Quantitative Modeling and Analytical Calculation of Anelasticity for a Cyber-Physical System

Hongfang Gong, Renfa Li, *Senior Member, IEEE*, Jiyao An, *Member, IEEE*, Yang Bai, and Keqin Li, *Fellow, IEEE*

***Abstract*—This paper investigates resource provisioning in cyber-physical systems (CPSs) by developing a new definition of anelasticity. A flat semi-dormant multicontroller (FSDMC) model is established on a special type of CPS platform named arbitrated networked control system with dual communication channels. A novel, quantitative, and formal definition of anelasticity for the FSDMC is proposed. A new finite capacity *M/M/c* queuing system with *N*-policy and asynchronous multiple working vacations of partial servers is established, and the FSDMC is modeled as a quasi-birth-and-death process to obtain the stationary probability distribution of the system. Based on the queueing model, we quantify various performance indices of the system to build a nonlinear cost-performance ratio (CPR) function. An optimization model is presented to minimize the CPR. A particle swarm optimization (PSO) algorithm is used to find the optimum solution of the optimization model and obtain the optimal configuration values of the system parameters under stability condition. By changing the system parameters, the sensitivity of the system performance indices and the CPR are analyzed, respectively. The unexpected workload varies randomly over time. Thus, an *M/M/1/K* queue is constructed in a Markovian environment by employing a three-state, irreducible Markov process. In this queue, the conditional average queue length and the probabilities of the three-state process are calculated. Then, the anelasticity value of the system is precisely determined. When the average arrival rate exceeds the average service rate in the queueing system, an optimal CPR unchanged adaptive algorithm based on PSO is designed to dynamically adjust the controller service rate. Extensive numerical results show the usefulness and effectiveness of the proposed techniques and exhibit that the system can maintain elastic invariance in adaptive adjustment parameters.**

***Index Terms*—Anelasticity, cost-performance ratio (CPR), cyber-physical system (CPS), queuing system, semi-dormant controller cluster.**

Manuscript received March 10, 2018; accepted July 17, 2018. This work was supported in part by the National Key Research and Development Plan of China under Grant 2016YFB0200405 and Grant 2012AA01A301-01, in part by the National Natural Science Foundation of China under Grant 61672217, Grant 61370097, Grant 61370095, and Grant 11771060, and in part by the Science Research Foundation of Hunan Provincial Education Department of China under Grant 17A003. This paper was recommended by Associate Editor F. Wang. *(Corresponding author: Renfa Li.)*

H. Gong is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China, also with the Key Laboratory for Embedded and Network Computing of Hunan Province, Hunan University, Changsha 410082, China, also with the National Supercomputing Center in Changsha, Hunan University, Changsha 410082, China, and also with the School of Mathematics and Statistics, Changsha University of Science and Technology, Changsha 410114, China (e-mail: ghongfang@126.com).

R. Li, J. An, and Y. Bai are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China, also with the Key Laboratory for Embedded and Network Computing of Hunan Province, Hunan University, Changsha 410082, China, and also with the National Supercomputing Center in Changsha, Hunan University, Changsha 410082, China (e-mail: lirenfa@hnu.edu.cn; jt_anbob@hnu.edu.cn; baiyang@hnu.edu.cn).

K. Li is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China, also with the Key Laboratory for Embedded and Network Computing of Hunan Province, Hunan University, Changsha 410082, China, also with the National Supercomputing Center in Changsha, Hunan University, Changsha 410082, China, and also with the Department of Computer Science, State University of New York, New Paltz, NY 12561 USA (e-mail: lik@newpaltz.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2861918

I. Introduction

A. Background

AUTOMOTIVE, avionics, and industrial automation systems are often distributed embedded systems (DESs), such as CAN, LIN, and FlexRay, with a large number of processing units (PUs) that communicate via shared buses. Such architectures are used to run a certain number of distributed control applications under multiple quality-of-control constraints [1]. In such a complex cyber-physical system (CPS), resource constraints, cost sensitivity, and time sensitivity are its important features [2], [3]. The CPU and network resources must be allocated to competing applications subject to the constraints on resources imposed by physical phenomena [4]. To achieve the minimum cost and optimal performance of the systems, the limited computational resources must be used as efficiently as possible. Hence, multiple distributed control applications in such systems require dynamic provisioning of resources based on application demand. An embedded computing platform relies on the mechanisms and capabilities for allocating/deallocating computing resources on demand to acquire a large number of PUs for handling workload surges or releasing PUs to avoid over-provisioning of resources. Such a dynamic resource provision and management feature is called elasticity [5]. Dustdar *et al.* [6] introduced the principle of elastic processes with cost, quality, and resources as the basic elasticity dimensions, thereby forming the foundations of elastic systems.

The problem of resource provisioning in CPSs has drawn research attention [7]. In [8], an elastic CPS (eCPS) was defined as an adaptive system that could further add/remove components at run-time, from computing resources to physical devices, to align their costs, quality, and resource usage to load and owner requirements. Moldovan *et al.* [9] introduced

elasticity to cyber-physical ecosystems to integrate computing processes, people, and things. They also outlined the vision over a new computing field, *elastic computing*, which pertains to the study of elastic systems. Candra *et al.* [10] presented a modeling tool named *elasticity profile*, which specifies constructs for modeling the elastic behavior of mixed systems with respect to tradeoffs among cost, quality, and resources. Schmidt *et al.* [3] identified challenges, opportunities, and benefits of large-scale CPSs that support cloud computing and elastic infrastructure. Palensky *et al.* [11] described the main challenges, tools and methods of continuous time-based and discrete event-based models of large scale cyber-physical energy systems under resource constraints. In [12], an elastic computation middle-ware was developed to federate computation resources on wearable, mobile, and connected devices in CPSs. However, these efforts neither involve the quantitative modeling of elasticity on CPS nor study the changes in the cost and performance of the system under different workloads.

In cloud computing, Wang *et al.* [13] proposed a computing resource allocation approach based on decentralized multiagent to address the problem of energy-aware concentration or migration resource provisioning. In [14], based on the Lyapunov optimization technique combined with the technique of weight perturbation, Fang *et al.* introduced a new stochastic control algorithm that dynamically allocates limited computing resources on mobile cloudlet platforms to reduce energy consumption and improve application performance. However, these efforts do not consider the problem of quantitative modeling of system elasticity and dynamic resource provisioning. Ai *et al.* [5] proposed an elasticity measurement model of the cloud platform by using a continuous-time Markov chain. Li [15] treated a cloud platform as a queueing system to analytically and comprehensively study elasticity, performance, and cost in cloud computing. However, this measurement model cannot be directly used in the elastic computing of DESs because of the time-varying delay caused by limited system resources.

B. Motivation

A special type of CPS named arbitrated networked control system (ANCS) was first presented in [16]. Gong *et al.* [17] extended the ANCS and proposed an ANCS with dual communication channels (see [17, Fig. 2]). In this novel special CPS, a flat semi-dormant multicontroller (FSDMC) model was proposed to consider load balancing and global consistency. To achieve a tradeoff between system performance and its implementation cost, the FSDMC was modeled as an *N/(d,c)-M/M/c/K/SMWV* queueing system to configure the system computing resources dynamically. However, in the FSDMC, two main problems need to be solved, namely, over-provisioning of resources and overloading of systems. In the *N/(d,c)-M/M/c/K/SMWV* system, synchronized vacations lead to over-provisioning of resources and incur unnecessary costs. In some cases, unexpected workload spikes are likely to occur due to unpredictable changes in the physical environment. In this paper, we focus on scaling resources dynamically in the FSDMC in accordance with dynamic external environment to improve resource utilization. We consider a typical distributed embedded architecture proposed in [1] (see [1, Fig. 1(a)]).

In the ANCS, the main problem that can be attributed to the gap between high-level control models and their actual implementations is the end-to-end delay between the signal sensed at the plant output and the signal sent to the control actuator [18]. In an elastic system, we call the time-dependent elastic behavior an anelastic behavior and consider this time-sensitive elastic system an anelastic system. The current Wikipedia definition of anelasticity in physics states that "an anelastic material is a special case of a viscoelastic material. Viscoelastic materials have elements of both viscous and elastic characteristics and, as such, exhibit time-dependent strain. An anelastic material will fully recover to its original state on the removal of load." When applied to computing, anelasticity naturally reflects the on-demand nature of time-related resource provisioning in DESs. Resource provisioning indicates making the resources seamlessly available to users as and when they are required [7]. In the FSDMC, when external forces (e.g., the number of packet-in message requests) are applied, the strain (e.g., increased resource utilization and average task response time) increase nonlinearly with time, that is, the FSDMC platform is deformed. A hysteresis occurs in the stress–strain due to the resource constraints of the shared bus. This hysteresis is the main difference between elastic and anelastic systems.

In this paper, we model the FSDMC as an *N/(d,c)-M/M/c/K/AMWV* queue to define auto-scaling mechanism for the resource management of FSDMC platform and obtain various performance measures to establish the nonlinear cost-performance ratio (CPR) function. When the workload increases in this queuing system, the system cost increases, whereas the system performance decreases. Thus, the strain (i.e., CPR) will change as the stress (i.e., arrival rate) increases. When the average arrival rate exceeds the average service rate, the controller service rate should be adjusted adaptively to maintain the CPR invariance of the system in a stable state. According to the match between the resource provisioning and workload, the FSDMC platform is in three states (three-state hereinafter), namely, over-provisioning, normal, and under-provisioning states [15]. We describe a Markovian environment using a three-state Markov process and construct an *M/M/1/K* queueing model with three phases in a Markovian environment [19] to obtain the quantitative model of anelasticity of the system under unexpected workload varying randomly over time. To our knowledge, we are the first to establish a quantitative and analytical model of anelasticity for a special CPS.

C. Our Contributions

The contributions of this paper are summarized as follows.

1) A novel, quantitative, and formal definition of anelasticity for the FSDMC on the special CPS is proposed. The quantization model is a two-tuple consisting of the average queue length in the normal state and the probability that the computing resources match the current workload.

2) We establish a finite capacity *M/M/c* queuing system with *N*-policy and asynchronous multiple working vacations (AMWV) of partial servers, namely, *N/(d,c)-M/M/c/K/AMWV*, where *c*, *d*, and *K* are described in [17]. This queuing system is used to quantify the various performance measures of the system, and to build an optimization model that minimizes the nonlinear CPR function. A particle swarm optimization (PSO) [20] algorithm is used to generate the optimum solution of the optimization model and obtain the optimal configuration values of the system parameters under stability condition.
3) We construct an *M/M/1/K* queue in a Markovian environment to calculate the anelasticity value of the FSDMC platform accurately. With the aid of the equilibrium probability distribution of this queue, the conditional average queue length and the probabilities of three-state are yielded, and the anelasticity value of the system is derived. When the average arrival rate exceeds the average service rate in the *N/(d,c)-M/M/c/K/AMWV* queueing system, we propose an optimal CPR unchanged adaptive algorithm based on PSO to adjust the controller service rate dynamically. As a result, the proposed queuing model is transformed into a classic *M/M/c/K* queue where the anelasticity remained unchanged.

The rest of this paper is organized as follows. Section II describes the related works. In Section III, we define an anelasticity CPS and the measurement of anelasticity in CPSs. In Section IV, we propose a novel queuing model, namely, *N/(d,c)-M/M/c/K/AMWV* queue, and compute the equilibrium distributions and the performance indices of the system by using matrix-geometric method based on a Markovian chain. In Section V, we construct an optimization model to minimize the average CPR in the system, and analyze the sensitivity of the system performance and the CPR using numerical results. In Section VI, we establish an *M/M/1/K* queue in a Markovian environment to calculate the anelasticity value of the FSDMC platform accurately, and we discuss the problems of the system adaptive and anelastic invariance. Finally, Section VII concludes this paper.

II. Related Works

In CPSs, provisioning the required resource is extremely challenging in composing a new service because of inherent heterogeneity in the resources and thus in the services. Therefore, the issues related to service composition and thus resource provisioning in CPSs must be investigated [7]. Resource provisioning in resource management and modeling has been rarely investigated. Gunes *et al.* [21] presented a detailed survey on concepts, applications, and challenges in CPSs, and indicated that a highly scalable system should provide scatter and gather mechanisms for workload balancing and effective communication protocols to improve the performance.

Ravindran and Li [22] presented two proactive resource allocation algorithms, namely, deadline-driven proactive and laxity-driven proactive, which proactively allocate resources to maximize the aggregate deadline-satisfied ratio for the future time interval in asynchronous real-time distributed systems. Marti *et al.* [23] developed a dynamic resource allocation system for control tasks, namely, Draco, based on feedback information from the plants. Tan *et al.* [24] developed utility-based resource configuration algorithms to allocate the limited resources among users to satisfy their specific quality of service requirements in wireless networks. To improve the utilization of computing resources for self-triggered control applications, Samii *et al.* [25] proposed a software-based middleware component for scheduling and optimization of control performance and CPU usage of multiple self-triggered control loops on a uniprocessor platform. Lozoya *et al.* [26] presented a performance evaluation framework that permits assessment whether recent state-of-the-art resource/performance-aware policies can be implemented in practice. The application of this framework showed that online management of computing resources is the key for allowing adaption of embedded control systems to varying demands while delivering specific control performance with tunable resource utilization. Chantem *et al.* [27] focused on the elastic task model to select task periods for adapting periodic real-time systems in the presence of uncertainty. However, these efforts only consider resource allocation performance optimization problems but ignore cost-effectiveness issues.

