SCIREA Journal of Information Science and Systems Science
http://www.scirea.org/journal/ISSS
February 12, 2022
Volume 6, Issue 1, February 2022
https://doi.org/10.54647/isss12197

Simulation Models of Modified Multiple Vacation Policy for Multi-Server Queuing Systems

Zhernovyi Yuriy
Faculty of Mechanics and Mathematics, Ivan Franko National University of Lviv, Ukraine
Email: yuriy.zhernovyy@lnu.edu.ua

Abstract

We consider a multi-server vacation queueing system that operates in the modified multiple vacation policy related to the number of busy servers. The condition for the transition to vacation mode at the time of the release of the server is the lack of customers in the queue and the implementation of one of the restrictions on the number of servers currently in use by customers (c_U -condition): $c_U < m$, where $m = \{1, ..., c\}$ and c is the number of servers. When the server returns from vacation it observes the following rule. If there is at least one customer in the queue, the server commences service and serves exhaustively before taking another vacation. If the server finds that there is no queue and the c_U -condition is satisfied, then it takes another vacation. Using GPSS World simulation models, we studied the dependencies of the system performance measures on the following parameters: the load factor, coefficient of variation of the inter-arrival times, value of c_U , duration of vacations, and number of servers. We tested the simulation models by comparing the results with the known ones obtained by an analytical method. The value of the simulation time, which makes it possible to obtain results corresponding to the stationary process, is determined. The results for systems with and without vacations are compared.

Keywords: *queueing system, vacations, modified multiple vacation policy, simulation model, GPSS World*

1. Introduction

A vacation queueing system is one in which a server may become unavailable for a random period of time from a primary service center. The time away from the primary service center is called a vacation, and customers who arrive while the server is on vacation will have to wait until he returns from vacation. A vacation can be the result of many factors. Dynamically developing the branch of wireless communication (Wi-Fi, LTE, sensor networks, etc.) forces the search for solutions that, in addition to modeling the packet processing itself, ensure the implementation of tasks resulting from the need for energy saving. In the field of modeling the energy-saving mode in wireless network nodes, queueing systems with various types of mechanisms of the periodic suspension of customer processing are used.

In some cases, the vacation can be the result of a server breakdown, which means that the system must be repaired and brought back to service. It can also be a deliberate action taken to utilize the server in a secondary service center when there are no customers present at the primary service center. Thus, server vacations are useful for those systems in which the server's idle time is utilized for other purposes, and this makes the queueing model to be applicable to a variety of real-world stochastic service systems.

There are different types of vacation queueing systems. In the *single vacation policy*, the server takes a vacation of a random duration when the queue is empty. At the end of the vacation, the server returns to the queueing system. If there is at least one customer waiting when the server returns from vacation, the server performs one of the following actions depending on the service policy:

a) Under the *exhaustive service policy*, the server will serve all waiting customers as well as those that arrive while he is still serving at the station. He takes another vacation when the queue becomes empty.

b) Under the *gated service policy*, the server will serve only those customers that he finds at the queue upon his return from vacation. At the end of their service, the server will commence another vacation and any customers that arrive while the server is already serving at the station will be served when the server returns from the vacation.

c) Under the *limited service policy*, the server will serve only a predefined maximum number of customers and then will commence another vacation. The single-service policy in which exactly one customer is served is a special type of this policy.

If the queue is empty on the server's return, the server waits to complete a busy period using one of the above service policies before taking another vacation.

In the *multiple vacation policy*, if the server returns from a vacation and finds the queue empty, he immediately commences another vacation. If there is at least one waiting customer, then he will commence service according to the prevailing service policy.

Queueing systems with server vacations have attracted the attention of many researchers since the idea was first discussed in the paper of Levy and Yechiali [1]. One can find a comprehensive review of many important analytical results for different vacation queueing models, for example, in [2-4].

Among many solutions, it is worth mentioning the *N*-policy threshold-type discipline, originally proposed in [5], in which the processing restarts after an idle period simultaneously with the *N*th arrival, and the *T*-policy, introduced in [6], in which the server is being activated exactly *T* time units after the last busy period. Moreover, in a fundamental paper [7], single and multiple vacation policies were proposed in which single or repeated independent vacations are taken until at least one customer present in the buffer is detected.

In the article, we consider the queueing model with multiple vacation periods launched after the completion of each busy period of the system. Queueing systems of this type can be used in the practical modeling of the functioning of, for example, computer and telecommunication network nodes (in particular wireless, e.g., based on Wi-Fi or LTE standards), in which an energy saving mechanism is implemented based on the cyclical checking the state of queue of packets waiting for processing). A multiple vacation period may also be used to perform some other operations for the service station (such as e.g., maintenance, software update, etc.). As one can observe, the literature on this type of model is extensive and constantly growing. However, the analytical results relate exclusively to the $M/G/1$-type queue or $M/M/c$ queue.

We consider a multi-server vacation queueing system that operates in the modified multiple vacation policy, which we agree to call c_U -policy. The condition for the transition to vacation mode at the time of the release of the server is the lack of customers in the queue (*Q*=0) and the implementation of one of the restrictions on the number of servers currently in use by customers (c_U -condition): $c_U < m,$ where $m = \{1,...,c\},$ and *c* is the number of servers. Since at this point in time *Q*=0, then $c_U = n,$ where *n* is the number of customers in the system. When the server returns from vacation it observes the following rule. If there is at least one customer in the queue, the server commences service and serves exhaustively before taking another vacation. If the server finds that there is no queue and the c_U -condition is satisfied, then it takes another vacation. This modification of the standard multiple vacation policy allows improving system performance measures and at the same time takes advantage of the use of vacations.

One of the methods for studying queuing systems is the simulation method, when the model simulates the operation of a real system, that is, the model reproduces the process of functioning of a real system in time. In many cases, simulation becomes the most effective and often practically the only available method for studying systems. For example, an efficient analysis of a *G*/*G*/*c* type queuing system by analytical methods is impossible, while such an analysis using simulation methods is not particularly difficult. In this paper, we use the GPSS World simulation system [8, 9].

GPSS (General-Purpose Simulation System) is a general process-oriented simulation software environment. GPSS World is a Microsoft Windows application designed to run on various Windows operating systems.

The main contributions of this paper are as follows.

1) We propose a modification of the multiple vacation policy (c_U -policy), which allows more flexible use of vacations, taking into account the state of the system at the time of the transition of each server to the vacations mode. Analytical results exist only for the *M*/*M*/*c* system in the case of the standard multiple vacation policy when c_U -condition has the form $c_U < c$ [10].

2) We construct the GPSS World simulation model of this multiple vacation c_U -policy for the *G*/*G*/*c* queueing system, which allows us to study the dependencies of the system performance measures on the following parameters: the load factor, coefficient of variation of inter-arrival

times, value of c_U, duration of vacations, number of servers, and so on. Using the constructed simulation models, we have the opportunity to compare the results for systems with and without vacations.

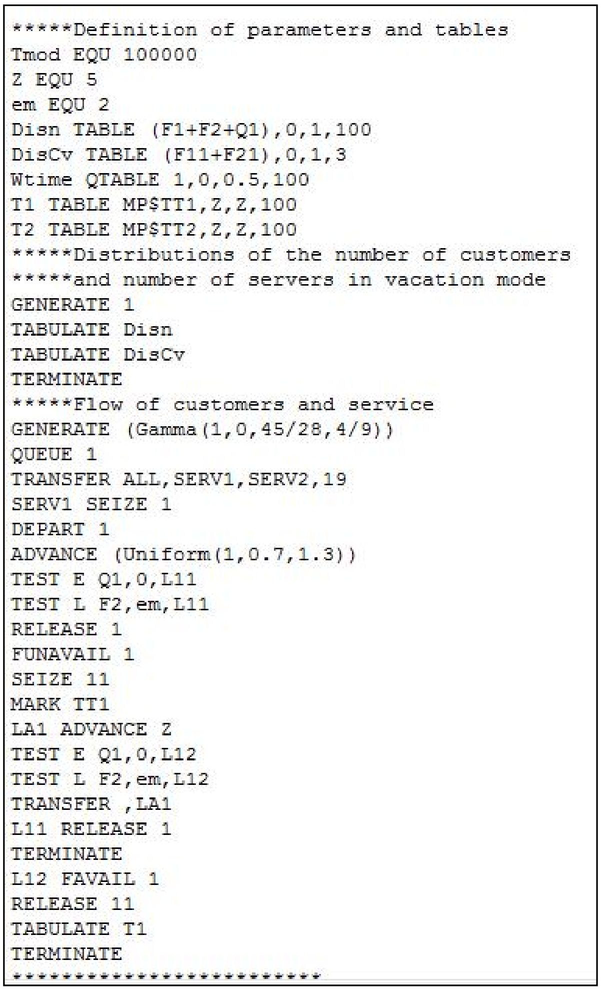
3) Using the constructed simulation models, we have the opportunity to obtain not only the average values of the system performance measures but also the distributions of all performance measures and with the help of graphic visualization to trace the behavior of these measures over time.

2. Simulation Models

2.1. Vacation System with c_U -Policy

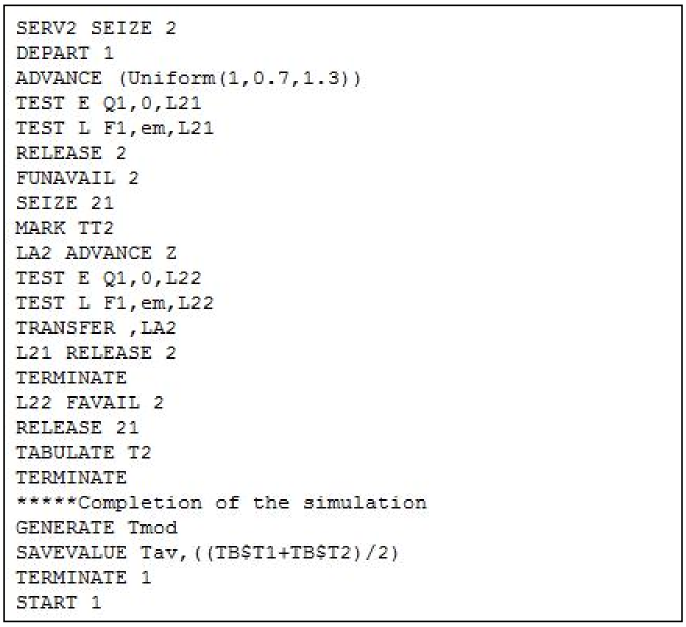
We consider a *G*/*G*/*c* queueing system in which both service times and the inter-arrival times have any given distributions. A random variable *X*, the time to serve a customer, has a general distribution with a cumulative distribution function (CDF) $F_X(x)$, and $F_Y(x)$ is CDF of inter-arrival time *Y*. A random variable *Z*, the duration of a vacation, is also assumed to have a general distribution with CDF $F_Z(x)$. We denote as $E(X)$, $E(Y)$, and $E(Z)$ the mean of the random variables $X, Y,$ and $Z,$ respectively. We assume that the service is organized according to the natural FIFO discipline. The considered queueing system is governed by the multiple vacation policy with c_U -condition. This is an additional condition for the server to switch to the vacations mode (or continue this mode) at the moment of its release (or the end of the previous vacation) when the queue is empty.

