Genetic Cost Optimization of the $GI/M/1/N$ Finite-Buffer Queue with a Single Vacation Policy

Marcin Gabryel[1], Robert K. Nowicki[1], Marcin Woźniak[2], and Wojciech M. Kempa[2]

[1] Institute of Computational Intelligence, Czestochowa University of Technology, Al. Armii Krajowej 36, 42-200 Czestochowa, Poland
[2] Institute of Mathematics, Silesian University of Technology, ul. Kaszubska 23, 44-100 Gliwice, Poland
{Marcin.Gabryel,Robert.Nowicki}@iisi.pcz.pl,
{Marcin.Wozniak,Wojciech.Kempa}@polsl.pl

Abstract. In the artice, problem of the cost optimization of the $GI/M/1/N$-type queue with finite buffer and a single vacation policy is analyzed. Basing on the explicit representation for the joint transform of the first busy period, first idle time and the number of packets transmitted during the first busy period and fixed values of unit costs of the server's functioning an optimal set of system parameters is found for exponentially distributed vacation period and 2-Erlang distribution of inter arrival times. The problem of optimization is solved using genetic algorithm. Different variants of the load of the system are considered as well.

Keywords: Busy period, finite-buffer queue, genetic algorithm, idle time, optimization, single vacation.

1 Introduction

Finite-buffer queues are being intensively studied nowadays due to their numerous applications in different real-life problems occurring in technical and economical sciences. They are used, first of all, in modelling of telecommunication and computer networks (ATM switches, IP routers etc.), however they can also be helpful in the investigation of manufacturing processes and in some problems occurring in logistics and transport.

From the observation follows that the stream of IP packets incoming to the Internet router or cells arriving into the ATM switch rather rarely can be described by a Poisson process. Moreover, due to permanent changing intensity of the traffic and some sophisticated phenomena characterizing the traffic, like the self-similarity or burstiness, the "classical" analysis limited to the stationary state of the system, is not sufficient for a thorough description of the system's operation. These arguments motivate the transient analysis of the system's operation basing on the non-Poisson process describing the input flow of packets.

L. Rutkowski et al. (Eds.): ICAISC 2013, Part II, LNAI 7895, pp. 12–23, 2013.

An extension of the class of finite-buffer queues are models with server's vacation, where the server takes a randomly distributed time during which the service process is blocked. Finite systems with server vacations can be used in modelling of SVC (switched virtual connection) networks, where the vacation period can be considered as a time needed for the server to release one SVC or the time for setting up any next SVC (see [11]).

In the paper we investigate the $GI/M/1$-type finite-buffer queuing system with single vacation policy and exhaustive service. Basing on the explicit representation for the joint transform of the first busy period, first idle time and the number of packets transmitted during the first busy period, and fixed values of unit costs of the server's functioning during the service and being idle, we find an optimal set of system parameters for exponentially distributed vacation period and 2-Erlang distribution of inter arrival times. In fact, we obtain results for different variants of the load of the system. The problem of optimization is solved using genetic algorithms.

Results for vacation queuing models can be found in monographs [14], [18] and e.g. in papers [2], [3], [11] and [12] and [15]. A batch-arrival $GI/G/1$-type system with infinite buffer and exponentially distributed single vacations is considered in [6] on the first vacation cycle, using the approach based on Wiener-Hopf-type factorization. Transient characteristics of the system with single vacations with Poisson arrivals and generally distributed service times are analyzed in [7], [8] and [9].

In [16] the classical linear cost structure is introduced in the queuing model. The problem of cost optimization is also investigated e.g. in [4]. In [10] the problem of the existence of the optimal vacation policy is solved. In [17] the formula for the total expected cost per unit time in the finite-buffer $M/G/1$-type system with removable server, working in the stationary regime, is obtained.

This paper presents an innovative application of genetic algorithms for positioning queuing systems. The authors of [1] and [19] presented other interesting applications of genetic algorithms to simulate examined objects. Built genetic simulation system helped to calculate optimal cost for modeled queuing system. Presented research describe system positioning in some possible scenarios, but described genetic method makes it also possible evaluate optimal values of system variables in other conditions.

