ where F = 1 +x +x^2 + **. . .** + x^{k+2} and x = λ_2/μ_2 is the necessary and sufficient condition for the system to be stable.

Proof

Let **Q** be an infinitesimal generator matrix for the queueing system (without retrial).

The stationary probability vector **X** satisfies

$$\mathbf{XQ = 0} \text{ and } \mathbf{Xe =} 1 \qquad (2)$$

Let R be the rate matrix satisfying the equation

$$\mathbf{A_0 + RA_1 + R^2A_2 = 0} \qquad (3)$$

The system is stable if sp(R) < 1.

The matrix R satisfies sp(R) < 1 if and only if

$$\mathbf{\Pi A_0 e} < \mathbf{\Pi A_2 e} \qquad (4)$$

where $\mathbf{\Pi} = (\pi_1, \pi_2, \ldots, \pi_k, \pi_{k+1}, \pi_{k+2}, \pi_{k+3})$ satisfies

$$\mathbf{\Pi A = 0} \text{ and } \mathbf{\Pi e =} 1 \qquad (5)$$

and $$\mathbf{A = A_0 + A_1 + A_2} \qquad (6)$$

A_0, A_1 and A_2 are square matrices of order k+3

$A_0 = (a_{ij})$

where $a_{ij} = \lambda_1$ for i = 1 and j = 1
$= \lambda_2$ for i = 1 and j = 3
$= 0$ otherwise

$$\mathbf{A_1} = \begin{pmatrix} T_2 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 \\ 0 & T_3 & 0 & \lambda_2 & 0 & \ldots & 0 & 0 \\ 0 & \lambda_2 & T_4 & 0 & 0 & \ldots & 0 & 0 \\ 0 & \mu_2 & 0 & T_3 & \lambda_2 & \ldots & 0 & 0 \\ 0 & 0 & 0 & \mu_2 & T_3 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & 0 & \ldots & T_3 & \lambda_2 \\ 0 & 0 & 0 & 0 & 0 & \ldots & \mu_2 & T_6 \end{pmatrix}$$

$\mathbf{A_2} = (a_{ij})$

where $a_{ij} = \mu_1$ if $i = 1$ and $j = 1$

$= \mu_2$ if $i = 2$ and $j = 1$

$= \alpha$ if $i = 3$ and $j = 1$

$= 0$ otherwise.

By substituting A_0, A_1, A_2 in equations (5) and (6), we get

$$\pi_i = x^{i-1}\pi_1 \text{ where } i = 2, 3, 4, \ldots, k+3$$

From (5), $\pi_1 + \pi_2 + \pi_3 + \pi_4 + \pi_5 + \ldots + \pi_{k+2} + \pi_{k+3} = 1$.

Substituting the values of π_i in the above equation, we get

$$\pi_1 = F^{-1} \text{ where } F = 1 + x + x^2 + \ldots + x^{k+2} .$$

From (4), $(\lambda_1/\mu_1) < F^{-1}$

$$\left(\frac{F\lambda_1}{\mu_1}\right) < 1$$

The inequality $\left(\frac{F\lambda_1}{\mu_1}\right) < 1$ is also a sufficient condition for the retrial queueing system to be stable.

Let Q_n be the number of customers in the orbit after the departure of n^{th} customer from the service station. We first prove the embedded Markov chain $\{Q_n, n \geq 0\}$ is ergodic if $\left(\frac{F\lambda_1}{\mu_1}\right) < 1$. It is readily to see that $\{Q_n, n \geq 0\}$ is irreducible and aperiodic. It remains to be proved that $\{Q_n, n \geq 0\}$ is positive recurrent. The irreducible and aperiodic Markov chain $\{Q_n, n \geq 0\}$ is positive recurrent if $|\psi_i| < \infty$ for all i and $\lim_{i \to \infty} \sup \psi_i < 0$ where

$$\psi_i = E(Q_{n+1} - Q_n / Q_n = i) ; \quad i = 0, 1, 2, 3, \ldots$$

$$\psi_i = \left(\frac{F\lambda_1}{\mu_1}\right) - \left(\frac{i\sigma}{\lambda_1 + \lambda_2 + i\sigma}\right)$$

if $\left(\frac{F\lambda_1}{\mu_1}\right) < 1$, then $|\psi_i| < \infty$ for all i and $\lim_{i \to \infty} \sup \psi_i < 0$.

Therefore the embedded Markov chain $\{Q_n , n \geq 0\}$ is ergodic.

As k → ∞, the above stability condition becomes $\left(\frac{\lambda_1}{\mu_1} + \frac{\lambda_2}{\mu_2}\right) < 1$.

8. Special Cases

1. This model becomes single server Retrial queueing system with Pre-emptive priority service discussed by Ayyappan and *et al* [5] if $\alpha \to \infty$.
2. This model becomes single server Retrial queueing system if $\lambda_2 \to 0$, $\mu_2 \to \infty$, $\alpha \to \infty$.
3. This model becomes single server classical queueing system if $\lambda_2 \to 0$, $\mu_2 \to \infty$, $\alpha \to \infty$ and $\sigma \to \infty$.