Figures 1 and 2 show a corresponding GPSS simulation model in the case when $c = m = 2,$ the random variables X and Y have the uniform and gamma distributions, respectively, and *Z*, the duration of a vacation is constant, $Z = 5.$ In this model, we assume that $E(X) = 1,$ the random variable X is uniformly distributed on the interval $[0.7, 1.3]$, the parameters of the gamma distribution of the random variable Y are as follows: $\alpha = 4/9, \beta = 45/28.$



```
*****Definition of parameters and tables
Tmod EQU 100000
Z EQU 5
em EQU 2
Disn TABLE (F1+F2+Q1),0,1,100
DisCv TABLE (F11+F21),0,1,3
Wtime QTABLE 1,0,0.5,100
T1 TABLE MP$TT1,Z,Z,100
T2 TABLE MP$TT2,Z,Z,100
*****Distributions of the number of customers
*****and number of servers in vacation mode
GENERATE 1
TABULATE Disn
TABULATE DisCv
TERMINATE
*****Flow of customers and service
GENERATE (Gamma(1,0,45/28,4/9))
QUEUE 1
TRANSFER ALL,SERV1,SERV2,19
SERV1 SEIZE 1
DEPART 1
ADVANCE (Uniform(1,0.7,1.3))
TEST E Q1,0,L11
TEST L F2,em,L11
RELEASE 1
FUNAVAIL 1
SEIZE 11
MARK TT1
LA1 ADVANCE Z
TEST E Q1,0,L12
TEST L F2,em,L12
TRANSFER ,LA1
L11 RELEASE 1
TERMINATE
L12 FAVAIL 1
RELEASE 11
TABULATE T1
TERMINATE
*************************
```

Figure 1. GPSS simulation model for the G/G/2 system with multiple vacation c_U-policy (Part 1)



```
SERV2 SEIZE 2
DEPART 1
ADVANCE (Uniform(1,0.7,1.3))
TEST E Q1,0,L21
TEST L F1,em,L21
RELEASE 2
FUNAVAIL 2
SEIZE 21
MARK TT2
LA2 ADVANCE Z
TEST E Q1,0,L22
TEST L F1,em,L22
TRANSFER ,LA2
L21 RELEASE 2
TERMINATE
L22 FAVAIL 2
RELEASE 21
TABULATE T2
TERMINATE
*****Completion of the simulation
GENERATE Tmod
SAVEVALUE Tav,((TB$T1+TB$T2)/2)
TERMINATE 1
START 1
```

Figure 2. GPSS simulation model for the G/G/2 system with multiple vacation c_U-policy (Part 2)

Let us denote for the random variable Y the probability density function, variance and coefficient of variation as $f_Y(t)$, $D(Y)$, and V, respectively, then for the gamma distribution, we have

$$f_Y(t) = \frac{t^{\alpha-1}}{\beta^{\alpha}\Gamma(\alpha)} e^{-\frac{t}{\beta}}, \quad \alpha > 0, \quad \beta > 0, \quad t \geq 0, \quad \Gamma(t) = \int_0^{\infty} x^{t-1} e^{-x} dx;$$

$$E(Y) = \alpha\beta, \qquad D(Y) = \alpha\beta^2, \qquad V = \frac{\sqrt{D(Y)}}{E(Y)} = \frac{1}{\sqrt{\alpha}}.$$

The simulation model is organized into three code groups, composed of a sequence of blocks and called block segments. Block segments generally start with a GENERATE block that inserts transactions into the simulation model and ends with a TERMINATE block that removes transactions from the simulation model. Such a block segment specifies a process, i.e., a life cycle, for transactions.

The model begins with the setting of the parameters and tables of distributions of those random variables that we plan to obtain as a result of the simulation. We use tables to obtain distributions of the number of customers in the system, the number of servers in vacation mode, waiting times in the queue, and vacation durations for each server. The first segment is designed to obtain tables of the distributions of the number of customers in the system and the number of servers in vacation mode. The second segment forms the basis of the model and is designed to simulate the process of arrival and service of customers. The third segment sets a simulation time, saves a value of the average duration of vacations over two servers, and stops the simulation process.

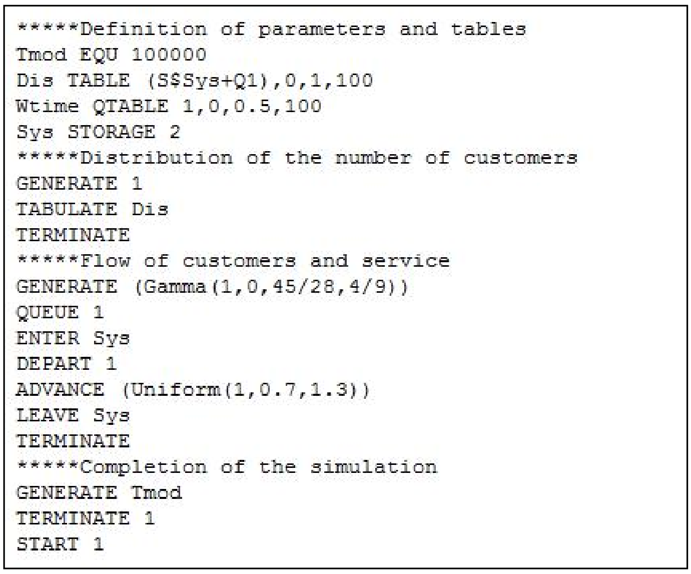
The main segment of the model consists of two sequentially located Facility Entities, each of which specifies the distribution of its service time in the corresponding ADVANCE block. The TRANSFER ALL block sequentially checks the possibility for an active transaction to enter each block located between labels SERV1 and SERV2 at every nineteenth below location. Thus, the possibility to enter each of the blocks SEIZE 1 and SEIZE 2, is tested. As a result, the customer enters the first free server. The difference between the numbers of the successive located SEIZE blocks is equal to 19. We can easily increase the number of sequentially located Facility Entities in accordance with the number of system servers. The combination of blocks SEIZE and RELEASE provides the operation of each Facility Entity. We use the FUNAVAIL and FAVAIL blocks to model the vacations of each server. In order to obtain statistical data associated with vacations, we introduce additional Facility Entities for each server using double-numbered blocks SEIZE and RELEASE. QUEUE and DEPART blocks are used to update the statistics associated with a queue. We use TEST blocks to organize a modified vacation policy related to the number of busy servers at the time the vacation mode is enabled. We find the duration of the vacations for each server using MARK and TABULATE blocks.

We can easily change in the model shown in Figures 1 and 2, not only the number of servers but also the type of distributions of random variables *X*, *Y*, and *Z*, which are set in blocks GENERATE and ADVANCE. We can also use different distributions of the random variables *Y* and *Z* for each server.

2.2. System without Vacations

We can obtain a simulation model for the system without vacations from the previous model if we substitute $m = 0$ (em EQU 0) in it. To reduce the calculation time, it is advisable to use the model shown in Figure 3.

In this model, a STORAGE command defines a Storage Entity named Sys with a total capacity of 2 units. By changing the capacity, we can set the desired value for the number of servers. Thus, we can define a multi-server queueing system, the functioning of which is provided by the ENTER and LEAVE blocks.



```
*****Definition of parameters and tables
Tmod EQU 100000
Dis TABLE (S$Sys+Q1),0,1,100
Wtime QTABLE 1,0,0.5,100
Sys STORAGE 2
*****Distribution of the number of customers
GENERATE 1
TABULATE Dis
TERMINATE
*****Flow of customers and service
GENERATE (Gamma(1,0,45/28,4/9))
QUEUE 1
ENTER Sys
DEPART 1
ADVANCE (Uniform(1,0.7,1.3))
LEAVE Sys
TERMINATE
*****Completion of the simulation
GENERATE Tmod
TERMINATE 1
START 1
```

Figure 3. GPSS simulation model for the G/G/2 system without vacations

2.3. Checking the Simulation Models and Choice the Simulation Time

Let us use analytical results for the *M*/*M*/*c* system in the case of the standard multiple vacation policy [10] to test the constructed simulation models and to choose the optimal value of the simulation time. We also use well-known solutions for the *M*/*M*/*c* system without vacations. The *M*/*M*/*c* system is a multi-server model with exponential inter-arrival times and service times. The duration of a vacation is also assumed in [10] to have the exponential distribution with mean $E(Z)$. The equality $E(Z)=0$ means that we are considering a system without vacations.

We denote by $N(t)=E\big(n(t)\big)$ a value of the average number of customers in the system in time t, then $N=N(\infty)$ denotes a stationary value of this indicator. We assume that the condition $\rho<1$ is satisfied, which ensures the existence of a stationary process of the number of customers in the system. Here $\rho=E(X)/\big(c\,E(Y)\big)$ is the load factor of the system.

Table 1 shows the comparison of the average number of customers in the *M*/*M*/*c* system obtained using simulation $\big(N(10^5)\big)$ and analytical models (N) for cases $c=5$ and $c=10$, for various values of ρ and $E(Z)$. The results are obtained for the simulation time $t_{\text{mod}}=10^5$. This time value seems to be optimal for further calculations from the point of view of the achieved accuracy of the results and the short time of model implementation.

We evaluate the accuracy of the results using the relative error calculated by the formula $\varDelta=\big(|\,N-N(10^5)\,|/N\big)\cdot 100\%$. According to Table 1, the relative error, $\varDelta$, does not exceed 4.07% and 1.44% for cases when $c=5$ and $c=10$, respectively.

Using the GPSS World tools, we are able to verify that the value of $N(t)$ approaches a stationary value N as time increases. Figure 4 shows this for the case when $c=10$, $\rho=0.6$, and $E(Z)=0.2$ on the time interval [0, 1000].

The GPSS World uses random number generators to sample random numbers for GENERATE and ADVANCE blocks. We can select which random number generator number is to be used as the source of the random number. The results obtained for different values of the random number generator may differ slightly from each other. In this work, we use the number of the random number generator, which is equal to 1.

Table 1. Comparison of the average number of customers in the *M*/*M*/*c* system obtained using simulation and analytical models

ρ	$E(Z)$	$c=5$			$c=10$		
		$N(10^5)$	N	Δ, %	$N(10^5)$	N	Δ, %
0.2	0	0.998	1.001	0.30	2.000	2.000	0.00
0.6	0	3.350	3.354	0.13	6.151	6.152	0.02
0.9	0	11.220	11.362	1.25	15.235	15.019	1.44
0.2	0.2	1.052	1.054	0.19	2.050	2.052	0.09
0.6	0.2	3.616	3.628	0.33	6.461	6.479	0.28
0.9	0.2	11.392	11.875	4.07	16.056	15.768	0.70
0.2	1	1.248	1.250	0.16	2.256	2.250	0.27
0.6	1	4.484	4.500	0.36	7.496	7.500	0.36
0.9	1	13.276	13.500	1.62	18.117	18.000	0.65
0.2	2	1.471	1.450	1.45	2.481	2.488	0.28
0.6	2	5.434	5.450	0.29	8.559	8.598	0.45
0.9	2	15.116	15.314	1.29	20.686	20.417	1.32

