Chapter 3

Service System with Emergency Vacation

3.1 Introduction

In the past few years, the study of different types of vacation policies opts by the service provider has put a very exclusive impact on the queueing and service systems. This correlation has also been due to its widespread applications in optimizing expected total cost, manpower, etc. in many practical areas, especially in computer systems, communications systems, flexible manufacturing systems, and supply chain systems, etc. Doshi [63] first surveyed the queueing models with vacation and summarized some general decomposition results with their respective methods. Jain and Agrawal [112] analyzed an Erlangian service $M/E_K/1$ queueing model with the concept of working vacation and determined a steady-state distribution of the number of customers in the system by implementing the generating function approach. Jain and Upadhyay [114] studied the concept of modified Bernoulli vacation schedule with N-policy and unreliable server and obtained queue-size distribution at random and departure points. Using a matrix geometric approach, [107] computed the steady-state queue-size distribution of the various queueing models incorporating several types of disruptions with the service provider's working vacation. In addition, [237] studied the steady-state behavior of the system by employing the supplementary variable technique and developed various system quality indices for the bulk arrivals which follow the Poisson process with varying arrival rates. Jain et al. [113] used the concept of two heterogeneous service providers in machining systems in which the first service provider provides the service according to N-policy whereas the other one has an independence to go for a vacation of the random period. They obtained the

[0]The partial content of this chapter has been published in
C. Shekhar, **S. Varshney**, and A. Kumar, "Optimal Control of a Service System with Emergency Vacation using Bat Algorithm", *Journal of Computational and Applied Mathematics*, vol. 364, pp. 112332, 2020, (Elsevier) (`https://doi.org/10.1016/j.cam.2019.06.048`).

transient-state-probabilities of the governing model by employing the Runge-Kutta method of fourth-order. To examine customers' equilibrium state and to study socially optimal joining balking strategies in a Markovian environment with multiple working vacations, [242] considered three precision levels of the system or delay information, namely, observable queue, the partially observable queue, and the unobservable queue, respectively. Liou [180] modeled a machine repair problem with multiple vacations and the working breakdown and implemented particle swarm optimization (PSO) to find the maximum profit of the developed optimization problem.

Recently, [140] used the queueing-theoretic approach for optimal allocation of manpower in the assembly line of manufacturing systems. Jain et al. [118] characterized the concept of F-policy together with a working vacation for the machining system and achieved the steady-state probabilities using the matrix method. Liu and Wang [182] examined strategic joining behavior of customers for a single-server Markovian queueing model with Bernoulli vacation and investigated Nash equilibrium strategies for fully observable and unobservable cases. Do et al. [62] extended [182] result for equilibrium behavior of customers with working vacations and a constant retrial rate and also explored social optimizing and Nash equilibrium strategies for joining the system using the closed-form representation of stationary probabilities. To ascertain the sensitivity of system performance indices, [108] demonstrated the numerical simulation with different system parameters and determined the optimum service rate and the threshold value (F) via conceptualizing the quasi-Newton method and direct-search method. Lately, [137] envisaged a single-channel finite-buffer queueing model with a general independent stream of customers, exponential processing time, and a working vacation policy and suggested the idea of an embedded Markov-chain and Volterra-type integral equations for computing the time-dependent queue-size-distribution. Recently, the vacation queueing models with classical optimization technique "quasi-Newton method" have been studied by many researchers (*cf.* [301], [131], [290], [126]).

Nowadays, the evolutionary algorithms, nature-inspired metaheuristic methods are extensively used in different scientific, computational, and engineering problems as they elucidate complex problems in an effective manner. The traditional linear and nonlinear optimization techniques cannot be formalized in such problems. Most of the heuristic and metaheuristic algorithms have been imitative from the behavior of physical and biological systems in nature and used Darwin̉s theory of survival of fittest. For example, based on the swarm behavior of bird and fish, the particle swarm optimization (PSO) was developed (*cf.* [67]), while bees algorithm was developed by the inspiration from social behavior of bees (*cf.* [208]). New algorithms are also developed recently, including cuckoo search (*cf.* [36]) and the firefly algorithm (*cf.* [307]) which were developed based on the brood reproductive strategy of cuckoo birds and flashing behavior of fireflies, respectively. Over the last decade, these algorithms have been successfully applied to solving different types of decision-making problems such as queueing problems, inventory problems, scheduling problems, combinatorial & numerical optimization, et cetera (*cf.* [322], [139], [211], [270], [15], [147]).

In this chapter, we employ the recently developed bat algorithm to achieve the optimal expected total cost with optimal values of decision parameters, which was first introduced by [308] and compare with results obtained through well-known metaheuristic techniques particle swarm optimization (PSO) and heuristic techniques quasi-Newton method. The bat algorithm is inspired by the echolocation behavior of microbats. The ability of the echolocation of microbats is much captivating as these bats can find their victim and differentiate between multiple types of insects even in complete darkness. Experimentally and computationally, it is more effective and efficient as compared to existing methods (*cf.* [308], [314], [98]). Due to its efficiency, bat algorithm is applicable in the diverse field of real-life optimization problems (*cf.* [195], [101], [275], [4]).