9. System Performance Measures

In this section we list some important performance measures along with their formulae. These measures are used to bring out the qualitative behaviour of the queueing model under study. Numerical studies have been dealt to study the following measures.

p(u,0,0) = Probability that there are u customers in the orbit, no customers in the high priority queue and server is free.

p(u,0,1) = Probability that there are u customers in the orbit, no customers in the high priority queue and server is busy with a low priority customer.

p(u,v,2) = Probability that there are u customers in the orbit , v customers in the high priority queue and server is busy with a high priority customer.

p(u,0,3) = Probability that there are u customers in the orbit and v customers in the high high priority queue and server is in vacation.

We can find the steady state probability vector **X** for various values of $\lambda_1, \lambda_2, \mu_1, \mu_2, \sigma, \alpha$ and k and the following system measures can be studied with these probabilities.

a. The probability mass function of server state

Let S(t) be the random variable which represents the server state at time t. In this model S(t) takes the values 0, 1, 2, 3.

P_0 = Probability that the server is idle $= \sum_{i=0}^{\infty} p(i,0,0)$

P_1 = Probability that the server is busy with a low priority customer $= \sum_{i=0}^{\infty} p(i,0,1)$

P_2 = Probability that the server is busy with a high priority customer $= \sum_{i=0}^{\infty}\sum_{j=0}^{k} p(i,j,2)$

P_3 = Probability that the server is in vacation $= \sum_{i=0}^{\infty} p(i,0,3)$

b. The probability mass function of number of customers in the orbit

Let X(t) be the random variable which represents the number of low priority customers in the orbit. X(t) takes the values 0, 1, 2, 3, . . .

Probability of no customers in the orbit

$$= p(0,0,0) + p(0,0,1) + \sum_{j=0}^{k} p(0,j,2) + p(0,0,3)$$

Probability of n customers in the orbit =

$$p(n,0,0) + p(n,0,1) + \sum_{j=0}^{k} p(n,j,2) + p(n,0,3)$$

c. The Probability mass function of number of high priority customers in the queue

Let H(t) be the random variable which represents the number of high priority customers in the queue at time t. In this model we assume that the capacity of high priority customers in the queue is finite and H(t) takes the values 0, 1, 2, 3 , . . . , k.

Probability of no customers in the high priority queue = $\sum_{i=0}^{\infty}\sum_{l=0}^{3} p(i,0,l)$

Probability of n customers in the high priority queue = $\sum_{i=0}^{\infty} p(i,n,2)$

d. The Mean number of high priority customers in the queue (MPQL)

$$\text{MPQL} = \sum_{j=1}^{k} j\left(\sum_{i=0}^{\infty} p(i,j,2)\right)$$

e. The Mean number of low priority customers in the orbit

$$\text{MNCO} = \sum_{i=0}^{\infty} i\left(p(i,0,0) + p(i,0,1) + \sum_{j=0}^{k} p(i,j,2) + p(i,0,3)\right)$$

f. The probability that the orbiting customer is blocked

$$\text{Blocking Probability} = \sum_{i=1}^{\infty}\left(\sum_{j=0}^{k} p(i,j,2) + p(i,0,1) + p(i,0,3)]\right)$$

g. The probability that an arriving customer enters the service station immediately

$$= \sum_{i=0}^{\infty} p(i,0,0)$$

10. Numerical Study

The values of parameters λ_1, λ_2, μ_1, μ_2 are chosen so that they satisfy the stability condition discussed in section 6.

The system performance measures of this model have been done and expressed in the form of tables which are shown below by finding the steady state probability vector **X** for various values for λ_1, λ_2, μ_1, μ_2, σ, α and k.

If $\lambda_1 = 10$ $\lambda_2=5$ $\mu_1 =20$ $\mu_2=25$ $\sigma = 100$ $\alpha = 100$ and k = 6, then the steady state probability vector is **X** = (x(0) , x(1), x(2) , . . . , x(M))

where

x(0) = [0.2147, 0.1430, 0.0337, 0.0322, 0.0046, 0.0006, 0.0001, 0.0000, 0.0000, 0.0000]
x(1) = [0.0286, 0.1078, 0.0340, 0.0028, 0.0060, 0.0010, 0.0002, 0.0000, 0.0000, 0.0000]
x(2) = [0.0103, 0.0759, 0.0267, 0.0002, 0.0055, 0.0011, 0.0002, 0.0000, 0.0000, 0.0000]