Xie *et al.* [28] investigated the cost minimization of effective use of resources on heterogeneous embedded systems without using fault tolerance. Chang and Chakraborty [29] developed approaches in automotive control systems design that consider implementation resources to improve the control performance and reduce the system costs for a given amount of resources. Liu *et al.* [30] considered task scheduling problem at computing center in CPSs, and analyzed the performance indicators of the computing center which is regarded as a multipriority multiserver queueing system. A comprehensive resource scheduling strategy was proposed to balance cost and system performance. In [31], an elastic thread management scheme based on external coordinator was implemented and provided in a middleware to avoid under-utilization of server resources. This scheme suits the dynamic behavior of systems that require supporting varying numbers of clients in a cost-effective manner. In [32], the joint optimization problem with energy efficiency and effective resource utilization was investigated for heterogeneous and distributed multicore embedded systems. To address fast energy equipartition for cyber-physical network systems, Zeng *et al.* [33] developed three novel hybrid stabilization techniques, and established an optimization-based network topology design framework. A good review of design techniques and applications on CPSs was described in [34]. These works only focused on the system cost problem but ignored the system performance under resource utilization and the quantitative analysis in resource provisioning. Sheng *et al.* [35] considered workflow system performance and resource cost balancing issues and used an *M/M/c* queuing network to model multidimensional workflow net (MWF-net) to calculate the probability density function of transaction instance

dwelling times. In [36], the workflow model was modeled as an *M/M/c-M/D/c* mixed queuing network where each activity is an independent *M/M/c* or *M/D/c* queuing system. The mean value and probability distribution function of transaction instance dwelling times were calculated using the mixed queuing network and MWF-net. However, due to the complex control structure characteristics of the workflow model, these techniques cannot be directly applied to the CPS resource provisioning problem in this paper.

Several new opportunities for extending the capabilities of CPSs have been provided by utilizing cloud resources in different approaches. In 2013, the National Institute of Standards and Technology defined cyber-physical cloud computing architectural framework as "a system environment that can rapidly build, modify, and provision auto-scale CPSs, composed of a set of cloud computing-based sensor, processing, control, and data services" [37]. Gai *et al.* [38] focused on the workload scheduling issue in CPS and proposed an approach for solving the limitations caused by heterogeneous cloud computing. In [39], a level value density task scheduling algorithm for CPSs on cloud platform was proposed to schedule and distinguish the high-volume and multiclass tasks more efficiently. The algorithm not only adds attributes of time and value to the tasks, but also describes the weighted relationship between these attributes by mathematical means. Fu *et al.* [40] proposed a dynamic resource scaling framework that included an accurate performance model based on the theory of Jackson open queueing networks and can handle arbitrary operator topologies for cloud-based real-time stream data analytics systems. An overview of research efforts on the integration of CPSs with cloud computing is presented in [41]. However, the development technology of CPS clouds still needs considerable time to reach a certain maturity.

III. Measurement of Anelasticity

In this paper, we still consider the DES proposed in [1] (see [1, Fig. 1(a)]), where sensor task T_s, controller task T_c, and actuator task T_a are performed in the respective ECUs, and the messages m_1 and m_2 are transmitted by the communication bus (e.g., FlexRay).

We consider the ANCS platform architecture with dual communication channels proposed in [17] (see [17, Fig. 2]). Multiple control applications are divided into a number of tasks that are mapped onto different PUs. All the controller nodes of the PUs are connected to the two channels of **A** and **B**, but other nodes are only connected to the channel **A** (see [17]). In addition to interacting with the corresponding sensor and actuator via the shared channel **A**, each controller needs to address the overload of other controllers via the channel **B**. The shared bus communication protocol is a hierarchical flexible TDMA/FP bus scheduling policy with an event-triggered protocol proposed in [17] (see [17, Fig. 5]).

According to the definition of eCPS in [8], we define an anelastic CPS as follows.

Definition 1: An anelastic CPS is an eCPS where strain (e.g., cost and performance) is delayed in stress (e.g., workload).

In [15], the elasticity of the system with dynamically variable workload is defined as the percentage of time (or probability) that the system is in the normal state. We use p_{over}, p_{normal}, and p_{under} to represent the probabilities that the system is in over-provisioning, normal, and under-provisioning states, respectively, and $p_{\text{over}} + p_{\text{normal}} + p_{\text{under}} = 1$. In the *M/M/1/K* queue in a Markovian environment, the conditional queue length densities are often utilized in the study of the time-dependent behavior of queues described by Markovian models [19]. $\text{AvgQL}_{\text{normal}}$ denotes the conditional average queue length when the *M/M/1/K* queue in a Markovian environment is in the normal state.

Definition 2: In an anelastic CPS, the anelasticity is calculated as follows:

$$\text{Anelasticity} = \left(\text{AvgQL}_{\text{normal}}, p_{\text{normal}}\right). \tag{1}$$

In the *N/(d,c)-M/M/c/K/AMWV* queue, a state is:

1) an over-provisioning state if $0 \le k < c$;
2) a normal state if $c \le k < N$;
3) an under-provisioning state if $N \le k \le K$, where k represents the number of packet-in messages in the system.

In the *M/M/1/K* queue in a Markovian environment, we use (k, i) to describe a state, where $k \ge 0$ is the number of packet-in messages in the system, and i takes the value from three states (or three phases, three-state or three-phase hereinafter), namely, over-provisioning, normal, and under-provisioning states; for simplicity, the three states are represented as 1, 2, and 3. The transitions among the three-state are described as follows.

1) $(k-1, i) \xrightarrow{\lambda_i} (k, i+1)$. This transition happens when a new packet-in message arrives in phase i.
2) $(k, i+1) \xrightarrow{\mu_i} (k-1, i)$. This transition happens when a packet-in message is completed in phase i, where $k = c$, $i = 1$ or $k = N$, $i = 2$; and λ_i and μ_i represent the arrival and service rates when the system is in phase i (or state i), respectively.

Under stability condition (see Section V-A), when the number of packet-in messages in the system is larger than or equal to the threshold N and the current state is under-provisioning, the d semi-dormant controllers return from the working vacations and proceed to serve the waiting packet-in messages. When the number of packet-in messages in the system is less than the number c of the controllers and the current state is over-provisioning, the idle controller immediately starts to take working vacation until the number of the working vacation controllers is equal to d.

IV. Queueing Model

A. Quasi-Birth-and-Death Process Model

In this section, we establish the *N/(d,c)-M/M/c/K/AMWV* queueing system, and analyze the implement cost and performance of the FSDMC model. For convenience, the packet-in message, controller, and semi-dormant state are referred to as *customer*, *server*, and *working vacation state*, respectively. In the FSDMC, it is assumed that the interarrival time, the service time during a normal busy period, and

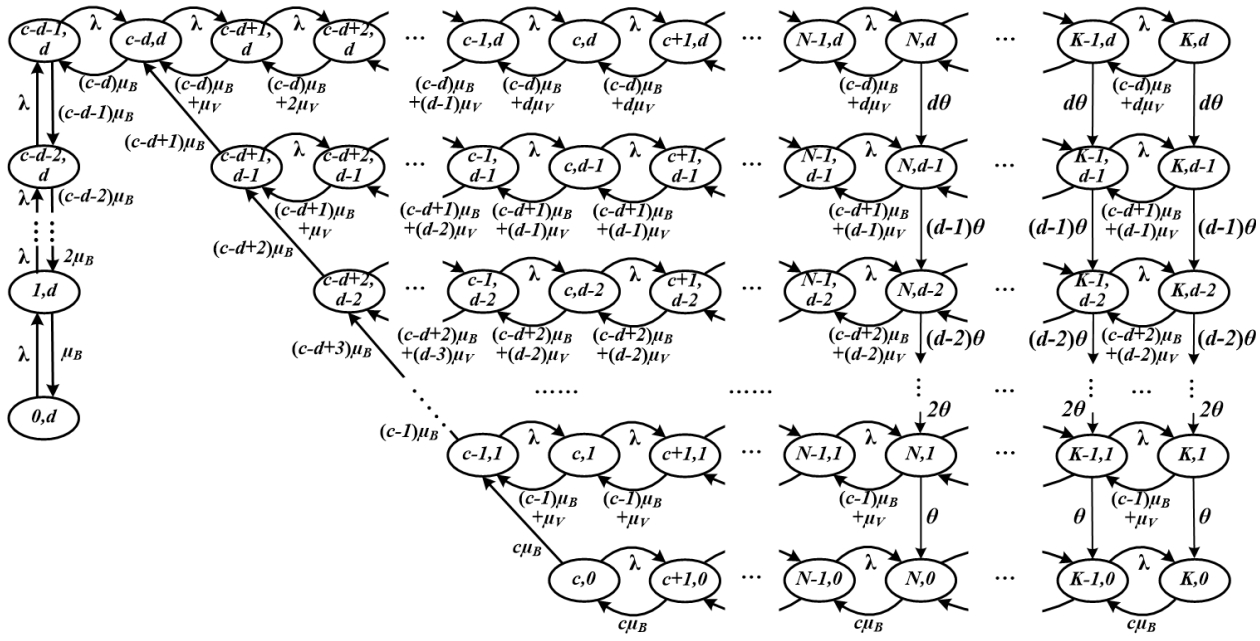


Fig. 1. State-transition-rate diagram for the *N/(d,c)-M/M/c/K/AMWV* queuing system.

the service time during a working vacation follow exponential distributions with parameter λ, μ_B, and μ_V ($\mu_V < \mu_B$) in the single global queue, respectively, [17].

The other features of the queueing system proposed in this paper are exactly the same as those of the queueing system presented in [17], except for the different ways of working vacations. Particularly, the former is an asynchronous way of multiple working vacations of partial servers, whereas the latter is a synchronous way. In the *N/(d,c)-M/M/c/K/AMWV* queue model, when the number of customers in the system is less than the number c of the servers, the idle server immediately starts to take working vacation with exponential rate θ until the number of the working vacation servers is equal to d. When the vacation of a server has been completed and if the number of customers in the system is less than the threshold N, then the vacation server takes another independent and identically distributed working vacation; otherwise, the vacation server returns and proceed to serve the waiting customers with service rate μ_B until the number of the customers is less than or equal to $c-d$.

In this paper, we assume that all interarrival, service, and vacation times are mutually independent, and first-come, first-served discipline is applied in our queueing model. Suppose that a server can serve only one customer at a time [17].

Let $L_v(t)$ and $J(t)$ be the number of packet-in messages and the number of multiple working vacation controllers in the system at time t, respectively. Then, $\{L_v(t), J(t)\}$ is a quasi-birth-and-death (QBD) process with the state space

$$\begin{aligned}\Omega = &\{(k,d) : 0 \le k \le c-d\}\\ &\cup \{(k,j) : c-d < k \le c-1, c-k \le j \le d\}\\ &\cup\{(k,j) : c \le k \le K, 0 \le j \le d\}.\end{aligned}$$

The state-transition-rate diagram for the queuing system is shown in Fig. 1. The lexicographical sequence for the states indicates that the infinitesimal generator Q of the process is the same as that of the *N/(d,c)-M/M/c/K/SMWV* queueing system (see [17, Sec. IV-B]).

Let $\sigma_j = (c-j)\mu_B + j\mu_V$, $\delta_{k,j} = (k-j)\mu_B + j\mu_V$, $0 \le j \le d$, $0 \le k \le c$, and $q = k-c+d$, then $\delta_{c,j} = \sigma_j$. The submatrices in Q are written as follows:

$$A_k = \begin{cases} -\big(\lambda + \delta_{k,0}\big), & 0 \le k \le c-d \\ -\lambda I - \mathrm{diag}\big(\delta_{k,q}, \delta_{k,q-1}, \ldots, \delta_{k,1}, \delta_{k,0}\big) & \\ & c-d < k \le c \\ -\lambda I - \mathrm{diag}(\sigma_d, \sigma_{d-1}, \ldots, \sigma_1, \sigma_0) & \\ & c < k \le N-1 \\ -\mathrm{diag}(\sigma_d, \sigma_{d-1}, \ldots, \sigma_1, \sigma_0) + \Lambda, & k = K, \end{cases}$$

$$B_k = \begin{cases} \delta_{k,0}, & 1 \le k \le c-d \\ \begin{pmatrix} \mathrm{diag}\big(\delta_{k,q}, \delta_{k,q-1}, \ldots, \delta_{k,1}\big) \\ \big(0, 0, \ldots, 0, \delta_{k,0}\big)_{1\times q} \end{pmatrix}_{(q+1)\times q} & \\ & c-d < k \le c \\ B, & c < k \le N-1 \end{cases}$$

$$C_k = \begin{cases} \lambda, & 0 \le k < c-d \\ \big(\lambda I_{(q+1)\times(q+1)} \ 0_{(q+1)\times 1}\big) & \\ & c-d \le k < c \\ \lambda I_{(d+1)\times(d+1)}, & c \le k \le N-1 \end{cases}$$

and

$$\begin{aligned} A &= -\lambda I - \mathrm{diag}(\sigma_d, \sigma_{d-1}, \ldots, \sigma_1, \sigma_0) + \Lambda, \ N \le k < K \\ B &= \mathrm{diag}(\sigma_d, \sigma_{d-1}, \ldots, \sigma_1, \sigma_0), \ N \le k \le K \\ C &= \lambda I_{(d+1)\times(d+1)}, \ N \le k < K \end{aligned}$$

where I is an appropriate order identity matrix, $0_{(q+1)\times 1}$ is a $(q+1)$-dimensional column vector of zeros, $\mathrm{diag}(\delta_{k,q}, \delta_{k,q-1}, \ldots, \delta_{k,1}, \delta_{k,0})$ and $\mathrm{diag}(\sigma_d, \sigma_{d-1}, \ldots, \sigma_1, \sigma_0)$

denote diagonal matrices, and

$$\Lambda = \begin{pmatrix} -d\theta & d\theta & & & & \\ & -(d-1)\theta & (d-1)\theta & & & \\ & & \ddots & \ddots & & \\ & & & -\theta & \theta & \\ & & & & 0 & \end{pmatrix}_{d+1}.$$

This QBD process can be analyzed by solving the minimal non-negative solution of the matrix quadratic equation

$$R^2B + RA + C = 0 \tag{2}$$

and R is called the rate matrix [19].