From the literature survey, it is clear that extensive studies have been done on the problem of service systems with several vacation policies due to their benefits in the service system. But, a research gap has been observed from the literature survey that no study has been done for the realistic feature, the emergency vacation of the service provider, and its damnation and condemnation so far. Inspired by this fact, we have presented this chapter in the general framework, including the concept of a single server's emergency vacation. Motivated by the results of the bat algorithm, we employ this nature-inspired algorithm to optimize the studied model with respect to the governing expected cost. To prove its excellence, a comparison with PSO and the quasi-Newton method has also been made. The main contribution is to develop the algorithm and MATLAB code for comparing the results of bat algorithm (BA), particle swarm optimization (PSO), and quasi-Newton method in terms of computation time, optimal cost, etc. and to suggest optimal design parameters of the studied service system. Also, numerical simulation has been presented to illustrate the effect of various service system parameters on performance indices.

The rest of the chapter is organized in the following manner. In section 3.2, we present assumptions and notations for the mathematical modeling of the studied service model and provide governing steady-state Chapman-Kolmogorov differential-difference equations. In section 3.3, we provide the matrix analytic solution technique to determine the steady-state probabilities vector. With obtained queue-size distribution, various service system performance measures are furnished in section

3.4. In section 3.5, the expected cost function is developed to achieve the optimal values of different decision parameters along with the optimal cost. We use the bat algorithm (BA) to achieve the optimal operating condition with the optimal expected cost of the service system in subsection 3.6.1. The particle swarm optimization algorithm and the quasi-Newton method are compared with the bat algorithm in establishing the excellence of the proposed technique in subsections 3.6.2 and 3.6.3, respectively. Some numerical simulations are executed to validate our modeling in the next section 3.7. Lastly, in section 3.8, conclusion and future perspectives are provided for highlighting the importance of the present study.

3.2 Model Description

In this section, we develop a finite capacity service model based on the emergency vacation policy of a single service provider. The service provider takes the vacation in an emergent critical situation without completing the ongoing service of the waiting customer during the working state in the service system. Apart from that, for modeling purpose, we assume following notations and assumptions

- The customers arrive in the service system according to an independent Poisson process with a mean arrival rate λ. If the service provider is idle, the arrived customer gets service immediately; otherwise, he joins the waiting line in the finite capacity of size K. On the exhaust of waiting space of the service system, the arrived customer is lost, customer.
- The service times of the single service provider are identically and independently exponentially distributed with the mean rate μ. The service provider follows *First Come First Serve* (*FCFS*) service discipline for choosing the customer from the waiting line.
- In the working state, the service provider may take an emergency vacation in any predicament situation without completing the ongoing service of the waiting customer. The occurrence of the emergent situation follows a Poisson process with a mean rate δ.
- The vacation times of the service provider are also identically and independently distributed and follow an exponential distribution with the meantime $1/\theta$.
- All processes and events are repeated all over again and independent of each other.

For the modeling purpose, the states of governing service model at time instant t are defined as

$$J(t)=\begin{cases}0, & \text{Server is on vacation at time instant } t\\ 1, & \text{Server is busy at time instant } t\end{cases}$$

$$N(t)=\text{Number of customers in the service system at time } t$$

Then, $\{J(t),N(t);t\geq 0\}$ represents the continuous-time Markov chain (CTMC) in the state space $\Theta\equiv\{(j,n);\ j=0,1 \text{ and } n=0,1,\ldots,K\}$. Hence, the state-probabilities at time t is outlined as

$$P_{0,n}(t)=\text{Prob}\{J(t)=0,N(t)=n;n=0,1,2,...,K\}$$
$$P_{1,n}(t)=\text{Prob}\{J(t)=1,N(t)=n;n=0,1,2,...,K\}$$

For the steady-state analysis, in equilibrium condition ($t\rightarrow\infty$), we have the steady-state probabilities as follows

$$P_{0,n}=\lim_{t\rightarrow\infty}P_{0,n}(t);\ n=0,1,2,\ldots,K \text{ and } P_{1,n}=\lim_{t\rightarrow\infty}P_{1,n}(t);\ n=0,1,2,\ldots,K$$

where

$P_{0,n}\equiv$ Probability that there are n customers in the service system and the server is on vacation.

$P_{1,n}\equiv$ Probability that there are n customers in the service system and the server is busy.

Now, by using the appropriate axioms of the birth and death process, the governing Chapman-Kolmogorov differential-difference equations are formulated to evaluate the steady-state probabilities associated with various conditions of the service system.

$$-(\lambda+\theta)P_{0,0}+\delta P_{1,0}=0 \tag{3.1}$$
$$-(\lambda+\theta)P_{0,n}+\lambda P_{0,n-1}+\delta P_{1,n}=0;\ 1\leq n\leq K-1 \tag{3.2}$$
$$-\theta P_{0,K}+\lambda P_{0,K-1}+\delta P_{1,K}=0 \tag{3.3}$$
$$-(\lambda+\delta)P_{1,0}+\theta P_{0,0}+\mu P_{1,1}=0 \tag{3.4}$$
$$-(\lambda+\delta+\mu)P_{1,n}+\lambda P_{1,n-1}+\theta P_{0,n}+\mu P_{1,n+1}=0;\ 1\leq n\leq K-1 \tag{3.5}$$
$$-(\delta+\mu)P_{1,K}+\lambda P_{1,K-1}+\theta P_{0,K}=0 \tag{3.6}$$

3.3 Matrix Analytic Solutions

It is extremely difficult to determine the closed-form expressions for the steady-state probabilities $P_{0,n}$ and $P_{1,n}$, $n = 0,1,2,\dots,K$ from the system of equations (3.1)-(3.6) due to multi-variable, multi-equation and multi-parameter stochastic problem. The matrix-analytic technique was introduced by [200] is widely used to deal with such type of queueing problems numerically. We execute the matrix-analytic technique to obtain the probability vectors $\Pi_j = [P_{0,j}, P_{1,j}]$; $j = 0,1,2,\dots,K$ and to improve the computation efficiency.