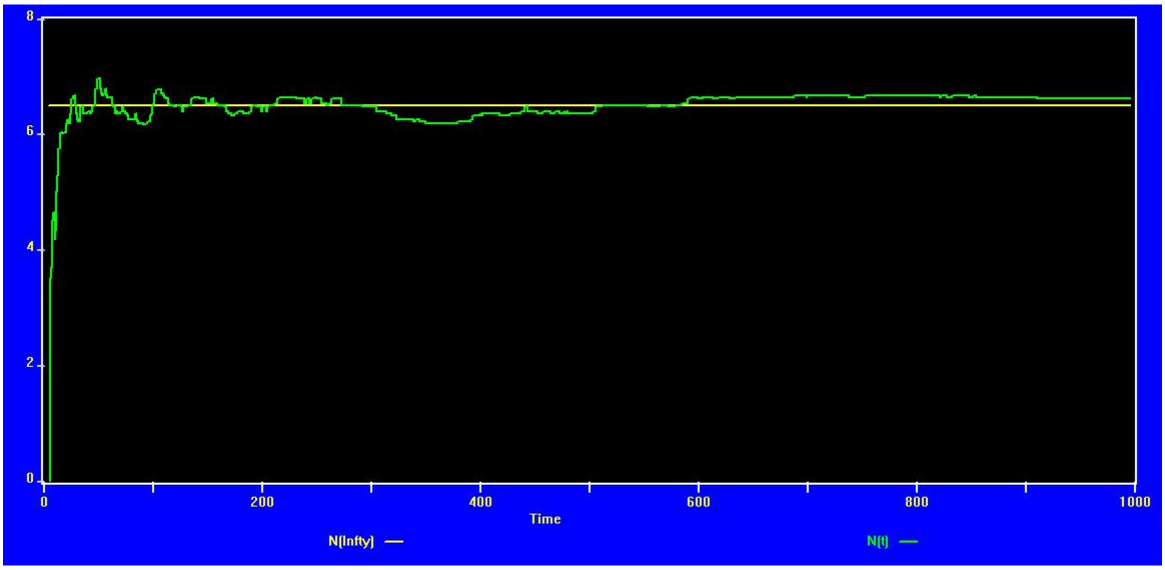


Figure 4. The approach of $N(t)$ to a stationary value N in the case when $c=10$, $\rho=0.6$, and $E(Z)=0.2$

3. Study of the Multiple Vacation Policy

3.1. Basic Designations

We consider the case when the random variables X and Y have the uniform and gamma distributions, respectively, and Z, the duration of a vacation, is constant, therefore $E(Z) = Z.$ We assume that $E(X) = 1,$ and the random variable X is uniformly distributed on the interval $[0.7, 1.3]$. Let us use the notation N, W and Z_k for stationary values of the average number of customers in the system, average waiting time in the queue, and average duration of multiple vacations for the kth server, respectively. Then

$$Z_{\text{av}} = \frac{1}{c} \cdot \sum_{k=1}^{c} Z_k$$

is the average duration of multiple vacations for the system as a whole. We denote by T_k the total time spent by the kth server in the vacation mode on the time interval $[0, t_{\text{mod}}],$ then

$$T_U = \frac{1}{c\,t_{\text{mod}}} \cdot \sum_{k=1}^{c} T_k$$

is the average relative time spent by the system server in the vacation mode. Let c_V denote the number of servers in vacation mode. We use the same simulation time $t_{\text{mod}} = 10^5$ in all calculations.

3.2. Dependencies of the System Performance Measures on *Z* for the Vacation c_U -Policy

Let us consider the case when $c = 5,$ $V = 1.5,$ and $\rho = 0.7.$ Figures 5−7 show dependencies of values of $W, Z_{\text{av}},$ and T_U , respectively, on Z, the duration of a vacation, for various *m* in the c_U -condition $c_U < m.$ The graphs show that W and Z_{av} are increasing approximately linear functions of Z. We have $T_U = 1 - \rho$ for the case of condition $c_U < c$ (see the line $T_U = 0.3$ in Figure 7) and T_U as an increasing function of Z for all other values of *m*. The values of $W, Z_{\text{av}},$ and T_U increase together with *m* for a fixed value of Z. If $Z = 0,$ then we have the value of W corresponding to the system without vacations.

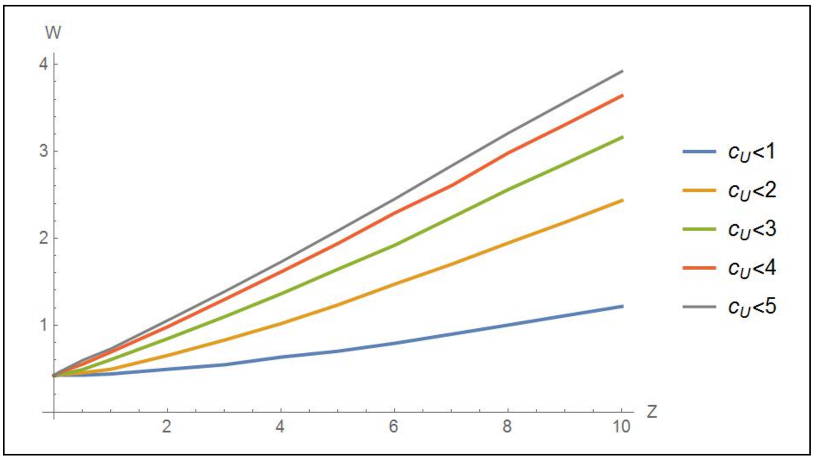


Figure 5. W **as a function of** Z **in the case when** $c = 5,\ V = 1.5,$ **and** $\rho = 0.7$

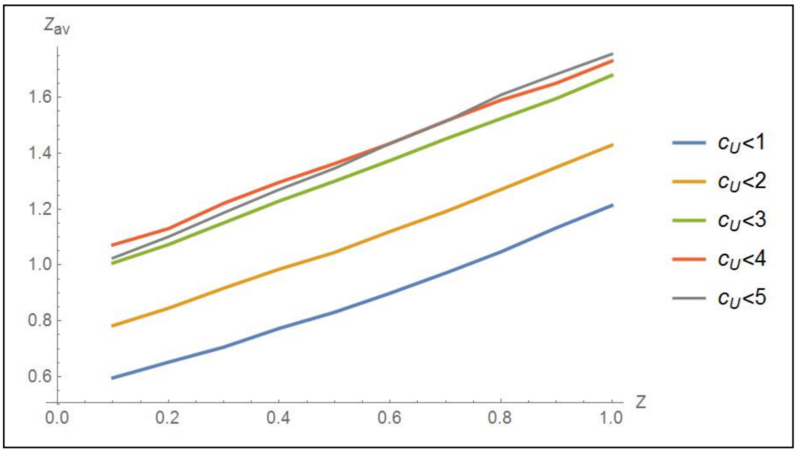


Figure 6. Z_{av} **as a function of** Z **in the case when** $c = 5,\ V = 1.5,$ **and** $\rho = 0.7$

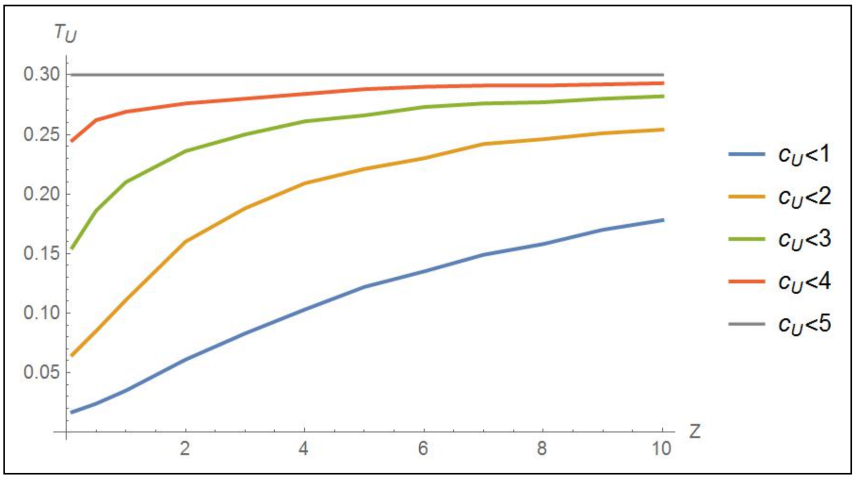


Figure 7. T_U **as a function of** Z **in the case when** $c = 5,\ V = 1.5,$ **and** $\rho = 0.7$

3.3. Dependencies of the System Performance Measures on ρ for the Vacation c_U-Policy

Let us consider the case when $c = 5$, $V = 1.5$, and $Z = 5$. Figures 8−11 show dependencies of values of W, Z_{av}, T_U, and $P\{Q = 0\}$, respectively, on ρ, the load factor, for various m in the c_U-condition $c_U < m$. Here $P\{Q = 0\}$ is the statistical probability of the absence of a queue at the moment of arrival of a customer. The dependencies for W and $P\{Q = 0\}$ on ρ are also considered for the system without vacations.

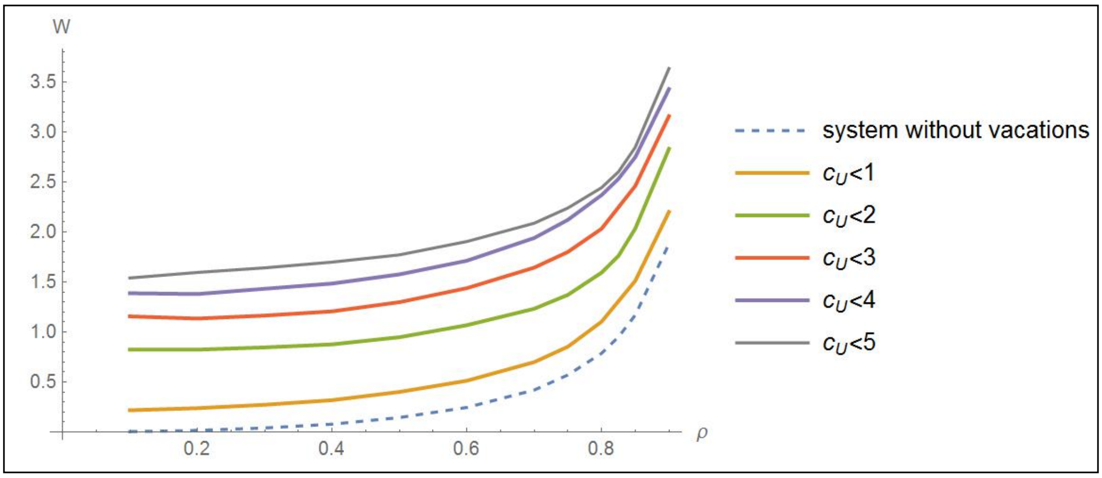


Figure 8. W **as a function of** ρ **in the case when** $c = 5$, $V = 1.5$, **and** $Z = 5$

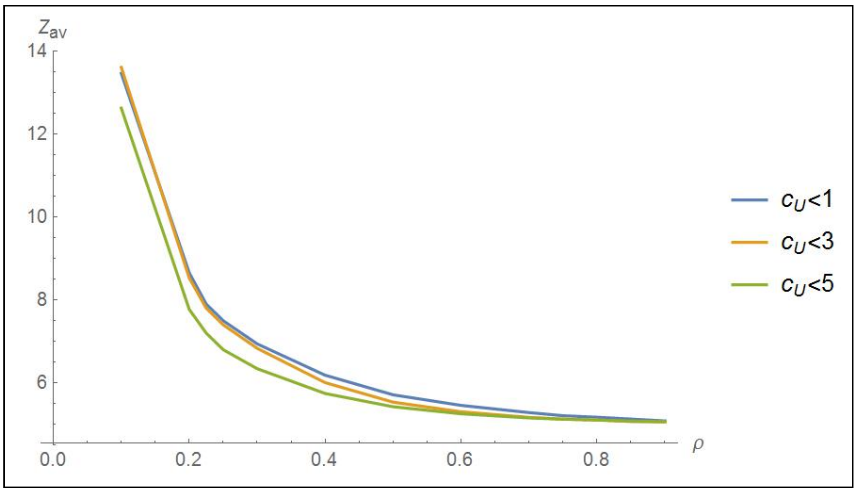


Figure 9. Z_{av} **as a function of** ρ **in the case when** $c = 5$, $V = 1.5$, **and** $Z = 5$