Theorem 1: If $\rho = \lambda\sigma_0^{-1} < 1$, then the matrix (2) possesses the minimal non-negative solution

$$R = \begin{pmatrix} r_d & r_{d,d-1} & r_{d,d-2} & \cdots & r_{d,1} & r_{d,0} \\ & r_{d-1} & r_{d-1,d-2} & \cdots & r_{d-1,1} & r_{d-1,0} \\ & & r_{d-2} & \cdots & r_{d-2,1} & r_{d-2,0} \\ & & & \ddots & \vdots & \vdots \\ & & & & r_1 & r_{1,0} \\ & & & & & r_0 \end{pmatrix} \tag{3}$$

where 1) the diagonal element r_k is a real root of the quadratic equation

$$\sigma_k z^2 - (\lambda + \sigma_k + k\theta)z + \lambda = 0, 0 \le k \le d \tag{4}$$

and $r_k = (1/2\sigma_k)(\lambda + \sigma_k + k\theta - \sqrt{\Delta})$, where $\Delta = (\lambda + \sigma_k + k\theta)^2 - 4\lambda\sigma_k$, $0 < r_k < 1$, $r_0 = \rho$, and r_k satisfies the following relationship:

$$\lambda + k\theta + \sigma_k(1 - r_k) = \sigma_k + \frac{k\theta}{1 - r_k} = \frac{\lambda}{r_k} \tag{5}$$

2) the nondiagonal element satisfies

$$r_{u,v} = \begin{cases} \frac{(v+1)\theta r_{u,v+1}}{[(\lambda+\sigma_v+v\theta)-\sigma_v(r_v+r_u)]}, & u = v+1 \\ \frac{(v+1)\theta r_{u,v+1} + \sigma_v \sum_{g=v+1}^{u-1} r_{u,g}r_{g,v}}{[(\lambda+\sigma_v+v\theta)-\sigma_v(r_v+r_u)]}, & v+1 < u \le d \end{cases} \tag{6}$$

where $u = d-i$, $v = d-j$, $0 \le i < d$, $i < j \le d$, and $r_{k,k} = r_k$, $0 \le k \le d$.

Proof: It is assumed that another real root of (4) is r_k^*. Then, we have

$$r_k, r_k^* = \frac{1}{2\sigma_k}\left(\lambda + \sigma_k + k\theta \pm \sqrt{\Delta}\right).$$

Obviously, the following inequalities hold:

$$(\lambda - \sigma_k + k\theta)^2 < \Delta < (\lambda + \sigma_k + k\theta)^2, \ \lambda > \sigma_k$$
$$(\sigma_k - \lambda + k\theta)^2 < \Delta < (\lambda + \sigma_k + k\theta)^2, \ \lambda < \sigma_k.$$

We can obtain $0 < r_k < 1$, $r_k^* \ge 1$, and $r_k = (1/2\sigma_k)(\lambda + \sigma_k + k\theta - \sqrt{\Delta})$.

If $k = 0$, then $r_0 = \rho$.

Substituting r_k into (4) yields (5).

In (2), each of matrices A, B, and C is an upper triangular matrix. Matrix R is also an upper triangular matrix. Hence, matrix R^2 can also be represented as an upper triangular matrix. Let R be denoted as (3). Then,

$$\left(R^2\right)_{i,j} = \begin{cases} r_{d-j}^2, & i = j, 0 \le j \le d \\ \sum_{h=i}^{j} r_{d-i,d-h}r_{d-h,d-j}, & 0 \le i < d, i < j \le d. \end{cases}$$

Set $u = d-i$, $v = d-j$, $g = d-h$, and $r_{k,k} = r_k$, $0 \le k \le d$. Substituting R^2 and R into the matrix (2), yields

$$\begin{cases} \sigma_v r_v^2 - (\lambda + \sigma_v + v\theta)r_v + \lambda = 0, & 0 \le v \le d \\ \sigma_v \sum_{g=v}^{u} r_{u,g}r_{g,v} + (v+1)\theta r_{u,v+1} - (\lambda + \sigma_v + v\theta)r_{u,v} = 0 & \\ \qquad 0 < u \le d, 0 \le v < u. & \end{cases} \tag{7}$$

In (7), the first equation is (4). We take r_v to be the root of (4) over the interval (0, 1), where $r_v = (1/2\sigma_v)(\lambda + \sigma_v + v\theta - \sqrt{\Delta})$, and $\Delta = (\lambda + \sigma_v + v\theta)^2 - 4\lambda\sigma_v$. In the second equation, if $u = v + 1$, then $\sum_{g=v}^{u} r_{u,g}r_{g,v} = r_{u,v}r_{v,v} + r_{u,u}r_{u,v}$; if $u > v+1$, then $\sum_{g=v}^{u} r_{u,g}r_{g,v} = r_{u,v}r_{v,v} + r_{u,u}r_{u,v} + \sum_{g=v+1}^{u-1} r_{u,g}r_{g,v}$. Accordingly, (6) can be easily obtained. Obviously, $\mathrm{SP}(R) = \max_{0 \le k \le d}\{r_k\} < 1$ if and only if $\rho < 1$. ■

B. Equilibrium Probability Distribution

If $\rho < 1$, then let (L_v, J) be the equilibrium limit of the QBD process $\{L_v(t), J(t)\}$. Let

$$\pi_{k,j} = \lim_{t \to \infty} P\{L_v(t) = k, J(t) = j\}, \ (k, j) \in \Omega.$$

Then

$$\pi_k = \begin{cases} \pi_{k,d}, & 0 \le k \le c-d \\ \left(\pi_{k,d}, \pi_{k,d-1}, \ldots, \pi_{k,c-k+1}, \pi_{k,c-k}\right)_{1 \times (k-c+d+1)} & \\ \qquad c-d < k \le c & \\ \left(\pi_{k,d}, \pi_{k,d-1}, \ldots, \pi_{k,1}, \pi_{k,0}\right)_{1 \times (d+1)}, & c < k \le K. \end{cases}$$

Theorem 2: If $\rho < 1$, then the equilibrium probability distribution $\pi_{k,j}$ of (L_v, J) is

$$\pi_{k,j} = \begin{cases} G\dfrac{\lambda^k}{\prod_{i=1}^{k}\delta_{i,0}}, & 0 \le k \le c-d, \ j = d \quad (8) \\ \dfrac{\lambda^{k+j-c}\pi_{c-j,j}}{\prod_{i=1}^{k+j-c}\delta_{c-j+i,i}} + \left[\pi_{N,j} - \pi_{c,j}\left(\dfrac{\lambda}{\sigma_j}\right)^{N-c}\right] & \\ \quad \times \sum_{h=0}^{k+j-c-1} \dfrac{\sigma_j\lambda^h}{\prod_{i=k-h}^{k}\delta_{i,i+j-c}} & \\ \qquad c-d+1 \le k \le c, c-k+1 \le j \le d \quad (9) \\ \dfrac{1}{\delta_{k,0}} \sum_{i=c-k+1}^{d} \sigma_i\left[\pi_{c,i}\left(\dfrac{\lambda}{\sigma_i}\right)^{N-c} - \pi_{N,i}\right] & \\ \qquad c-d+1 \le k < c, \ j = c-k \quad (10) \\ \pi_{c,j}\left(\dfrac{\lambda}{\sigma_j}\right)^{k-c}, & c < k < N, 1 \le j \le d \quad (11) \\ \pi_{c,0}\dfrac{1-\rho^{k-c+1}}{1-\rho}, & c < k < N, j = 0 \quad (12) \end{cases}$$

and

$$\pi_{N,j} = \begin{cases} r_d\pi_{c,d}\left(\dfrac{\lambda}{\sigma_d}\right)^{N-1-c}, & j = d \quad (13) \\ \dfrac{r_j}{\lambda}\left[\sigma_j \sum_{i=j+1}^{d} \pi_{N,i}r_{i,j} + (j+1)\theta\pi_{N,j+1}\right] & \\ \quad + r_j\pi_{c,j}\left(\dfrac{\lambda}{\sigma_j}\right)^{N-1-c}, & 1 \le j \le d-1 \quad (14) \\ \pi_{c,0}\dfrac{1-\rho^{N-c+1}}{1-\rho}, & j = 0 \quad (15) \end{cases}$$

where

$$\pi_{c,j}=\begin{cases}\dfrac{G\frac{\lambda^c}{\prod_{w=1}^{c-d}\delta_{w,0}\prod_{i=1}^{d}\delta_{c-d+i,i}}}{1+(\lambda-\sigma_d r_d)\left(\frac{\lambda}{\sigma_d}\right)^{N-1-c}\Phi(d)}, & j=d \quad (16)\\ \dfrac{\frac{r_j\sigma_j}{\lambda}\left[\sigma_j\sum_{i=j+1}^{d}\pi_{N,i}r_{i,j}+(j+1)\theta\pi_{N,j+1}\right]}{\frac{1}{\Phi(j)}+(\lambda-\sigma_j r_j)\left(\frac{\lambda}{\sigma_j}\right)^{N-1-c}} \\ \quad+\dfrac{\frac{\lambda^j\pi_{c-j,j}}{\Phi(j)\prod_{i=1}^{j}\delta_{c-j+i,i}}}{\frac{1}{\Phi(j)}+(\lambda-\sigma_j r_j)\left(\frac{\lambda}{\sigma_j}\right)^{N-1-c}}, & 1\le j\le d-1 \quad (17)\\ \sum_{i=1}^{d}\pi_{N,i}r_{i,0}+\dfrac{\theta}{\sigma_0}\pi_{N,1}, & j=0 \quad (18)\end{cases}$$

where $\Phi(u)=\sum_{h=0}^{u-1}\lambda^h(\prod_{i=c-h}^{c}\delta_{i,i+u-c})^{-1}$, and the constant factor G can be yielded by the normalization condition.

Proof: α-dimensional row vector $(\pi_0,\pi_1,\pi_2,\ldots,\pi_N)$ satisfies the set of equations

$$(\pi_0,\pi_1,\pi_2,\ldots,\pi_N)B[R]=0$$

where $\alpha=(1/2)d^2+(N-c-(1/2))d+N+1$, and $B[R]$ is an $\alpha\times\alpha$ square matrix shown in [17, Th. 2].

