



A Chemical Queueing Framework for Bulk Service Systems with Rework, Starting Failure and Server Vacations: An Adaptive Neuro-Fuzzy Inference System Approach

R. LOKESH¹ and S.KARPAGAM^{2*}

^{1,2}Department of Mathematics, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai-600062, India.

*Corresponding author E-mail: karpagammaths19@gmail.com

<http://dx.doi.org/10.13005/ojc/410540>

(Received: May 17, 2025; Accepted: October 25, 2025)

This paper describes a chemical queueing framework for modeling bulk service systems that include important operational aspects of starting failure, rework, inspection, repair, and a number of vacations. By applying queueing theory to chemical and pharmaceutical production situations, the model accounts for the real-time dynamics of batch processing systems compared to the deterministic approaches. Using the supplementary variable technique, steady-state analysis was used to identify stability criteria. Additionally, a cost model and a few performance metrics are developed. Numerical illustrations are provided to help visualize the impact of various characteristics of the system. In addition, the Adaptive Neuro-Fuzzy Inference System (ANFIS) is used for the validation and prediction of the system behavior.

Keywords: Chemical queueing, Bulk service, Rework, Starting failure, ANFIS. MSC: 60K25, 68M20, 90B22.

INTRODUCTION

Chemical queueing is a growing interdisciplinarity that brings some concepts of queueing theory, which have been applied previously to computer science, operations research, and telecommunications, into the modeling of chemical processes and reaction engineering. Chemical species, reaction stages, and process controls are elements of a structured queueing system in this approach. In the same way that jobs join a queue waiting to be handled by servers, the molecules arrive at a reaction site (server) and react through well-

defined mechanisms to produce products (output). Each reaction step, each catalyst interaction, or phase transition is then a service process, with the kinetics as service rates and concentration profiles as queue lengths.

The use of queueing theory for chemical systems has been gradually gaining momentum as researchers look for more dynamic and realistic tools to describe reaction behavior beyond the classical rate equations. Tsitkov *et al.*,¹ presented a queueing-based method of considering enzyme cascade reactions, explaining that queue models exhibit



superior capability to represent discrete stochasticity of enzyme activity relative to conventional continuous framework work. Similarly, Kandemir-Cavas *et al.*,² used the theory of queues to biosynthesize caulerpenyne, a toxic secondary metabolite produced from *Caulerpa taxifolia*, effectively modeled fluctuations in metabolite levels, and predicted the time to reach maximum concentration.

Conolly *et al.*,³ provided contributions to discussions on the foundations of chemical queueing through a direct modeling approach and prepared the ground for later formalizations. Following this, Tarabia *et al.*,⁴ developed a new power series approach to obtain the transient state distributions in chemical queueing systems, which expands ordinary queue setups, such as the M/M/1/ ∞ model, in broader chemical settings. Extending this further, Alshreef and Tarabia⁵ included more complex actions, including catastrophic failures and server repairs, through generating functions and Laplace transforms to give robust analytical expressions for transients dynamics.

Starting failure was a very important aspect of the queueing systems studied in the present papers, and it occurred while the customer arrived and the server did not start the service. Shanmugam and Saravananarajan⁶ examined a model, an unreliable server was prone to starting failure upon arrival or departure of a new customer from the orbit, where the customer was sent to the orbit to be serviced later. This concept was also discussed in the Karpagam and Ayyappan⁷ bulk queueing system study, where starting failure was included in addition to repair, multiple vacations and a stand-by server. Ayyappan *et al.*,⁸ proposed a mitigation strategy such that the batch of customers only when the main server was in repair state due to a starting failure.

The bulk queueing system in the manufacturing research of Karpagam *et al.*,⁹ which included a starting failure, showed its applicability in other production/manufacturing systems where the rework of defective items was needed. The study of Venkatesan and Paramasivam¹⁰ was concerned with energy saving approaches for cellular networks, which introduced a Markov-based model for base station sleep strategies that considered disaster scenarios and repairable servers. Their model showed that the power savings at base stations increased with light sleep cycles. In contrast, the article by

Ayyappan and Meena¹¹ presented an integrated general single-server queueing model with multiple states, such as server vacation, repair, breakdown, degrading service, starting failure, and close-down. For additional information on the approach of starting failures, researchers can read¹²⁻¹⁶.

Arivudainambi and Gowsalya¹⁷ have studied an M/G/1 retrial queue with Bernoulli vacation, two types of service, and starting failure. They gave necessary and sufficient conditions for steady state and solved the model by the supplementary variable technique. Ke and Chang¹⁸ analyzed general retrial times, starting failures, heterogeneous service, Bernoulli vacation schedules batch arrival retrial queues. In their model, some of the features were server breakdowns and orbital queueing for batch arrivals. They obtained steady-state distribution for the state of the server and the number of customers in the system/orbit. Mokaddis *et al.*,¹⁹ investigated an M/G/1 retrial queue with Bernoulli feedback and a single vacation where the starting failures were assumed.

Das and Pradhan²⁰ have analyzed server failures, close-downs, and multiple vacations due to low availability of customers, obtaining simplified probabilities and performance figures. In contrast, Sikdar²¹ considered a Geom/G/1/K system with L-limited service and multiple vacations, where the server, having Bernoulli arrivals, went on vacation when the system was empty or after serving L packets. Karpagam *et al.*,²² studied an M^[X]/G/1 queue with optional second service, feedback, and Bernoulli vacation to consider batch arrivals and flexible service.

Mahanta *et al.*,²³ introduced a more comprehensive single-server model that had two service representatives. Three post-service options available for customers at this model were presented as below: re-service, feedback, or exiting the system. Karpagam²⁴ considered a bulk service queue with rework on faulty items, server breakdowns, and repairs, so that the server becomes operational after the completion of the repairs. The implication of this model is of specific relevance to the manufacturing and production industries. Kalaiselvi and Saravananarajan²⁵ suggested a single-server retrial queue model, with optional re-service and working vacations, which used advanced cost optimization methods to increase operational efficiency.

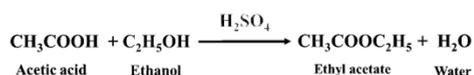
Marjasz *et al.*,²⁶ investigated a finite-buffer queueing system subjected to batch arrivals and multiple vacation policies in the context of system resilience and departure dynamics. Their research considers the application of stress testing and predictive analysis to analyze how different parameters, such as the duration of vacations, influence system behavior. Haridass and Arumuganathan²⁷ studied a bulk queueing system that features multiple vacations, setup time, and the server's choice of admitting reservice. In their model, the server goes on vacation if the queue length is lower than some threshold and needs time to be set up after getting out of a vacation. Jeyakumar and Arumuganathan²⁸ examined a non-Markovian queue with multiple vacations, as well as a controlled optional re-service.