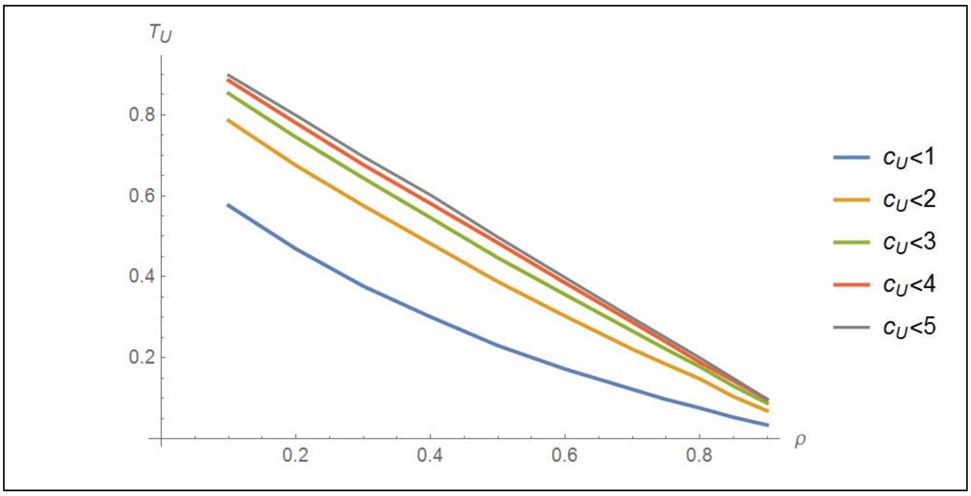


Figure 10. T_U **as a function of** ρ **in the case when** $c = 5,\ V = 1.5,$ **and** $Z = 5$

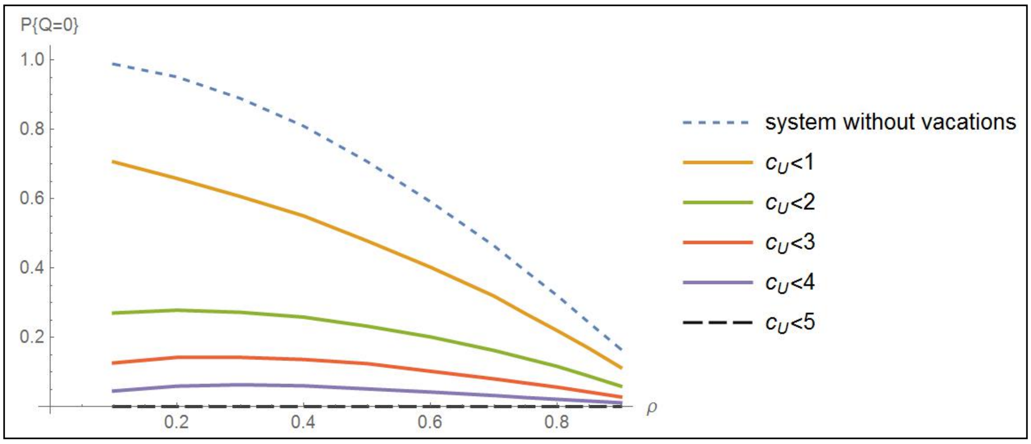


Figure 11. $P\{Q = 0\}$ **as a function of** ρ **in the case when** $c = 5,\ V = 1.5,$ **and** $Z = 5$

The graphs show that T_U and $P\{Q=0\}$ are approximately linear functions of ρ. In the case of condition $c_U < c$ (when the standard multiple vacation policy is enforced), only the first arrival to each server finds an empty queue, so the value of $P\{Q=0\}$ is practically zero. The average waiting time, W, is increasing function of ρ, and other dependencies are decreasing functions. The values of Z_{av} change slightly when the c_U-condition changes for a fixed value of ρ. The values of W and T_U increase together with *m*, and the value of $P\{Q=0\}$ decreases as *m* increases for a fixed value of ρ. The values of W and $P\{Q=0\}$, when *m* decreases, approach the corresponding values for the system without vacations.

3.4. Dependencies of the System Performance Measures on *V* for the Vacation c_U -Policy

Let us consider the case when $c = 5,\ \rho = 0.7,$ and $Z = 5$. Figures 12−14 show dependencies of values of W, T_U, and $P\{Q=0\}$, respectively, on V, the coefficient of variation of inter-arrival time, for various m in the c_U-condition. The dependencies of W and $P\{Q=0\}$ on V are also considered for the system without vacations.

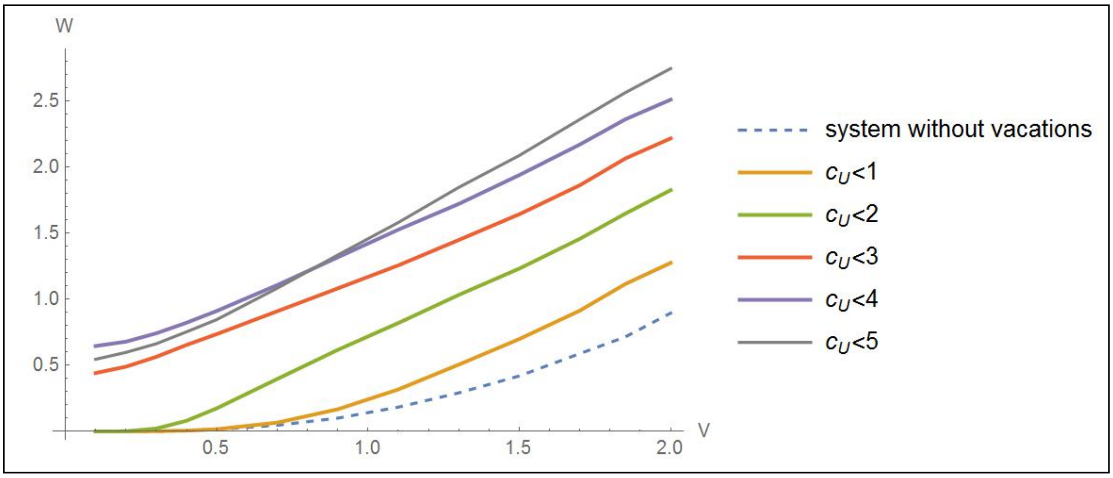


Figure 12. W **as a function of** V **in the case when** $c=5$, $\rho=0.7$, **and** $Z=5$

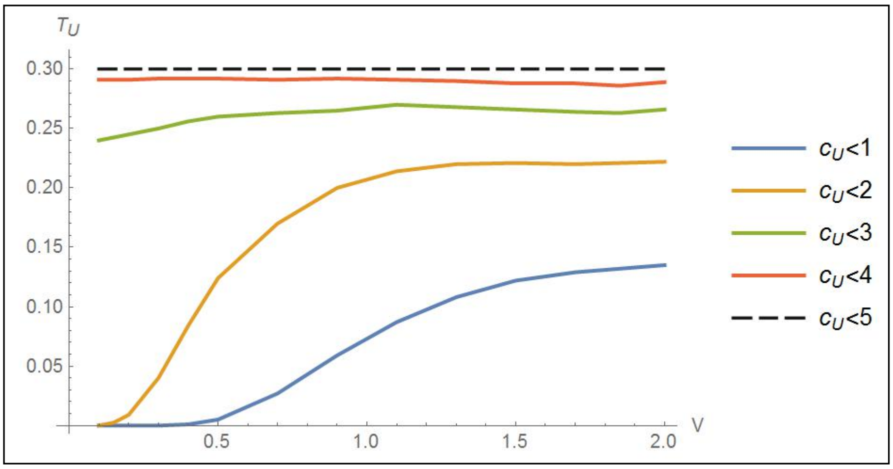


Figure 13. T_U **as a function of** V **in the case when** $c=5$, $\rho=0.7$, **and** $Z=5$

The graphs show that W is increasing approximately linear function of V. We have $T_U=1-\rho$ for the case of condition $c_U<c$ (see the dashed line in Figure 13) and T_U as an increasing function of V for $c_U<m$, $m=1$, 2. For other values of m, the values of T_U almost do not depend on V. In the case of condition $c_U<c$, the value of $P\{Q=0\}$ is practically zero. Probability $P\{Q=0\}$ decreases as a function of V and as a function of m for a fixed value of V, taking the largest value for the system without vacations. The values of W and T_U increase together with m for a fixed value of V. The smallest value of W is reached for the system without vacations.

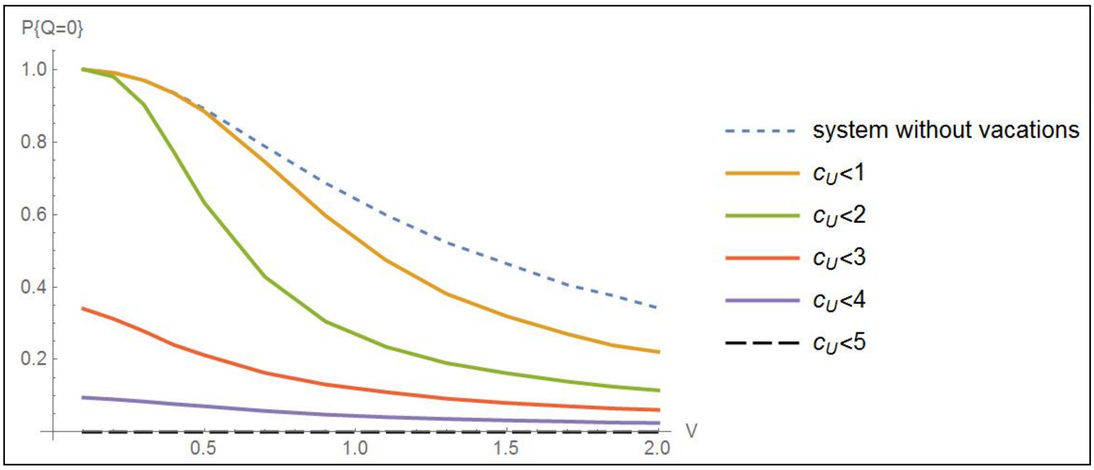


Figure 14. $P\{Q=0\}$ **as a function of** V **in the case when** $c=5$, $\rho=0.7$, **and** $Z=5$

3.5. Dependencies of the System Performance Measures on Z for Various Values of V

Let us consider the case when $c=5$, $\rho=0.7$, and $c_U<c$. Figures 15 and 16 show dependencies of values of N and Z_{av}, respectively, on Z for various V. The graphs show that N and Z_{av} are increasing approximately linear functions of Z. The values of N and Z_{av} increase together with V for a fixed value of Z.