Now, to solve the flow balance Chapman-Kolmogorov equations for the stationary distribution, we build up the block tri-diagonal transition rate matrix $\mathbf{Q}$ in the following form

$$\mathbf{Q} = \begin{bmatrix} \mathbf{A}_0 & \mathbf{B} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{C} & \mathbf{A}_1 & \mathbf{B} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{C} & \mathbf{A}_1 & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_1 & \mathbf{B} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{C} & \mathbf{A}_1 & \mathbf{B} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{C} & \mathbf{A}_2 \end{bmatrix}$$

where $\mathbf{A}_0$, $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{B}$, and $\mathbf{C}$ are the square matrices of order 2. Matrices $\mathbf{A}_0$, $\mathbf{A}_1$, and $\mathbf{A}_2$ are the generator matrices of the environmental process whereas $\mathbf{B}$, $\mathbf{C}$ are the diagonal matrices having elements λ and μ, respectively due to the characteristic of Markov process. Each block element of the rate matrix $\mathbf{Q}$ is listed as follows

$$\mathbf{A}_0 = \begin{bmatrix} -(\lambda+\theta) & \theta \\ \delta & -(\lambda+\delta) \end{bmatrix}, \mathbf{A}_1 = \begin{bmatrix} -(\lambda+\theta) & \theta \\ \delta & -(\lambda+\delta+\mu) \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} -\theta & \theta \\ \delta & -(\lambda+\delta) \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}$$

and

$$\mathbf{C} = \begin{bmatrix} 0 & 0 \\ 0 & \mu \end{bmatrix}$$

Let Π represents the steady-state probability vector associated with the rate matrix

Q which is partitioned as $\Pi = [\Pi_0, \Pi_1, \Pi_2, \ldots, \Pi_K]$, where each sub-vector $\Pi_j = \left[P_{0,j}, P_{1,j}\right]$ is of dimension (1 × 2). Now, by solving the matrix form of the homogeneous system of linear equations $\Pi Q = \mathbf{0}$ and normalization condition $\Pi e = 1$ where **e** is the column vector having all its elements 1, we find the steady-state probability equations in vector form as

$$\Pi_0 \mathbf{A}_0 + \Pi_1 \mathbf{C} = \mathbf{0} \tag{3.7}$$

$$\Pi_{j-1}\mathbf{B} + \Pi_j \mathbf{A}_1 + \Pi_{j+1}\mathbf{C} = \mathbf{0};\ j = 1,2,\ldots,K-1 \tag{3.8}$$

$$\Pi_{K-1}\mathbf{B} + \Pi_K \mathbf{A}_2 = \mathbf{0} \tag{3.9}$$

Now, after recursively substitution, we get

$$\Pi_0 = \Pi_1 \mathbf{C}\left(-\mathbf{A}_0^{-1}\right) = \Pi_1 \mathbf{X}_0 \tag{3.10}$$

$$\Pi_j = \Pi_{j+1}\left\{-\mathbf{C}\left(\mathbf{X}_{j-1}\mathbf{B} + \mathbf{A}_1\right)^{-1}\right\} = \Pi_{j+1}\mathbf{X}_j;\ j =, 1,2,\ldots,K-1 \tag{3.11}$$

Again, we can express the steady-state probability vector Π_j in the product form of $\mathbf{X}_j;\ j = 0,1,2,\ldots,K-1$ as

$$\Pi_j = \Pi_K\left[\mathbf{X}_{K-1}\mathbf{X}_{K-2}\mathbf{X}_{K-3}\cdots\mathbf{X}_{j+2}\mathbf{X}_{j+1}\mathbf{X}_j\right];\ j = 0,1,2,\ldots,K-1$$

$$\Pi_j = \Pi_K \prod_{n=K-1}^{j} \mathbf{X}_j = \Pi_K \Phi_j;\ j = 0,1,2,\ldots,K-1 \tag{3.12}$$

Let $\mathbf{u} = [1,\, 1]^T$ be a column vector of order 2 having all entries 1. The normalization condition is re-expressed as

$$\sum_{j=0}^{K} \Pi_j \mathbf{u} = 1$$

Hence,

$$[\Pi_0 + \Pi_1 + \Pi_2 + \cdots + \Pi_{K-1} + \Pi_K]\,\mathbf{u} = 1$$

$$[\Pi_K\Phi_0 + \Pi_K\Phi_1 + \Pi_K\Phi_2 + \cdots + \Pi_K\Phi_{K-1} + \Pi_K]\,\mathbf{u} = 1$$

$$\Pi_K\,[\Phi_0 + \Phi_1 + \Phi_2 + \cdots + \Phi_{K-1} + \mathbf{I}]\,\mathbf{u} = 1$$

So, finally we get the closed form expression of the normalization condition as

$$\Pi_K \left[\sum_{j=0}^{K-1} \Phi_j + \mathbf{I}\right] \mathbf{u} = 1 \tag{3.13}$$

Therefore, from eq^n(3.9), the numerical value of Π_K can be easily determined. Hence, the numerical value of all other steady-state probabilities is also obtained by substituting Π_K in eq^n(3.12). From these steady-state probabilities, we develop some standard performance measures and cost function to validate our modeling and methodology.