x(3) = [0.0049, 0.0530, 0.0196, 0.0000, 0.0044, 0.0009, 0.0002, 0.0000, 0.0000, 0.0000]
x(4) = [0.0026, 0.0369, 0.0139, 0.0000, 0.0033, 0.0007, 0.0002, 0.0000, 0.0000, 0.0000]
x(5) = [0.0015, 0.0256, 0.0098, 0.0000, 0.0024, 0.0006, 0.0001, 0.0000, 0.0000, 0.0000]
x(6) = [0.0009, 0.0178, 0.0068, 0.0000, 0.0017, 0.0004, 0.0001, 0.0000, 0.0000, 0.0000]
x(7) = [0.0005, 0.0123, 0.0048, 0.0000, 0.0012, 0.0003, 0.0001, 0.0000, 0.0000, 0.0000]
x(8) = [0.0003, 0.0085, 0.0033, 0.0000, 0.0008, 0.0002, 0.0001, 0.0000, 0.0000, 0.0000]
x(9) = [0.0002, 0.0059, 0.0023, 0.0000, 0.0006, 0.0002, 0.0000, 0.0000, 0.0000, 0.0000]
x(10) = [0.0001,0.0041, 0.0016, 0.0000, 0.0004, 0.0001, 0.0000, 0.0000, 0.0000, 0.0000]

Similarly, we can find x (n) for $n \geq 10$ and it is noticed that $x(n) \to 0$ as $n \to \infty$. For the numerical parameters chosen above, $x(n) \to 0$ for $n \geq 37$ and the sum of the steady state probabilities becomes 0.9999999999. In the same manner, we can find steady state probability vector **X** for all values of $\lambda_1, \lambda_2, \mu_1, \mu_2, \sigma$, α and k.

System Performance Measures

Probability that the server is idle = 0.264736
Probability that the server is busy with a low priority customer = 0.500000
Probability that the server is busy with a high priority customer = 0.199998
Probability that there are 0 customers in the orbit = 0.428921
Probability that there are 1 customers in the orbit = 0.180424
Probability that there are 2 customers in the orbit = 0.120041
Probability that there are 3 customers in the orbit = 0.083086
Probability that there are 4 customers in the orbit = 0.057689
Probability that there are 5 customers in the orbit = 0.040027
Probability that there are 6 customers in the orbit = 0.027742
Probability that there are 7 customers in the orbit = 0.019207
Probability that there are 8 customers in the orbit = 0.013284
Probability that there are 9 customers in the orbit = 0.009180
Probability that there are 10 customers in the orbit = 0.006339
Probability that there are 11 customers in the orbit = 0.004374
Probability that there are 12 customers in the orbit = 0.003016
Probability that there are 13 customers in the orbit = 0.002079
Probability that there are 14 customers in the orbit = 0.001432
Probability that there are 15 customers in the orbit = 0.000986
Probability that there are 16 customers in the orbit = 0.000679
Probability that there are 17 customers in the orbit = 0.000467
Probability that there are 18 customers in the orbit = 0.000321

Probability that there are 19 customers in the orbit = 0.000221
Probability that there are 20 customers in the orbit = 0.000152
Probability that there are 21 customers in the orbit = 0.000104
Probability that there are 22 customers in the orbit = 0.000072
Probability that there are 23 customers in the orbit = 0.000049
Probability that there are 24 customers in the orbit = 0.000034
Probability that there are 25 customers in the orbit = 0.000023
Probability that there are 26 customers in the orbit = 0.000016
Probability that there are 27 customers in the orbit = 0.000011
Probability that there are 28 customers in the orbit = 0.000007
Probability that there are 29 customers in the orbit = 0.000005
Probability that there are 30 customers in the orbit = 0.000004
Probability that there are 31 customers in the orbit = 0.000002
Probability that there are 32 customers in the orbit = 0.000002
Probability that there are 33 customers in the orbit = 0.000001
Probability that there are 34 customers in the orbit = 0.000001
Probability that there are 35 customers in the orbit = 0.000001
Mean number of customers in the orbit = 1.839531
Probability that the orbiting customer is blocked = 0.391682
Probability that there are 0 customers in the high priority queue = 0.960002
Probability that there are 1 customers in the high priority queue = 0.032000
Probability that there are 2 customers in the high priority queue = 0.006400
Probability that there are 3 customers in the high priority queue = 0.001280
Probability that there are 4 customers in the high priority queue = 0.000256
Probability that there are 5 customers in the high priority queue = 0.000051
Probability that there are 6 customers in the high priority queue = 0.000010
Mean number of customers in the high priority queue = 0.049982

Tables 1, 2, 3 and 4 show the impact of σ and k over Mean number of customers in the orbit and Mean number of customers in the high priority queue. Further, we infer the following

- Mean number of customers in the orbit decreases as retrial rate σ increases.
- Mean number of customers in the high priority queue increases as k increases.
- This model becomes standard pre-emptive priority queueing model with vacation interruptions if σ is large.

Table 5 shows the effect of vacation rate over the system. As α is large, the model coincides with single server Retrial queueing system with pre-emptive priority service discussed by Ayyappan and *et al* **[5]**.