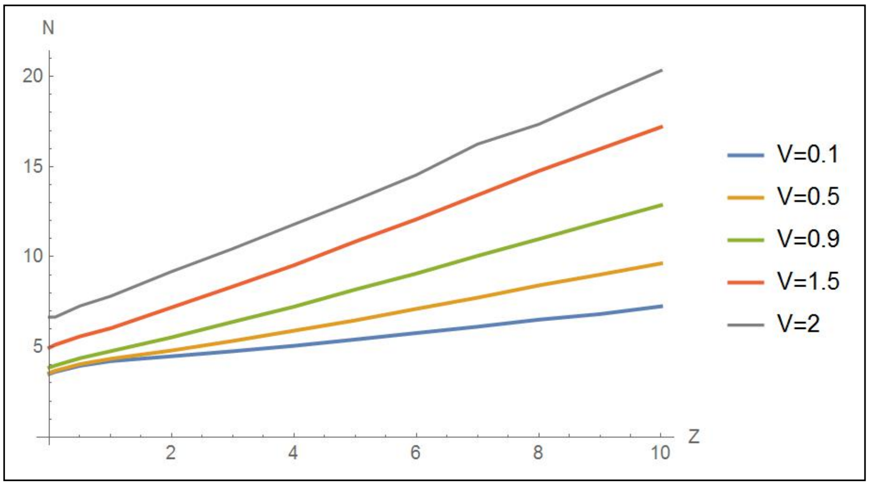


Figure 15. N **as a function of** Z **in the case when** $c=5$, $\rho=0.7$, **and** $c_U<c$

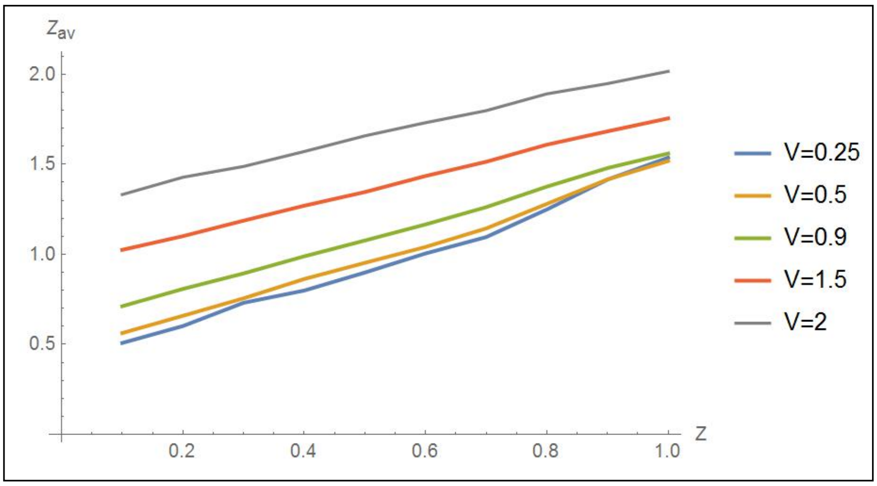


Figure 16. Z_{av} **as a function of** Z **in the case when** $c = 5,$ $\rho = 0.7,$ **and** $c_U < c$

3.6. Dependencies of the System Performance Measures on ρ for Various Values of V

Let us consider the case when $c = 5,$ $Z = 5,$ and $c_U < c.$ Figures 17−19 show dependencies of values of $N,$ $W,$ and $Z_{\text{av}},$ respectively, on ρ for various $V.$ The graphs show that N and W are increasing functions of $\rho.$ At the same time, Z_{av} decreases as a function of $\rho.$ For values of $V < 1,$ the value of W is almost independent of $\rho.$ The values of N and W increase together with V for a fixed value of $\rho.$ The values of Z_{av} are almost independent of $V.$

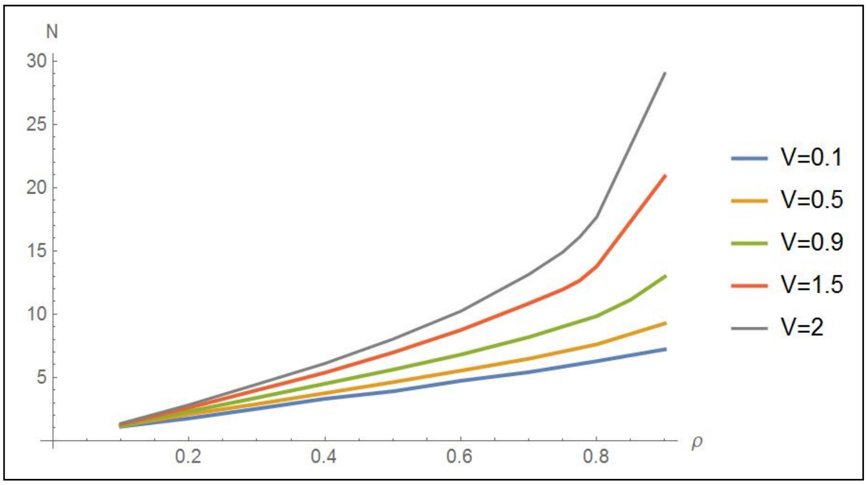


Figure 17. N **as a function of** ρ **in the case when** $c = 5,$ $Z = 5,$ **and** $c_U < c$

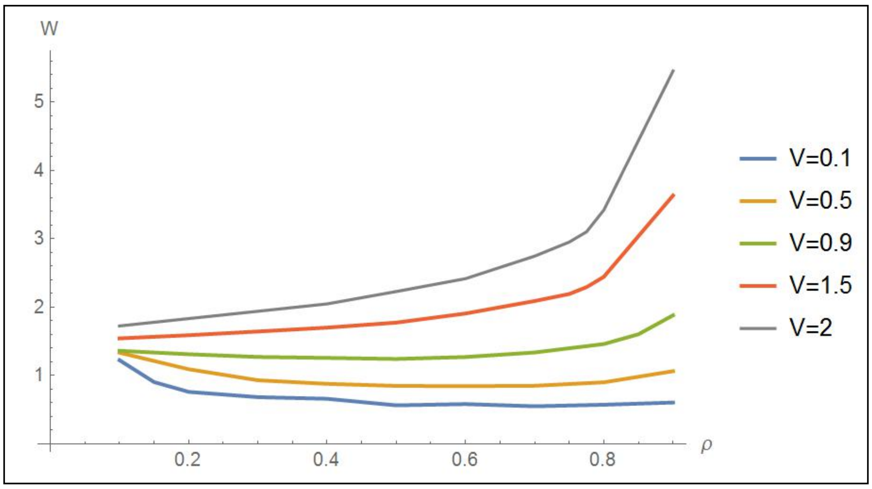


Figure 18. W **as a function of** ρ **in the case when** $c = 5,\ Z = 5,$ **and** $c_U < c$

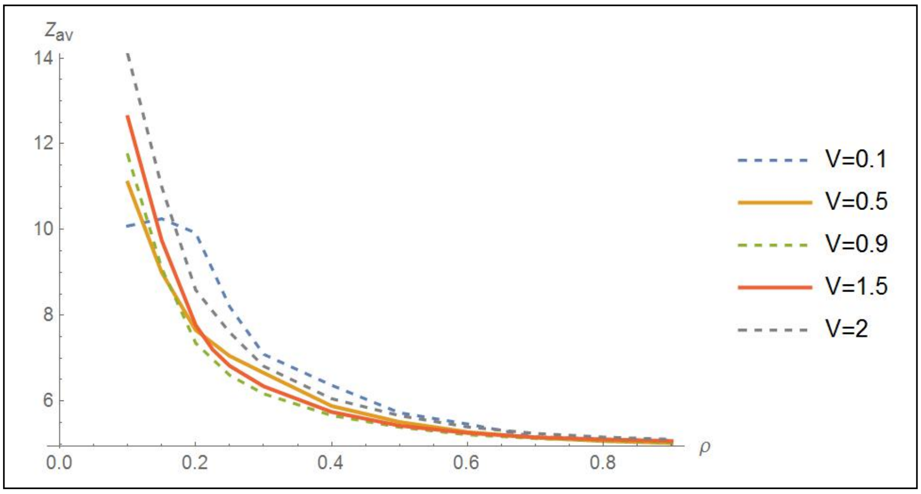


Figure 19. Z_{av} **as a function of** ρ **in the case when** $c = 5,\ Z = 5,$ **and** $c_U < c$

3.7. Dependencies of the System Performance Measures on *V* for Various Values of *c*

Let us consider the case when $\rho = 0.7,\ Z = 5,$ and $c_U < c.$ Figures 20−22 show dependencies of values of $N, W,$ and Z_{av}, respectively, on V for various c. Dependencies of $P\{Q = 0\}$ on V for various c are shown in Figure 23. The graphs show that $N, W,$ and Z_{av} are increasing functions of V. For the dependence of Z_{av} on V, this statement applies only to the single-server system, but for other values of c, Z_{av} is almost independent of V.

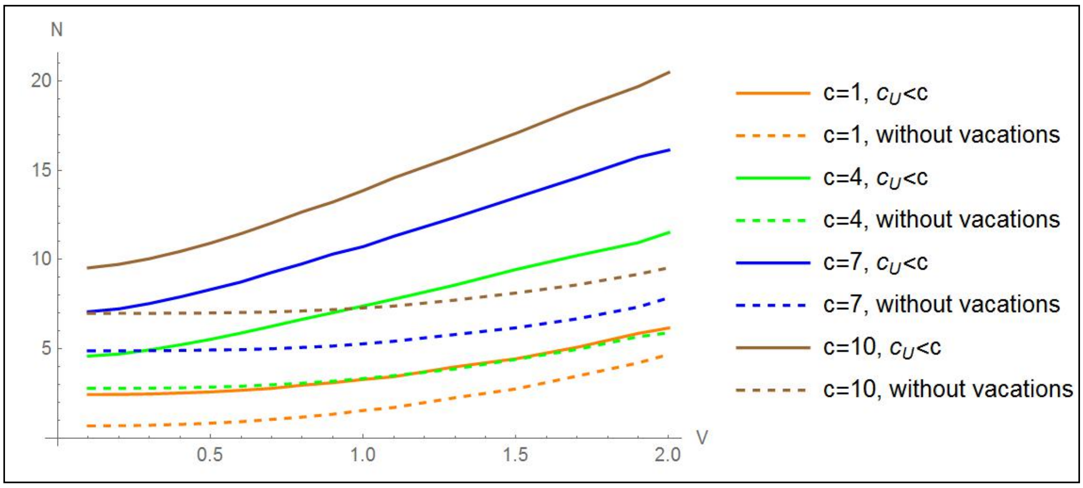


Figure 20. N **as a function of** V **in the case when** $\rho = 0.7$ **and** $Z = 5$

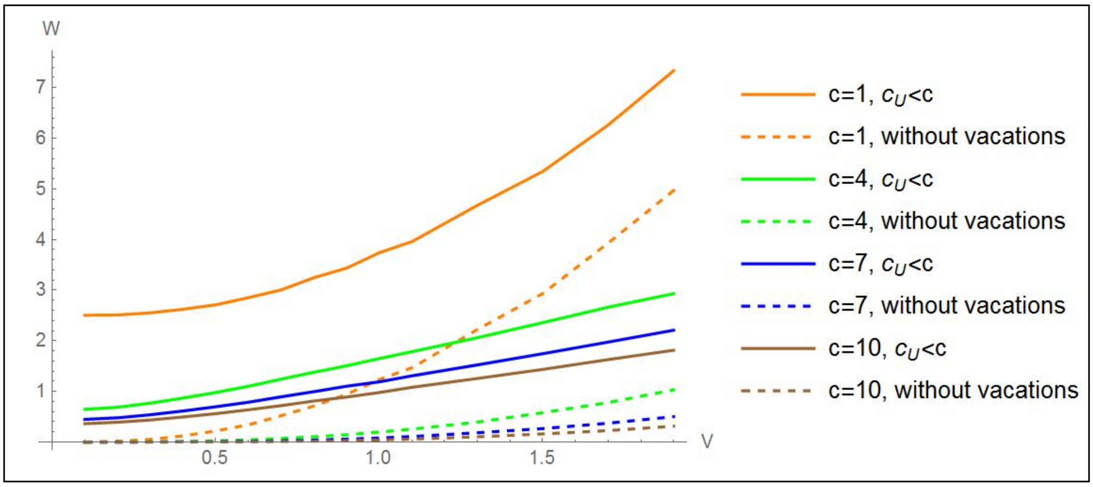


Figure 21. W **as a function of** V **in the case when** $\rho = 0.7$ **and** $Z = 5$