3.4 System Performance Measures

To determine the characteristics of the service system and examine the performance, we delineate various service system performance measures in terms of the steady-state probabilities of various states of the service system obtained in the previous section. Let $\mathbf{u}_1$ and $\mathbf{u}_2$ be column vectors of order 2 having element $[1,0]^T$ and $[0, 1]^T$ respectively. The closed-form expressions are as follows

- Expected number of customers in the service system

$$L_S = \sum_{j=0}^{K} j\Pi_j\mathbf{u} \tag{3.14}$$

- Expected number of customers in the queue

$$L_Q = \sum_{j=1}^{K} (j-1)\Pi_j\mathbf{u} \tag{3.15}$$

- Probability that the service provider is on emergency vacation

$$P_V = \sum_{j=0}^{K} \Pi_j\mathbf{u}_1 \tag{3.16}$$

- Probability that the service provider is in busy state

$$P_B = \sum_{j=0}^{K} \Pi_j\mathbf{u}_2 \tag{3.17}$$

- Throughput of the service system

$$\tau_P = \sum_{j=1}^{K} \mu\Pi_j\mathbf{u}_2 \tag{3.18}$$

- Expected waiting time of the customers in the service system

$$W_S = \frac{L_S}{\lambda} \tag{3.19}$$

3.5 Cost Function

The system designers and decision-makers are, in general, interested in determining the optimal values of the decision parameters to reduce the expected total cost of the service system. Therefore, we develop a cost function by considering various cost elements incurred in various situations and two decision parameters (μ, θ), which are continuous. Following cost elements associated with various activities are considered and defined as follows

$C_h \equiv$ Holding cost incurred for each customer present in the service system

$C_b \equiv$ Cost incurred for the working server in a busy state

$C_m \equiv$ Cost incurred to provide the service to each customer with rate μ

$C_t \equiv$ Cost incurred by the service system when the service provider is on vacation

$C_d \equiv$ Cost incurred when the server opts the emergency vacation

Using the above-defined cost elements, we develop the cost function as follows

$$TC(\mu,\theta) = C_h L_S + C_b P_B + C_m \mu + C_t \theta + C_d \delta \tag{3.20}$$

Cost minimization problem of the conceived model can be mathematically described as an unconstrained problem as follows

$$TC(\mu^*, \theta^*) = \min_{(\mu,\theta)} TC(\mu, \theta) \tag{3.21}$$

3.6 Optimal Analysis

The most real-world service system's optimization problem is highly nonlinear under various complex constraints. Different cost elements are often conflicting, and even for a single objective, sometimes, the optimal solution may not exist at all. The existing classical optimization techniques are not suitable for such problems. We need some efficient alternative algorithms for finding the near-optimal solution to such a problem. The alternative algorithm must be gradient-free, or gradient can be computed numerically to overcome the difficulty of computation of the gradient of such a complex issue. Algorithms with stochastic mechanisms were often denoted as heuristic techniques in the past. However, recent literature tends to refer to them as metaheuristics. Likewise, all metaheuristic algorithms utilize a specific tradeoff of randomization and neighborhood search. Quality solutions to severe optimization problems can be found in a reasonable amount of time; however, there is no assurance that optimal solutions can be reached. In this chapter, we employ the latest metaheuristic method bat algorithm for the optimal analysis, and results are compared with the finding of the well-known metaheuristic method, particle swarm optimization (PSO), and semi-classical optimization technique quasi-Newton method under the same governing design parameters of service system. For the economic analysis purpose, the system design parameters μ, θ & the cost function TC are used in place of x_1, x_2 and f respectively in the following subsections.

3.6.1 Bat Algorithm

For implementing the bat algorithm for the governing cost optimization problem, refer the section 1.10.5.

3.6.2 Particle Swarm Optimization

For the brief study and assumptions of the PSO algorithm, refer the section 1.10.3.

3.6.3 Quasi-Newton Method

For implementing the semi-classical optimization technique, the quasi-Newton method, and viewing its iterative steps, refer the section 1.10.1.

3.7 Numerical Results

In this section, we present the quantitative evaluation for the queue-based service system depending on the service provider's emergency vacation policy by examining various system performance indices in the context of several decision parameters, especially the vacation time, service rate, etc. By depicting numerical simulation results through multiple graphs and table, we validate our formulation and methodology which is helpful for the decision-makers in the designing of the real-time service system as an electrical grid system, evaluation, and operation of the system and in improving the system performance, etc.

For the queueing analysis purpose, we fix the default values of several system parameters as follows: $K = 15$, $\lambda = 1.5$, $\mu = 2.5$, $\delta = 0.05$, and $\theta = 2$. Using the MATLAB programming language and applying the matrix method, we obtain the steady-state probabilities of various states of the service system.