Mechanistic Flow of Bulk Chemical Reactions

A bulk esterification reaction, where acetic acid (CH_3COOH) reacts with ethanol ($\text{C}_2\text{H}_5\text{OH}$) to give ethyl acetate ($\text{CH}_3\text{COOC}_2\text{H}_5$) and water (H_2O), can be visualized as a queueing model. The arrival phase concerns the input of molecules of reactants, acetic acid and ethanol, which are poured into the reactor in controlled amounts. The chemical formula of these molecules is shown below:

- Acetic Acid (CH_3COOH)
- Ethanol ($\text{C}_2\text{H}_5\text{OH}$)

The server in this context is the chemical reactor, where the reaction occurs under the influence of a catalyst such as sulfuric acid (H_2SO_4). The molecular interaction proceeds as follows:



In this, the ethyl acetate ($\text{CH}_3\text{COOC}_2\text{H}_5$) produced is the product and the water (H_2O) is the by-product. Once the reaction is over, the resultant output, which is ethyl acetate and water, goes under inspection to ensure that it is pure. The products are tested on whether they are at the required standards because, if they are not, they undergo a rework process that may include the addition of more reactants or reworking to attain the required purity and yield.

Detailed model description

The proposed model is a bulk service queueing system, which has fundamental mechanisms

like regular service, inspection, rework, starting failure, repair, and multiple vacations running under the General Bulk Service Rule (GBSR), in which jobs are served in groups with a first-come-first-served (FCFS) discipline, where jobs are served in the order of their arrival. The server starts regular service only if the queue length $Q \geq a$. Once the service is completed, the batch is subjected to an inspection. When the batch is identified as faulty with probability ' γ ', it is taken for the rework process. After the rework has been completed, the server re-checks the queue:

- If $Q \geq a$, it immediately proceeds to the next service.
- If $Q < a$, the server takes a vacation.

In the case of a non-faulty batch (with probability ' $1-\gamma$ '), the server also checks the queue. If the number of waiting for jobs is large enough (that is, $Q \geq a$), the server proceeds with the next batch. Otherwise, it goes on vacation. When the server is on vacation, if the queue size remains less than a , the server stays idle or continues in vacation mode for a greater number of arrivals. However, if at the end of vacation the queue length exceeds at least ' a ', the server tries to continue service. At this point, there is a probability ' δ ' of failure to start. When faced with a starting failure, the server is taken for repair, which makes it active and provides regular service. If there is no failure (probability ' $1-\delta$ '), the server will start its service without interruption.

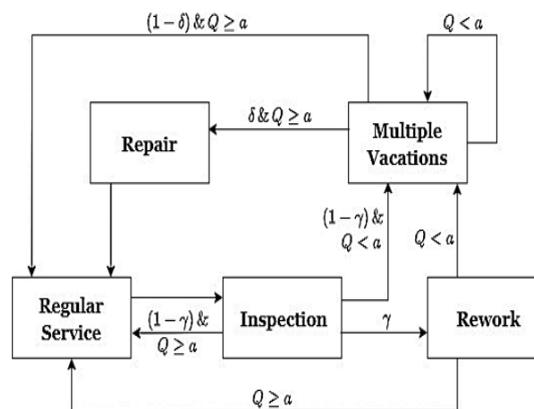


Fig. 1. Schematic representation of the model

All operational times, such as normal service, inspection, rework, vacation, repair, etc., are also assumed to be of general (arbitrary) distributions, rendering the model relevant to real systems where there is bulk processing with concern for reliability and control of quality. The schematic representation of the model shown in Figure 1.

Implementation of the model in real Life

In the pharmaceutical industry, the bulk queueing system model is the most efficient tool to process the flow of raw materials, the drug production process, quality inspection and rework. This model combines the main ideas of several areas of chemistry (especially organic, inorganic, pharmaceutical, analytical, material, and green chemistry) and includes operational dynamics such as the vacation period, the start failures, and the repair strategies reflecting the actual production situations.

The system starts with the arrival of raw materials, which are batched in the form of active pharmaceutical ingredients (APIs), solvents, and excipients. These are materials that queue up until some predetermined threshold 'a' is reached, which causes the start of regular service operations by the server. If the threshold 'a' isn't met, the server goes into vacation mode, which is applied in pharmaceutical plants for processes, clean-in-place (CIP), and sterilization-in-place (SIP)-guided by green chemistry to reduce the use of solvents and environmental impact. When the threshold is eventually achieved, the server is preparing to start regular service. However, there may be two scenarios possible at the point of service initiation:

Starting Failure Occurs: In some instances, the server may not start due to a mechanical, chemical, or calibration problem. This may occur due to the inadequacy of controlling temperature, inactivity of catalysts, malfunctioning of valves, or system errors of sensors. When a starting failure is detected, the server switches to a repair mode that entails failure diagnosis and cure. For example, catalyst chemistry could provide for resubstitution or reactivation of the catalyst, while electrochemistry and material chemistry may lead to restoration of the pH sensors or metal components. Once the repair is completed, the server starts the batch service.

No Starting Failure: When all conditions are normal (correct pressure, temperature and reagent levels, and operating equipment), the (server) starts service without a hitch. The batch undergoes such processes as mixing, granulation, or synthesis according to organic chemistry and pharmaceutical chemistry. These may take place by esterification, acetylation, or hydrogenation processes, depending on the drug to be prepared.

After processing, the batches are inspected by using analytical and biophysical chemistry methods such as HPLC, UV-Vis, FTIR, and NMR to determine the purity, potency, and structural integrity of the products. When a batch fails inspection, rework operations are ramped up, getting the chemistry and thermochemistry of solutions to make the batch conform to specification. After rework, batches are transferred into a packaging and storage area, with the assumption of conformance to predefined recovery standards and procedures.

During the process, chemoinformatics and computational chemistry help in the prediction of batch performance, optimization of reaction pathways, and scheduling operations effectively. This superior model, including handling of vacation time, threshold-based batch control, potential starting failures, and repair actions, offers an extremely realistic and chemically integrated perspective of contemporary pharmaceutical production. It guarantees not only the efficiency of the works but also compliance with Good Manufacturing Practices (GMP) and sustainability of the environment.

Notations

Let X be the batch size random variable, and g_i be the probability of 'i' materials that arrive in a batch, using $X(z)$ as the probability generating function. The number of jobs present at the service station and queue at time 't' is denoted by N_s and N_q , respectively. The notation are listed below in Table 1.

Table 1: Notations

	Cumulative Distribution Function	Probability Density Function	Laplace-Stieltjes Transform	Remaining Time
Regular service	$S_m(v)$	$S_m(v)$	$S_m(\vartheta)$	$S_m^0(t)$
Inspection	$S_q(v)$	$S_q(v)$	$S_q(v)$	$S_q^0(t)$
Rework	$S_r(v)$	$S_r(v)$	$S_r(v)$	$S_r^0(t)$
Repair	$S_i(v)$	$S_i(v)$	$S_i(v)$	$S_i^0(t)$
Vacation	$W(v)$	$W(v)$	$W(v)$	$W^0(t)$

$B(t) = 1, 2, 3, 4$ and 5 denotes the server in regular service, inspection, rework, repair, and vacation respectively.

$\xi(t) = k$, if the server is on k^{th} vacation.