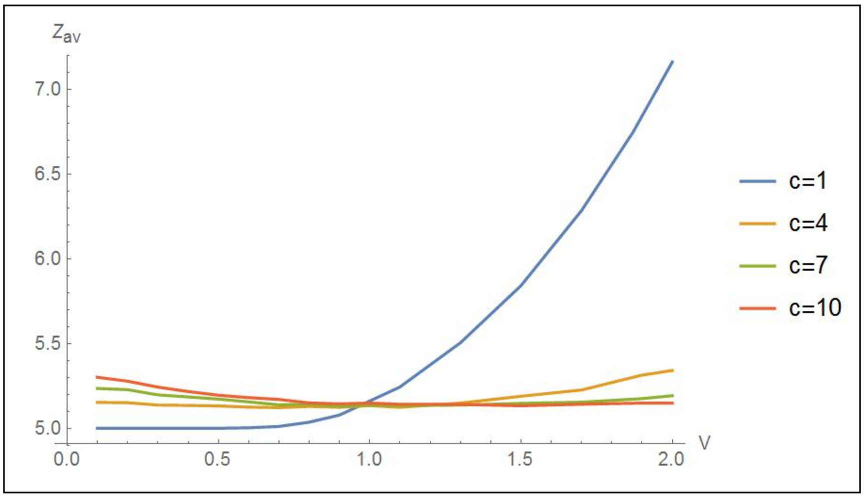


Figure 22. Z_{av} **as a function of** V **in the case when** $\rho = 0.7$, $Z = 5$, **and** $c_U < c$

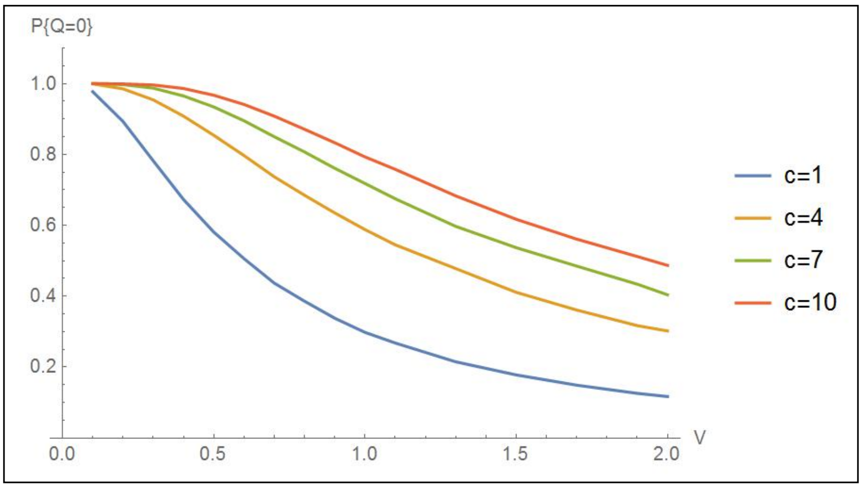


Figure 23. $P\{Q=0\}$ **as a function of** V **for the system without vacation, when** $\rho = 0.7$

For the system without vacations, the dependencies of $P\{Q=0\}$ on V are decreasing functions. The values of N and $P\{Q=0\}$ increase as the number of servers increases, while the value of W decreases. It is quite natural that for the system without vacations, the values of N and W are smaller than the corresponding values for the vacation system.

3.8. Dependencies of the System Performance Measures on ρ for Various Values of c

Let us consider the case when $V = 1.5, Z = 5,$ and $c_U < c.$ Figures 24 and 25 show dependencies of values of N and W, respectively, on ρ for various $c.$ The same dependencies are also considered for the system without vacations. The graphs show that N and W are increasing functions of $\rho,$ but for the multi-server systems, the values of W increase only for values of ρ approaching 1.

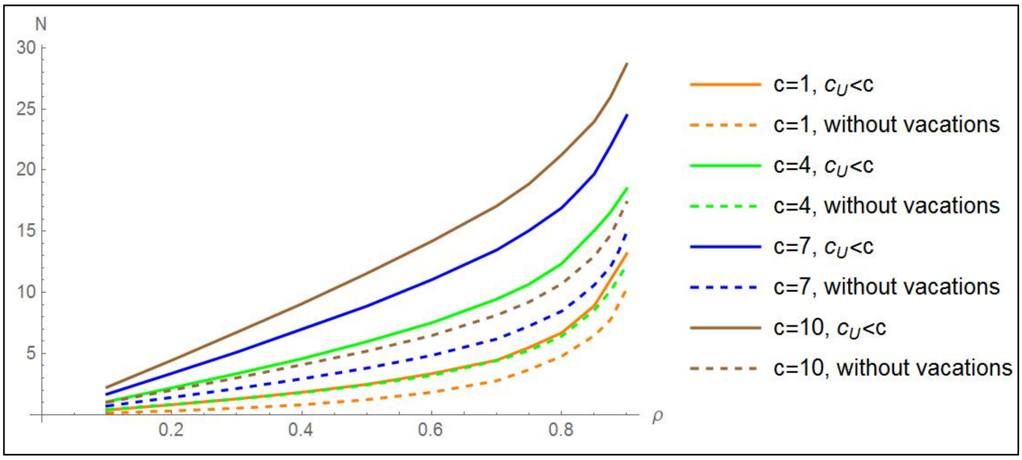


Figure 24. N **as a function of** ρ **in the case when** $V = 1.5$ **and** $Z = 5$

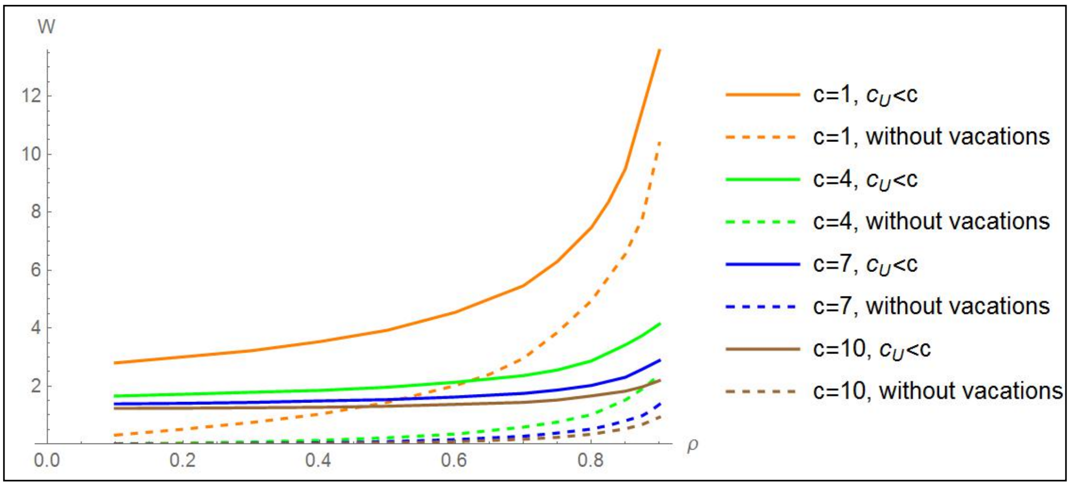


Figure 25. W **as a function of** ρ **in the case when** $V = 1.5$ **and** $Z = 5$

The value of N increases as the number of servers increases, while the value of W decreases. It is quite natural that for the system without vacations, the values of N and W are smaller than the corresponding values for the vacation system.

3.9. Examples of Obtaining Distributions of the System Performance Measures

We can get the distributions of those random variables for which tables are given in a simulation model. GPSS World tools make it possible to obtain graphic representations of distribution tables in the form of histograms and use graphs to track the dynamics of changes in random variables over time. The constructed simulation model contains tables to obtain distributions of n, the number of customers in the system, c_V, the number of servers in vacation mode, W, waiting times in the queue, and T_k $(k = 1,...,c)$, vacation durations for each server.

Let us consider the case when $c = 5$ and $Z = 5$. Figures 26–29 show changing values of c_V for four different datasets: 1) $V = 0.5$, $\rho = 0.7$, $c_U < 1$; 2) $V = 1.5$, $\rho = 0.7$, $c_U < 1$; 3) $V = 0.5$, $\rho = 0.7$, $c_U < 5$; 4) $V = 0.5$, $\rho = 0.3$, $c_U < 1$, respectively. Comparing the graphs in Figures 26 and 27, we conclude that an increase in V, the coefficient of variation, with the other parameters being equal, leads to an increase in the duration, frequency of vacations, and maximum values of c_V. An increase in the value of m in condition $c_U < m$ or a decrease in ρ, the system load factor, lead to the same result (here it is necessary to compare Figures 26 and 28, and Figures 26 and 29, respectively).

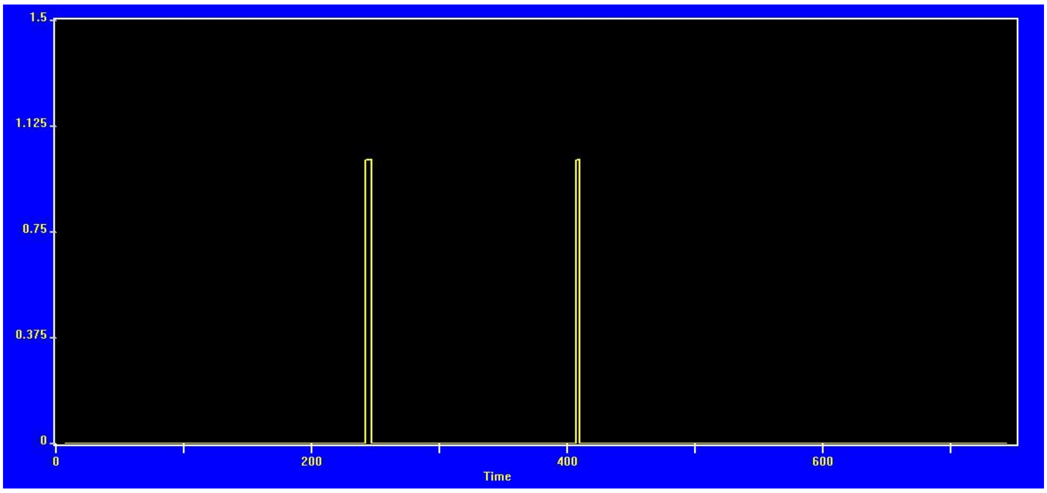


Figure 26. Changing c_V on the time interval [0, 750] in the case when $V = 0.5$, $\rho = 0.7$, and $c_U < 1$