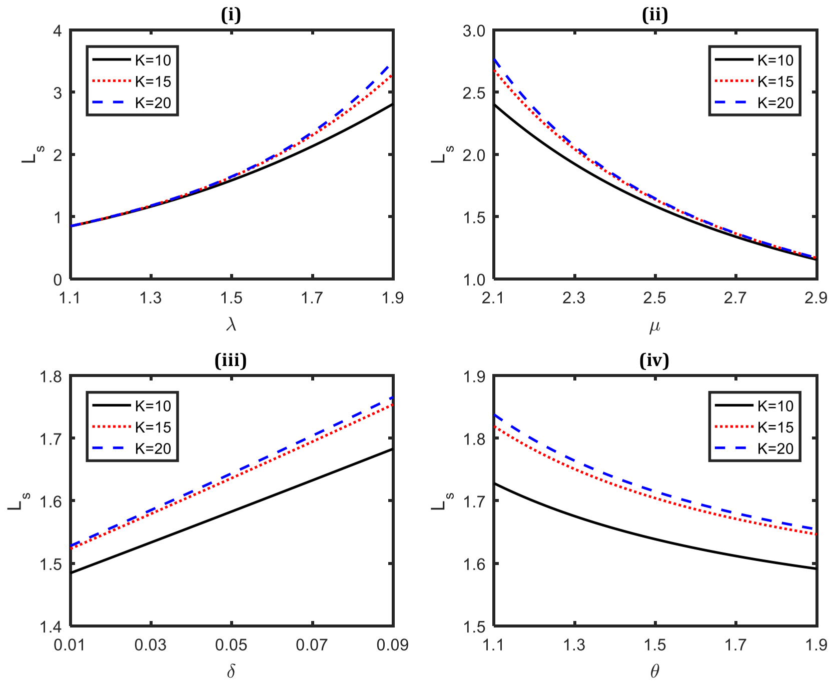

Figure 3.1: Expected number of customers in the service system (L_S) wrt (i) λ, (ii) μ, (iii) δ, and (iv) θ for different values of K.

For the parametric analysis of the expected number of the customers in the service system (L_S), in Fig. 3.1, we vary the values of governing parameters namely the arrival rate of customers in the service system (λ), service provider's service rate (μ), the rate of occurrence of the emergent situation (δ) and vacation meantime ($1/\theta$) respectively for different values of system capacity (K). For the incremental change in the values of λ and δ, the value of the expected number of customers in the service system (L_S) is clearly increased. Similarly, as we increase the value of μ and θ, the value of L_S decreases, as intuitively expected.

Fig. 3.2 shows the effect of parameters λ, μ, δ, and θ along with the different values of the capacity of the service system (K) on the throughput of the service system (τ_P). In Fig. 3.2(ii) and 3.2(iv), we observe that as the value of parameters μ and θ increases with the increased values of K, the value of the expected number of served customers *i.e.* throughput of the service system (τ_P) increases. Similarly, the throughput (τ_P) decreases on increasing the value of δ. It shows the impact of the emergency vacation on the service system. From the present study, we suggest the provision of a standby service facility in the service system for uninterrupted service facility. In addition, for a fixed value of δ, the throughput of the service system increases for an increasing value of K, which is quite obvious.

The variability in the expected total cost (TC) defined in the eq^n(3.20) wrt various system parameters (K, λ), (μ, θ), (δ, μ) and (δ, θ) is demonstrated in Fig. 3.3.

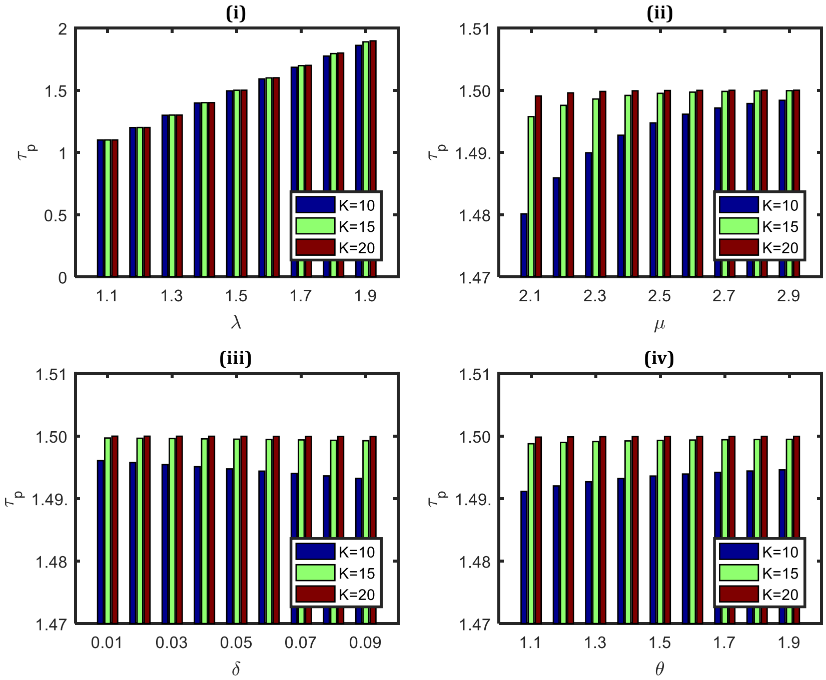

Figure 3.2: Throughput of the service system (τ_P) wrt (i) λ, (ii) μ, (iii) δ, and (iv) θ for different values of K.