We define the following state probabilities as follows :

$$M_{i,h}(v,t)\Delta t = \Pr\{\xi_s(t)=i, \xi_q(t)=h, v \leq S_m^0(t) < v+\Delta t, B(t)=1\},$$

where $a \leq i \leq b, h \geq 0$;

$$Q_h(v,t)\Delta t = \Pr\{\xi_q(t)=h, v \leq S_q^0(t) < v+t, \chi(t)=2\}, \text{ where } h \geq 0;$$

$$R_h(v,t)\Delta t = \Pr\{\xi_r(t)=h, v \leq S_r^0(t) < v+t, \chi(t)=3\}, \text{ where } h \geq 0;$$

$$F_i(v,t)\Delta t = \Pr\{\xi_f(t)=i, v \leq S_f^0(t) < v+\Delta t, \chi(t)=4\}, \text{ where } i \geq \alpha;$$

$$V_{k,h}(v,t)\Delta t = \Pr\{\xi_q(t)=h, v \leq W^0(t) < v+\Delta t, \chi(t)=5, \xi(t)=k\},$$

where $h \geq 0, k \geq 1$.

Steady state queue size distribution

Cox²⁹ established the supplementary variable approach. Using the supplementary variable approach, we establish the steady state difference-differential equations for the suggested model at different time epochs in the way described below:

$$-M_{i,0}(v) = -\lambda M_{i,0}(v) + R_i(0)s_m(v) + (1-\gamma)Q_i(0)s_m(v) + F_i(0)s_m(v)$$

$$+ (1-\delta) \sum_{k=1}^{\infty} V_{k,i}(0)s_m(v), \quad a \leq i \leq b, \quad (1)$$

$$-M_{i,h}(v) = -\lambda M_{i,h}(v) + \sum_{l=1}^h M_{i,h-l}(v)\lambda g_l, \quad h \geq 1, \quad a \leq i \leq b-1 \quad (2)$$

$$-M_{b,h}(v) = -\lambda M_{b,h}(v) + R_{b+h}(0)s_m(v) + (1-\gamma)Q_{b+h}(0)s_m(v) + F_{b+h}(0)s_m(v)$$

$$+ (1-\delta) \sum_{k=1}^{\infty} V_{k,b+h}(0)s_m(v) + \sum_{l=1}^h B_{b,h-l}(v)\lambda g_l, \quad h \geq 1, \quad (3)$$

$$-Q_0(v) = -\lambda Q_0(v) + \sum_{l=a}^b M_{l,0}(0)s_q(v), \quad (4)$$

$$-Q_h(v) = -\lambda Q_h(v) + \sum_{l=a}^b M_{l,h}(0)s_q(v) + \sum_{l=1}^h Q_{h-l}(v)\lambda g_l, \quad h \geq 1 \quad (5)$$

$$-R_0(v) = -\lambda R_0(v) + Q_0(0)s_r(v) \quad (6)$$

$$-R_h(v) = -\lambda R_h(v) + Q_h(0)s_r(v) + \sum_{l=1}^h R_{h-l}(v)\lambda g_l, \quad h \geq 1 \quad (7)$$

$$-F_a(v) = -\lambda F_a(v) + \delta \sum_{k=1}^{\infty} V_{k,a}(0)s_f(v) \quad (8)$$

$$-F_l(v) = -\lambda F_l(v) + \delta \sum_{k=1}^{\infty} V_{k,l}(0)s_f(v) + \sum_{l=1}^{l-1} F_{l-l}(v)\lambda g_l, \quad l \geq \alpha \quad (9)$$

$$-V_{1,0}(v) = -\lambda V_{1,0}(v) + (1-\gamma)Q_0(0)w_v(v) + R_0(0)w_v(v) \quad (10)$$

$$-V_{1,l}(v) = -\lambda V_{1,l}(v) + [(1-\gamma)Q_l(0) + R_l(0)w_v(v) + \sum_{i=1}^l V_{1,i-l}(v)\lambda g_i], \quad 1 \leq l \leq \alpha-1 \quad (11)$$

$$-V_{1,l}(v) = -\lambda V_{1,l}(v) + \sum_{i=1}^l V_{1,l-i}(v)\lambda g_i, \quad l \geq \alpha, \quad (12)$$

$$-V_{k,0}(v) = -\lambda V_{k,0}(v) + V_{k-1,0}(0)w_v(v), \quad k \geq 2 \quad (13)$$

$$-V_{k,l}(v) = -\lambda V_{k,l}(v) + V_{k-1,l}(0)w_v(v) + \sum_{i=1}^l V_{k,l-i}(v)\lambda g_i, \quad 1 \leq l \leq \alpha-1, \quad k \geq 2 \quad (14)$$

$$-V_{k,l}(v) = -\lambda V_{k,l}(v) + \sum_{i=1}^l V_{l,l-i}(v)\lambda g_i, \quad l \geq \alpha, \quad k \geq 2 \quad (15)$$

The LST is defined as

$$M_{i,h}(\vartheta) = \int_0^{\infty} e^{-\vartheta v} M_{i,h}(v) dv.$$

Similarly, the LST F, Q, R and V are defined using the same manner. Using the LST on both sides of equations (1) - (15), we obtain

$$\vartheta M_{i,0}(\vartheta) - M_{i,0}(0) = \lambda M_{i,0}(\vartheta) - F_i(0)S_m(\vartheta) - (1-\gamma)Q_i(0)S_m(\vartheta) - R_i(0)S_m(\vartheta)$$

$$- (1-\delta) \sum_{k=1}^{\infty} V_{k,i}(0)S_m(\vartheta), \quad a \leq i \leq b \quad (16)$$

$$\vartheta M_{i,h}(\vartheta) - M_{i,h}(0) = \lambda M_{i,h}(\vartheta) - \sum_{l=1}^h M_{i,h-l}(\vartheta)\lambda g_l, \quad a \leq i \leq b-1, \quad h \geq 1 \quad (17)$$

$$\vartheta M_{b,h}(\vartheta) - M_{b,h}(0) = \lambda M_{b,h}(\vartheta) - F_{b+h}(0)S_m(\vartheta) - \sum_{l=1}^h M_{b,h-l}(\vartheta)\lambda g_l - R_{b+h}(0)S_m(\vartheta)$$

$$- (1-\delta) \sum_{k=1}^{\infty} V_{k,b+h}(0)S_m(\vartheta) - (1-\gamma)Q_{b+h}(0)S_m(\vartheta), \quad h \geq 1 \quad (18)$$

$$\vartheta Q_0(\vartheta) - Q_0(0) = \lambda Q_0(\vartheta) - \sum_{l=a}^b M_{l,0}(0)S_q(\vartheta) \quad (19)$$

$$\vartheta Q_h(\vartheta) - Q_h(0) = \lambda Q_h(\vartheta) - \sum_{l=a}^b M_{l,h}(0)S_q(\vartheta) - \sum_{l=1}^h Q_{h-l}(\vartheta)\lambda g_l, \quad h \geq 1 \quad (20)$$

$$\vartheta R_0(\vartheta) - R_0(0) = \lambda R_0(\vartheta) - \gamma Q_0(0)S_r(\vartheta) \quad (21)$$

$$\vartheta R_h(\vartheta) - R_h(0) = \lambda R_h(\vartheta) - \gamma Q_h(0)S_r(\vartheta) - \sum_{l=1}^h R_{h-l}(\vartheta)\lambda g_l, \quad h \geq 1 \quad (22)$$

$$\vartheta F_a(\vartheta) - F_a(0) = \lambda F_a(\vartheta) - \sum_{k=1}^{\infty} V_{k,a}(0)S_f(\vartheta) \quad (23)$$

$$\vartheta F_l(\vartheta) - F_l(0) = \lambda F_l(\vartheta) - \sum_{k=1}^{\infty} V_{k,l}(0)S_f(\vartheta) - \sum_{l=1}^{l-1} F_{l-l}(\vartheta)\lambda g_l, \quad h \geq 1, \quad l \geq \alpha \quad (24)$$

$$\vartheta V_{1,0}(\vartheta) - V_{1,0}(0) = \lambda V_{1,0}(\vartheta) - (1-\gamma)Q_0(0)W(\vartheta) - R_0(0)W(\vartheta) \quad (25)$$

$$V_{1,h}(\vartheta) - V_{1,h}(0) = \lambda V_{1,h}(\vartheta) - (1-\gamma)Q_h(0)W(\vartheta) - R_h(0)W(\vartheta) -$$

$$\sum_{i=1}^h M_{1,h-i}(\vartheta) \lambda g_i, 1 \leq h \leq \alpha-1 \quad (26)$$

$$\vartheta V_{1,h}(\vartheta) - V_{1,h}(0) = \lambda V_{1,h}(\vartheta) - \sum_{i=1}^h M_{1,h-i}(\vartheta) \lambda g_i, h \geq \alpha \quad (27)$$

$$\vartheta V_{k,0}(\vartheta) - V_{k,0}(0) = \lambda V_{k,0}(\vartheta) - V_{k-1,0}(0) W(\vartheta), k \geq 2 \quad (28)$$

$$\vartheta V_{k,h}(\vartheta) - V_{k,h}(0) = \lambda V_{k,h}(\vartheta) - V_{k-1,h}(0) W(\vartheta) - \sum_{i=1}^h V_{k,h-i}(\vartheta) \lambda g_i, 1 \leq h \leq \alpha-1, k \geq 2 \quad (29)$$

$$\vartheta V_{k,h}(\vartheta) - V_{k,h}(0) = \lambda V_{k,h}(\vartheta) - \sum_{i=1}^h V_{k,h-i}(\vartheta) \lambda g_i, h \geq \alpha, k \geq 2 \quad (30)$$

The probability generating functions listed below are defined:

$$\left. \begin{aligned} \tilde{M}_l(z, \vartheta) &= \sum_{h=0}^{\infty} \tilde{M}_{l,h}(\vartheta) z^h, M_l(z, 0) = \sum_{h=0}^{\infty} M_{l,h}(0) z^h, a \leq l \leq b, \\ \tilde{Q}(z, \vartheta) &= \sum_{h=0}^{\infty} \tilde{Q}_h(\vartheta) z^h, Q(z, 0) = \sum_{h=0}^{\infty} Q_h(0) z^h, \\ \tilde{R}(z, \vartheta) &= \sum_{h=0}^{\infty} \tilde{R}_h(\vartheta) z^h, R(z, 0) = \sum_{h=0}^{\infty} R_h(0) z^h, \\ \tilde{F}(z, \vartheta) &= \sum_{h=0}^{\infty} \tilde{F}_h(\vartheta) z^h, F(z, 0) = \sum_{h=0}^{\infty} F_h(0) z^h, \\ \tilde{V}_l(z, \vartheta) &= \sum_{h=0}^{\infty} \tilde{V}_{l,h}(\vartheta) z^h, V_l(z, 0) = \sum_{h=0}^{\infty} V_{l,h}(0) z^h, l \geq 1. \end{aligned} \right\} \quad (31)$$

By multiplying equations (16) to (30) with suitable powers of z^h and summing over h , then by using (31) and after some mathematical manipulations, we get

$$(\vartheta - u(z))Q(z, \vartheta) = (S_q(u(z)) - S_q(\vartheta)) \sum_{l=a}^b M_l(z, 0) \quad (32)$$

$$(\vartheta - u(z))R(z, \vartheta) = (S_r(u(z)) - S_r(\vartheta)) \gamma Q(z, 0) \quad (33)$$

$$(\vartheta - u(z))F(z, \vartheta) = (S_f(u(z)) - S_f(\vartheta)) \sum_{k=1}^{\infty} [V_k(z, 0) + \sum_{l=0}^{a-1} V_{k,l}(0) z^l] \quad (34)$$

$$(\vartheta - u(z))V_1(z, \vartheta) = (W(u(z)) - W(\vartheta)) \sum_{l=0}^{a-1} [(1-\gamma)q_l + r_l] z^l, \quad (35)$$

$$(\vartheta - u(z))V_k(z, \vartheta) = (W(u(z)) - W(\vartheta)) \sum_{l=0}^{a-1} V_{k-1,l}(0) z^l, k \geq 2 \quad (36)$$

$$(\vartheta - u(z))M_l(z, \vartheta) = (S_m(u(z)) - S_m(\vartheta)) c_l, \alpha \leq l \leq b-1 \quad (37)$$

$$(\vartheta - u(z))B_1(z)M_b(z, \vartheta) = A(z)(S_m(u(z)) - S_m(\vartheta)) \quad (38)$$

Where

$$u(z) = (1-X(z))\lambda, r_f = R_f(0), q_i = Q_i(0), f_i = F_i(0), v_i = \sum_{k=1}^{\infty} V_{k,i}(0),$$

$$c_i = (1-\gamma)q_i + r_i + f_i + (1-\delta)v_i, d_i = (1-\gamma)q_i + r_i + v_i,$$

$$B_i(z) = z^{b-\gamma} S_m(u(z)) S_q(u(z)) S_r(u(z)) - (1-\gamma) S_m(u(z)) S_q(u(z))$$

$$A(z) = [\gamma S_f(z) + (1-\gamma)] S_q(z) S_m(z) \sum_{l=a}^{b-1} c_l - \sum_{l=0}^{b-1} c_l z^l + (1-\delta S_f(z)) W(z) \sum_{l=0}^{a-1} d_l z^l$$

Probability-generating function of the queue size

Let $P(z)$ be the PGF of the queue size at

an arbitrary time epoch. Then,

$$P(z) = \sum_{l=a}^b M_l(z, 0) + Q(z, 0) + R(z, 0) + F(z, 0) + \sum_{k=1}^{\infty} V_k(z, 0) \quad (39)$$

By substituting $\vartheta = 0$ in equations (32)-(38), then the equation (83) becomes

$$P(z) = \frac{B_2(z) \sum_{l=a}^{b-1} c_l [z^b - z^l] + [B_1(z) - B_2(z)] B_3(z) \sum_{l=0}^{a-1} d_l z^l}{u(z) B_1(z)} \quad (40)$$

Where

$$B_2(z) = (1-\gamma) S_m(u(z)) S_q(u(z)) S_r(u(z)) - (1-\gamma) S_m(u(z)) S_q(u(z))$$

$$B_3(z) = (1-W(u(z))) + \delta W(u(z)) (1-S_f(u(z)))$$

Steady state condition

The probability-generating function (PGF) must satisfy the condition $P(1)=1$. This requirement is met by setting a term equal to 1 and applying L'Hopital's rule in (40). Consequently, the steady-state condition for the model's existence is given by

$$\rho = \frac{\lambda X_1(E(S_m) + E(S_q) + \gamma E(S_r))}{b} < 1.$$

Performance measures

In this section, we derive various performance metrics for the concerned queueing model as follows.

Expected queue length

The expected queue length E_{q1} at an arbitrary time epoch is

$$E_{q1} = \left(\frac{dP(z)}{dz} \right)_{z=1} = \frac{Dr_{q1}^{(II)} Nr_{q1}^{(III)} - Dr_{q1}^{(III)} Nr_{q1}^{(II)}}{3(Dr_{q1}^{(II)})^2}. \quad (41)$$

Where

$$Dr_{q1}^{(II)} = 2(-\lambda X_1) B_4, Dr_{q1}^{(III)} = 3(-\lambda X_1) B_5 + 3(-\lambda X_2) B_4,$$

$$B_4 = b - S_{m1} - S_{q1} + S_{r1}, S_{m1} = X_1 E(S_m), S_{q1} = X_1 E(S_q)$$

$$B_5 = b(b-1) - S_{m2} - S_{q2} - 2S_{m1} S_{q1} - (2S_{r1} S_{q1} + 2S_{r1} S_{m1} + S_{r2}),$$

$$Nr_{q1}^{(II)} = 2B_6(b-1) \sum_{l=a}^{b-1} c_l - 2B_6[B_4 - B_6] \sum_{l=0}^{a-1} d_l, S_{r1} = X_1 E(S_r),$$

$$Nr_{q1}^{(III)} = 3[B_6[b(b-1) - l(l-1)] + B_7(b-1)] \sum_{l=a}^{b-1} c_l - [6lW_1[B_4 - B_6] + 3W_2[B_4 - B_6] + 3W_1[B_5 - B_7]] \sum_{l=0}^{a-1} d_l$$

$$S_{r2} = X_2 E(S_r) + (\lambda X_1)^2 E(S_r^2), S_f^2 = X_2 E(S_f) + (\lambda X_1)^2 E(S_f^2),$$

$$B_6 = -S_{m1} - S_{q1} + S_{r1}, S_{b1} = \lambda X_1 E(S_b), S_{f1} = \lambda X_1 E(S_f),$$

$$B_7 = -S_{m2} - S_{q2} - 2S_{m1}S_{q1} - \gamma(2S_{r1}S_{q1} + 2S_{r1}S_{m1} + S_{r2}),$$

$$S_{m2} = \lambda X_2 E(S_m) + (\lambda X_1)^2 E(S_m^2), S_{b2} = X_2 E(S_m) + (\lambda X_1)^2 E(S_m^2),$$

$$B_8 = -W_1 - \delta S_{f1}, B_9 = -W_2 - \delta S_{f2} - 2\delta W_1 S_{f1}, W_1 = \lambda X_1 E(W)$$

$$S_{q2} = \lambda X_2 E(S_q) + (\lambda X_1)^2 E(S_q^2), W_2 = \lambda X_2 E(W) + (\lambda X_1)^2 E(W^2)$$

The length of the server expected busy time:

The expected length of the busy period E_{bp} is given by

$$E_{bp} = \frac{E(T)}{\sum_{i=0}^{\alpha-1} d_i} \tag{42}$$

Where $E(T) = E(S_m) + E(S_q) + E(S_r) + E(S_f)$, $E(S_m)$ =mean regular service time, $E(S_q)$ =mean inspection time, $E(S_r)$ =mean rework time and $E(S_f)$ =mean repair time.

Expected duration of the dormant period

As a result of multiple vacation processes, let I be the random variable for the inactive period of time. Then, the anticipated duration of the inactive period is provided by

$$E_{ip} = \frac{E(W_1)}{1 - \sum_{i=0}^{\alpha-1} \sum_{j=0}^i \gamma_i [d_{i-i} - \sum_{j=0}^{i-1} \gamma_j d_{i-i-j}]} \tag{43}$$

Where $E(W_1)$ =mean vacation time.

The expected duration of line waiting

We have obtained the result by applying Little's formula

$$E_{wt} = \frac{E_{qi}}{\lambda X_1} \tag{44}$$

Where X_1 =mean batch size and E_{qi} is given in (41).

Cost model

Cost analysis serves as a critical tool in any practical scenario and every step of the operation of the system. System managers therefore want to reduce total average cost in order to operate efficiently and economically. In this context, we build up a cost model for the proposed queuing system and formulate the expression for the total average cost. The unit-cost system elements consist of the startup (C_s), holding (C_h), operating (C_o), vacation reward (C_v), repair (C_r), and rework (C_f) values. All these parameters collectively determine the financial performance and reliability of the overall operation. The duration of a cycle is defined as the sum of the idle period and the

busy period, from Equations (42) and (43).

The expected length of a cycle, denoted by E_{tc} , is given by:

$$E_{tc} = E_{ip} + E_{bp}$$

The total average cost per unit is given by,

$$T_{ac} = [C_s + \frac{C_{rp} E(S_f) E_{bp}}{E_{tc}} + \gamma C_r E(S_r) - C_v E_{ip}] \frac{1}{E_{tc}} + C_h E_{qi} + C_o \rho.$$

Where ρ is given in section 4.1.

Numerical Illustration

A numerical illustration is performed for the purpose of demonstrating the practical applicability of the proposed bulk queueing model in a real-life chemical or pharmaceutical production environment under specific, real-life assumptions. In this context, batch arrivals (raw materials or reagent lots) are following a geometric distribution with a mean of 2, and the production (service) process is started at the time when the queue number of items reaches the minimum threshold, ' $\alpha=5$ ', and can handle up to ' $b=8$ ' items per service cycle, corresponding to the capacity planning in chemical.

It is assumed that the server, which performs work such as regular service, inspection, and rework, works according to exponentially distributed processing times, taking rates μ_1 , μ_2 , and μ_3 , as parameters. Additionally, vacation time when the server stands idle if the batch threshold has not been met and repair time that emulates potential start-up problems because of instability in temperature, catalysts, or sensors are exponentially distributed with rates τ and η .

It is necessary to fulfill the stability condition to be able to choose the parameters. In MATLAB, some of the key performance indicators, like expected waiting time (E_{wt}), expected busy period (E_{bp}), expected idle time (E_{ip}) and expected queue length (E_{qi}) are determined to ascertain the effects of change in the system parameters on overall operational A cost analysis is also incorporated in a comprehensive assessment with fixed unit costs. startup cost, holding cost ($C_h=0.25$), operation cost ($C_o=4$), reward per service ($C_v=1.5$), rework cost ($C_f=2$), and repair cost ($C_r=0.9$).

Table 2: The impact of arrival rate (λ) on E_{qi} , E_{wt} , E_{bp} , E_{ip} and T_{ac}

λ	E_{qi}	E_{wt}	E_{bp}	E_{ip}	T_{ac}
4	6.3842	0.798	0.3198	0.7611	8.3101
5	8.014	0.8014	0.3622	0.7128	9.6445
6	9.963	0.8303	0.4165	0.668	11.102
7	12.3937	0.8853	0.4888	0.6295	12.736
8	15.6054	0.9753	0.5898	0.5973	14.6824
9	20.1823	1.1212	0.7415	0.5705	17.2202

Table 2 presents the comparison between arrival rate vs. performance measures and total average cost. We fix the default parameters for the numerical results summarized in Table 2 as $\mu_1=8, \mu_2=7, \mu_3=5, \eta=3, \tau=2, \gamma=0.2$ and $\delta=0.2$. E_{qi} , E_{wt} , E_{bp} and T_{ac} increases as we increase the values of λ but E_{ip} decreases.

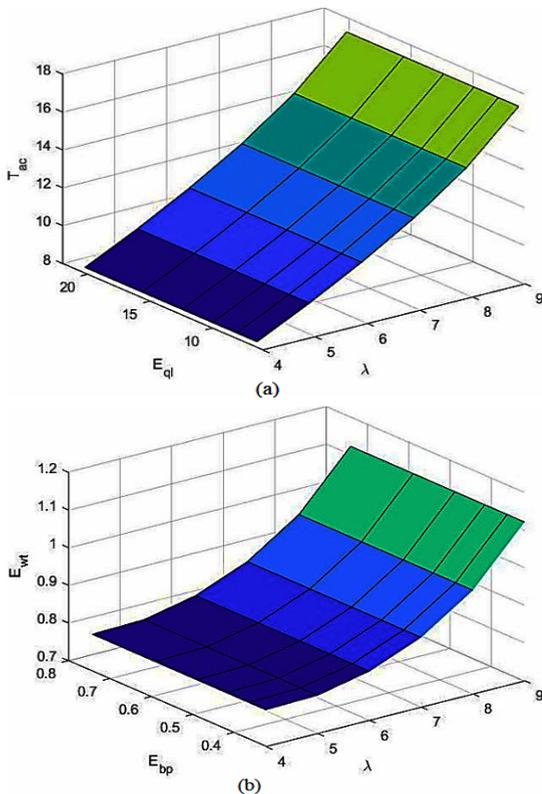


Fig. 2. λ verses (a) E_{qi} , T_{ac} and (b) E_{bp} , E_{wt}

Figure 2(a-b) shows the impact of the arrival rate (λ) on the total average cost (T_{ac}), expected queue duration (E_{qi}), expected busy period (E_{bp}) and expected waiting time (E_{wt}) in the bulk service pharmaceutical system. In this context, λ is the rate of arrival of new batch production in the system, for example, the arrival of raw material for the granulation, coating or formulation processes. With an increase in λ , the number of batches increases in the queue waiting to reach the bulk service threshold.

This increases the expected queue length and busy period of the server (e.g., mixer, reactor, tablet press), which directly affects the system performance. As arrivals are now more frequent, the server is constantly busy with no idle or recovery time, hence helping to saturate the system. This congestion causes the average waiting time to increase and the total cost to be higher as processing times, energy wasted, and scheduling delays occur due to a longer processing duration.

Table 3: The impact of faulty probability (γ) on E_{qi} , E_{wt} , E_{bp} , E_{ip} and T_{ac}

γ	E_{qi}	E_{wt}	E_{bp}	E_{ip}	T_{ac}
0.2	8.014	0.8014	0.3622	0.7128	9.6445
0.3	8.3995	0.84	0.3993	0.6987	9.9517
0.4	8.8174	0.8817	0.44	0.6853	10.2637
0.5	9.2723	0.9272	0.4848	0.6725	10.5827
0.6	9.7694	0.9769	0.5342	0.6603	10.9114
0.7	10.3151	1.0315	0.589	0.6485	11.253

The influence of performance metrics and the total average cost for various faulty probability is shown in Table 3. The study was carried out on a system with $\lambda=5, \mu_1=8, \mu_2=7, \mu_3=5, \eta=3, \tau=2$, and $\delta=0.2$. It can be seen that when the probability of rework increases, E_{qi} , E_{bp} , T_{ac} and E_{wt} will increase and decrease with the duration of the inactive period (E_{ip}).

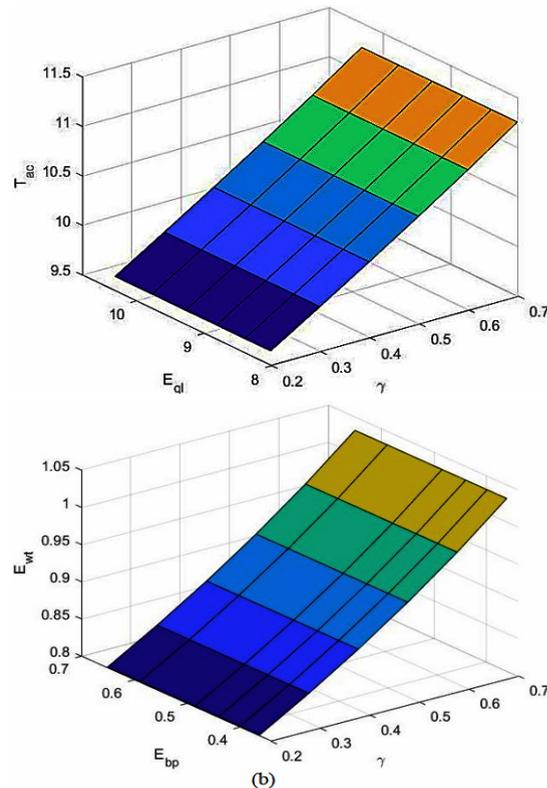


Fig. 3. γ verses (a) E_{qi} , T_{ac} and (b) E_{bp} , E_{wt}

Figure 3(a-b) represents the impact of the probability of failure (γ) on the total average cost (T_{ac}), expected queue duration (E_{qt}), expected busy period (E_{bp}), and expected waiting time (E_{wt}) in a pharmaceutical manufacturing environment under a bulk queueing system. In this model, once a batch is through primary processing, it is inspected. If it is found to be faulty (γ), it is taken for the rework process immediately and then transferred to the package.

This rework does not affect the server (e.g., blending unit, granulator, or tablet press), but it extends the time batches are in the system. As γ increases, a greater number of batches are sent to rework, requiring longer busy periods, longer queues, and greater congestion in the system. As a result, the average cost increases with increased processing time, higher energy consumption, and more labor used in reworked batches.

Computing of ANFIS

Jang³⁰ proposed the Adaptive Neuro-Fuzzy Inference System (ANFIS), a multi-layer architecture in the combination of principles of neural networks and fuzzy logic, which results in a

fuzzy inference system. Mabrook *et al.*,³¹ used the ANFIS algorithm to optimize the decision-making process by correctly identifying occupied and free communication channels. Ahuja *et al.*,³² investigated a queueing model (QM) with an unreliable server and multi-stage services, where the ANFIS method was utilized to estimate important performance parameters. Recently, Madhu Jain and Sibasish Dhibar³³ studied the combination of ANFIS with metaheuristic optimization techniques to construct a strategic joining policy in systems with re-attempts and vacation times.