1.1 Queuing Model

In the paper we study a finite-buffer $GI/M/1/N$-type queuing system with inter arrival times with generally distributed random variables with a distribution function $F(\cdot)$, and exponential service times with mean μ^{-1}. The system capacity equals N i.e. we have the $(N-1)$-places in buffer and one place in service. We assume that the system starts working at $t = 0$ with at least one packet present. After each busy period the server takes a single vacation, with a general type distribution function $V(\cdot)$, during which the service process is stopped. If at the end of the vacation there is no packet in the buffer queue, the server is being activated (is on standby) and waits for the first arrival to start the service process.

If the system is not empty at the vacation completion epoch, then the service process is being initialized immediately and the new busy period begins.

We assume that sequences of successive inter arrival times, service times and the single vacation duration are totally independent random variables.

Let us introduce the following notations:

- τ_1 — the first busy period of the system (starting at $t = 0$);
- δ_1 — the first idle time of the system (consisting of the first vacation time v_1 and the first server standby time q_1);
- $h(\tau_1)$ — the number of packets completely served during τ_1;
- $X(t)$ — the number of packets present in the system at time t.

1.2 Auxiliary Results

In [13] the explicit formula for the conditional joint characteristic function of random variables τ_1, δ_1 and $h(\tau_1)$ is derived.

Define

$$B_n(s, \varrho, z) = \mathbf{E}\{e^{-s\tau_1 - \varrho\delta_1} z^{h(\tau_1)} \,|\, X(0) = n\}, \tag{1}$$

where $1 \le n \le N$, $s \ge 0$, $\varrho \ge 0$ and $|z| \le 1$.

Introduce the following functions:

$$f(s) = \int_0^\infty e^{-st} dF(t), \quad s > 0 \tag{2}$$

and besides

$$a_n(s, z) = \int_0^\infty \frac{(z\mu t)^n}{n!} e^{-(\mu+s)t} dF(t), \quad n \ge 0, \tag{3}$$

$$\begin{aligned}\Psi_n(s, \varrho, z) = &-\frac{(z\mu)^n}{(n-1)!}\Big[\int_0^\infty dF(t)\int_0^t x^{n-1} e^{-(\mu+s)x} \\ &\times \left(e^{-\varrho(t-x)} V(t-x) + \int_{t-x}^\infty e^{-\varrho y} dV(y)\Big) dx\right].\end{aligned} \tag{4}$$

Moreover, basing on the sequence $\big(a_n(s, z)\big)$ defined in Eq. (3), we defined recursively the following sequence:

$$\begin{aligned} &R_0(s, z) = 0, \quad R_1(s, z) = a_0^{-1}(s, z), \\ &R_{n+1}(s, z) = R_1(s, z)(R_n(s, z) - \sum_{k=0}^{n} a_{k+1}(s, z) R_{n-k}(s, z)). \end{aligned} \tag{5}$$

Finally, let

$$D(s, \varrho, z) = \sum_{k=1}^{N-1} a_k(s, z) \sum_{i=2}^{N-k+1} R_{N-k+1-i}(s, z)\Psi_i(s, \varrho, z), \tag{6}$$

$$G(s, \varrho, z) = \Psi_N(s, \varrho, z) + \big(1 - f(\mu + s)\big) \sum_{k=2}^{N} R_{N-k}(s, z)\Psi_k(s, \varrho, z) \tag{7}$$

and

$$H(s,z) = \big(1 - f(\mu + s)\big) R_{N-1}(s,z) - \sum_{k=1}^{N-1} a_k(s,z) R_{N-k}(s,z). \tag{8}$$

The following theorem is true.

Theorem 1. *For $s \geq 0$, $\varrho \geq 0$ and $|z| \leq 1$ the following formulae hold true:*

$$\begin{aligned} B_1(s,\varrho,z) &= \mathbf{E}\{e^{-s\tau_1 - \varrho\delta_1} z^{h(\tau_1)} \mid X(0) = 1\} \\ &= \frac{D(s,\varrho,z) - G(s,\varrho,z)}{H(s,z)} - \Psi_1(s,\varrho,z) \end{aligned} \tag{9}$$

and

$$\begin{aligned} &B_n(s,\varrho,z) = \mathbf{E}\{e^{-s\tau_1 - \varrho\delta_1} z^{h(\tau_1)} \mid X(0) = n\} \\ &= \frac{D(s,\varrho,z) - G(s,\varrho,z)}{H(s,z)} R_{n-1}(s,z) - \sum_{k=2}^{n} R_{n-k}(s,z) \Psi_k(s,\varrho,z), \quad 2 \leq n \leq N. \end{aligned} \tag{10}$$

where $n \geq 1$ and $a_n(s,z)$, $\Psi_n(s,\varrho,z)$, $D(s,\varrho,z)$, $G(s,\varrho,z)$ and $H(s,z)$ are defined in (3), (4), (6), (7) and (8) respectively.