Substituting $B[R]$ into the above equation yields α linear equations, as shown as follows:

$$-\lambda\pi_{0,d}+\delta_{1,0}\pi_{1,d}=0,\ k=0,\ j=d \quad (19)$$

$$\lambda\pi_{k-1,d}-\big(\lambda+\delta_{k,0}\big)\pi_{k,d}+\delta_{k+1,0}\pi_{k+1,d}=0,\quad 1\le k\le c-d-1,\ j=d \quad (20)$$

$$\lambda\pi_{c-d-1,d}-\big(\lambda+\delta_{c-d,0}\big)\pi_{c-d,d}+\delta_{c-d+1,0}\pi_{c-d+1,d-1}+\delta_{c-d+1,1}\pi_{c-d+1,d}=0,\ k=c-d,\ j=d \quad (21)$$

$$-\big(\lambda+\delta_{k,0}\big)\pi_{k,j}+\delta_{k+1,0}\pi_{k+1,j-1}+\delta_{k+1,1}\pi_{k+1,j}=0,\quad c-d<k\le c-1,\ j=c-k \quad (22)$$

$$\lambda\pi_{k-1,j}-\big(\lambda+\delta_{k,k+j-c}\big)\pi_{k,j}+\delta_{k+1,k+j-c+1}\pi_{k+1,j}=0,\quad c-d<k\le c-1,\ c-k+1\le j\le d \quad (23)$$

$$-\big(\lambda+\delta_{c,0}\big)\pi_{c,0}+\delta_{c,0}\pi_{c+1,0}=0,\ k=c,\ j=0 \quad (24)$$

$$\lambda\pi_{k-1,0}-(\lambda+\sigma_0)\pi_{k,0}+\sigma_0\pi_{k+1,0}=0,\quad c+1\le k\le N-1,\ j=0 \quad (25)$$

$$\lambda\pi_{k-1,j}-\big(\lambda+\sigma_j\big)\pi_{k,j}+\sigma_j\pi_{k+1,j}=0,\quad c\le k\le N-1,\ 1\le j\le d \quad (26)$$

$$\lambda\pi_{N-1,j}+\sum_{i=j+1}^{d}\pi_{N,i}r_{i,j}\sigma_j+(j+1)\theta\pi_{N,j+1}+\big[r_j\sigma_j-\big(\lambda+\sigma_j+j\theta\big)\big]\pi_{N,j}=0,\ k=N,\ 0\le j\le d-1 \quad (27)$$

$$\lambda\pi_{N-1,d}+[r_d\sigma_d-(\lambda+\sigma_d+d\theta)]\pi_{N,d}=0,\quad k=N,\ j=d. \quad (28)$$

From (20), we have

$$\delta_{k,0}\pi_{k,d}-\lambda\pi_{k-1,d}=\delta_{1,0}\pi_{1,d}-\lambda\pi_{0,d},\ 1\le k\le c-d. \quad (29)$$

From (19) and (29), we obtain (8).

From (24), (25), and $\delta_{c,0}=\sigma_0$, we get

$$\pi_{k+1,0}=\rho\pi_{k,0}+\pi_{c,0},\ c<k<N. \quad (30)$$

When (30) is iterated, we obtain (12).

Therefore, (26) can be satisfied by (11).

From (5), (11), and (28), we obtain (13), that is,

$$\pi_{N,d}=r_d\pi_{N-1,d}=r_d\pi_{c,d}\left(\frac{\lambda}{\sigma_d}\right)^{N-1-c}.$$

Substituting (11) and (12) into (27) yields (14) and (15), respectively. From (26), we have

$$\sigma_j\pi_{N,j}-\lambda\pi_{N-1,j}=\sigma_j\pi_{c,j}-\lambda\pi_{c-1,j},\ 1\le j\le d. \quad (31)$$

From (23), (31), and $\delta_{c,j}=\sigma_j$, we obtain

$$\delta_{k,k+j-c}\pi_{k,j}=\lambda\pi_{k-1,j}+\delta_{c,j}\pi_{N,j}-\lambda\pi_{N-1,j},\quad c-d<k\le c,\ c-k+1\le j\le d. \quad (32)$$

Iterating (32) yields (9). In (9), let $k=c, j=d$. Substituting (8) and (13) into (9) yields (16).

From (20) and (21), we obtain

$$\pi_{c-d+1,d-1}=\lambda\pi_{N-1,d}-\delta_{c,d}\pi_{N,d}=\frac{\lambda-\delta_{c,d}r_d}{\delta_{c-d+1,0}}\left(\frac{\lambda}{\sigma_d}\right)^{N-1-c}\pi_{c,d}. \quad (33)$$

From (22), we obtain

$$\pi_{k,j}=\frac{\lambda+\delta_{k-1,0}}{\delta_{k,0}}\pi_{k-1,j+1}-\frac{\delta_{k,1}}{\delta_{k,0}}\pi_{k,j+1},\quad c-d+2\le k\le c-1, j=c-k. \quad (34)$$

Substituting (9) into (34) yields (35)

$$\pi_{k,j}=\frac{\delta_{k-1,0}\pi_{k-1,j+1}}{\delta_{k,0}}+\frac{\sigma_{j+1}}{\delta_{k,0}}\left[\pi_{c,j+1}\left(\frac{\lambda}{\sigma_{j+1}}\right)^{N-c}-\pi_{N,j+1}\right]. \quad (35)$$

Iterating (35) yields

$$\pi_{k,j}=\frac{1}{\delta_{k,0}}\sum_{i=c-k+1}^{d}\sigma_i\left[\pi_{c,i}\left(\frac{\lambda}{\sigma_i}\right)^{N-c}-\pi_{N,i}\right],\quad c-d+2\le k\le c-1,\ j=c-k. \quad (36)$$

Therefore, (33) satisfies (36) when $k=c-d+1$ and $j=d-1$ in (36). Thus, (10) can be derived from (33) and (36).

In (9), let $1\le j\le d-1$. Then, $s=c-j$. From (9), (10), and (14), we obtain (17).

Equation (18) can be expressed from (12) and (15).

Finally, G satisfies the normalization condition as follows:

$$\sum_{k=0}^{c-d-1}\pi_{k,d}+\sum_{j=0}^{d}\sum_{k=c-j}^{K}\pi_{k,j}=1.$$

■

Theorem 3: If $k>N$, then the equilibrium probability distribution of (L_v, J) is

$$\pi_{k,d}=\pi_{N,d}r_d^{k-N},\ N<k<K,\ j=d \quad (37)$$

$$\varphi_k^{(d)} = \pi_{N,d} \sum_{i=1}^{k-N} r_d^{i-1} \eta H^{k-N-i} + \varphi_N^{(d)} H^{k-N} \quad N < k < K,\ 0 \le j \le d-1 \tag{38}$$

$$\pi_{K,d} = \frac{\lambda r_d^{K-1-N}}{\sigma_d + d\theta} \pi_{N,d},\ j = d \tag{39}$$

$$\pi_{K,j} = \lambda \sum_{h=j}^{d-1} \left[\frac{1}{(h+1)\theta} \prod_{i=j}^{h} \frac{(i+1)\theta}{\sigma_i + i\theta} \pi_{K-1,h} \right] + \frac{\lambda r_d^{K-1-N}}{(d+1)\theta} \prod_{i=j}^{d} \frac{(i+1)\theta}{\sigma_i + i\theta} \pi_{N,d},\ 0 \le j \le d-1 \tag{40}$$

where $\varphi_k^{(d)} = (\pi_{k,d-1}, \pi_{k,d-2}, \ldots, \pi_{k,1}, \pi_{k,0})_{1\times d}$, $\eta = (r_{d,d-1}, r_{d,d-2}, \ldots, r_{d,1}, r_{d,0})$, and $R = \begin{pmatrix} r_d & \eta \\ 0 & H \end{pmatrix}$.

Proof: When $N < k < K$, with the matrix-geometric solution [19], we obtain $\pi_k = \pi_N R^{k-N}$, that is,

$$\left(\pi_{k,d}, \varphi_k^{(d)}\right) = \left(\pi_{N,d}, \varphi_N^{(d)}\right) R^{k-N}. \tag{41}$$

We can obtain the k power of the rate matrix R as follows:

$$R^k = \begin{pmatrix} r_d^k & \sum_{i=1}^{k} r_d^{i-1} \eta H^{k-i} \\ 0 & H^k \end{pmatrix}. \tag{42}$$

Substituting (42) into (41) yields (37) and (38), respectively. When $k = K$, based on the equilibrium equations, we have

$$\lambda \pi_{K-1,d} - (\sigma_d + d\theta)\pi_{K,d} = 0,\ j = d \tag{43}$$

$$\lambda \pi_{K-1,j} - \left(\sigma_j + j\theta\right)\pi_{K,j} + (j+1)\theta \pi_{K,j+1} = 0 \quad 0 \le j \le d-1. \tag{44}$$

From (43), we obtain (39), that is,

$$\pi_{K,d} = \frac{\lambda}{\sigma_d + d\theta} \pi_{K-1,d} = \frac{\lambda r_d^{K-1-N}}{\sigma_d + d\theta} \pi_{N,d}.$$

From (44), we have

$$\pi_{K,0} = \lambda \sum_{h=0}^{j-1} \frac{h! \theta^h}{\prod_{i=0}^{h} (\sigma_i + i\theta)} \pi_{K-1,h} + \frac{j! \theta^j}{\prod_{i=0}^{j-1} (\sigma_i + i\theta)} \pi_{K,j},\ 1 \le j \le d. \tag{45}$$

In (45), let $j = d$. Then, obtain the next expression

$$\pi_{K,j} = \lambda \sum_{h=j}^{d-1} \left[\frac{1}{(h+1)\theta} \prod_{i=j}^{h} \frac{(i+1)\theta}{\sigma_i + i\theta} \pi_{K-1,h} \right] + \prod_{i=j}^{d-1} \frac{(i+1)\theta}{\sigma_i + i\theta} \pi_{K,d},\ 0 \le j \le d-1. \tag{46}$$

Substituting (39) into (46) yields (40). ■

C. Performance Measures

Using the above equilibrium distribution, the performance indices of the *N/(d,c)-M/M/c/K/AMWV* queueing system can be numerically evaluated as follows.

1) The mean number of customers in the system is

$$E[L_s] = \sum_{i=0}^{c-d-1} i\pi_{i,d} + \sum_{j=0}^{d} \sum_{i=c-j}^{K} i\pi_{i,j}. \tag{47}$$

2) The average number of customers in the queue is

$$E\left[L_q\right] = \sum_{j=0}^{d} \sum_{i=c-j+1}^{K} (i - c + j)\pi_{i,j}. \tag{48}$$

3) The expected number of servers during normal busy periods is

$$E[NB] = c - E[WV] - E[I]. \tag{49}$$

4) The expected number of servers during working vacation periods is

$$E[WV] = d \sum_{i=0}^{c-d-1} \pi_{i,d} + \sum_{j=1}^{d} \sum_{i=c-j}^{K} j\pi_{i,j}. \tag{50}$$

5) The mean number of servers during idle periods is

$$E[I] = \sum_{i=0}^{c-d} (c - d - i)\pi_{i,d}. \tag{51}$$

6) The average waiting time that the customer sojourns in the queue is

$$E[T_q] = \frac{E[L_q]}{\lambda(1 - P_{\text{loss}})}. \tag{52}$$

7) The loss probability in the system is

$$P_{\text{loss}} = \sum_{j=0}^{d} \pi_{K,j}. \tag{53}$$

V. Optimization and Sensitivity Analysis

A. Minimizing Average CPR

1) Cost: We utilize the concept of crew-service equipment by White *et al.* [42]. The average cost function per unit time is given by

$$F(d, N, K, \mu_B, \mu_V, \lambda, \theta) = C_h E[L_s] + C_b E[NB] + C_v E[WV] + C_i E[I] + C_d E\left[T_q\right] + C_l P_{\text{loss}} \tag{54}$$

where the cost parameters C_h, C_b, C_v, C_i, C_d, and C_l are defined in [17], and $E[L_s]$, $E[NB]$, $E[WV]$, $E[I]$, $E[T_q]$, P_{loss} are given by (47) and (49)–(53), respectively.

2) Performance: We use the reciprocal of the average delay time that the customer stays in the system as the performance index (see [15, Sec. V]): $P = (1/E[T])$, where $E[T] = (E[L_s]/[\lambda(1 - P_{\text{loss}})])$.

3) CPR: In accordance with [15, eq. (16)] and (54), we can define the CPR of the FSDMC as $\text{CPR} = (F/P) = F \times E[T]$. We consider the optimal model as follows:

$$\begin{aligned} &\text{Minimize} \quad \text{CPR}(d, N, K, \mu_B, \mu_V, \lambda, \theta) \\ &\text{subject to} \quad \begin{cases} 0 < d < c < N < K < K', \ \mu_V < \mu_B \\ {\mu'}_B \le \mu_B \le {\mu''}_B, \ {\mu'}_V \le \mu_V \le {\mu''}_V \\ \lambda' \le \lambda \le \lambda'', \theta' \le \theta \le \theta'' \\ E[NB] \le c - d \end{cases} \end{aligned} \tag{55}$$

where the lower and upper bounds of the parameters are described in [17, Sec. V].

We aim to determine the optimal values of parameters d, N, K, μ_B, μ_V, λ, θ in the system to provide the configuration of the system in steady state. Thus, a PSO algorithm [20] is implemented to address the optimization problem numerically.

4) Optimal System Configuration: We solve the optimization model (55) to find the minimum value of the CPR in the system by using PSO algorithm. We select various parameter values of the system in [17] as follows: $C_h = 1$ mW, $C_b = 35$ mW, $C_v = 15$ mW, $C_i = 10$ mW, $C_d = 8$ mW, $C_l = 10000$ mW, and $c = 10$. The system parameters are selected as $d \in [1, c]$, $N \in [c + 1, 25]$, $K \in [N + 1, 60]$, $\mu_B \in [1.6, 3.4]$, $\mu_v \in [0.5, 1.5]$, $\lambda \in [4, 30]$, and $\theta \in [0.01, 10]$, and they meet the constraint conditions in the optimization model (55).