Table 1: System performance measures for λ_1=10, λ_2=5, μ_1=20, μ_2=25, α =100 & k=2

σ	P_0	P_1	P_2	P_3	MNCO	MPQL
10	0.2941	0.5000	0.1987	0.0072	4.6528	0.0449
20	0.2840	0.5000	0.1987	0.0173	3.0863	0.0449
30	0.2780	0.5000	0.1987	0.0233	2.5604	0.0449
40	0.2742	0.5000	0.1987	0.0270	2.2958	0.0449
50	0.2717	0.5000	0.1987	0.0296	2.1364	0.0449
60	0.2699	0.5000	0.1987	0.0314	2.0297	0.0449
70	0.2685	0.5000	0.1987	0.0328	1.9534	0.0449
80	0.2674	0.5000	0.1987	0.0339	1.8960	0.0449
90	0.2666	0.5000	0.1987	0.0347	1.8513	0.0449
100	0.2659	0.5000	0.1987	0.0354	1.8155	0.0449
200	0.2625	0.5000	0.1987	0.0388	1.6538	0.0449
300	0.2613	0.5000	0.1987	0.0400	1.5996	0.0449
400	0.2607	0.5000	0.1987	0.0406	1.5725	0.0449
500	0.2603	0.5000	0.1987	0.0410	1.5563	0.0449
600	0.2600	0.5000	0.1987	0.0412	1.5454	0.0449
700	0.2599	0.5000	0.1987	0.0414	1.5377	0.0449
800	0.2597	0.5000	0.1987	0.0416	1.5318	0.0449
900	0.2596	0.5000	0.1987	0.0417	1.5273	0.0449
1000	0.2595	0.5000	0.1987	0.0417	1.5237	0.0449
2000	0.2592	0.5000	0.1987	0.0421	1.5074	0.0449
3000	0.2590	0.5000	0.1987	0.0423	1.5019	0.0449
4000	0.2590	0.5000	0.1987	0.0423	1.4992	0.0449
5000	0.2589	0.5000	0.1987	0.0424	1.4976	0.0449
6000	0.2589	0.5000	0.1987	0.0424	1.4965	0.0449
7000	0.2589	0.5000	0.1987	0.0424	1.4957	0.0449
8000	0.2589	0.5000	0.1987	0.0424	1.4951	0.0449
9000	0.2589	0.5000	0.1987	0.0424	1.4947	0.0449

Table 2: System performance measures for λ_1=10, λ_2=5, μ_1=20, μ_2=25, α=100 & k = 4

σ	P_0	P_1	P_2	P_3	MNCO	MPQL
10	0.2929	0.5000	0.1999	0.0072	4.6909	0.0497
20	0.2828	0.5000	0.1999	0.0172	3.1159	0.0497
30	0.2769	0.5000	0.1999	0.0232	2.5872	0.0497
40	0.2731	0.5000	0.1999	0.0269	2.3212	0.0497
50	0.2706	0.5000	0.1999	0.0294	2.1609	0.0497
60	0.2688	0.5000	0.1999	0.0313	2.0537	0.0497
70	0.2674	0.5000	0.1999	0.0326	1.9769	0.0497
80	0.2663	0.5000	0.1999	0.0337	1.9192	0.0497
90	0.2655	0.5000	0.1999	0.0346	1.8743	0.0497
100	0.2648	0.5000	0.1999	0.0353	1.8383	0.0497
200	0.2614	0.5000	0.1999	0.0386	1.6757	0.0497
300	0.2602	0.5000	0.1999	0.0398	1.6213	0.0497
400	0.2596	0.5000	0.1999	0.0404	1.5941	0.0497
500	0.2592	0.5000	0.1999	0.0408	1.5777	0.0497
600	0.2590	0.5000	0.1999	0.0411	1.5668	0.0497
700	0.2588	0.5000	0.1999	0.0412	1.5590	0.0497
800	0.2587	0.5000	0.1999	0.0414	1.5532	0.0497
900	0.2586	0.5000	0.1999	0.0415	1.5486	0.0497
1000	0.2585	0.5000	0.1999	0.0416	1.5450	0.0497
2000	0.2581	0.5000	0.1999	0.0420	1.5286	0.0497
3000	0.2580	0.5000	0.1999	0.0421	1.5231	0.0497
4000	0.2579	0.5000	0.1999	0.0421	1.5204	0.0497
5000	0.2579	0.5000	0.1999	0.0422	1.5187	0.0497
6000	0.2578	0.5000	0.1999	0.0422	1.5177	0.0497
7000	0.2578	0.5000	0.1999	0.0422	1.5169	0.0497
8000	0.2578	0.5000	0.1999	0.0422	1.5163	0.0497
9000	0.2578	0.5000	0.1999	0.0423	1.5158	0.0497

Table 3: System performance measures for λ_1=10, λ_2=5, μ_1=20, μ_2=25, α=100 & k=6