1.3 Cost Optimization Problem

In the paper we are interested in the solution of the following optimization problem. Consider the following equation

$$Q_n(c_1) = r(\tau_1)\mathbf{E}_n\tau_1 + r(\delta_1)\mathbf{E}_n\delta_1, \tag{11}$$

in which $r(\tau_1)$ and $r(\delta_1)$ denote, respectively, the fixed unit costs of the system's operation during the first busy period τ_1 and the first idle time δ_1, $\mathbf{E}_n\tau_1$ and $\mathbf{E}_n\delta_1$ are, respectively, the means of the first busy period τ_1 and the first idle time δ_1 on condition that the system starts evolution with n packets present. Besides, $Q_n(c_1)$ is the total cost of the operation of the system during the first cycle c_1 on condition that $X(0) = n$.

Of course the unit cost of the operation of the system during the first cycle c_1, on condition that the system contains exactly n packets at the opening, one can expressed as

$$r_n(c_1) = \frac{Q_n(c_1)}{\mathbf{E}_n(c_1)} = \frac{r(\tau_1)\mathbf{E}_n\tau_1 + r(\delta_1)\mathbf{E}_n\delta_1}{\mathbf{E}_n\tau_1 + \mathbf{E}_n\delta_1}. \tag{12}$$

Let us consider the system in which inter arrival times are exponentially distributed and the vacation period has 2-Erlang distribution. Namely, let us take

$$F(t) = 1 - e^{-\lambda t}, \quad \lambda > 0,\, t > 0, \tag{13}$$

and

$$V(t) = 1 - e^{-\alpha t}(1 + \alpha t), \quad \alpha > 0, t > 0. \tag{14}$$

For such a system we use genetic optimization for finding the optimal set of "input" system parameters λ, μ and α, to minimize the costs $Q_n(c_1)$ and/or $r_n(c_1)$. In fact we will take into consideration different variants of the load of the system: the case of under-load, critically load and overload.

1.4 Genetic Cost Optimization

Very important phase in the process of designing queuing systems is positioning for optimal costs of work. To perform this operation we must know possible malfunctions and optimal work conditions. To do this we may apply the knowledge that comes from human experts or our previous experience. In many projects this is a main source which determines the prototypes. However, usually it is insufficient. The best way is to test the system in practice or perform computer simulations. First way may be vary long in time and cost a lot. Therefore best solution is to use computer. The question is, which method to use? We would like to present positioning of queuing system by the use of genetic algorithm. Computational intelligence, especially genetic algorithms, are very effective. In the literature [1] and [19] the authors described use of genetic algorithm to create knowledge about technical systems. Presented use of computational intelligence help to simulate the object states and build decision support systems. In this paper we present possible way to use genetic to calculate and position queuing system. However queuing systems have complicated mathematical models therefore genetic calculation is still not easy. We present simulation results of $GI/M/1/N$ finite-buffer queue with a single vacation policy. We used standard genetic algorithm to built a dedicated evolutionary simulation system based on mathematical model described in Section 1.2. In the research we used formula (12) to optimize cost of the system. Genetic simulation system was searching for best values of $GI/M/1/N$ queuing system variables that make it work with lowest costs in the specified time unit. The research results provide type of knowledge that describes examples of proper operation of the system in some possible scenarios. This type of knowledge is necessary for its tuning and evaluating.

2 Research Results

In this section we will present research results for optimum values of the examined $GI/M/1/N$ queue system. Based on the research results, we may predict possible time of system response in each case and optimize cost of work $r_n(c_1)$, described in Eq. (12). In the research, we have examined the system for optimum values in 100 samplings. Presented optimal results are average values. We will determine time prediction based on the following assumptions:

- Average service time: $T_{service} = \frac{1}{\mu}$,

- Average time between packages income into the system: $T_{income} = \frac{2}{\lambda}$,
- Average vacation time: $T_{vacation} = \frac{1}{\alpha}$,
- Examined system size: $N =$ buffer size $+1$.

In Table 1 we present optimum values for all system parameters μ, λ and α. However in the real environment we are able to set some values of the system we use. Therefore we have also tried to optimized values of parameters μ, λ and α in few possible scenarios. Each scenario was defined and then, there were optimized values of system parameters and cost of work according to given assumptions. In each scenario there were 100 genetic optimization experiments and the results are given as average value of optimization for all the experiments.