In our experiment, the optimal values of the decision variables are yielded as follows: $d = 4$, $N = 20$, $K = 42$, $\mu_B = 3.4$, $\mu_V = 1.5$, $\lambda = 16.9337$, $\theta = 6.6058$. The minimum value of CPR is obtained, that is, $\text{CPR}_0 = 53.1134$. We configure the system using the optimal values of parameters. At this point, we call this system a *stable system*. Notably, in the solving process, the optimal values of μ_B and μ_V are always taken from the upper bounds μ''_B and μ''_V of the respective constraints, because the CPR decreases with the increase of μ_B or μ_V (see Figs. 3 and 4 and Tables V and VI). Therefore, we adopt the values of μ_B and μ_V in [17] as the values of parameters μ_B and μ_V in our system, respectively.

B. Sensitivity Analysis of Performance Indices

Under stability condition, the numerical results show that changes in system parameters have an impact on system performance. Let $K = 42$ and consider that system parameters take different values in the following three cases.

Case 1: $\lambda = 16.9$, $\theta = 6.6$, $d = 4$, $c = 10$, $N = 20$, and changing the values of (μ_B, μ_V).

Case 2: $\mu_B = 3.4$, $\mu_V = 1.5$, $d = 4$, $c = 10$, $N = 20$, and changing the values of (λ, θ).

Case 3: $\lambda = 16.9$, $\theta = 6.6$, $\mu_B = 3.4$, $\mu_V = 1.5$, and changing the values of (d, c, N).

The numerical results for the above three cases are shown in Tables I–III, respectively. Table I shows that, as the value of μ_B or μ_V increase, $E[L_s]$, $E[L_q]$, and $E[NB]$ decrease and $E[WV]$ slightly decreases but $E[I]$ increases. As shown in Table II: 1) $E[L_s]$ and $E[NB]$ increase with the increase of λ but slightly increase with the increase of θ; 2) $E[L_q]$ and $E[WV]$ increase with the increase of λ but slightly decrease with the increase of θ; and 3) $E[I]$ decreases with the increase of λ but does not change with the increase of θ. Table III shows that: 1) $E[L_s]$, $E[L_q]$, and $E[WV]$ drastically increase as d increases; however, $E[NB]$ and $E[I]$ drastically decrease as d increases; 2) that as the values of c increase, $E[L_s]$ and $E[WV]$ decrease and $E[L_q]$ drastically decreases; meanwhile, $E[NB]$ increases and $E[I]$ drastically increases; and 3) that similar to 1), $E[L_s]$, $E[L_q]$, and $E[WV]$ slightly increase but $E[NB]$ and $E[I]$ slightly decrease with the increase of N. These results are consistent with the actual situation for the *N/(d,c)-M/M/c/K/AMWV* queue.

TABLE I
SYSTEM PERFORMANCE INDICES FOR DIFFERENT VALUES OF (μ_B, μ_V) WITH $\lambda = 16.9$, $\theta = 6.6$, $d = 4$, $c = 10$, AND $N = 20$

(μ_B, μ_v)	(2.4, 1.5)	(3.4, 1.5)	(4.4, 1.5)	(3.4, 0.5)	(3.4, 2.5)
$E[L_s]$	9.2830	5.5363	4.0266	6.2840	5.1866
$E[L_q]$	3.3622	0.8914	0.2710	1.4613	0.6358
$E[NB]$	5.1571	4.2040	3.5380	4.3220	4.1467
$E[WV]$	4.4462	4.4362	4.2175	4.4683	4.4033
$E[I]$	0.3967	1.3598	2.2444	1.2097	1.4500

TABLE II
SYSTEM PERFORMANCE INDICES FOR DIFFERENT VALUES OF (λ, θ) WITH $\mu_B = 3.4$, $\mu_V = 1.5$, $d = 4$, $c = 10$, AND $N = 20$

(λ, θ)	(15.9, 6.6)	(16.9, 6.6)	(17.9, 6.6)	(16.9, 5.6)	(16.9, 7.6)
$E[L_s]$	5.0840	5.536280	6.0379	5.536249	5.536423
$E[L_q]$	0.6585	0.891352	1.1843	0.891786	0.891037
$E[NB]$	4.0486	4.204032	4.3480	4.203491	4.204562
$E[WV]$	4.3752	4.436159	4.4940	4.436711	4.435634
$E[I]$	1.5763	1.359808	1.1580	1.359798	1.359803

TABLE III
SYSTEM PERFORMANCE INDICES FOR DIFFERENT VALUES OF (d, c, N) WITH $\lambda = 16.9$, $\theta = 6.6$, $\mu_B = 3.4$, AND $\mu_V = 1.5$

(d, c, N)	(4, 10, 20)	(7, 10, 20)	(4, 15, 20)	(4, 10, 15)
$E[L_s]$	5.5363	8.0472	4.9762	5.4805
$E[L_q]$	0.8914	4.7934	0.0098	0.8135
$E[NB]$	4.2040	2.3067	4.9526	4.2314
$E[WV]$	4.4362	7.6232	4.0138	4.3995
$E[I]$	1.3598	0.0700	6.0336	1.3691

C. Sensitivity Analysis of CPR

Under stability condition, some results from our numerical experiment show that the CPR is impacted by the change in system parameters. We fix the values of the cost parameters as follows: $C_h = 1$ mW, $C_b = 35$ mW, $C_v = 15$ mW, $C_i = 10$ mW, $C_d = 8$ mW, $C_l = 10000$ mW, controller number $c = 10$, and capacity $K = 42$. Consider that system parameters take different values in the following eight cases.

Case 4: $N = 20$, $\mu_B = 3.4$, $\mu_V = 1.5$, $\theta = 6.6$, and changing the values of λ in the interval [1.0, 30.0] and $d = 4, 7, 9$.

Case 5: $N = 20$, $\mu_V = 1.5$, $\lambda = 16.9$, $\theta = 6.6$, and changing the values of μ_B in the interval [2.0, 20.0] and $d = 4, 7, 9$.

Case 6: $N = 20$, $\mu_B = 3.4$, $\lambda = 16.9$, $\theta = 6.6$, and changing the values of μ_V in the interval [0, 10.0] and $d = 4, 7, 9$.

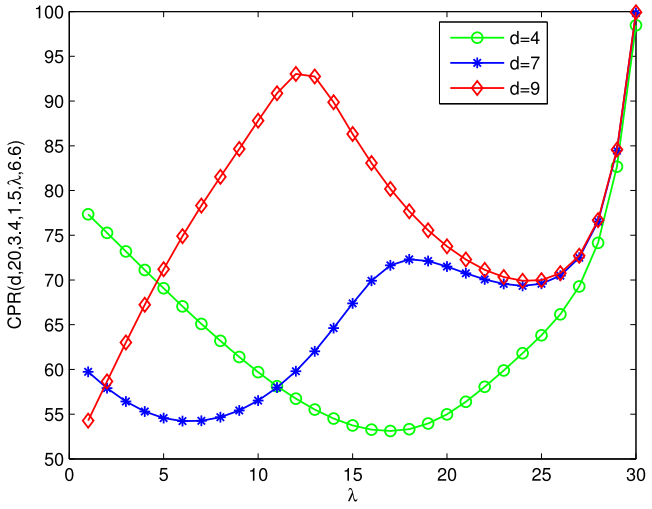


Fig. 2. Plot of CPR(d, 20, 3.4, 1.5, λ, 6.6) versus λ and d.

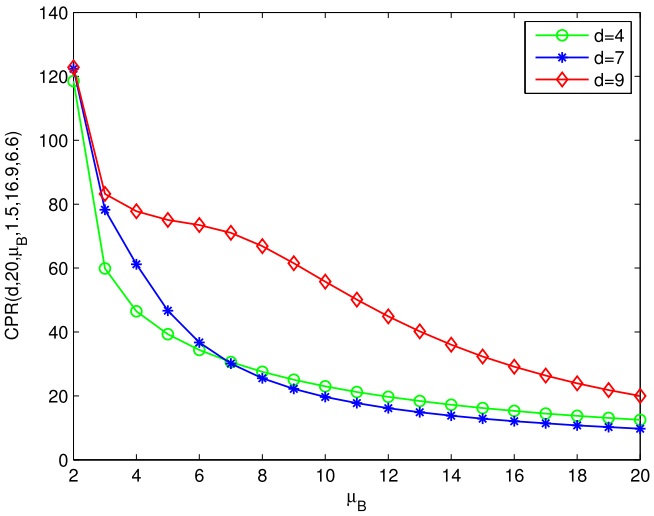


Fig. 3. Plot of CPR(d, 20, μ_B, 1.5, 16.9, 6.6) versus μ_B and d.

Case 7: $N = 20$, $\mu_B = 3.4$, $\mu_V = 1.5$, $\lambda = 16.9$, and changing the values of θ in the interval [2.0, 20.0] and $d = 4, 7, 9$.

Case 8: $d = 4$, $\mu_B = 3.4$, $\mu_V = 1.5$, $\theta = 6.6$, and changing the values of (λ, N).

Case 9: $d = 4$, $\mu_V = 1.5$, $\lambda = 16.9$, $\theta = 6.6$, and changing the values of (μ_B, N).

Case 10: $d = 4$, $\mu_B = 3.4$, $\lambda = 16.9$, $\theta = 6.6$, and changing the values of (μ_V, N).

Case 11: $d = 4$, $\mu_B = 3.4$, $\mu_V = 1.5$, $\lambda = 16.9$, and changing the values of (θ, N).

The experimental results of CPR are reported in Figs. 2–5 for cases 4–7 and in Tables IV–VII for cases 8–11, respectively. As shown in Fig. 2 and Table IV, CPR significantly fluctuates as λ and d increase, but increases with N. As shown in Figs. 3 and 4, CPR drastically decreases as μ_B and μ_V increase but shows fluctuations as d changes. Tables V and VI show that CPR slightly increases with the increase of N. From Fig. 5 and Table VII, CPR increases with d and N but slightly changes with the increase of θ.

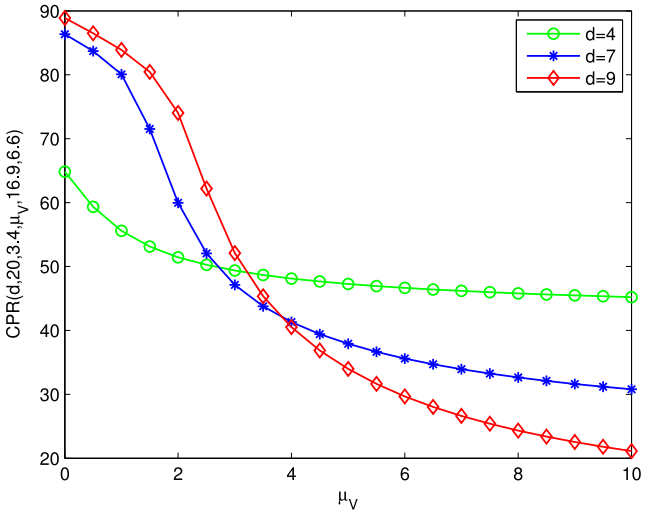


Fig. 4. Plot of CPR(d, 20, 3.4, μ_V, 16.9, 6.6) versus μ_V and d.

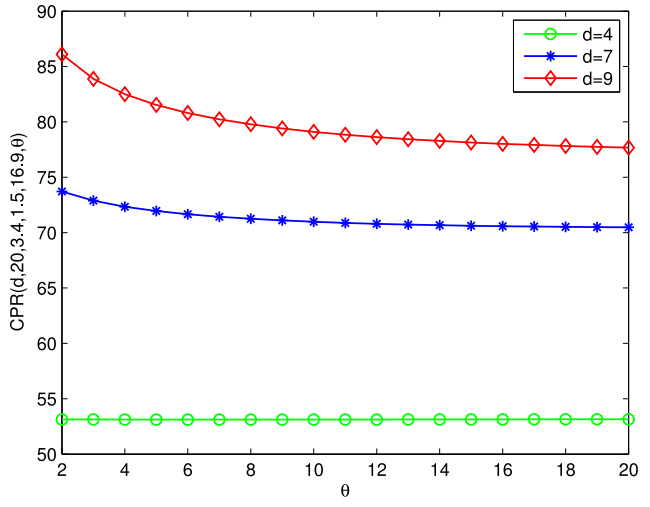


Fig. 5. Plot of CPR(d, 20, 3.4, 1.5, 16.9, θ) versus θ and d.

TABLE IV
CPR FOR DIFFERENT VALUES OF (λ, N) WITH $d = 4$, $\mu_B = 3.4$, $\mu_V = 1.5$, AND $\theta = 6.6$

(λ,N)	(14.9,20)	(16.9,20)	(18.9,20)	(14.9,15)	(14.9,25)	(18.9,15)	(18.9,25)
CPR	53.8112	53.1136	53.8698	53.6692	53.8200	52.2426	54.2922

TABLE V
CPR FOR DIFFERENT VALUES OF (μ_B, N) WITH $d = 4$, $\mu_V = 1.5$, $\lambda = 16.9$, AND $\theta = 6.6$

(μ_B, N)	(2.4, 20)	(3.4, 20)	(4.4, 20)	(3.4, 15)	(3.4, 25)
CPR	78.5825	53.1136	43.2096	52.5643	53.1849

VI. QUANTITATIVE MODEL OF ANELASTICITY

A. M/M/1/K Queue in Markovian Environment

In this section, we consider an *M/M/1/K* queue model in a Markovian environment, where K denotes the system capacity. The Markovian environment is described by a three-state, irreducible Markov renewal process with generator $\bar{Q}$. For simplicity, the three states are represented by states 1, 2, and 3.