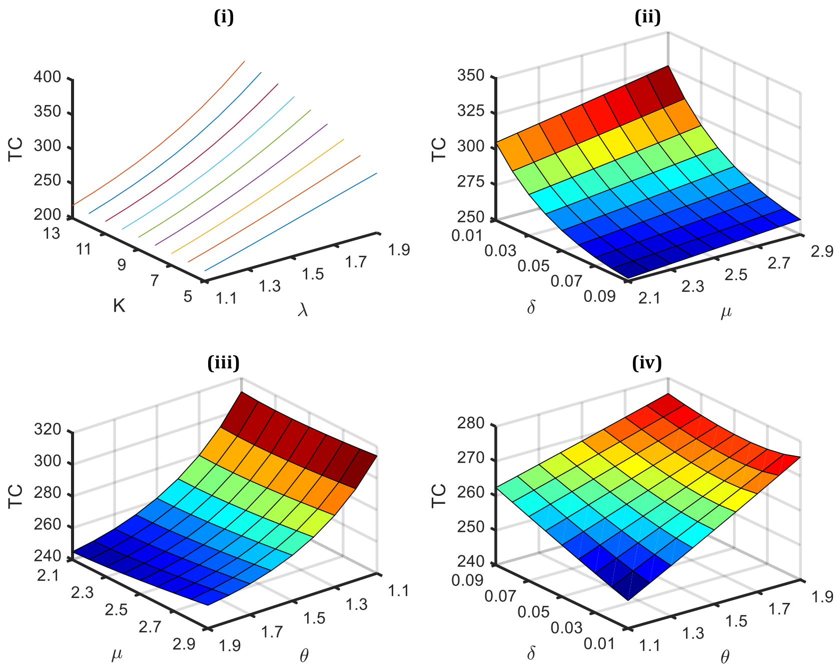

Figure 3.3: The total expected cost of the service system (TC) wrt system parameters (i) (K,λ), (ii) (δ,μ), (iii) (μ,θ), and (iv) (δ,θ).

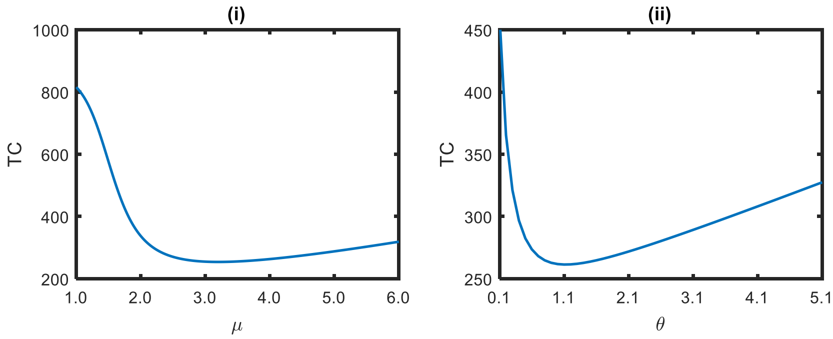

Figure 3.4: Convex nature of the total expected cost function (TC) wrt the decision parameters (i) μ, and (ii) θ.

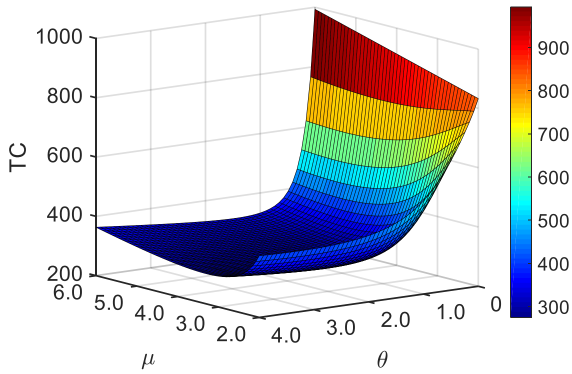

Figure 3.5: Surface plot of expected cost function (TC) wrt μ, and θ.

For the Fig. 3.3, we take the default values of several system parameters similar to Figs. 3.1–3.2 along with several unit cost elements associated with their respective performance indices as $C_h = 50$, $C_m = 40$, $C_t = 20$, $C_b = 80$, and $C_d = 10$. It is easily depicted that the expected total cost (TC) varies subsequently with the incremental change in the parameters. It is noticed from Fig. 3.3(i) that the increasing values of the expected total cost are obtained with the incremental changes in K and λ that is quite obvious. The value of expected total cost (TC) decreases with the increasing values of θ and μ as observed in Fig. 3.3(iii). It prompts that just increasing the service facility always not renders the better service. For the optimal service as per cost and throughput, optimal design parameter(s) have to be determined. It also depicts the convex curvature of TC which inspire us to determine the optimal value of service system design parameter(s).