σ	P_0	P_1	P_2	P_3	MNCO	MPQL
10	0.2928	0.5000	0.2000	0.0072	4.6927	0.0500
20	0.2828	0.5000	0.2000	0.0172	3.1174	0.0500
30	0.2768	0.5000	0.2000	0.0232	2.5886	0.0500
40	0.2731	0.5000	0.2000	0.0269	2.3225	0.0500
50	0.2706	0.5000	0.2000	0.0294	2.1622	0.0500
60	0.2687	0.5000	0.2000	0.0313	2.0550	0.0500
70	0.2674	0.5000	0.2000	0.0326	1.9782	0.0500
80	0.2663	0.5000	0.2000	0.0337	1.9205	0.0500
90	0.2654	0.5000	0.2000	0.0346	1.8755	0.0500
100	0.2647	0.5000	0.2000	0.0353	1.8395	0.0500
200	0.2614	0.5000	0.2000	0.0386	1.6769	0.0500
300	0.2602	0.5000	0.2000	0.0398	1.6225	0.0500
400	0.2596	0.5000	0.2000	0.0404	1.5953	0.0500
500	0.2592	0.5000	0.2000	0.0408	1.5789	0.0500
600	0.2589	0.5000	0.2000	0.0411	1.5680	0.0500
700	0.2588	0.5000	0.2000	0.0412	1.5602	0.0500
800	0.2586	0.5000	0.2000	0.0414	1.5544	0.0500
900	0.2585	0.5000	0.2000	0.0415	1.5498	0.0500
1000	0.2584	0.5000	0.2000	0.0416	1.5462	0.0500
2000	0.2581	0.5000	0.2000	0.0419	1.5298	0.0500
3000	0.2579	0.5000	0.2000	0.0421	1.5243	0.0500
4000	0.2579	0.5000	0.2000	0.0421	1.5216	0.0500
5000	0.2578	0.5000	0.2000	0.0422	1.5199	0.0500
6000	0.2578	0.5000	0.2000	0.0422	1.5188	0.0500
7000	0.2578	0.5000	0.2000	0.0422	1.5180	0.0500
8000	0.2578	0.5000	0.2000	0.0422	1.5175	0.0500
9000	0.2578	0.5000	0.2000	0.0422	1.5170	0.0500

Table 4: System performance measures for λ_1=10, λ_2=5, μ_1=20, μ_2=25 α=100 and k=8

σ	P_0	P_1	P_2	P_3	MNCO	MPQL
10	0.2928	0.5000	0.2000	0.0072	4.6928	0.0500
20	0.2828	0.5000	0.2000	0.0172	3.1175	0.0500
30	0.2768	0.5000	0.2000	0.0232	2.5886	0.0500
40	0.2731	0.5000	0.2000	0.0269	2.3226	0.0500
50	0.2706	0.5000	0.2000	0.0294	2.1623	0.0500
60	0.2687	0.5000	0.2000	0.0313	2.0550	0.0500
70	0.2674	0.5000	0.2000	0.0326	1.9783	0.0500
80	0.2663	0.5000	0.2000	0.0337	1.9205	0.0500
90	0.2654	0.5000	0.2000	0.0346	1.8756	0.0500
100	0.2647	0.5000	0.2000	0.0353	1.8396	0.0500
200	0.2614	0.5000	0.2000	0.0386	1.6770	0.0500
300	0.2602	0.5000	0.2000	0.0398	1.6226	0.0500
400	0.2596	0.5000	0.2000	0.0404	1.5953	0.0500
500	0.2592	0.5000	0.2000	0.0408	1.5790	0.0500
600	0.2589	0.5000	0.2000	0.0411	1.5681	0.0500
700	0.2588	0.5000	0.2000	0.0412	1.5603	0.0500
800	0.2586	0.5000	0.2000	0.0414	1.5544	0.0500
900	0.2585	0.5000	0.2000	0.0415	1.5499	0.0500
1000	0.2584	0.5000	0.2000	0.0416	1.5462	0.0500
2000	0.2581	0.5000	0.2000	0.0419	1.5298	0.0500
3000	0.2579	0.5000	0.2000	0.0421	1.5244	0.0500
4000	0.2579	0.5000	0.2000	0.0421	1.5216	0.0500
5000	0.2578	0.5000	0.2000	0.0422	1.5200	0.0500
6000	0.2578	0.5000	0.2000	0.0422	1.5189	0.0500
7000	0.2578	0.5000	0.2000	0.0422	1.5181	0.0500
8000	0.2578	0.5000	0.2000	0.0422	1.5175	0.0500
9000	0.2578	0.5000	0.2000	0.0422	1.5171	0.0500

Table 5: Effect of vacation rate over the system for $\lambda_1 = 10$, $\lambda_2 = 5$, $\mu_1 = 20$, $\mu_2 = 25$, $\sigma = 100$ and $k = 6$ and various values of α