Table 1. Optimimal parameters μ, λ and α for lowest value of Eq.(12)

	μ	λ	α
Average optimal value	2.104179284	0.049898874	13092.21989
optimal $r_n(c_1)$	0.011857087		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.475244675	40.08106516	$7.63812E^{-05}$

Scenario 1

In this scenario, to genetic cost optimization we have assumed that the system handles incoming packets in a constant time what means that, average service time: $T_{service} = \frac{1}{\mu}$ is constant. Therefore we have set the value of μ parameter and optimized other system parameters. In this scenario all system parameters were optimized for set values: $\mu = 100$, $\mu = 1$ or $\mu = 0.01$. Research results are shown in Table 2.

Scenario 2

In this scenario, to genetic cost optimization we have assumed that packets come into the system with some regularity, time between packages income into the system: $T_{income} = \frac{2}{\lambda}$ is constant. Therefore we have set the value of λ parameter and optimized other system parameters. In this scenario all system parameters were optimized for set values: $\lambda = 100$, $\lambda = 1$ or $\lambda = 0.01$. Research results are shown in Table 3.

Scenario 3

In this scenario, to genetic cost optimization we have assumed that system need to stop serving packets with some regularity, vacation time: $T_{vacation} = \frac{1}{\alpha}$ is constant. Therefore we have set the value of α parameter and optimized other system parameters. In this scenario all system parameters were optimized for set values: $\alpha = 100$, $\alpha = 1$ or $\alpha = 0.01$. Research results are shown in Table 4. Moreover we have also analyzed some more complicated scenarios. In the research were examined possible situations where service time, packets income time or vacation time is set and cost of system work must be adequate.

Table 2. Optimal parameters λ and α for set $\mu = 100$, $\mu = 1$, $\mu = 0.01$ and lowest cost value of Eq.(12)

	μ	λ	α
Average optimal value	100	0.441648853	1.092121241
optimal $r_n(c_1)$	0.002208244		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.01	4.528484533	0.915649255
Average optimal value	1	0.185442954	1.456477569
optimal $r_n(c_1)$	0.092721477		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	1	10.78498781	0.686587985
Average optimal value	0.01	0.258147693	1.594055468
optimal $r_n(c_1)$	12.90738467		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	100	7.747502887	0.627330742

Table 3. Optimal parameters μ and α for set $\lambda = 100$, $\lambda = 1$, $\lambda = 0.01$ and lowest cost value of Eq.(12)

	μ	λ	α
Average optimal value	4.059353298	100	$2.3177E^{-06}$
optimal $r_n(c_1)$	12.3172329		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.246344658	0.02	431462.2255
Average optimal value	27.30386864	1	0.000002924
optimal $r_n(c_1)$	0.018312423		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.036624847	2	341997.264
Average optimal value	1.356968521	0.01	1.503990611
optimal $r_n(c_1)$	0.003684684		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.736936771	200	0.664897768

Scenario 4

In this scenario, to genetic cost optimization we have assumed, similar to scenario 2, that service time: $T_{service} = \frac{1}{\mu}$ is constant. Therefore we have set the value of μ parameter and optimized other system parameters. Moreover we have assumed that cost of system work is $r_n(c_1)$ is also defined in some way. In this scenario all system parameters were optimized for set values: $\mu = 100$, $\mu = 1$ or $\mu = 0.01$ and $r_n(c_1) < 1$ or $r_n(c_1) > 1$. Research results are shown in Table 5 - Table 6. We have also optimized cost of the system for set time of packets income an vacation.

Table 4. Optimal parameters μ and λ for set $\alpha = 100$, $\alpha = 1$, $\alpha = 0.01$ and lowest cost value of Eq.(12)

	μ	λ	α
Average optimal value	1.16928266	0.177362545	100
optimal $r_n(c_1)$	0.075842459		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.855225203	11.27633797	0.01
Average optimal value	1.247463554	0.172708087	1
optimal $r_n(c_1)$	0.069223701		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.801626626	11.58023365	1
Average optimal value	1.227674838	0.206532185	0.01
optimal $r_n(c_1)$	0.084115182		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.814547931	9.683720705	100

Table 5. Optimal parameters α and λ for set μ and lowest cost value of Eq.(12)< 1

	μ	λ	α
Average optimal value	100	0.400070456	1.192386163
optimal $r_n(c_1) < 1$	0.002000352		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.01	4.999119455	0.838654482
Average optimal value	1	0.223777444	1.300852544
optimal $r_n(c_1) < 1$	0.111888722		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	1	8.937451265	0.768726636
Average optimal value	0.01	0.008013686	7.000048983
optimal $r_n(c_1) < 1$	0.4006843		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	100	249.5730429	0.142856143

Scenario 5

In this scenario, to genetic cost optimization we have assumed, similar to scenario 2, that time between packages income into the system: $T_{income} = \frac{2}{\lambda}$ is constant. Therefore we have set the value of λ parameter and optimized other system parameters. Moreover we have assumed that cost of system work $r_n(c_1)$ is also defined in some way. In this scenario all system parameters were optimized for set values: $\lambda = 100$, $\lambda = 1$ or $\lambda = 0.01$ and $r_n(c_1) < 1$ or $r_n(c_1) > 1$. Research results are shown in Table 7 - Teble 8. Last scenario present research results on optimized cost of the system for set vacation time.

Table 6. Optimal parameters α and λ for set μ and lowest cost value of Eq.(12)> 1

	μ	λ	α
Average optimal value	100	1310.416361	0.000056554
optimal $r_n(c_1) > 1$	6.552081805		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.01	0.001526232	17682.21523
Average optimal value	1	16.55791579	0.000000459
optimal $r_n(c_1) > 1$	8.278957897		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	1	0.120788149	2178649.237
Average optimal value	0.01	0.232515381	1.47718659
optimal $r_n(c_1) > 1$	11.62576905		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	100	8.601581501	0.676962549

Table 7. Optimal parameters α and μ for set λ and lowest cost value of Eq.(12)< 1

	μ	λ	α
Average optimal value	1598.611383	100	0.000278277
optimal $r_n(c_1) < 1$	0.031277145		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.000625543	0.02	3593.541687
Average optimal value	1.311446436	1	0.184496128
optimal $r_n(c_1) < 1$	0.070340703		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.762516846	2	10.8403359
Average optimal value	1.715058427	0.01	1.357447045
optimal $r_n(c_1) < 1$	0.002915353		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.583070515	200	0.736676988

Scenario 6

In this scenario, to genetic cost optimization we have assumed, similar to scenario 2, that vacation time: $T_{vacation} = \frac{1}{\alpha}$ is constant. Therefore we have set the value of α parameter and optimized other system parameters. Moreover we have assumed that cost of system work $r_n(c_1)$ is also defined in some way. In this scenario all system parameters were optimized for set values: $\alpha = 100$, $\alpha = 1$ or $\alpha = 0.01$ and $r_n(c_1) < 1$ or $r_n(c_1) > 1$. Research results are shown in Table 9 - Table 10.

Table 8. Optimal parameters α and μ for set λ and lowest cost value of Eq.(12)> 1

	μ	λ	α
Average optimal value	4.772254414	100	0.000003217
optimal $r_n(c_1) > 1$	10.47722851		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.20954457	0.02	310848.6167
Average optimal value	0.288431013	1	0.000006832
optimal $r_n(c_1) > 1$	1.733516777		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	3.467033554	2	146370.0234
Average optimal value	0.002060192	0.01	25.22387757
optimal $r_n(c_1) > 1$	2.426958264		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	485.3916528	200	0.039644975

Table 9. Optimal parameters λ and μ for set α and lowest cost value of Eq.(12)< 1

	μ	λ	α
Average optimal value	1.416040912	0.216560935	100
optimal $r_n(c_1) < 1$	0.076467047		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.706194285	9.235275974	0.01
Average optimal value	1.737398608	0.115104772	1
optimal $r_n(c_1) < 1$	0.033125608		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.575573156	17.37547423	1
Average optimal value	1.34126515	0.180202828	0.01
optimal $r_n(c_1) < 1$	0.067176437		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	0.745564738	11.09860496	100

3 Final Remarks

In the article, we have proposed a new method for queuing systems positioning. Genetic algorithms are useful in simulations or to generate a collection of representative samples as presented by the authors of [1] and [19], which can be used by decision support systems. They are also very effective for positioning queuing systems. This method can be useful when we have a model of positioned object and because of its complexity classical model calculations are merely feasible. Conducted experiments confirm its usefulness to simulate examined object in

Table 10. Optimal parameters λ and μ for set α and lowest cost value of Eq.(12)> 1

	μ	λ	α
Average optimal value	0.129801621	0.448907478	100
optimal $r_n(c_1) > 1$	1.729205978		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	7.704064035	4.455261046	0.01
Average optimal value	0.153689121	0.479652145	1
optimal $r_n(c_1) > 1$	1.560462256		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	6.506641417	4.169688431	1
Average optimal value	0.10366026	0.462297133	0.01
optimal $r_n(c_1) > 1$	2.229866744		
Time	$T_{service}$	T_{income}	$T_{vacation}$
[s]	9.646898435	4.326221941	100

many possible scenarios. An important restriction is only to carry out a large number of simulations to determine the best description of the simulated object. It is therefore time-consuming procedure. Further work should be carried out in the direction of reducing time-consuming solution, eg. by using some knowledge prior to generating the initial population.

References

1. Gabryel, M., Woźniak, M., Nowicki, R.K.: Creating learning sets for control systems using an evolutionary method. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) EC 2012 and SIDE 2012. LNCS, vol. 7269, pp. 206–213. Springer, Heidelberg (2012)
2. Gupta, U.C., Banik, A.D., Pathak, S.S.: Complete analysis of $MAP/G/1/N$ queue with single (multiple) vacation(s) under limited service discipline. Journal of Applied Mathematics and Stochastic Analysis 3, 353–373 (2005)
3. Gupta, U.C., Sikdar, K.: Computing queue length distributions in $MAP/G/1/N$ queue under single and multiple vacation. Appl. Math. Comput. 174(2), 1498–1525 (2006)
4. Kella, O.: Optimal control of the vacation scheme in an $M/G/1$ queue. Oper. Res. 38(4), 724–728 (1990)
5. Kempa, W.M.: $GI/G/1/\infty$ batch arrival queuing system with a single exponential vacation. Math. Method. Oper. Res. 69(1), 81–97 (2009)
6. Kempa, W.M.: Some new results for departure process in the $M^X/G/1$ queuing system with a single vacation and exhaustive service. Stoch. Anal. Appl. 28(1), 26–43 (2009)
7. Kempa, W.M.: Characteristics of vacation cycle in the batch arrival queuing system with single vacations and exhaustive service. Int. J. Appl. Math. 23(4), 747–758 (2010)

8. Kempa, W.M.: On departure process in the batch arrival queue with single vacation and setup time. Annales UMCS, Informatica 10(1), 93–102 (2010)
9. Kempa, W.M.: The virtual waiting time in a finite-buffer queue with a single vacation policy. In: Al-Begain, K., Fiems, D., Vincent, J.-M. (eds.) ASMTA 2012. LNCS, vol. 7314, pp. 47–60. Springer, Heidelberg (2012)
10. Lillo, R.E.: Optimal operating policy for an $M/G/1$ exhaustive server-vacation model. Methodol. Comput. Appl. 2(2), 153–167 (2000)
11. Niu, Z., Takahashi, Y.: A finite-capacity queue with exhaustive vacation/close-down/setup times and Markovian arrival processes. Queueing Syst. 31, 1–23 (1999)
12. Niu, Z., Shu, T., Takahashi, Y.: A vacation queue with setup and close-down times and batch Markovian arrival processes. Perform. Evaluation 54(3), 225–248 (2003)
13. Nowicki, R.K., Kempa, W.M., Woźniak, M.: Characteristics of vacation cycle in a finite-buffer queue with single vacation policy - analytical study with optimization. Int. J. Appl. Math. Comp. Sci. (submitted)
14. Takagi, H.: Queueing Analysis, vol. 1: Vacation and Priority Systems, vol. 2. Finite Systems. North-Holland, Amsterdam (1993)
15. Takagi, H.: $M/G/1/N$ queues with server vacations and exhaustive service. Oper. Res. 42(5), 926–939 (1994)
16. Teghem Jr., J.: Control of the service process in a queueing system. Eur. J. Oper. Res. 23, 141–158 (1986)
17. Teghem Jr., J.: Optimal control of a removable server in an $M/G/1$ queue with finite capacity. Eur. J. Oper. Res. 31, 358–367 (1987)
18. Tian, N., Zhang, Z.G.: Vacation queueing models. Theory and applications. Springer, New York (2006)
19. Woźniak, M.: Model of artificial intelligence for active control of selected vehicle dynamic characteristics, PhD. Thesis, Czestochowa University of Technology, Department of Mechanical Engineering and Computer Science, Czestochowa (2012)