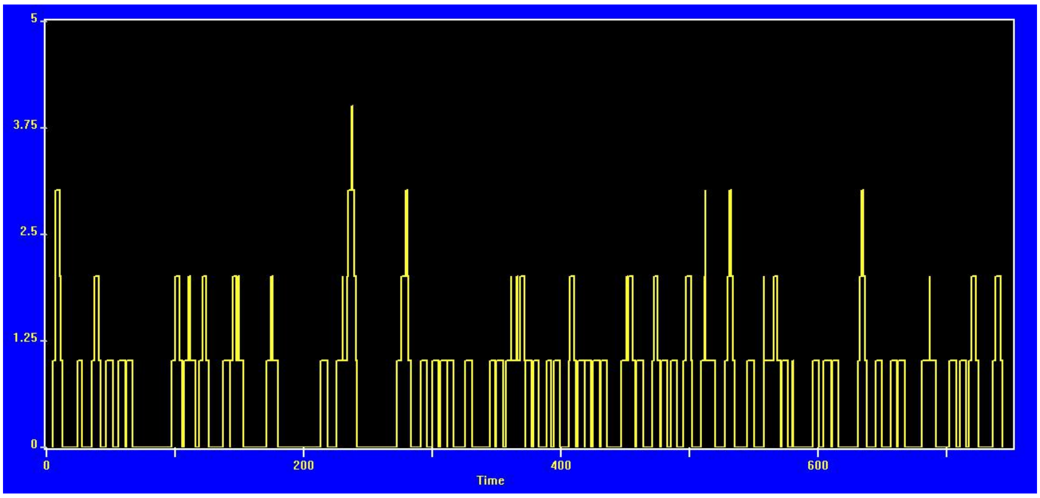


Figure 27. Changing c_V on the time interval [0, 750] in the case when $V = 1.5$, $\rho = 0.7$, and $c_U < 1$

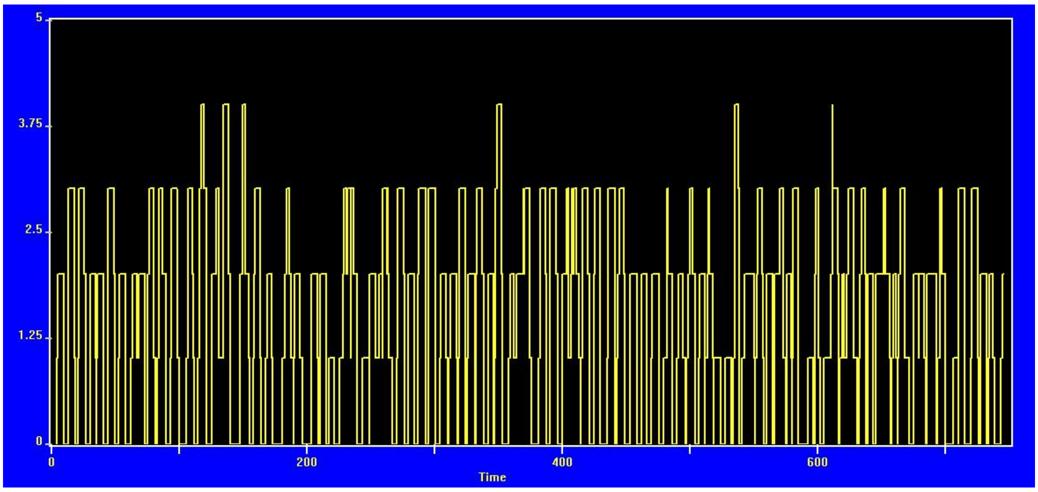


Figure 28. Changing c_V on the time interval [0, 750] in the case when $V = 0.5$, $\rho = 0.7$, and $c_U < 5$

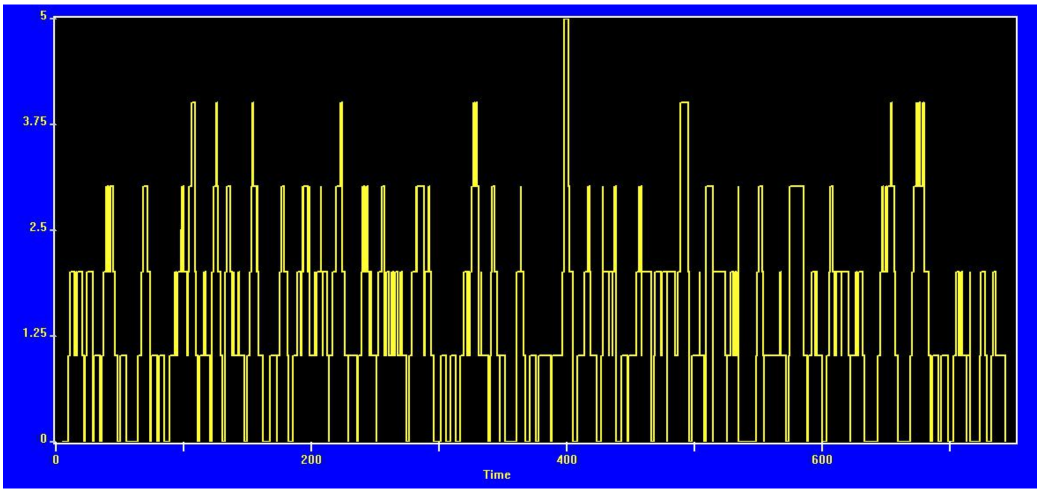


Figure 29. Changing c_V on the time interval [0, 750] in the case when $V = 0.5$, $\rho = 0.3$, and $c_U < 1$

The dependencies presented in the graphs are consistent with the histograms of the distributions of the random variables n, c_V, W, and T_k $(k = 1,...,c)$ shown in Figures 30−33.

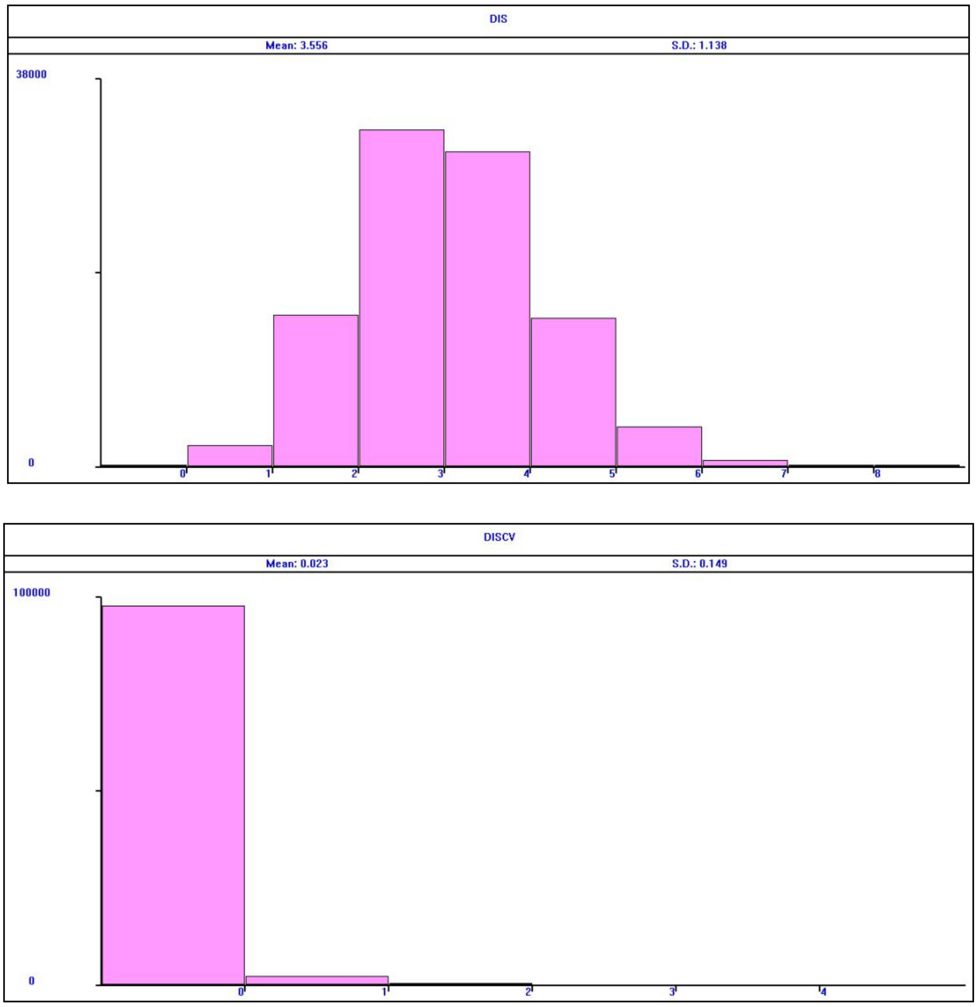
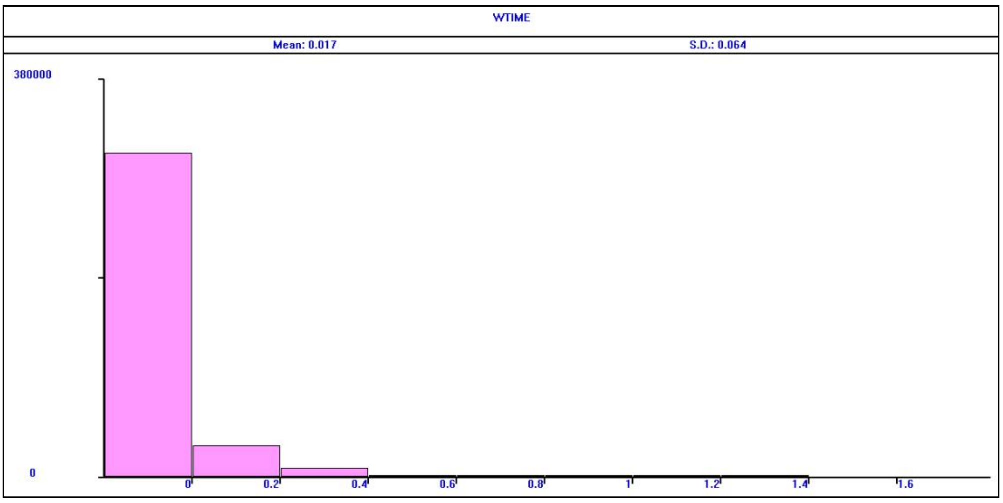
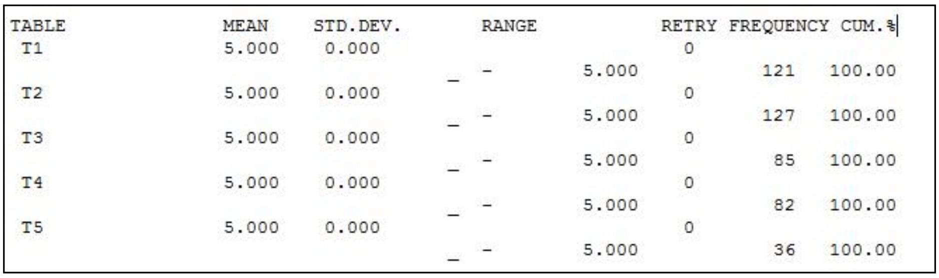


Figure 30. Distributions of n and c_V in the case when $V = 0.5$, $\rho = 0.7$, and $c_U < 1$





```
TABLE                 MEAN    STD.DEV.        RANGE            RETRY FREQUENCY CUM.%
 T1                   5.000    0.000                               0
                                           _  -          5.000           121   100.00
 T2                   5.000    0.000                               0
                                           _  -          5.000           127   100.00
 T3                   5.000    0.000                               0
                                           _  -          5.000            85   100.00
 T4                   5.000    0.000                               0
                                           _  -          5.000            82   100.00
 T5                   5.000    0.000                               0
                                           _  -          5.000            36   100.00
```