The ANFIS model that extends the Sugeno-type fuzzy inference systems is trained by fuzzy if-then rules with Gaussian membership functions. The training system uses input-output data pairs that are generated using analytical methods. A comparative analysis of performance measurements and a total average cost analysis is carried out through the comparison of numerical results obtained from both analytical and ANFIS-based models. These methods are applied to test the impact of system parameters on performance metrics, such as the lengths of queues and the efficacy of the servers.

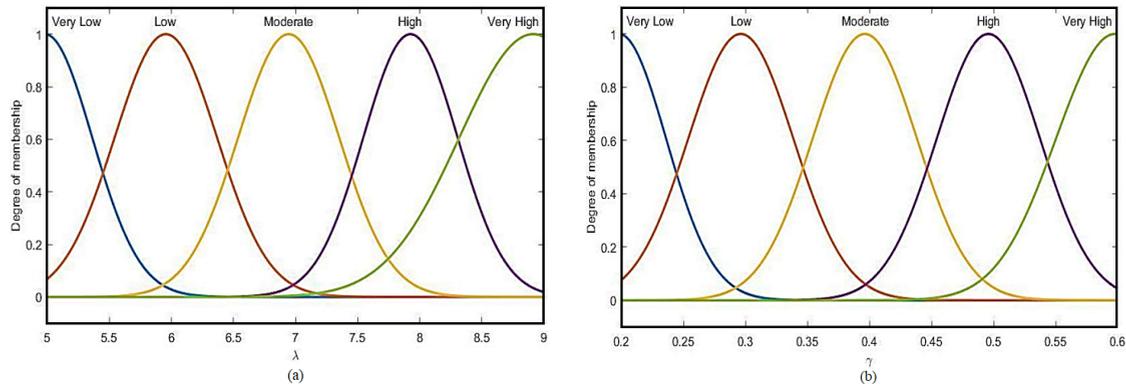


Fig. 4. The tuned Gaussian shape membership functions for input variable (a) λ and (b) γ

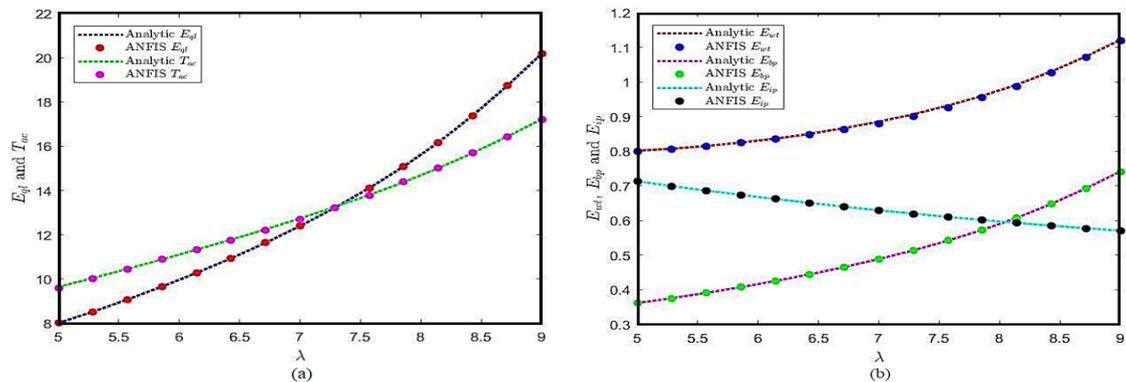
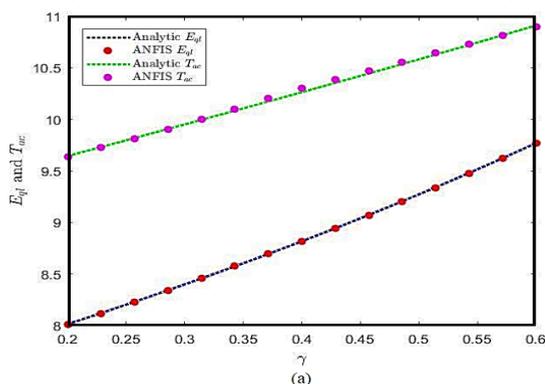


Fig. 5. Relation between analytic and ANFIS the comparison of different performance metrics and total average cost with varying arrival rate

ANFIS is mostly based on the neuro-adaptive learning methods, which iteratively adapt membership functions that allow applications in time-series predictions and control systems. Some inputs are expressed as linguistic variables and are mapped



to the fuzzy system built with the ANFIS model. In Fig. 4(a-b) it is possible to show the Gaussian membership functions for each input variable as well as the number of membership functions, their parameters, and the corresponding linguistic labels.

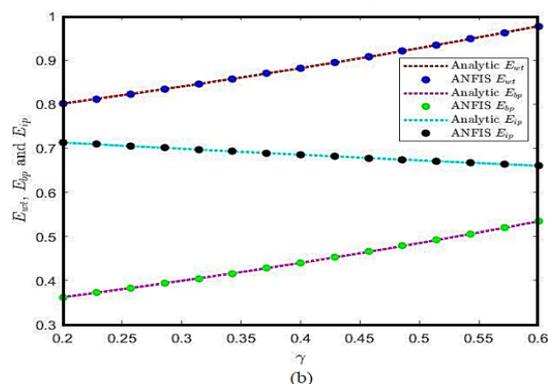


Fig. 6. Relation between analytic and ANFIS the comparison of different performance metrics and total average cost with varying faulty probability

We compute ANFIS-generated values for various configurations of these E_{qt} , E_{wt} , E_{bp} , E_{ip} , and T_{ac} for ascertained values of λ and γ in MATLAB. In Fig. 5(a-b) and 6(a-b), the dot marks are for ANFIS results, while discrete lines represent results obtained from the analytical calculations. Eventually, the results indicate an excellent correlation between the ANFIS and the analytical results, thus verifying the validity and accuracy of the ANFIS-based method.

CONCLUSION

This work presents a bulk queuing framework, which is chemically inspired and effectively captures major operational components like starting failures, rework, inspection, repair, and vacation periods. By placing batch chemical processes in a queueing structure, the model can represent the real-world situations within the pharmaceuticals and fine chemicals industries where the reliability of production and quality control is imperative. In addition, the supplementary variable approach used is observed to be an effective and reliable analytical method when compared to

other methods. The stability requirement has been investigated in terms of its required and sufficient conditions.

Moreover, constructive analytical results with numerical examples can be applicable in developing solutions for numerous situations in the real world. A comparison analysis between ANFIS-generated results and those of traditional analytical solutions shows high accuracy and consistency, presenting the robustness of the model. Future extensions could be retrieval queues accompanied by setup times, priority mechanisms, catastrophes, and altered Bernoulli vacations, which would make the model even more capable of fitting complex operational situations.

ACKNOWLEDGMENT

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare no conflict of interest.

REFERENCES

1. Tsitkov, S.; Pesenti, T.; Palacci, H.; Blanchet, J.; Hess, H. A queueing theory-based perspective of the kinetics of "channeled" enzyme cascade reactions., *ACS Catal.*, **2018**, 8(11), 10721–10731.
2. Kandemir-Cavas, C.; Cavas, L.; Yokes, M.B.; Hlynka, M.; Schell, R.; Yuradoc, K. A novel application of queueing theory on the Caulerpenyne secreted by invasive *Caulerpa taxifolia* (Vahl) C. Agardh (Ulvoophyceae, Caulerpales): a preliminary study., *Mediterr. Mar. Sci.*, **2008**, 9, 67–76.