Fig. 3.4 prompts the convex nature of the expected total cost function (3.21) with respect to decision parameter(s). In Fig. 3.4, the optimal values of μ^* and θ^* for the default value of system parameters and unit cost elements similar to Fig. 3.3 are almost nearby 3.2 and 1.1 respectively. For employing the bat algorithm, in Fig. 3.5,

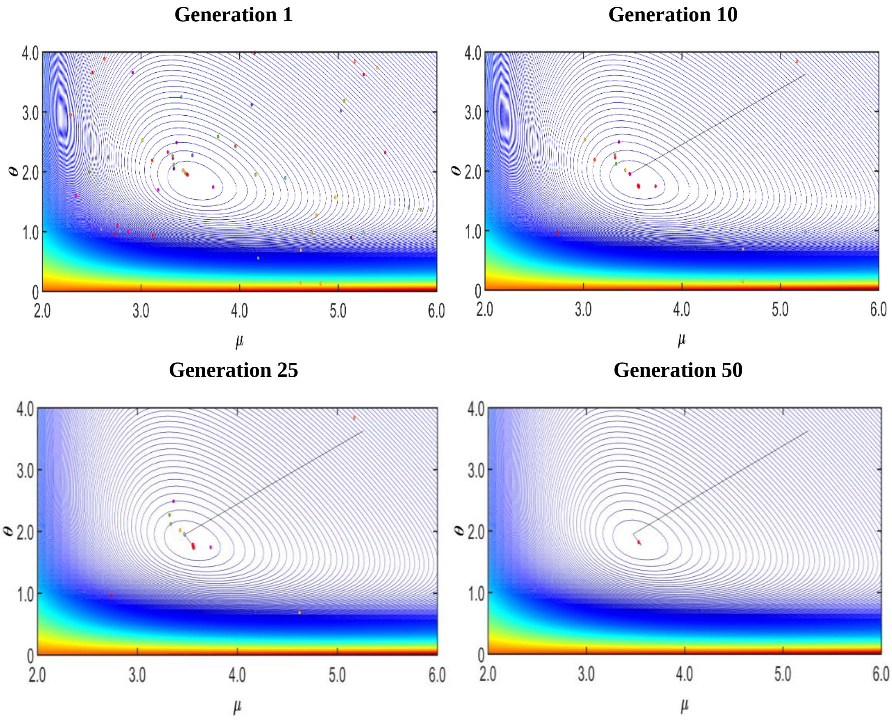

Figure 3.6: Various generations of bat algorithm (BA).

we portray the three-dimensional surface plot generated through the joint variation of μ and θ for the following default values of service system parameters $K = 15$, $\lambda = 1.5$, & $\delta = 0.3$ and associated cost elements as in Fig. 3.3. It also prompts the convex nature of TC wrt μ and θ. As per the restriction of service system resources, the analyst can fix design parameters for the optimal service cost.

We set the lower and upper bounds of both decision parameters μ and θ as [2 6] and [0 4] and the default parameters of bat algorithm as $\rho = 0.01$, $\varepsilon \in [-1, 1]$, and $\vartheta \in [0, \in 1]$, respectively. In Fig. 3.6, we delineate some selected generations of bat algorithm in the search space for illustrative purpose and determine the optimal values of decision parameters besides optimal expected total cost. Because the bat algorithm is an agent-based optimization technique, we easily observe that in an initial generation (generation 1) all the search agents randomly scattered in the whole space and come closer and closer by exploring the untouched region as in generations 10, 25 and 50. This implies the robustness of bat algorithm for all such experiments and shows the capability of bat algorithm to converge to the optimum within a reasonable time. With the aid of bat algorithm, we obtain the coordinates of the best position of the best agent as $[\mu^*, \theta^*] = [3.531007, 1.815887]$ along with the corresponding optimal value of the fitness function *i.e.* minimal expected total cost $TC^* = 275.026994$.

For the depth analysis, we experiment for varied values and results are tabulated in Table 3.1. We describe all numerical experiments for bat algorithm by taking

Table 3.1: Optimal values of (μ^*, θ^*) with minimal expected cost TC^* using bat algorithm.

(K, λ, δ)	μ^*	θ^*	TC^*	Mean $\left\{\frac{TC}{TC^*}\right\}$	Max $\left\{\frac{TC}{TC^*}\right\}$	CPU time
(15, 1.5, 0.05)	3.287475	0.944954	241.122689	1.0000169493	1.0000508477	83.9457
(20, 1.5, 0.05)	3.291865	0.954716	241.276807	1.0000098567	1.0000278245	108.4874
(25, 1.5, 0.05)	3.292196	0.959599	241.297391	1.0000157655	1.0000472966	133.2196
(15, 2.0, 0.05)	4.016472	1.054427	285.087039	1.0000252268	1.0000756804	81.9455
(15, 2.5, 0.05)	4.711412	1.141210	325.658005	1.0000000112	1.0000000335	83.3716
(15, 1.5, 0.10)	3.354009	1.216581	251.103217	1.0000030215	1.0000077631	86.8598
(15, 1.5, 0.20)	3.452486	1.565497	264.722735	1.0000000027	1.0000000082	83.3169
(15, 1.5, 0.30)	3.531007	1.815887	275.026994	1.0000000002	1.0000000006	82.6382

50 search agents, 50 generations and 20 runs for each experiment independent to the others. For confirming the bat algorithm' robust nature, we also present the statistical parameters namely mean and maximum of the ratio of optimal TC in all runs and optimal TC in each. From Table 3.1, we observe that the mean and maximum values of $\left[\frac{TC}{TC^*}\right]$ varies from 1.0000000002 to 1.0000756804. This shows that the searching quality of bat algorithm is very well for all test instances.