TABLE VI
CPR FOR DIFFERENT VALUES OF (μ_V, N) WITH $d = 4$, $\mu_B = 3.4$, $\lambda = 16.9$, AND $\theta = 6.6$

(μ_V, N)	(0.5, 20)	(1.5, 20)	(2.5, 20)	(0.5, 15)	(0.5, 25)
CPR	59.3303	53.1136	50.2562	56.4197	60.4087

TABLE VII
CPR FOR DIFFERENT VALUES OF (θ, N) WITH $d = 4$, $\mu_B = 3.4$, $\mu_V = 1.5$, AND $\lambda = 16.9$

(θ, N)	(4.6, 20)	(6.6, 20)	(8.6, 20)	(8.6, 15)	(8.6, 25)
CPR	53.1163	53.1136	53.1153	52.541	53.186

When a transition from state i to state j occurs in process $\bar{Q}$, the average Poisson arrival rate λ_i, and average service rate μ_i instantaneously change to new values of λ_j and μ_j, respectively, where $1 \le i, j \le 3$.

In accordance with the transitions among the three-state (see Section III), matrix $\bar{Q}$ is written as

$$\bar{Q} = \begin{pmatrix} -\mu_1 & \mu_1 & 0 \\ \mu_2 & -2\mu_2 & \mu_2 \\ 0 & \mu_3 & -\mu_3 \end{pmatrix}.$$

In $\bar{Q}$, the mean service rate μ_i is computed by

$$\mu_i = E[NB_i]\mu_B + E[WV_i]\mu_V, \ \ i = 1, 2, 3 \tag{56}$$

where $E[NB_i]$ and $E[WV_i]$ represent the expected number of servers during normal busy and working vacation periods in the ith state, respectively. Their values can be obtained from (49)–(51).

Let $\tilde{\pi} = (\tilde{\pi}_1, \tilde{\pi}_2, \tilde{\pi}_3)$ be the stationary probability vector of Markov process $\bar{Q}$. Then,

$$\tilde{\pi}\bar{Q} = 0, \ \ \tilde{\pi}e = 1. \tag{57}$$

From (57), we obtain

$$\tilde{\pi} = (\mu_2\mu_3, \mu_3\mu_1, \mu_1\mu_2)(\mu_1\mu_2 + \mu_2\mu_3 + \mu_3\mu_1)^{-1}.$$

By a standard construction, the *M/M/1/K* queue may be studied as a QBD process with generator $\tilde{Q}$, given by

$$\tilde{Q} = \begin{pmatrix} \bar{Q}-\Delta(\lambda) & \Delta(\lambda) & & & & \\ \Delta(\mu) & \bar{Q}-\Delta(\lambda+\mu) & \Delta(\lambda) & & & \\ & \ddots & \ddots & \ddots & & \\ & & \Delta(\mu) & \bar{Q}-\Delta(\lambda+\mu) & \Delta(\lambda) & \\ & & & \Delta(\mu) & \bar{Q}-\Delta(\mu) & \end{pmatrix}$$

where $\Delta(\lambda)$ and $\Delta(\mu)$ represent diagonal matrices $\mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)$ and $\mathrm{diag}(\mu_1, \mu_2, \mu_3)$, respectively, (see [19, Sec. VI-2]).

Matrix $\tilde{R}$ is the minimal non-negative solution of the matrix quadratic equation

$$\tilde{R}^2\Delta(\mu) + \tilde{R}\big[\bar{Q} - \Delta(\lambda + \mu)\big] + \Delta(\lambda) = 0 \tag{58}$$

where $\tilde{R}$ can be solved by employing the fast Newton–Shamanskii iteration algorithm [43].

Let $X = (x_0, x_1, x_2, \ldots, x_K)$ be the stationary probability vector of the stable queue, where $x_k = (x_{k,1}, x_{k,2}, x_{k,3})$, $0 \le k \le K$. In $x_{k,j}$, k and j denote the queue length and the environmental state, respectively, $j = 1, 2, 3$.

Theorem 4: If $\tilde{\rho} = \tilde{\pi}\vec{\lambda}(\tilde{\pi}\vec{\mu})^{-1} < 1$, then the stationary probability distribution x_k is

$$x_k = \begin{cases} \tilde{G}\tilde{\pi}\big(I - \tilde{R}\big)\tilde{R}^k, & 0 \le k < K \\ \tilde{G}\tilde{\pi}\big(I - \tilde{R}\big)\tilde{R}^{K-1}\Delta(\lambda)\big[\Delta(\mu) - \bar{Q}\big]^{-1}, & k = K \end{cases} \tag{59}$$

where $\vec{\lambda} = (\lambda_1, \lambda_2, \lambda_3)^T$, $\vec{\mu} = (\mu_1, \mu_2, \mu_3)^T$, and

$$\tilde{G} = \Big[\tilde{\pi}\big(I - \tilde{R}^K\big)e + \tilde{\pi}\big(I - \tilde{R}\big)\tilde{R}^{K-1}\Delta(\lambda) \times \big(\Delta(\mu) - \bar{Q}\big)^{-1}e\Big]^{-1}. \tag{60}$$

Proof: From [19, Th. 6.2.1], we have

$$x_k = \tilde{G}\tilde{\pi}\big(I - \tilde{R}\big)\tilde{R}^k, \ \ 0 \le k < K \tag{61}$$

where $\tilde{G}$ denotes a pending constant.

From $x_k\tilde{Q} = 0$, we can obtain the equilibrium equation as follows:

$$x_{K-1}\Delta(\lambda) + x_K\big[\bar{Q} - \Delta(\mu)\big] = 0. \tag{62}$$

From (61) and (62), we obtain the second expression in (59).

The pending constant $\tilde{G}$ can be obtained by the normalization condition

$$\sum_{k=0}^{K-1}\tilde{G}\tilde{\pi}\big(I - \tilde{R}\big)\tilde{R}^k e + \tilde{G}\tilde{\pi}\,(I - \tilde{R})\tilde{R}^{K-1}\Delta(\lambda) \times \big[\Delta(\mu) - \bar{Q}\big]^{-1}e = 1. \tag{63}$$

From (63), (60) can be obtained. ■

B. Sensitivity Analysis of Elasticity

From (59), the probabilities that the system is in the three states are calculated using (64), respectively,

$$p_{\text{over}} = \sum_{j=1}^{3}\sum_{i=0}^{c-1}x_{i,j}, \quad p_{\text{normal}} = \sum_{j=1}^{3}\sum_{i=c}^{N-1}x_{i,j}$$
$$p_{\text{under}} = \sum_{j=1}^{3}\sum_{i=N}^{K}x_{i,j}. \tag{64}$$

In the case where the traffic intensity $\tilde{\rho} = (\tilde{\pi}\vec{\lambda})(\tilde{\pi}\vec{\mu})^{-1} < 1$, we use numerical data to demonstrate that elasticity is impacted by the change in the basic parameters of the system. We fix the controller number $c = 10$, the semi-dormant controller number $d = 4$, the threshold value $N = 20$, and the capacity $K = 42$. Consider that system parameters take different values in the following four cases.

Case 12: $\mu_B = 3.4$, $\mu_V = 1.5$, $\theta = 6.6$, and changing the values of λ in the interval [5.0, 35.0].

Case 13: $\lambda = 16.9$, $\mu_V = 1.5$, $\theta = 6.6$, and changing the values of μ_B in the interval [2.0, 20.0].

Case 14: $\lambda = 16.9$, $\mu_B = 3.4$, $\theta = 6.6$, and changing the values of μ_V in the interval [0, 10.0].

Case 15: $\lambda = 16.9$, $\mu_B = 3.4$, $\mu_V = 1.5$, and changing the values of θ in the interval [2.0, 20.0].

The numerical results of the elasticity are depicted in Figs. 6–8 for cases 12–14, respectively. As shown in Fig. 6,

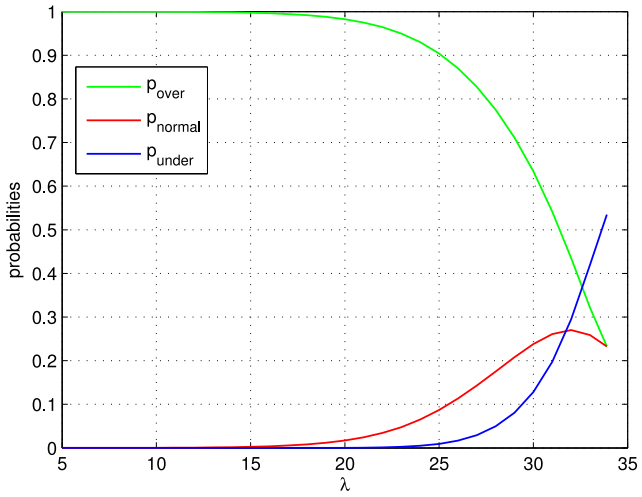


Fig. 6. Plot of p_{over}, p_{normal}, and p_{under} versus λ for $\rho < 1$.

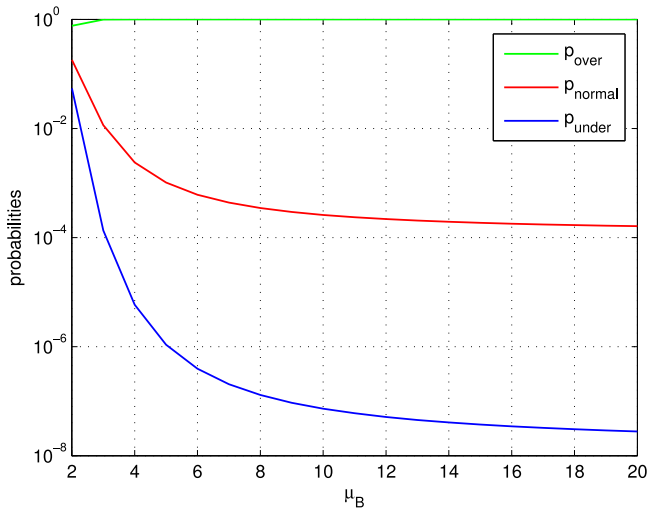


Fig. 7. Plot of p_{over}, p_{normal}, and p_{under} versus μ_B for $\rho < 1$.

as λ increases, p_{over} drastically decreases and p_{under} drastically increases but p_{normal} changes, that is, initially increases and then slightly decreases. From Fig. 7, as μ_B increases, p_{normal} and p_{under} decrease but p_{over} slightly increases. As shown in Fig. 8, as μ_V increases, the change in the elasticity probabilities is the same as that in Fig. 7. However, for case 15, as θ increases, the three probabilities remain unchanged, and $p_{\text{over}} = 0.9945$, $p_{\text{normal}} = 0.5452e-02$, and $p_{\text{under}} = 0.3038e-04$. Therefore, the elasticity of the system exhibits no correlation with the working vacations of the controller.

C. Adaptivity Analysis

In this section, we analyze the adaptive capability of the system. In the *N/(d,c)-M/M/c/K/AMWV* queueing system, when $\rho = \lambda\sigma_0^{-1} < 1$, the system changes the number d of semi-dormant controllers to adapt to the workload changes (see Fig. 2). However, for $\rho \geq 1$, we increase the mean service rate μ_B of the controller in the system during a normal busy period and let $d = 0$ and $\mu_V = \mu_B$. Accordingly, the

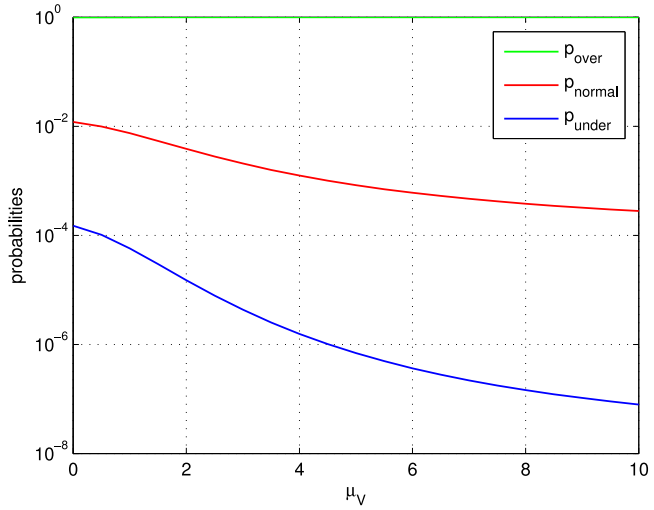


Fig. 8. Plot of p_{over}, p_{normal}, and p_{under} versus μ_V for $\rho < 1$.

queueing system becomes a classic queuing system *M/M/c/K*. As the average arrival rate decreases and when the number of customers in the system is less than c, the idle controller starts to take working vacations until the number of the semi-dormant controllers is equal to d. At this point, the system returns to a stable state again.

As the average arrival rate λ increases to $\rho \geq 1$, the system adaptively adjusts the controller service rate μ_B during a normal busy period while keeping the optimal value CPR_0 of CPR at a stable state. The optimal CPR unchanged adaptive algorithm based on PSO is summarized as Algorithm 1.