α	P_0	P_1	P_2	P_3	MNCO	MPQL
10	0.1243	0.5000	0.2000	0.1757	2.2656	0.0500
20	0.1741	0.5000	0.2000	0.1259	2.0263	0.0500
30	0.2035	0.5000	0.2000	0.0965	1.9406	0.0500
40	0.2223	0.5000	0.2000	0.0777	1.9001	0.0500
50	0.2351	0.5000	0.2000	0.0649	1.8776	0.0500
60	0.2444	0.5000	0.2000	0.0556	1.8637	0.0500
70	0.2513	0.5000	0.2000	0.0487	1.8544	0.0500
80	0.2568	0.5000	0.2000	0.0432	1.8479	0.0500
90	0.2612	0.5000	0.2000	0.0388	1.8432	0.0500
100	0.2647	0.5000	0.2000	0.0353	1.8395	0.0500
200	0.2817	0.5000	0.2000	0.0183	1.8257	0.0500
300	0.2876	0.5000	0.2000	0.0124	1.8221	0.0500
400	0.2907	0.5000	0.2000	0.0093	1.8205	0.0500
500	0.2925	0.5000	0.2000	0.0075	1.8196	0.0500
600	0.2938	0.5000	0.2000	0.0063	1.8190	0.0500
700	0.2946	0.5000	0.2000	0.0054	1.8186	0.0500
800	0.2953	0.5000	0.2000	0.0047	1.8183	0.0500
900	0.2958	0.5000	0.2000	0.0042	1.8181	0.0500
1000	0.2962	0.5000	0.2000	0.0038	1.8180	0.0500
2000	0.2981	0.5000	0.2000	0.0019	1.8172	0.0500
3000	0.2987	0.5000	0.2000	0.0013	1.8170	0.0500
4000	0.2991	0.5000	0.2000	0.0009	1.8169	0.0500
5000	0.2992	0.5000	0.2000	0.0008	1.8168	0.0500
6000	0.2994	0.5000	0.2000	0.0006	1.8168	0.0500
7000	0.2995	0.5000	0.2000	0.0005	1.8168	0.0500
8000	0.2995	0.5000	0.2000	0.0005	1.8167	0.0500
9000	0.2996	0.5000	0.2000	0.0004	1.8167	0.0500

11. Graphical Study

Figures 1, 2 and 3 show the effect of retrial rate σ over the Mean number of customers in the orbit. The following figures show that the mean number of customers in the orbit decreases as retrial rate σ increases and this model becomes single server classical queueing system with vacation interruptions under pre-emptive priority service if σ is large.

Figure 1: Mean number of low priority customers in the orbit for $\lambda_1 = 10$, $\lambda_2 = 5$, $\mu_1 = 20$, $\mu_2 = 25$, $\alpha = 100$, $k = 6$ and σ various from 10 to 90

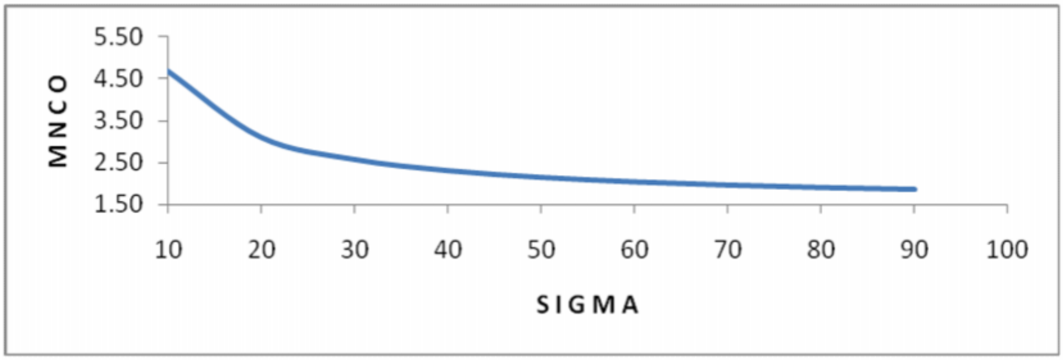


Figure 2: Mean number of low priority customers in the orbit for $\lambda_1 = 10$, $\lambda_2 = 5$, $\mu_1 = 20$, $\mu_2 = 25$, $\alpha = 100$, k = 4 and σ various from 100 to 900

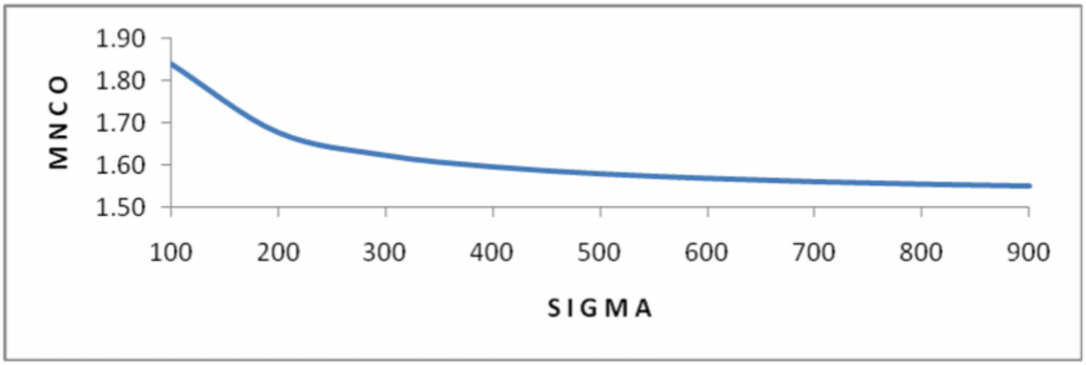
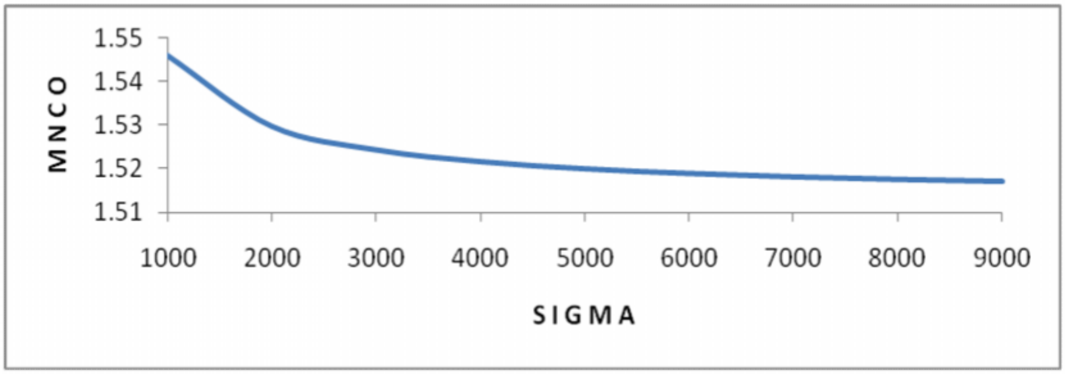


Figure 3: Mean number of low priority customers in the orbit for $\lambda_1 = 10$, $\lambda_2 = 5$, $\mu_1 = 20$, $\mu_2 = 25$, $\alpha = 100$, k = 4 and σ various from 1000 to 9000



Figures 4, 5 and 6 show the effect of vacation rate α over system. The following figures show that the mean number of customers in the orbit decreases as vacation rate α increases and this model becomes single server Retrial queueing system with pre-emptive priority service if α is large.

Figure 4: Mean number of low priority customers in the orbit for $\lambda_1 = 10$, $\lambda_2 = 5$, $\mu_1 = 20$, $\mu_2 = 25$, $\sigma = 100$, k = 6 and α various from 10 to 90

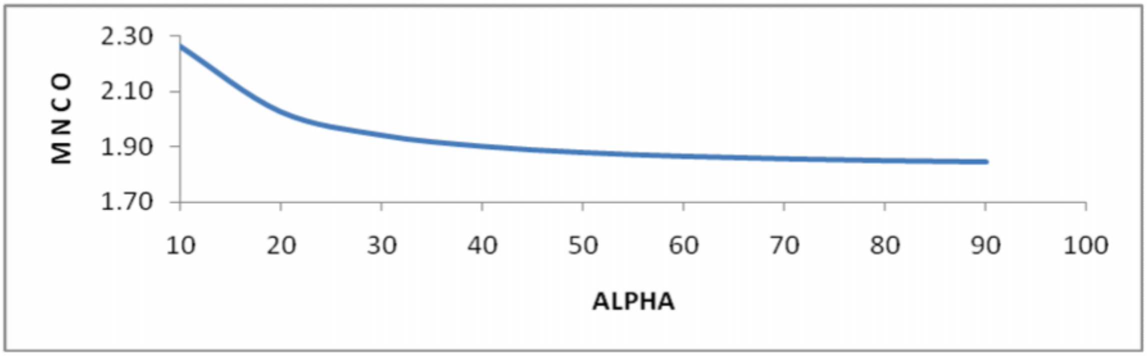


Figure 5: Mean number of low priority customers in the orbit for λ_1 = 10, λ_2 = 5, μ_1 = 20, μ_2 = 25, σ = 100, k = 6 and α various from 100 to 900