Figure 31. Distributions of W and T_k $(k = 1,...5)$ in the case when $V = 0.5,$ $\rho = 0.7,$ and $c_U < 1$

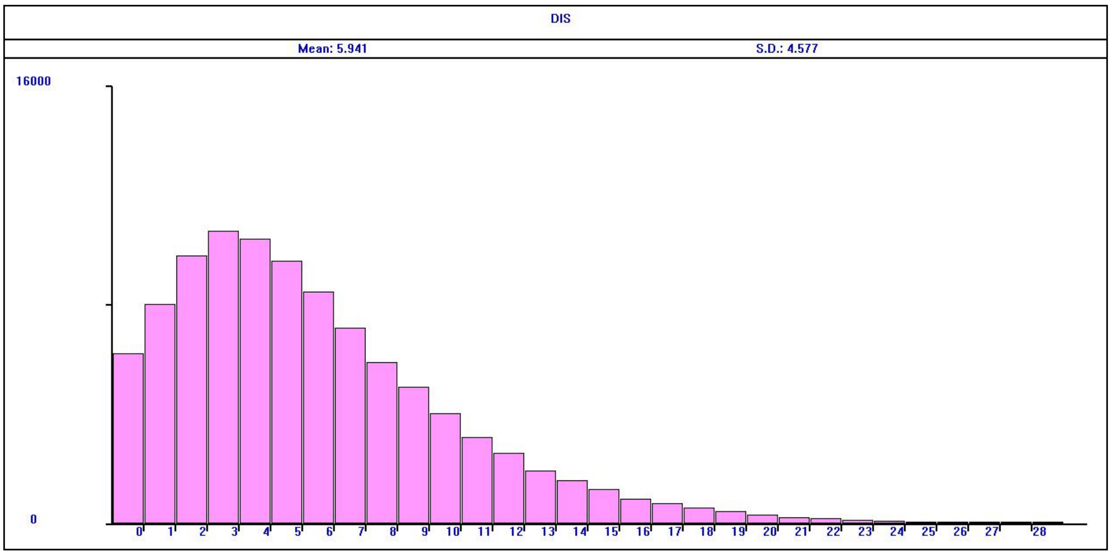
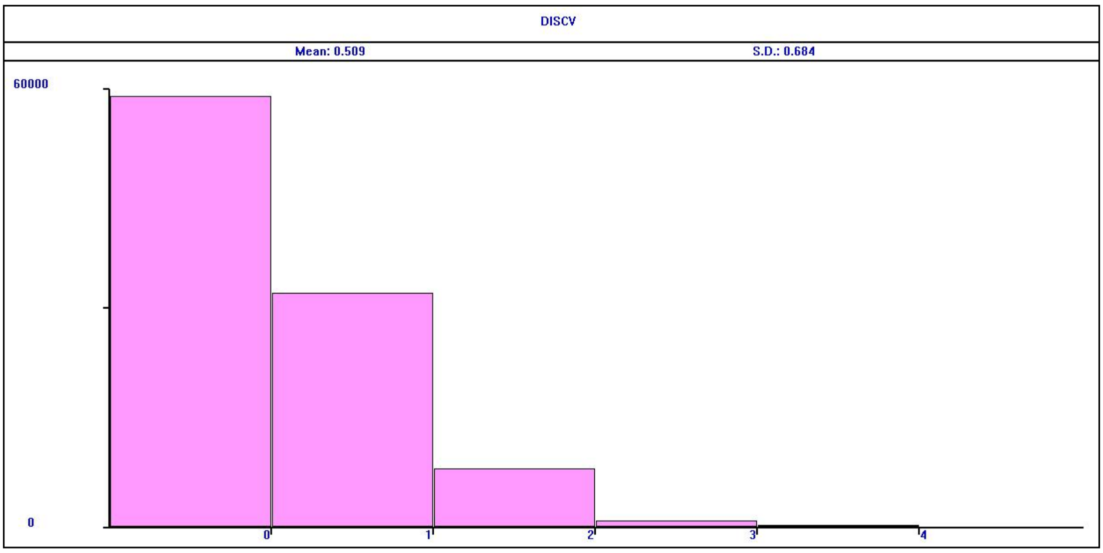
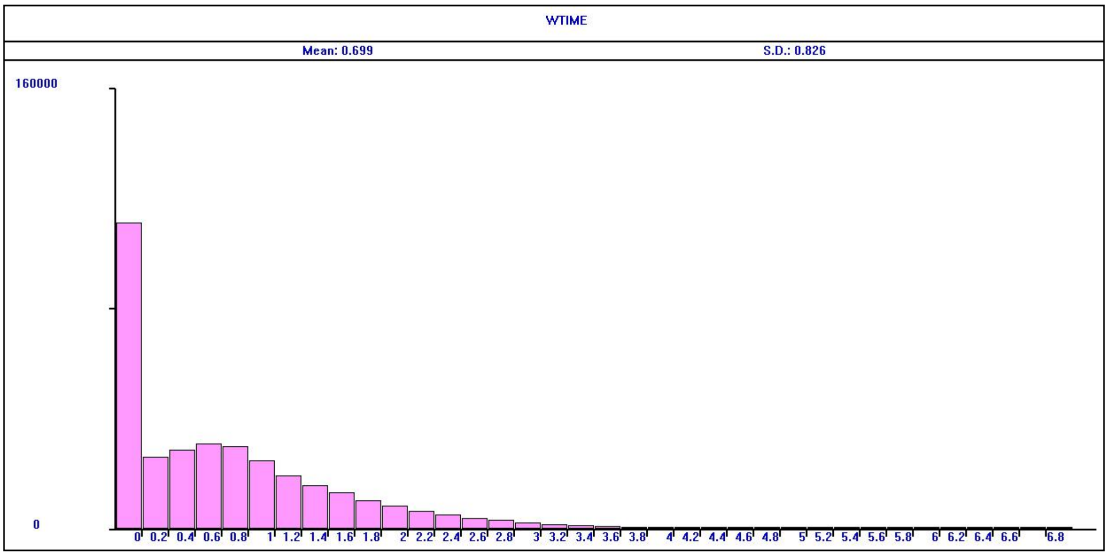
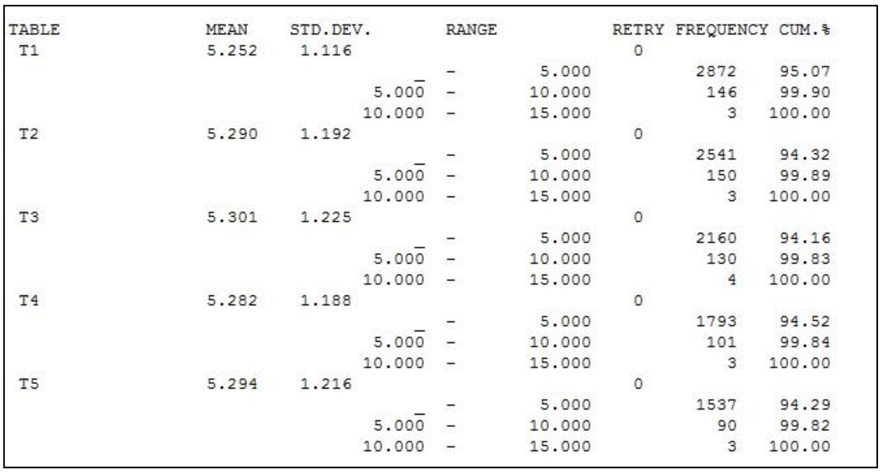


Figure 32. Distribution of n in the case when $V = 1.5,$ $\rho = 0.7,$ and $c_U < 1$







```
TABLE              MEAN    STD.DEV.       RANGE            RETRY FREQUENCY CUM.%
 T1                5.252    1.116                            0
                                      _   -         5.000             2872   95.07
                                 5.000    -        10.000              146   99.90
                                10.000    -        15.000                3  100.00
 T2                5.290    1.192                            0
                                      _   -         5.000             2541   94.32
                                 5.000    -        10.000              150   99.89
                                10.000    -        15.000                3  100.00
 T3                5.301    1.225                            0
                                      _   -         5.000             2160   94.16
                                 5.000    -        10.000              130   99.83
                                10.000    -        15.000                4  100.00
 T4                5.282    1.188                            0
                                      _   -         5.000             1793   94.52
                                 5.000    -        10.000              101   99.84
                                10.000    -        15.000                3  100.00
 T5                5.294    1.216                            0
                                      _   -         5.000             1537   94.29
                                 5.000    -        10.000               90   99.82
                                10.000    -        15.000                3  100.00
```

Figure 33. Distributions of c_V, W, and T_k $(k = 1,...5)$ in the case when $V = 1.5$, $\rho = 0.7$, and $c_U < 1$

4. Conclusion

A simulation model is constructed that allows describing the stochastic process of the functioning of a vacation queuing system. A modification of the multiple vacation policy (c_U -policy) proposed by us allows more flexible use of vacations, taking into account the number of servers in use by customers at the time each server switches to vacation mode. The application of the c_U -policy helps to reduce the degradation in system performance compared to the system without vacations. The constructed simulation model gives us the fundamental opportunity to predict the impact of each of the input parameters on the system performance measures for the *G*/*G*/*c* system with arbitrary inter-arrival times and service times distributions. Calculations showed good convergence of our simulation results with the results of the analytical model for the *M*/*M*/*c* system, which exists only in the case of the standard multiple vacation policy.

The obtained results for the *G*/*G*/*c* system showed a significant dependence of the system performance measures on the value of the coefficient of variation of the inter-arrival times distribution. This dependence increases even more if we take into account the possibility of changing the coefficients of variation of service times and the duration of vacations. Therefore, it is difficult to say about the practical application of the results of analytical models, which were obtained only for the *M*/*G*/1 and *M*/*M*/*c* systems.

References

[1] Levy, Y. and Yechiali, U., “Utilization of Idle Time in an M/G/1 Queueing System,” *Management Science*, 22 (2). 202-211. Oct.1975.

[2] Doshi, B.T., “Queueing Systems with Vacations − a Survey,” *Queueing Systems*, 1, 29-66. June.1986.

[3] Ibe, O., “M/G/1 Vacation Queueing Systems with Server Timeout,” *American Journal of Operations Research*, 5 (2). 77-88. Mar.2015.

[4] Takagi, H., *Queueing Analysis: A Foundation of Performance Evaluation, Vacation and Priority Systems*, Part I, Volume I; North-Holland, Amsterdam, 1991.

[5] Yadin, M. and Naor, P., “Queueing system with a removable service station,” *Journ. of the Operational Research Society*, 14 (4). 393-405. 1963.

[6] Heyman, D.P., “The T-policy for the M/G/1 queue,” *Manag. Sci.*, 23 (7). 775–778. Mar.1977.

[7] Levy, Y. and Yechiali, U., “Utilization of idle time in an M/G/1 queueing system,” *Manag. Sci*., 22 (2). 202–211. Oct.1975.

[8] Birta, L.G. and Arbez, G., *Modelling and Simulation: Exploring Dynamic System Behaviour*, 3rd edition, Springer Nature, Switzerland, 2019, 491-520.

[9] Zhernovyi, Yu., *Creating models of queueing systems using GPSS World: Programs, detailed explanations and analysis of results*, LAP Lambert Academic Publishing, Saarbrücken, 2015, 220 p.

[10] Levy, Y. and Yechiali, U., “An M/M/s queue with servers vacations,” Information Systems and Operational Research, 14 (2). 153-163. 1976.