3. Conolly, B.W.; Parthasarathy, P.R.; Dharmaraja, S. A chemical queue., *Math. Sci.*, **1997**, *22*, 83–91.
4. Tarabia, A.M.K.; Takagi, H.; El-Baz, A.H. Transient solution of a non-empty chemical queueing system., *Math. Methods Oper. Res.*, **2008**, *70*(1), 77–98.
5. Alshreef, G.; Tarabia, A. Transient analysis of chemical queue with catastrophes and server repair., *Inf. Sci. Lett.*, **2023**, *12*, 541–545.
6. Shanmugam, B.; Saravanarajan, M.C. Unpredictable two-stage retrial model featuring impatient customers pinned Bernoulli working vacations and starting failure., *Ann. Oper. Res.*, **2025**, 1–30.
7. Karpagam, S.; Ayyappan, G. A bulk queueing system with starting failure and repair, multiple vacation, stand-by server, closedown and n-policy., *Thai J. Math.*, **2022**, *20*(2), 905–936.
8. Ayyappan, G.; Nirmala, M.; Karpagam, S. Analysis of repairable single server bulk queue with standby server, two phase heterogeneous service, starting failure and multiple vacation., *Int. J. Appl. Comput. Math.*, **2020**, *6*, 52.
9. Karpagam, S.; Ayyappan, G.; Somasundaram, B. A bulk queueing system with rework in manufacturing industry with starting failure and single vacation., *Int. J. Appl. Comput. Math.*, **2020**, *6*, 174.
10. Venkatesan, N.; Paramasivam, R. Markov Modelling of Small Cell Base Station Sleep Strategies with Disaster and Repairable Server., *Contemp. Math.*, **2025**, *6*(2), 1520–1539.
11. Ayyappan, G.; Meena, S. Phase type queueing model of server vacation, repair and degrading service with breakdown, starting failure and close-down., *Reliab. Theory Appl.*, **2023**, *18*(1), 464–483.
12. Yang, D.Y.; Wu, C.H. Performance analysis and optimization of a retrial queue with working vacations and starting failures., *Math. Comput. Model. Dyn. Syst.*, **2019**, *25*(5), 463–481.
13. Yang, D.Y.; Ke, J.C.; Wu, C.H. The multi-server retrial system with Bernoulli feedback and starting failures., *Int. J. Comput. Math.*, **2015**, *92*(5), 954–969.
14. Kumar, B.K.; Madheswari, S.P.; Vijayakumar, A. The M/G/1 retrial queue with feedback and starting failures., *Appl. Math. Model.*, **2002**, *26*(11), 1057–1075.
15. Yang, T.; Li, H. The M/G/1 retrial queue with the server subject to starting failures., *Queueing Syst.*, **1994**, *16*(1), 83–96.
16. Keerthiga, S.; Indhira, K. Cost optimization for system stability and orbital search under working vacation and starting failure by using the ANFIS soft computing., *Ain Shams Eng. J.*, **2024**, *15*(8), 102847.
17. Arivudainambi, D.; Gowsalya, M. Analysis of an M/G/1 retrial queue with Bernoulli vacation, two types of service and starting failure., *Int. J. Artif. Intell. Soft Comput.*, **2017**, *6*(3), 222–249.
18. Ke, J.C.; Chang, F.M. $M^X(G1, G2)/1$ retrial queue under Bernoulli vacation schedules with general repeated attempts and starting failures., *Appl. Math. Model.*, **2009**, *33*(7), 3186–3196.
19. Mokaddis, G.S.; Metwally, S.A.; Zaki, B.M. A feedback retrial queueing system with starting failures and single vacation., *J. Appl. Sci. Eng.*, **2007**, *10*(3), 183–192.
20. Das, S.; Pradhan, S. Steady-state analysis of batch-size-dependent service queue with server breakdown, close-down time and multiple vacations., *RAIRO-Oper. Res.*, **2025**, *59*(2), 743–770.
21. Sikdar, K. A single server finite buffer Geom/G/1/K queue with L limited service and multiple vacation policy., *Int. J. Math. Oper. Res.*, **2024**, *27*(3), 328–353.
22. Karpagam, S.; Somasundaram, B.; Mary, A. K. S., & Lokesh, R. Analysis Of MX/G/1 Queue With Optional Second Service, Feedback And Bernoulli Vacation., *Reliab. Theory Appl.*, **2024**, *19*(79), 583-594.
23. Mahanta, S.; Kumar, N.; Choudhury, G. Study of a two types of general heterogeneous service queueing system in a single server with optional repeated service and feedback queue., *Hacet. J. Math. Stat.*, **2024**, *53*(3), 851–878.
24. Karpagam, S. Analysis of bulk service queueing system with rework, unreliable server, resuming service and two kinds of multiple vacation., *Yugosl. J. Oper. Res.*, **2023**, *33*(1), 17–39.
25. Kalaiselvi, J.; Saravanarajan, M.C. Particle Swarm Optimization of a Single Server Retrial Queue with Delayed Repair Under Working Vacation From Optional Service to ReService., *Contemp. Math.*, **2025**, *6*(1), 743–771.

26. Marjasz, R.; Kempa, W.M.; Kovtun, V. Quantification of system resilience through stress testing using a predictive analysis of departure dynamics in a queue with multiple vacation policy., *Sci. Rep.*, **2025**, *15*(1), 10484.
27. Haridass, M.; Arumuganathan, R. A batch service queueing system with multiple vacations, setup time and server's choice of admitting reservice., *Int. J. Oper. Res.*, **2012**, *14*(2), 156–186.
28. Jeyakumar, S.; Arumuganathan, R. A non-Markovian bulk queue with multiple vacations and control policy on request for re-service., *Qual. Technol. Quant. Manag.*, **2011**, *8*(3), 253–269.
29. Cox, D.R. The analysis of non-Markovian stochastic processes by the inclusion of supplementary variables., *Proc. Comput. Philos. Soc.*, **1965**, *51*, 433–441.
30. Jang, J.S. ANFIS: adaptive-network-based fuzzy inference system., *IEEE Trans. Syst. Man Cybern.*, **1993**, *23*(3), 665–685.
31. Mabrook, M.M.; Taha, H.A.; Hussein, A.I. Cooperative spectrum sensing optimization based adaptive neuro-fuzzy inference system (ANFIS) in cognitive radio networks., *J. Ambient Intell., Humaniz. Comput.*, **2022**, *13*(7), 3643–3654.
32. Ahuja, A.; Jain, A.; Jain, M. Transient analysis and ANFIS computing of unreliable single server queueing model with multiple stage service and functioning vacation., *Math. Comput. Simul.*, **2022**, *192*, 464–490.
33. Jain, M.; Dhibar, S. ANFIS and metaheuristic optimization for strategic joining policy with re-attempt and vacation., *Math. Comput. Simul.*, **2023**, *211*, 57–84.