In Algorithm 1, rand() represents a random function in the closed interval [0, 1]. Algorithm 1 is essentially an extension of the PSO algorithm. Different from the PSO algorithm, Algorithm 1 sets the upper and lower bounds of the decision variables, and continuously changes the upper and lower bounds by the dichotomy to narrow the search space of the decision variable. Given the minimum value CPR_0 of the objective function, the termination condition of the algorithm is that the global optimum value of the CPR is equal to the minimum value CPR_0 within the given precision. The global optimal position of the particles is obtained by alternately reducing the range of values of the particles while the optimal value of CPR alternates close to the minimum value CPR_0 on both sides. In Algorithm 1, we use a population of 100 particles and a mutation rate of 0.5. Let, high = 20, low = 1.6, and $\text{CPR}_0 = 53.1134$.

In accordance with Algorithm 1, let $d = 0$, $N = 20$, $K = 42$, and vary the values of λ from 34.0 to 65.0. The values of μ_V and θ are negligible, and we can obtain the relationship between μ_B and λ, as shown in Fig. 9. μ_B initially decreases and then significantly increases with the increase of λ.

From (64), when varying the values of λ from 34.0 to 65.0 while μ_B changes accordingly, probabilities p_{over}, p_{normal}, and p_{under} are influenced, as shown in Fig. 10. As λ increases, p_{normal} and p_{under} increase but p_{over} decreases.

Algorithm 1 Optimal CPR Unchanged Adaptive Algorithm Based on PSO

Input: The average arrival rate λ, and tolerance $\varepsilon = 10^{-7}$.
Output: The average arrival rate μ_B of the controller during a normal busy period.

Initialize the upper bounds *high* and the lower bounds *low*, respectively, and $low < high$;
Set the initial value of the CPR is CPR_0;
while TRUE **do**
  for each particle i **do**
    $x_i \leftarrow low + rand() \times (high - low)$;
    $v_i \leftarrow low + rand() \times (high - low)$;
    Evaluate particle i using (55), and set $pBest_i = x_i$;
  end for
  $Best \leftarrow \min\{pBest_i\}$;
  for each particle i **do**
    Update the velocity and position of particle i, and evaluate particle i;
    Update the global optimal position *gBest*, and $gCPR \leftarrow CPR(gBest)$;
  end for
  if $abs(gCPR - CPR_0) < \varepsilon$ **then**
    break;
  else
    $mid \leftarrow (high + low)/2$;
    if $gCPR \leq CPR_0$ **then**
      $temp \leftarrow high$;
      $high \leftarrow mid$;
    else
      $low \leftarrow high$;
      $high \leftarrow temp$;
    end if
  end if
end while
return *gBest*.

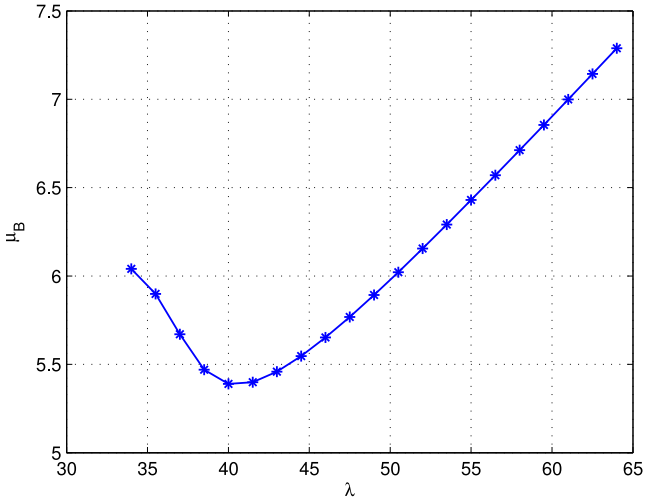


Fig. 9. Plot of μ_B versus λ adaptively.

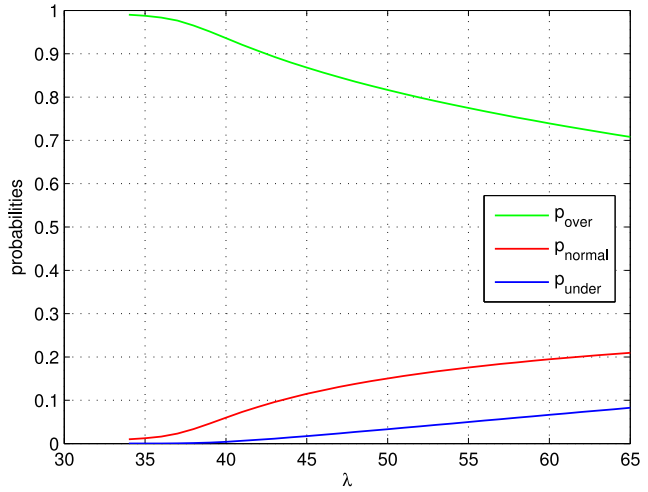


Fig. 10. Plot of p_{over}, p_{normal}, and p_{under} versus λ while μ_B changes accordingly.

In the *M/M/1/K* queue model in a Markovian environment, the conditional queue length densities $q_k(j)$ are given by $q_k(j) = P\{L = k|J = j\} = (x_{k,j}/\tilde{\pi}_j)$, $0 \leq k \leq K$, $1 \leq j \leq 3$. From (1), we obtain that the anelasticity value of the system under stability condition is Anelasticity $= (1.4172, 5.4617e-03)$.

Our experiment exhibits that as λ increases, no linear relationship is found between the conditional average queue length $\text{AvgQL}_{\text{normal}}$ of the queue in Phase 2 and elasticity p_{normal}. As shown in Fig. 11, for $\lambda \in [1, 34)$ (i.e., $\rho = \lambda\sigma_0^{-1} < 1$), as λ increases, the conditional average queue length of the queue in Phase 2 increases but elasticity p_{normal} initially increases and then slightly decreases after one value. However, for $\lambda \in [34, 65]$ (i.e., $\rho = \lambda\sigma_0^{-1} \geq 1$), after adaptively adjusting the service rate value of the controller as $\tilde{\mu}_B$ in the queuing system, we obtain $\rho = \lambda(c\tilde{\mu}_B)^{-1} < 1$. Thus, when the conditional average queue length of the queue in Phase 2 increases, elasticity p_{normal}l, which is denoted by the asterisk, still increases and is the same as that before the adaptive adjustment of the parameters in the system. Therefore, the system can maintain elastic invariance in adaptive adjustment parameters when the average arrival rate of the system exceeds the average service rate.

D. Comparison and Analysis

In [15], the models for calculating probabilities p_{over}, p_{normal}, and p_{under} are expressed by (4), (5), and (6), respectively. We exhibit that the probabilities using the models in [15] are influenced by varying the values of λ from 5.0 to 34.0, as shown in Fig. 12. As the average arrival rate λ continues to increase, λ is greater than or equal to $c\mu_B$ at some cases, where $\mu_B = 3.4$. Therefore, the system adaptively adjusts the controller service rate μ_B in accordance with Algorithm 1. Under the variation in arrival rate λ from 34 to 65 and the change in service rate μ_B, the plot of p_{over}, p_{normal}, and p_{under} versus λ is shown in Fig. 13.

As shown in Fig. 12, as λ increases, the volatility of probability p_{normal} is greater than that shown in Fig. 6 and that the probability value is larger than that shown in Fig. 6. Similarly, the comparison of Figs. 10 and 13 shows that the

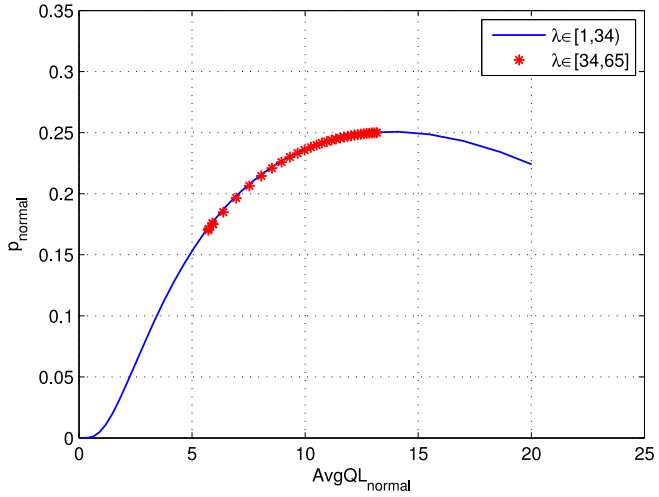


Fig. 11. Plot of $\text{AvgQL}_{\text{normal}}$ versus p_{normal} in the normal state for $\lambda \in [1, 34) \cup [34, 65]$.

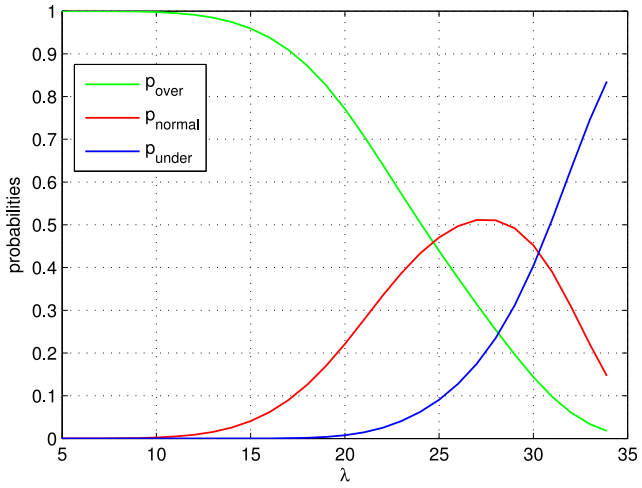


Fig. 12. Plot of p_{over}, p_{normal}, and p_{under} versus λ using models in [15].

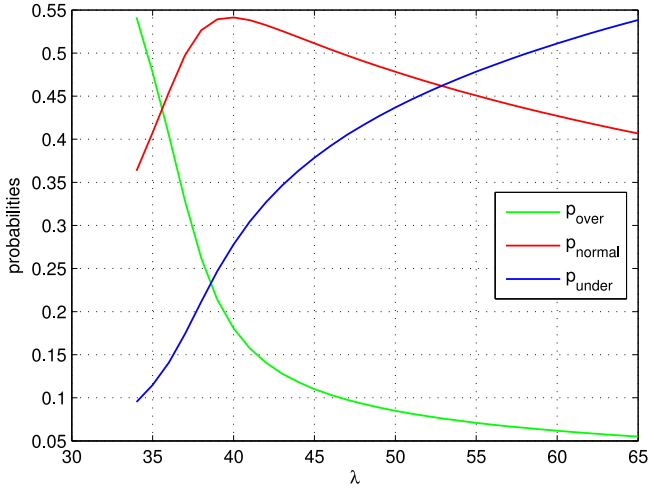


Fig. 13. Plot of p_{over}, p_{normal}, and p_{under} versus λ while μ_B changes accordingly using models in [15].

probability p_{normal} in Fig. 13 is larger than that in Fig. 10. Moreover, as λ increases, p_{normal} initially increases and then decreases (Fig. 13), whereas p_{normal} continues to increase steadily (Fig. 10). Our elastic model is therefore in line with the actual CPS.

TABLE VIII
COMPARISON RESULTS OF THE COST FUNCTION F AND THE NOMINAL TIME DELAY τ_{nom} FOR THE TWO MODELS

Cost/Perform.	*N/(d,c)-M/M/c/K/AMWV*	*N/(d,c)-M/M/c/K/SMWV*
F	234.74mW	237.07mW
τ_{nom}	2.3560e-04s	2.9259e-04s

We compare the *N/(d,c)-M/M/c/K/AMWV* queueing system proposed in this paper with the *N/(d,c)-M/M/c/K/SMWV* queueing system in [17] by using the cost function F and the nominal time delay function τ_{nom}. In the current study, F and τ_{nom} are given by (54) and (52), respectively. We use the values of the system parameters in [17] as the input of the two models. The results of the comparison are shown in Table VIII. From Table VIII, F and τ_{nom} of the queue model in this paper are less than those in [17], respectively. The results show that our model exhibits low cost and high performance.

VII. CONCLUSION

In this paper, we considered resource provisioning under uncertain workload condition on a special CPS. In the shared-bus DESs, an on-demand nature of time-related resource provisioning existed. The anelasticity naturally reflected the on-demand nature with the hysteresis characteristic between stress and strain. We proposed a novel, quantitative, and formal definition of anelasticity model for the special CPS. The optimal values of various parameters in such a system under stability condition were also solved. We established the FSDMC as an *N/(d,c)-M/M/c/K/AMWV* queue model to minimize the CPR of the system. Then, we obtained the respective service rates of three states, namely, over-provisioning, normal, and under-provisioning states. An *M/M/1/K* queue model in a Markovian environment was developed to calculate the anelasticity value of the FSDMC platform accurately. When the average arrival rate exceeded the average service rate in the queueing system, an adaptive adjustment algorithm based on PSO was proposed to adjust the controller service rate dynamically while maintaining the minimum value of the system CPR unchanged under stability condition.