Next, a comparative analysis between the bat algorithm, PSO algorithm, and the quasi-Newton method is performed with the varied values of system parameters in Tables 3.1–3.3 for the validity of the proposed algorithm with respect to well-known algorithms. The default parameters of PSO algorithm are taken as $\kappa_1 = 2$, $\kappa_2 = 2$, and $\omega_2 = 0.5$, respectively. Computation time and optimal result are key factors for comparision of the effieciency and utility of the algorithm. It is clearly observed that the approximate optimal solutions for decision parameters and the corresponding minimal expected cost obtained by the bat algorithm, PSO algorithm, and quasi-Newton method are very close to each other. The CPU time (in seconds) for BA algorithm is lesser than the PSO algorithm and corresponding minimal expected cost obtained by bat algorithm converges significantly in more effective manner for all test instances as well. This shows that the efficiency of bat algorithm is higher than the PSO algorithm because the PSO algorithm has numerous calculations/computations in updating the candidate solution. The results of bat algorithm is also better than semi-classical quasi-Newton method for each experiment. Newton method requires the computation of gradient or direction of optimality which we obtain numerically due to complex nature of the problem. It involves the high grade of approximation which lowers its efficiency.

From the above experiments and discussions, we ascertain that our proposed bat algorithm is superior to PSO and quasi-Newton, which we, in general, employ a lot for optimization problems having high-grade of non-linearity. It is also observed that the optimal design parameter setup is vital to minimize the expected total cost incurred in providing the service to the customers.

3.8 Conclusion and Future Prospective

The novelty of the present study is to introduce the new vacation policy, emergency vacation, and to observe its effects on performance through the queue-theoretic approach. In this chapter, we examine the impact of the emergency vacation of the service provider in a finite capacity service system. To obtain steady-state probabilities, we delineate the Chapman-Kolmogorov differential-difference system of

Table 3.2: Optimal values of (μ^*, θ^*) with minimal expected cost TC^* using PSO algorithm.

(K, λ, δ)	μ^*	θ^*	TC^*	Mean $\left\{\frac{TC}{TC^*}\right\}$	Max $\left\{\frac{TC}{TC^*}\right\}$	CPU time
(15, 1.5, 0.05)	3.287345	0.944665	241.122691	1.0000001162	1.0000003168	83.9956
(20, 1.5, 0.05)	3.291571	0.957683	241.276809	1.0000000929	1.0000001577	108.4139
(25, 1.5, 0.05)	3.292297	0.959511	241.297391	1.0000001632	1.0000003633	133.9529
(15, 2.0, 0.05)	4.015875	1.053792	285.087057	1.0000000684	1.0000001178	82.0087
(15, 2.5, 0.05)	4.711580	1.141199	325.658006	1.0000002071	1.0000004207	83.3897
(15, 1.5, 0.10)	3.353806	1.216816	251.103218	1.0000000164	1.0000000277	83.8269
(15, 1.5, 0.20)	3.452291	1.565708	264.722736	1.0000000346	1.0000001031	83.7293
(15, 1.5, 0.30)	3.530982	1.816141	275.026996	1.0000000604	1.0000000146	82.7166

Table 3.3: Optimal values of (μ^*, θ^*) with minimal expected cost TC^* using quasi-Newton method.

(K, λ, δ)	(15,1.5,0.05)	(20,1.5,0.05)	(25,1.5,0.05)	(15,2.0,0.05)	(15,2.5,0.05)	(15,1.5,0.1)	(15,1.5,0.2)	(15,1.5,0.3)
(μ_0, θ_0)	(4,2)	(4,2)	(4,2)	(4,2)	(4,2)	(4,2)	(4,2)	(4,2)
Total Iteration	7	8	8	8	8	8	6	6
μ^*	3.287408	3.291688	3.292243	4.016466	4.711430	3.354016	3.452469	3.531034
θ^*	1.944937	0.957355	0.959616	1.054416	1.141150	1.216619	1.565514	1.815838
$TC(\mu^*, \theta^*)$	241.122835	241.276808	241.297391	285.087040	325.658010	251.103217	264.722736	275.026995

equations and employ the matrix-analytic method. We deduce various system performance indices and also develop the expected total cost function of the service system. The numerical simulation of several system performance measures has also been done to study the effects of different system parameters. In the end, various numerical illustrations have provided to illustrate and achieve optimal solutions. We create the unconstrained cost optimization problem and find the optimal values of decision parameters (μ, θ) using a bat algorithm, which is metaheuristic in nature. The comparison between bat algorithm, PSO algorithm, and quasi-Newton method has also been undertaken to achieve the optimal operating condition with minimal expected cost. We ascertain that the proposed bat algorithm is more efficient for the optimal analysis of optimization problem involved in such a real-time problem as compared to some well-known algorithms.

Moreover, the analysis of the expected total cost of the service system exhibits the validity and profitability of the developed model very effectively, which will be helpful in reducing the cost of maintenance by system designers and decision-makers, which is the highly desirable feature of any organization. This work can be further extended by incorporating some hypotheses like the bulk arrival and machining system etc.