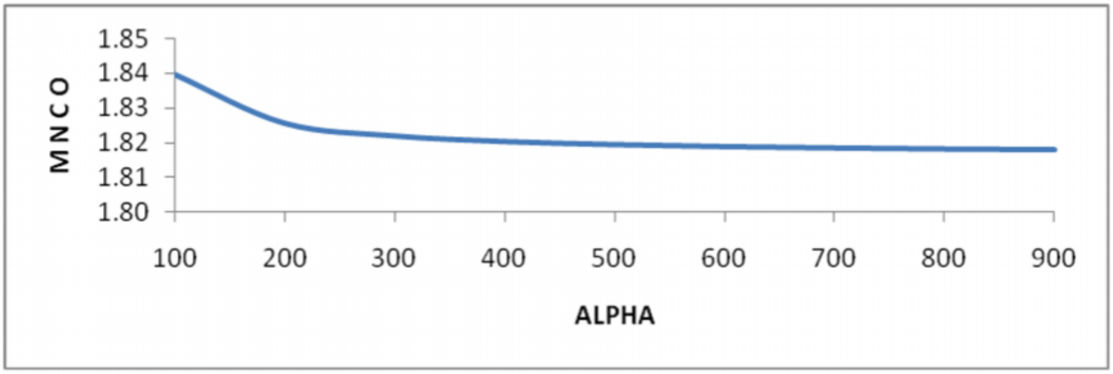
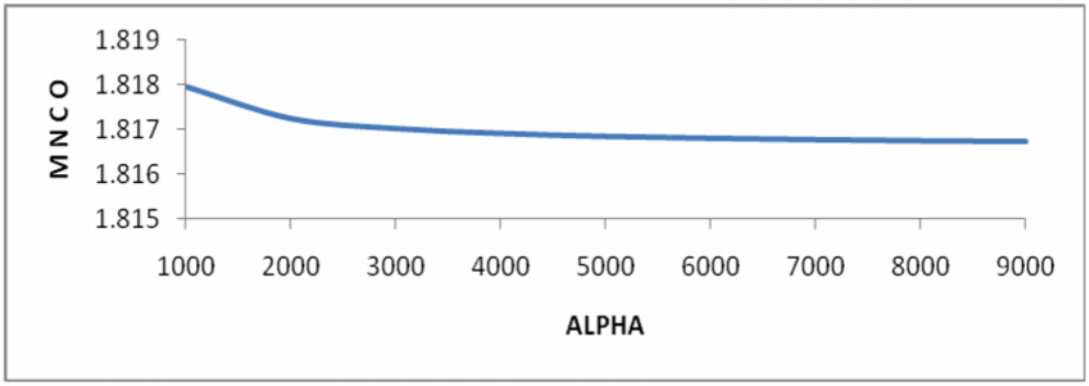


Figure 6: Mean number of low priority customers in the orbit for λ_1 = 10, λ_2 = 5, μ_1 = 20, μ_2 = 25, σ = 100, k = 6 and α various from 1000 to 9000



12. Conclusion

It is observed from numerical study that Mean number of low priority customers in the orbit decreases as the retrial rate increases, the probabilities for the server being busy with low and high priority customers are independent on retrial rate. The impact of vacation rate over the system implies that as the vacation rate increases this model becomes retrial queueing system with pre-emptive priority service. This paper can further be extended by introducing various parameters like different types of vacation policies and finite source input.

References

1. Artalejo, J.R. (1999*b*). A classified bibliography of research on retrial queues: Progress in 1990-1999. *Top*, 7, 187-211.

2. Artalejo, J.R. (1995). A queueing system with returning customers and waiting line. *Operations* Research *Letters* 17, 191-199.

3. Choi B.D. and Y. Chang. *Single server retrial queues with priority calls*, Mathematical and Computer Modelling, 30, No. 3-4, pp.7-32, 1999.

4. Dequan Yue, Chunyan and Wuyi Yue (2006), The Matrix Geometric solution of the M/E_k/1 queue with balking and state dependent service, Nonlinear Dynamics and Systems Theory, 6(3) (2006) 295–308.

5. Diamond, J.E. and A.S. Alfa (1995). Matrix analytical methods for M/PH/1 retrial queues. *Stochastic Models* 11,447-470.

6. Falin, G.I. (1990). A survey of retrial queues. *Queueing Systems 7,* 127-167.

7. Falin, G.I. and J.G.C. Templeton (1997). *Retrial Queues.* Chapman and Hall, London.

8. Falin, G.I. J.R. Artalejo and M. Martin (1993). On the single server retrial queue with priority customers. *Queueing Systems* 14, 439-455.

9. G. Ayyappan, A. Muthu Ganapathi Subramanian and Gopal sekar (2009). Single server Retrial queueing system with Pre-emptive priority service – Proceedings of International Conference on Mathematical Methods and Computations, Tiruchirapalli, India and published in the Proceedings of the above Conference, *pp 328-346, Allied Publications, India.*

10. Gomez-Corral. A. *A Bibliographical guide to the analysis of retrial queues through matrix analytic technique,* Annals of Operationas, 141, pp. 163-191, 2006.

11. Jihong Li and Naishuo TiaN (2007), The M/M/1 queue with working vacations and vacation interruptions, Journal of Systems Science and Systems Engineering, Vol. 16, No. 1, March 2007.

12. Marcel F. Neuts, Matrix Geometric Solutions in Stochastic Models an algorithmic approach.

13. Neuts, M.F. and B.M. Rao (1990). Numerical investigation of a multiserver retrial model. *Queueing Systems* 7, 169-190.

14. Tian van do and Ram Chakka (2010), An efficient method to compute the rate matrix for retrial queues with large number of servers, Applied Mathematics Letters, Vol. 23, issue 5, may 2010, pp 636-645.