REFERENCES

[1] A. M. Annaswamy, D. Soudbakhsh, R. Schneider, D. Goswami, and S. Chakraborty, "Arbitrated network control systems: A co-design of control and platform for cyber-physical systems," in *Proc. Workshop Control Cyber Phys. Syst.*, Baltimore, MD, USA, Mar. 2013, pp. 339–356.

[2] S. Chakraborty *et al.*, "Automotive cyber-physical systems: A tutorial introduction," *IEEE Design Test*, vol. 33, no. 4, pp. 92–108, Aug. 2016.

[3] D. C. Schmidt, J. White, and C. D. Gill, "Elastic infrastructure to support computing clouds for large-scale cyber-physical systems," in *Proc. 17th IEEE Int. Symp. Object Compon. Service Oriented Real Time Distrib. Comput. (ISORC)*, Reno, NV, USA, Jun. 2014, pp. 56–63.

[4] J. Balasubramanian, "Resource-aware deployment, configuration, and adaptation for fault-tolerant distributed real-time embedded systems," Ph.D. dissertation, Inst. Softw. Integr. Syst., Vanderbilt Uni., Nashville, TN, USA, Dec. 2009.

[5] W. Ai *et al.*, "On elasticity measurement in cloud computing," *Sci. Program.*, vol. 2016, Jun. 2016, Art. no. 7519507.

[6] S. Dustdar, Y. Guo, B. Satzger, and H.-L. Truong, "Principles of elastic processes," *IEEE Internet Comput.*, vol. 15, no. 5, pp. 66–71, Sep./Oct. 2011.

[7] S. Nikam and R. Ingle, "Resource provisioning algorithms for service composition in cyber physical systems," in *Proc. Int. Conf. Adv. Comput. Commun. Informat. (ICACCI)*, New Delhi, India, Sep. 2014, pp. 2797–2802.

[8] D. Moldovan and H.-L. Truong, "A platform for run-time health verification of elastic cyber-physical systems," in *Proc. 24th IEEE Int. Symp. Model. Anal. Simulat. Comput. Telecommun. Syst. (MASCOTS)*, London, U.K., Sep. 2016, pp. 379–384.

[9] D. Moldovan, G. Copil, and S. Dustdar, "Elastic systems: Towards cyber-physical ecosystems of people, processes, and things," *Comput. Stand. Interfaces*, vol. 57, no. 4, pp. 76–82, Mar. 2018.

[10] M. Z. C. Candra, H.-L. Truong, and S. Dustdar, "Modeling elasticity trade-offs in adaptive mixed systems," in *Proc. 22nd IEEE Int. Workshop Enabling Technol. Infrastruct. Collaborative Enterprises (WETICE)*, Hammamet, Tunisia, Jun. 2013, pp. 21–26.

[11] P. Palensky, E. Widl, and A. Elsheikh, "Simulating cyber-physical energy systems: Challenges, tools and methods," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 3, pp. 318–326, Mar. 2014.

[12] C.-S. Shih, Y.-H. Wang, C.-M. Yang, and S.-H. Chao, "Elastic computation middleware for interactive wearable devices in cyber-physical systems," in *Proc. 3rd IEEE Int. Conf. Cyber Phys. Syst. Netw. Appl. (CPSNA)*, Hong Kong, Aug. 2015, pp. 1–6.

[13] W. Wang, Y. Jiang, and W. Wu, "Multiagent-based resource allocation for energy minimization in cloud computing systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 2, pp. 205–220, Feb. 2017.

[14] W. Fang, X. Yao, X. Zhao, J. Yin, and N. Xiong, "A stochastic control approach to maximize profit on service provisioning for mobile cloudlet platforms," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 4, pp. 522–534, Apr. 2018.

[15] K. Li, "Quantitative modeling and analytical calculation of elasticity in cloud computing," *IEEE Trans. Cloud Comput.*, to be published.

[16] A. Annaswamy, S. Chakraborty, D. Soudbakhsh, D. Goswami, and H. Voit, "The arbitrated networked control systems approach to designing cyber-physical systems," *IFAC Proc. Vol.*, vol. 45, no. 26, pp. 174–179, Sep. 2012.

[17] H. Gong, R. Li, J. An, W. Chen, and K. Li, "Scheduling algorithms of flat semi-dormant multi-controllers for a cyber-physical system," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 1665–1680, Aug. 2017.

[18] D. Soudbakhsh, L. T. X. Phan, O. Sokolsky, I. Lee, and A. Annaswamy, "Co-design of control and platform with dropped signals," in *Proc. 4th ACM/IEEE Int. Conf. Cyber Phys. Syst. (ICCPS)*, Philadelphia, PA, USA, Apr. 2013, pp. 129–140.

[19] M. F. Neuts, *Matrix Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Baltimore, MD, USA: Johns Hopkins Univ. Press, 1981.

[20] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Perth, WA, Australia, Nov. 1995, pp. 1942–1948.

[21] V. Gunes, S. Peter, T. Givargis, and F. Vahid, "A survey on concepts, applications, and challenges in cyber-physical systems," *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 12, pp. 4242–4268, Dec. 2014.

[22] B. Ravindran and P. Li, "DPR, LPR: Proactive resource allocation algorithms for asynchronous real-time distributed systems," *IEEE Trans. Comput.*, vol. 53, no. 2, pp. 201–216, Feb. 2004.

[23] P. Marti, C. Lin, S. A. Brandt, M. Velasco, and J. M. Fuertes, "Draco: Efficient resource management for resource-constrained control tasks," *IEEE Trans. Comput.*, vol. 58, no. 1, pp. 90–105, Jan. 2009.

[24] L. Tan, Z. Zhu, F. Ge, and N. Xiong, "Utility maximization resource allocation in wireless networks: Methods and algorithms," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 7, pp. 1018–1034, Jul. 2015.

[25] S. Samii, P. Eles, Z. Peng, P. Tabuada, and A. Cervin, "Dynamic scheduling and control-quality optimization of self-triggered control applications," in *Proc. 31st IEEE Real Time Syst. Symp. (RTSS)*, San Diego, CA, USA, Nov. 2010, pp. 95–104.

[26] C. Lozoya, P. Martí, M. Velasco, J. M. Fuertes, and E. X. Martin, "Resource and performance trade-offs in real-time embedded control systems," *Real Time Syst.*, vol. 49, no. 3, pp. 267–307, May 2013.

[27] T. Chantem, X. S. Hu, and M. D. Lemmon, "Generalized elastic scheduling for real-time tasks," *IEEE Trans. Comput.*, vol. 58, no. 4, pp. 480–495, Apr. 2009.

[28] G. Xie *et al.*, "Resource consumption cost minimization of reliable parallel applications on heterogeneous embedded systems," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 1629–1640, Aug. 2017.

[29] W. Chang and S. Chakraborty, "Resource-aware automotive control systems design: A cyber-physical systems approach," *Found. Trend® Electron. Design Autom.*, vol. 10, no. 4, pp. 249–369, Dec. 2016.

[30] C. Liu, L. Zhang, and D. Zhang, "Task scheduling in cyber-physical systems," in *Proc. 11th IEEE Int. Conf. Ubiquitous Intell. Comput. (UIC)*, Dec. 2014, pp. 319–324.

[31] M. Garcia-Valls, "A proposal for cost-effective server usage in CPS in the presence of dynamic client requests," in *Proc. 19th IEEE Int. Symp. Real Time Distrib. Comput. (ISORC)*, York, U.K., May 2016, pp. 19–26.

[32] J. Huang *et al.*, "Energy-efficient resource utilization for heterogeneous embedded computing systems," *IEEE Trans. Comput.*, vol. 66, no. 9, pp. 1518–1531, Sep. 2017.

[33] X. Zeng, Z. Liu, and Q. Hui, "Energy equipartition stabilization and cascading resilience optimization for geospatially distributed cyber-physical network systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 1, pp. 25–43, Jan. 2015.

[34] S. K. Khaitan and J. D. McCalley, "Design techniques and applications of cyber physical systems: A survey," *IEEE Syst. J.*, vol. 9, no. 2, pp. 350–365, Jun. 2015.

[35] L. Sheng, F. Yushun, and L. Huiping, "Dwelling time probability density distribution of instances in a workflow model," *Comput. Ind. Eng.*, vol. 57, no. 3, pp. 874–879, Oct. 2009.

[36] S. Liu and Y. Fan, "Workflow performance analysis and simulation based on multidimensional workflow net," *Comput. Ind.*, vol. 65, no. 2, pp. 333–344, Feb. 2014.

[37] E. D. Simmon *et al.*, "A vision of cyber-physical cloud computing for smart networked systems," NIST, Gaithersburg, MD, USA, Rep. NISTIR 7951, Aug. 2013.

[38] K. Gai, M. Qiu, H. Zhao, and X. Sun, "Resource management in sustainable cyber-physical systems using heterogeneous cloud computing," *IEEE Trans. Sustain. Comput.*, vol. 3, no. 2, pp. 60–72, Apr./Jun. 2018.

[39] L. Kuang and L. Zhang, "Level value density task scheduling algorithm for cyber physical systems on cloud," in *Proc. 14th IEEE Int. Conf. Ubiquitous Intell. Comput. (UIC)*, San Francisco, CA, USA, Aug. 2017, pp. 1–8.

[40] T. Z. J. Fu *et al.*, "DRS: Auto-scaling for real-time stream analytics," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3338–3352, Dec. 2017.

[41] R. Chaari *et al.*, "Cyber-physical systems clouds: A survey," *Comput. Netw.*, vol. 108, pp. 260–278, Oct. 2016.

[42] J. A. White, J. W. Schmidt, and G. K. Bennett, *Analysis of Queueing Systems*. New York, NY, USA: Acad. Press, 1975.

[43] P.-C. Guo, "A fast Newton–Shamanskii iteration for a matrix equation arising from M/G/1-type Markov chains," *Math. Problems Eng.*, vol. 2017, Oct. 2017, Art. no. 4018239.

Hongfang Gong received the B.S. degree in mathematics from the Changsha University of Science and Technology, Changsha, China, in 1991 and the M.E. degree in computer application from Hunan University, Changsha, in 2004, where he is currently pursuing the Ph.D. degree in computer science and technology with the Key Laboratory for Embedded and Network Computing of Hunan Province.

He is currently an Associate Professor of information science with the Changsha University of Science and Technology. His current research interests include cyber-physical systems, embedded computing system, and distributed control systems.

Renfa Li (M'05–SM'10) received the M.S. degree in electronic engineering from Tianjin University, Tianjin, China, in 1987 and the Ph.D. degree in electronic engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2002.

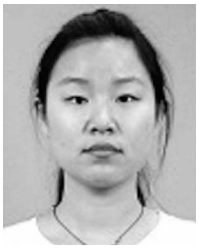
He is a Full Professor with the College of Computer Science and Electronic Engineering and the Dean of the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. He is the Director of the Key Laboratory for Embedded and Network Computing of Hunan Province, Changsha. He is also an Expert Committee Member with National Supercomputing Center, Changsha. His current research interests include computer architecture, embedded computing system, cyber-physical systems, and Internet of Things.

Mr. Li is a Senior Member of ACM and a member of the Council of China Computer Federation.

Jiyao An (M'12) received the M.Sc. degree in mathematics from Xiangtan University, Xiangtan, China, in 1998 and the Ph.D. degree in mechanical engineering from Hunan University, Changsha, China, in 2012.

He was a Visiting Scholar with the Department of Applied Mathematics, University of Waterloo, Waterloo, ON, Canada, from 2013 to 2014. In 2000, he joined the College of Computer Science and Electronic Engineering, Hunan University, where he is currently a Full Professor. He has published over 50 papers in international and domestic journals and refereed conference papers. His current research interests include cyber-physical systems, Takagi–Sugeno fuzzy systems, parallel and distributed computing, and computational intelligence.

Dr. An is a member of the ACM and a Senior Member of CCF. He is an active reviewer of international journals.



Yang Bai received the B.S. degree in communication engineering and M.S. degree in computer technology from Hunan University, Changsha, China, in 2013 and 2016, respectively, where she is currently pursuing the Ph.D. degree in computer science and technology with the Key Laboratory for Embedded and Network Computing of Hunan Province.

Her current research interests include service computing, embedded systems, and cyber-physical systems.

Keqin Li (M'90–SM'96–F'15) received the Ph.D. degree in computer science from the University of Houston, Houston, TX, USA, in 1990.

He is a Distinguished Professor of computer science with the State University of New York, New Paltz, NY, USA. He has published over 570 journal articles, book chapters, and refereed conference papers. His current research interests include parallel computing and high-performance computing, distributed computing, energy-efficient computing and communication, heterogeneous computing systems, cloud computing, big data computing, CPU–GPU hybrid and cooperative computing, multicore computing, storage and file systems, wireless communication networks, sensor networks, peer-to-peer file sharing systems, mobile computing, service computing, Internet of Things, and cyber-physical systems.

Mr. Li was a recipient of the several best paper awards. He is currently serving or has served on the editorial boards for the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON SERVICES COMPUTING, and IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING.