INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, e-ISSN 1841-9844, 14(5), 647-659, October 2019.

Exploring Analytical Models for Performability Evaluation of Virtualized Servers using Dynamic Resource

Y Kirsal

Yonal Kirsal*
Department of Electronics and Communication Engineering
European University of Lefke, Lefke, North Cyprus
TR-10 Mersin, Turkey
*Corresponding author:ykirsal@eul.edu.tr

Abstract: Virtualization of resources is a widely accepted technique to optimize resources in recent technologies. Virtualization allows users to execute their services on the same physical machine, keeping these services isolated from each other. This paper proposes the analytical models for performability evaluation of virtualized servers with dynamic resource utilization. The performance and avalability models are considered separately due to the behaviour of the proposed system. The well-known Markov Reward Model (MRM) is used for the solution of the analytical model considered together with an exact spectral expansion and product form solution. The dynamic resource utilization is employed to enhance the QoS of the proposed model which is another major issue in the performance characterization of virtulazilation. In this paper, the performability output parameters, such as mean queue length, mean response time and blocking probability are computed and presented for the proposed model. In addition, the performability results obtained from the analytical models are validated by the simulation (DES) results to show the accuracy and effectiveness of the proposed work. The results indicate the proposed modelling results show good agreement with DES and understand the factors are very important to improve the QoS.

Keywords: Analytical models, markov reward model, performability evaluation, virtulazilation, dynamic resource utilization.

1 Introduction

With the rapid growth of network applications, the management of resources and bandwidth in future system is becoming more complicated. Virtualization is one of the techniques that serve large number of tasks and multimedia applications with high demands by using a single physical machine [12]. Virtualization allows users to better utilize the resources that networks have available in a physical machine. The physical machine considered for virtualization purpose has a lot more resources than a personal computer such as a lot more CPU, RAM and the hard disk space. Thus, all of those resources can be emulated inside a software and with this software many virtual machines can be put on a single physical machine [18].

Virtualization allows service providers to lower their capital expenditures instead of buying many physical machines. In addition, centralized management with virtualization can be easily reachable instead of having to reach many individual physical machines providers only to manage one physical machine. In addition, all the updates or patches that the network to do for each of those individual machines can be done through from one central location. So, with the centralized management service providers have lower operational expenses because it takes less effort to manage these multiple machines [1]. However, putting multiple machines on one machine is not anything new. Currently, service providers have servers running these virtual machines for their subscribers in cloud computing environment and need additional computing resources for different server. Thus, they begin to add other virtual machines and these could be larger virtual machines depending on the business need [15].

Obtaining the best QoS requirements and improve the system performance of virtualized servers in cloud computing is a challenging task. Even though it is not a new topic, QoS modelling and evaluation of virtualized servers in such environment is still one of the key issues in order to obtain QoS measurements. Modelling and performance evaluation of such system in an analytical point of view help to service providers to predict the complex system behaviour. This might lead to improve the system QoS characterization in different aspects. Therefore, the main focus in this paper is to model and analyse the QoS of the proposed work considering the availability issues of a physical and virtual machines together with dynamic resource utilization. In other words, the physical and virtual machine failures and repairs are considered in analytical point of view. In addition, dynamic resource utilization is also employed to enhance the QoS of the proposed model which is another major issue in the performance characterization. Using the proposed model, important performability measures, such as mean queue length, mean response time and blocking probability can be computed. The well-known Markov Reward Model (MRM) is used for the solution of the analytical model considered together with an exact spectral expansion and product form solution. The rest of the paper is organized as follows: Section II gives motivation of the proposed model. Section III describes the proposed model and the solution approach. Numerical results and discussions are given in Section IV. Conclusions and future-work are provided in Section VI.

2 Motivation

With the widespread deployment and development of cloud computing facilities, the importance of virtualized cloud systems has significantly grown. The resource allocation is one of the important factors which affect the QoS of the virtualized servers [7]. The modeling of resource allocation and evaluation in cloud computing is considered in [2, 6, 13, 19, 21]. An analytic probabilistic model is presented in [2] to calculate profit in a virtualized cloud data center. The proposed model accurately calculate the arrival rates based on the external and internal requests for virtualized cloud data centers. In [13] the authors proposed a web application model via simulation to get the behavioral patterns of different users and an performance analysis is done for resource utilization in cloud computing. The energy-efficient scheduling of virtual machine resource reservations in the cloud data center is presented in [19] focusing on CPU applications. The main aim is to schedule all reservations non-preemptively considering the limitations of a physical machine capacities to minimize the total energy consumption. The authors in [21] proposed an effective evolutionary approach for virtual machine allocation that can maximize the energy efficiency of a cloud data center. This is done by designing a simplified simulation using CloudSim that speed up the process of the proposed work. However, those proposed models do not consider availability issues of both virtual and physical machines for dynamic resource utilization.

In addition, many existing studies mainly focus on performance analysis of virtual servers in cloud computing systems [5, 6, 10, 11, 17, 20]. Performance modelling and evaluation of the virtual servers in the cloud computing is considered in [5]. However, two virtual machines and a physical machine have been considered for performance evaluation which is not practical case. In [6] and [17] a performance management system is proposed based on an analytical queuing model on cloud. In those studies, the web applications are modeled as queues and virtual machines are modeled as service centers. Thus, the queuing theory models are applied to dynamically create and remove virtual machines in the cloud to enhance the system performance. The performance models for service migration have been addressed in [20] to predict the virtual machine migration time and the resources availability. However, the analysis done above studies considers only the pure performance model which ignore failure and recovery behaviour of the system. But in reality such systems are prone to failures due to hardware and software failures. An optimization method for the scheduling of scientific work flows on cloud systems is presented in [16]. A performability model is presented which provides the fitnesses of explored solutions by use of a meta-heuristic algorithm. In [9] a hierarchical model is employed for analysis of virtual machines failure and recovery with respect to the system behavior. In addition, the proposed model used in [9] for virtual machine mode was Continuous Time Markov Chain (CTMC) and a stochastic reward nets (SNR) is employed in [8] to model and analysis of the availability of a virtualized system.

In summary, however, the performability analysis of virtual machines with physical machine failure and recovery behaviour together with dynamic resource utilization have not been considered in none of the presented studies for an analytic probabilistic model. In this paper, the proposed model considers analytical models for performance and availability issues together with dynamic resource utilization in an attempt to understand, and improve system QoS. In addition, the analytical models results are validated by the discrete event simulation (DES) results to show the accuracy and effectiveness of the proposed work. The results indicate the proposed modelling results show good agreement with DES.

3 Proposed model and solution approach

In this section, the proposed analytical models and the solution approaches are presented to evaluate the QoS of the virtualized machines considering availability of physical as well as virtual machines together with dynamic resource utilization. In order to obtain realistic QoS output measurements dynamic resource utilization is also considered which can maximize the memory usage depending on the number of requests and failures within the proposed model. The main idea is to enhance the overall performance of the entire system by sharing the available resources. All the virtual machines used by the same physical machine are perfectly coordinated and synchronized. Hence, the tasks can be serviced more efficiently since the service ability will change dynamically when the virtual machine failures occur. The performance and availability models are considered separately due to the behavior of the proposed model. The exact spectral expansion solution approach is used to obtain steady state probabilities for the availability model, on the other hand, the steady state probabilities of the performance model are obtained by using the product form solution approach. Therefore, the MRM is employed to obtain performability output parameters for the proposed model. In MRM, the steady state probabilities of performance model are obtained (π_i^P) and are passed as reward rates to the availability model.

3.1 Model description

The handling of an incoming tasks for the proposed system with single physical machine (PM), a number of parallel homogeneous virtual machines (VMs) and a finite queue, L are illustrated in Fig. 1. Hence the total system capacity is K where K=L+VM. As shown in the figure, the arriving tasks reach first in a finite buffer and then distributed by the PM with a mean service time $1/\mu$ according to virtual machine availability. The PM will distribute the tasks for processing if and only if any of the VMs are available to handle a new task. Hence, each VM can also service and process the tasks in the system with a mean service time $1/\mu$. A VM can only serve one task at a time.

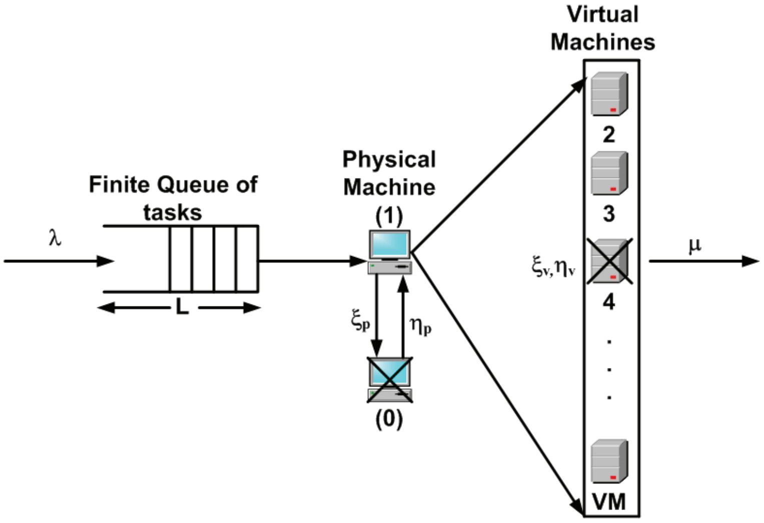


Figure 1: The proposed system

The resource allocations of such systems can be implemented in various ways depending on the system architecture. The dynamic resource allocation is presented which assures fair sharing of the CPU or the memory. The proposed allocation considers the avalability of the virtual machines and number of VMs in the system. It determines if the VM is busy or idle, if it is idle, it starts to transmit. If the VM is busy, the physical machine sends the tasks to VMs when they available. However, if the VM fails the shared CPU will not be wasted and can be dynamically used by the others available VMs until the failed VM will be repaired. Thus, the marginal distribution of the number of operative VMs is easily seen to be binomial [14]:

$$q_{i,.} = \sum_{j=0}^{\infty} q_{i,j} = \binom{VM}{i} \left(\frac{\eta_v}{\eta_v + \xi_v}\right)^i \left(\frac{\xi_v}{\eta_v + \xi_v}\right)^{VM-i} \quad where \quad i = 0, 1, \ldots, VM. \tag{1}$$

Where q_i is the probability that i, VM are operative. Hence, the processing capacity of the system, which is defined as the average number of operative virtual machines, is equal to E(I)=VM($\eta_n/\eta_n+\xi_n$). The η_n and ξ_n are recovery and failure rates of virtual machines in the proposed system. The failure and recovery behaviour of the proposed system is explained in section 3.2 in more detail. However, for a good QoS measurements in such system the availability of virtual machines and impact of on system performance should be considered. Therefore, in order to get more realistic results, dynamic resource utilization (effective service rate) for the proposed system is calculated considering all possible combinations of the number of operative VMs. Then the expected effective service rate is calculated as follows:

$$\mu_{eff} = \sum_{i=0}^{VM} q_i \cdot \mu_i \tag{2}$$

Equation 2 is used to calculate effective service rate of the proposed work when a VM failure occurs and used to optimize the resource allocation in the proposed work. In other words, Let's define T as a service time of a virtual machine and T_{eff} as dynamic service time of virtual machine with a failure with means μ and μ_{eff}, respectively. Hence, for dynamic resource utilization the service time of a task is equal to the smaller one between T and T_{eff}. Since the random variables

T and T_{eff} have exponential distribution, the service times of a task is exponentially distributed with mean;

$$E[T] = \frac{1}{\mu} = E[min(T, T_{eff})] = \frac{1}{\mu + \mu_{eff}} \tag{3}$$

As shown in Fig. 1, the tasks join the system with the Poisson distribution, hence the average arrival rate is λ [4, 10, 11]. On the other hand, the physical and virtual machines in the system are prone to failures, each VMs and PM get corrupted with an average rate of ξ_v and ξ_p, respectively. The failed VMs and a PM then repaired with an average repair rate of η_v and η_p, respectively. The repair priority is given to PM since the VMs can not operate when a PM failure occurs. If there are tasks in the queue, the operative VM cannot be in a pending state. However, when any VM fails due to the corruption, new tasks can be served by the first available VM. It is also assumed that if an operative VM fails, it becomes available again at the breakpoint. If all VMs are busy or failed, the queue is growing with average rate of λ. In order to obtain more realistic QoS output parameters both performance and availability issues are considered together with dynamic resource allocation. The performance and availability models considered are presented with the resulting MRM solution approach in detail following sections.

3.2 The availability model of the proposed system

The availability model shows possible VMs and a PM failures and repairs in the proposed system. An exact spectral expansion solution approach is used for availability model to obtain the steady state probabilities. The both failures and repairs behaviour are shown in Fig. 2. As mentioned before, the distribution of time intervals between VMs and a PM failures are exponential and given by mean $1/\xi_v$ and $1/\xi_p$, respectively. At the end of the VMs and a PM failures, the VMs and a PM require an exponentially distributed repair time with mean $1/\eta_v$ and $1/\eta_p$. As clearly seen from the Fig. 2 that multi-repairman facility is assumed for all of the VMs in order to get realistic QoS measurements.

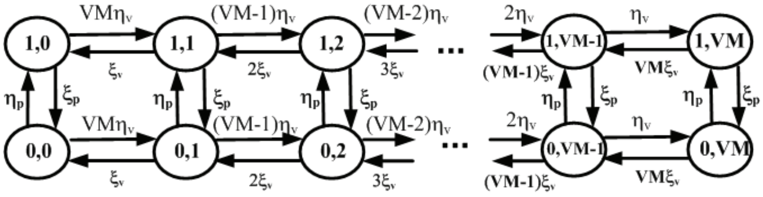


Figure 2: The availability model considered

Assume that $\pi^A_{i,j}$ represents the steady state probabilities of the availability model. Hence, the state of the availability model at time t can be described by a pair of integer valued random variables, $I(t)$ and $J(t)$, specifying the VMs and the PM failures and repairs, respectively. As clearly seen from the Fig. 2, $J(t)$ describes the PM failures and has two modes. Let $J(t)$=0,1 denote a binary value indicating whether or not the system is down due to a PM failure. In other words, *0* indicates the whole system is down due to a PM failure and *1* indicates all VMs are operate. On the other hand, $I(t)$ represents the operative states of the VMs and there are $VM + 1$ configurations. Thus, $I(t) = 0,\ 1,\ \cdots,\ VM$. Hence, $Z = [I(t), J(t)]$; $t \geq 0$ is an irreducible Markov process on a lattice strip that models the system. Its state space is (0, 1, $\cdots$, $VM + 1$) x (0, 1). In other words, the failure and repair behaviour of the VMs can be represented in the horizontal as well as the failure and recovery behaviour of the PM in the vertial direction of a lattice strip. The three matrixes of the spectral expansion can be defined when the state diagram is obtained. Hence, A, B, and C are the matrixes that indicate of the transition rates of the proposed model. The definition and the formation of the matrixes can be found in [3, 11]. Therefore, clearly, the elements of A,B and C depend on the parameters VM, ξ_v, η_v, η_p,ξ_p. The transition matrices of a system with VM machines are of size $(VM+1)\times(VM+1)$. It is possible to specify the numbering of the matrices as (0,1,2,$\cdots$,$VM+1$) for states (0,1,2,$\cdots$,$VM+1$) respectively. The state transition matrices A, A_j, B, B_j, C, and C_j can be given as follows:

$$A = A_j = \begin{pmatrix} 0 & VM\eta_v & 0 & 0 & 0 & 0 & 0 \\ \xi_v & 0 & (VM-1)\eta_v & 0 & 0 & 0 & 0 \\ 0 & 2\xi_v & 0 & (VM-2)\eta_v & 0 & 0 & 0 \\ 0 & 0 & 3\xi_v & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 2\eta_v & 0 \\ 0 & 0 & 0 & 0 & (VM-2)\xi_v & 0 & \eta_v \\ 0 & 0 & 0 & 0 & 0 & (VM-1)\xi_v & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & VM\xi_v \end{pmatrix}$$

$$B = B_j = \begin{pmatrix} \eta_p & 0 & 0 & 0 & 0 \\ 0 & \eta_p & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & \eta_p & 0 \\ 0 & 0 & 0 & 0 & \eta_p \end{pmatrix}$$

$$C = C_j = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \xi_p & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \xi_p & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \xi_p & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \xi_p & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Therefore, the proposed system can be solved using the well-known exact spectral expansion solution method. Thus, the steady state probabilities for the availability model, $\pi^A_{i,j}$ can be obtained using the steady state solution. The solution is given for systems with bounded queuing capacities. The steady-state probabilities of the system considered can be expressed as:

$$p_{i,j} = lim_{t\to\infty}^{P(I(t)=i,J(t)=j)}; \quad 0 \le i \le VM+1, \quad 0 \le j \le 1 \tag{4}$$

Let's define certain diagonal matrices of size $(VM+1)\times(VM+1)$ as follows:

$$D^A_j(i,i) = \sum_{k=0}^{VM+1} A_j(i,k); \quad D^A(i,i) = \sum_{k=0}^{VM+1} A(i,k); \tag{5}$$

$$D^B_j(i,i) = \sum_{k=0}^{VM+1} B_j(i,k); \quad D^B(i,i) = \sum_{k=0}^{VM+1} B(i,k); \tag{6}$$

$$D^C_j(i,i) = \sum_{k=0}^{VM+1} C_j(i,k); \quad D^C(i,i) = \sum_{k=0}^{VM+1} C(i,k); \tag{7}$$

and $Q_0 = B$, $Q_1 = A - D^A - D^B - D^C$, $Q_2 = C$. Hence, all state probabilities in a row can be defined as:

$$v_j = (p_{0,j}, p_{1,j}, \cdots, p_{VM+1,j}); j = 0, 1 \tag{8}$$

The steady-state balance equations for bounded queuing systems ($0 \leq j \leq 1$) can now be written as follows:

$$v_0[D_0^A + D_0^B] = v_0 A_0 + v_1 C_1 \tag{9}$$

$$v_j[D_j^A + D_j^B + D_j^C] = v_{j-1}B_{j-1} + v_j A_j + v_{j+1}C_{j+1}; \quad 0 \leq j \leq 1 \tag{10}$$

$$v_L[D^A + D^C] = v_{L-1}B + v_L A \tag{11}$$

The normalizing equation is given as follows:

$$\sum_{j=0}^{L} v_j e = \sum_{j=0}^{1} \sum_{i=0}^{VM+1} P(i,j) = 1.0 \tag{12}$$

From the equations above, the following equation can be written:

$$v_j Q_0 + v_{j+1} Q_1 + v_{j+2} Q_2 = 0; \quad 0 \leq j \leq 1 \tag{13}$$

Furthermore, the characteristic matrix polynomial $Q(\lambda)$ can be defined as:

$$Q_\lambda = Q_0 + Q_1\lambda + Q_2\lambda^2; \quad \bar{Q}_\beta = Q_2 + Q_1\beta + Q_0\beta^2; \tag{14}$$

where

$$\Psi Q_\lambda = 0; \quad |Q_\lambda| = 0; \quad \phi\bar{Q}_\beta = 0; |\bar{Q}_\beta| = 0; \tag{15}$$

λ and Ψ are eigenvalues and left-eigenvectors of Q_λ and β and ϕ are eigenvalues and left-eigenvectors of $\bar{Q}_\beta$, respectively. Note that, ϕ and β are vectors defined as:

$$\phi = \phi_0, \phi_1, \ldots, \phi_{S+1} \tag{16}$$

$$\beta = \beta_0, \beta_1, \ldots, \beta_{S+1} \tag{17}$$

Furthermore, $v_j = \sum_{k=0}^{VM+1} (a_k \Psi_k \lambda_k^{j+1} + b_k \phi_k(i) \beta_k^{2-j})$,0≤j≤ 1 and in the state probability form,

$$p_{i,j} = \sum_{k=0}^{VM+1} (a_k \Psi_k \lambda_k^{j-M+1} + b_k \phi_k(i) \beta_k^{2-j}) \quad 0 \leq j \leq 1 \tag{18}$$

Where $\lambda_k (k = 0, 1, \ldots, VM + 1)$ and $\beta_k (k = 0, 1, \ldots, VM + 1)$ are $VM + 1$ eigenvalues [3,4,11]. The more details of the exact spectral expansion method can be found in [3]. Therefore, all steady state probabilities of the availability model, $\pi_{i,j}^A$ can be obtained using the exact spectral expansion method.

3.3 The performance model of the system

The performance model considers the arrival and service rates transitions of the single PM and multiple VMs in this section. The state transition diagram of performance model of a proposed system is given in Fig. 3.

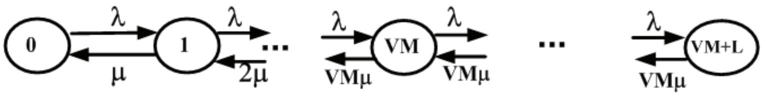


Figure 3: The performance model considered

Let's define the states i (i=0,1,2,$\cdots$, VM+L) as the number of tasks in the system at time t. In order to obtain perofrmability measures of the proposed model, the performance model is solved using general product form solution technique to get steady state probabilities. Once all π_i^P obtained, they pass as reward rates to the availability model. ρ is the traffic intensity in the system where ρ=λ/μ. Hence, the state probabilities, π_i^P can be obtained and are given in equation 19 [9, 10].

$$\pi_i^P = \begin{cases} \frac{\rho^i}{i!} \cdot \pi_0^P & 0 \leq i \leq VM \\ \frac{\rho^i}{VM^{i-VM}.VM!} \cdot \pi_0^P & VM < i \leq VM+L \end{cases} \tag{19}$$

In equation 19, π_i^P is the probability that there are i tasks in the proposed system and π_0^P can be defined as follows:

$$\pi_0^P = \left[\sum_{i=0}^{VM-1} \frac{\rho^i}{i!} + \sum_{i=VM}^{VM+L} \frac{\rho^i}{VM^{i-VM}.VM!} \right]^{-1} \tag{20}$$

Hence, the average number of tasks in the proposed system, N_{VM} can then be calculated as $N_{VM} = \sum_{i=0}^{VM+L} i \cdot \pi_i^P$ which gives:

$$N_{VM} = \left[\sum_{i=0}^{VM} i \cdot \frac{\rho^i}{i!} + \sum_{i=VM}^{VM+L} i \cdot \frac{\rho^i}{VM^{i-VM}.VM!} \right] \cdot \pi_0^P \tag{21}$$

Similarly, the blocking probability P_B of the system can be calculated as:

$$P_B = P(VM+L) = \frac{\lambda^{VM+K}}{VM^L VM! \mu^{VM+L}} \cdot \pi_0^P \tag{22}$$

Using Little's formula, the average value of time of a task in a the system is can be calculated as follows:

$$MRT = \frac{MQL}{(1-P_B)\lambda} \tag{23}$$

Therefore, a MRM approach can be used to obtain the overall performability output parameters such as mean queue length, blocking probability and mean response time. Equation 21 gives the MQL assuming that all VMs are operative. However, since only i VMs are operative at any time, the MQL can now be represented by N_i where i is the number of operative VM. Thus overall MQL can be calculated as follows;

$$MQL = \sum_{i=0}^{K} N_i \sum_{j=0}^{1} \pi_{i,j}^A \tag{24}$$

Similarly, the blocking probability, P_B and MRT of the proposed system can be evaluated as follow:

$$P_B = \sum_{i=0}^{K} P_{B,i} \sum_{j=0}^{1} \pi_{i,j}^{A} \tag{25}$$

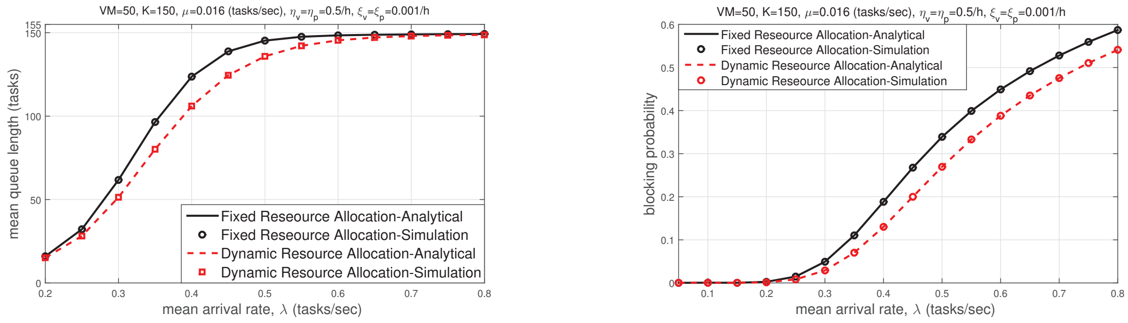
$$MRT = \sum_{i=0}^{K} MRT_i \sum_{j=0}^{1} \pi_{i,j}^{A} \tag{26}$$

4 Results and discussion

In this section, numerical results are presented to understand the beahviour of the system and show the affect of the availability issues for virtualized servers with dynamic resource utilization. The performance and availability models are considered separately and a MRM is employed for steady state solution. As stated before, the results obtained from the analytical model and the simulation for each analysis are presented in order to validate as well as to show accuracy of the proposed analytical model. Numerical results are presented here for performability measures of virtualized servers with dynamic resource utilization. The parameters used are mainly taken from the literature in order to be consistent [2, 4–6, 8–13, 17, 19, 20]. The mean service rate is mainly application dependent.

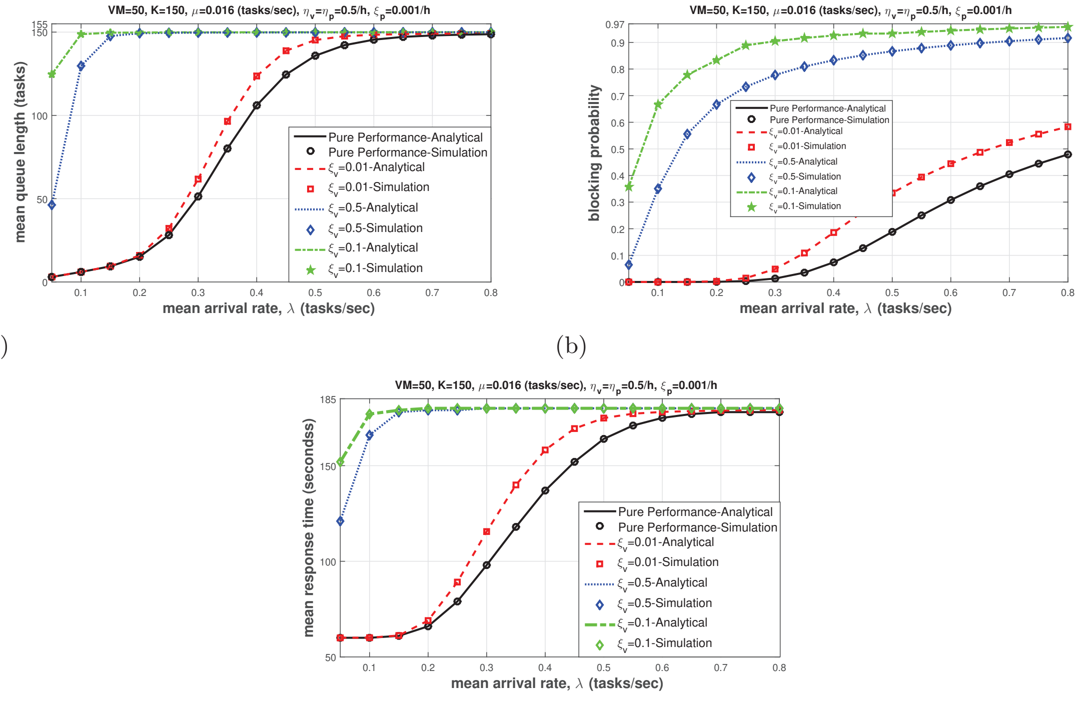
In Fig. 4 QoS results are presented as a function of the arrival rate for the proposed system with VM=50 and L=100 where, K=VM+L=150. The other parameters are $\mu = 0.016(tasks/sec)$, η_v=$\eta_p = 0.5/h$,ξ_v=$\xi_p = 0.001/h$ and mean arrival rate per tasks varies from 0.05 tasks per second. QoS output parameters, mean queue length and blocking probability, have been computed for both fixed resource allocation (FRA) and dynamic resource allocation (DRA) in Fig. 4(a) and 4(b), respectively in order to show efficiency of the dynamic resource allocation in the proposed system. It can be clearly seen that dynamic resource allocation gives more promising performance results compared to fixed resource allocation scheme in terms of mean queue length and blocking probability. Fig. 4 shows that as the rate of the incoming tasks increases, mean queue length and blocking probability results are also increases for both reservation schemes. However, when the virtual as well as the physical machine failures are considered, the system with dynamic resource allocation policy performs better than fixed resource allocation especially for the loaded system. This is due to the service utilization continues to increase by dynamic resource allocation. Thus, the tasks can be serviced since the service ability will change dynamically when the virtual machine failures occur. On the other hand, all virtual machines have an equal share of resources in FRA and some of the resources will be wasted due to failures. As the rate of the incoming requests increases this difference becomes less evident since the queuing capacity, K=150 becomes the main limiting factor. Hence, please note that DRA is used for the rest of the analysis since it gives better QoS output results compare to FRA when the both failures are considered.

In Figs. 5 and 6, QoS output measurements are presented as a function of mean arrival rate for the proposed model with different virtual machines and a physical machine failure rates, respectively. The mean queue length, blocking probability and mean response time results are shown in Fig. 5(a), 5(b) and 5(c), respectively with different virtual machines failures. The parameters are VM=50, K=150, $\mu = 0.016(tasks/sec)$, η_v=$\eta_p = 0.5/h$, $\xi_p = 0.001/h$ and mean arrival rate per tasks varies from 0.05 tasks per second. As clearly seen that even though the dynamic resource utilization is used, the failures of the virtual machines significantly effect the system performance. In Fig. 5 the pure perforamnce results gives best QoS output measurements



(a) (b)

Figure 4: QoS results of the proposed model with fixed resource allocation and dynamic resource allocation: (a) Mean queue length; (b) Blocking probability



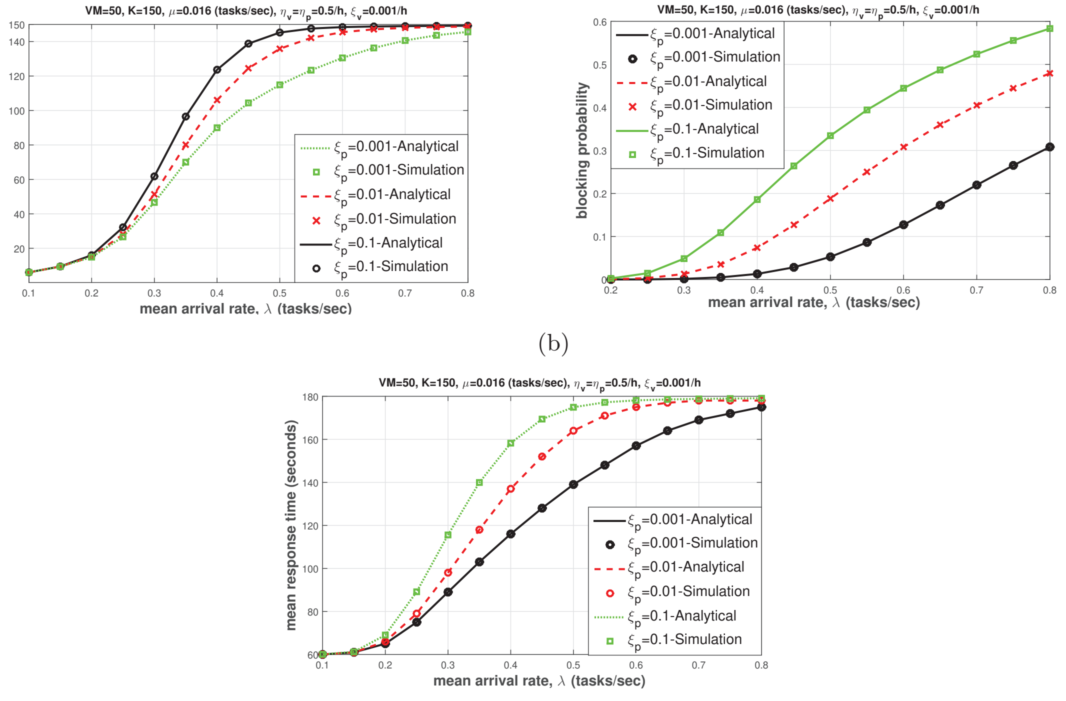
(a) (b)

(c)

Figure 5: QoS results of the proposed model with different failure rates: (a) Mean queue length; (b) Blocking probability; (c) Mean response time

for each analysis since the system never fails. However, the results show in Fig. 5 that as virtual machine failure rate increases the QoS degradation becomes more evident. For instance, for $\xi_v = 0.1/h$ and $\xi_v = 0.5/h$ the all QoS values dramatically increases. In other words, the tasks will be piled up in the system quickly as shown in Fig. 5(a) when the virtual machine failure increases. The system will not serve the tasks and the system reaches the maximum capacity quickly (i.e, K=150). In Fig. 5(b), the system started to block incoming tasks due to frequent virtual machine failures hence the blocking probability increases as the virtual machine failure increases. Similarly, the system will not able to serve the tasks due to virtual machine failures hence the mean response time of the proposed system also increases as shown in Fig. 5(c).

On the other hand, in Fig. 6 the affect of the physical machine failures are shown. Similarly



(a) (b)

(c)

Figure 6: QoS results of the proposed model with physical machine failure rates: (a) Mean queue length; (b) Blocking probability; (c) Mean response time

to the Fig. 5, the mean queue length, the blocking probability and the mean response time results are presented in Fig. 6 with a physical machine failure and repair. The parameters used are the same as used in Fig. 5. However, in this analysis the virtual machine failures are kept constant, $\xi_v = 0.001/h$ and a physical machine failure varies from 0.001 to 0.1. As clearly seen from the Fig. 6, the system QoS degrades with a physical machine failure in terms of mean queue length, blocking probability and mean response time in Figs. 6(a), 6(b) and 6(c), respectively. When a physical machine fails all of the virtual machines do not operate, hence the failure of the physical machine limits the access to the virtual machines. As the failure rate of a physical machine increases all of the performability measures increases and the QoS is getting worst. In other words, less physical machine failure, the better the system performance gets. In the case of a physical machine fails, it will cause of blocking the incoming tasks as shown in Fig. 6(b), increase of the tasks in the queue as well as late responses from the system as shown in Fig. 6(a) and 6(c), respectively.

Therefore, it is very important to understand the failure and recovery of both VM and PM as well as dynamic resource utilization so that such practical systems could be incorporated improved in terms of QoS for such systems.

5 Conclusions

In this paper, the analytical models have been modelled for performance and availability issues together with dynamic resource utilization in order to understand and improve system QoS parameters. The behaviour of failure and recovery of the virtual machines and a physical machine are considered as an availability model and the exact spectral expansion solution approach is used to obtain steady state probabilities. The steady state probabilities of the performance model are obtained by using the product form solution approach. Therefore, the MRM is employed to obtain performability output parameters for the proposed model. In addition, dynamic resource utilization is also employed to enhance the QoS of the proposed model.

The main focus in the analysis is given to performability output parameters such as mean queue length, blocking probability and mean response time. Numerical results obtained clearly show that, the virtual machine and especially a physical machine failures affect the system QoS significantly. The failures of a physical machine cause more significant performance degradation. It can be clearly seen from the numerical results that dynamic resource allocation gives more promising performance results compared to fixed resource allocation. The QoS results obtained from the analytical model are compared to the DES results in order to show the accuracy and effectiveness of the proposed models. Findings show that the analytical models and a MRM technique presented show good agreement with DES. The models presented in this paper are flexible and useful for modelling systems with similar behaviour and queuing considerations.

Bibliography

[1] Borangiu, T.; Trentesaux, D.; Thomas, A.; Leitao, P.; Barata, J. (2019). Digital transformation of manufacturing through cloud services and resource virtualization, *Computers in Industry*, 108, 150-162, 2019.

[2] Bi, J.; Yuan, H.; Tan, W.; Zhou, M.C.; Fan, Y.; Zhang, J.; Li, J.G. (2017). Application-Aware Dynamic Fine-Grained Resource Provisioning in a Virtualized Cloud Data Center, *IEEE Transactions on Automation Science and Engineering*, 14(2), 1172–1184, 2017.

[3] Chakka, R. (1995). *Performance and reliability modelling of computing systems using spectral expansion*, Ph.D. thesis, University of Newcastle, Upon Tyne, UK, 1995.

[4] Ever, Y.K.; Kirsal, Y.; Ever, E.; Gemikonakli, O. (2015). Analytical modelling and performability evaluation of multi channel WLANs with global failures. *International Journal of Computers Communications & Control*, 10(10), 551–566, 2015.

[5] Gemikonakli, O.; Ever, E.; Gemikonakli, E. (2009). Performance modelling of virtualized servers. *International Conference on Computer Modelling and Simulation*, 434–438, 2009.

[6] Goswami, V.; Patra, S.S.; Mund, G.B. (2012). Performance analysis of cloud with queue dependent virtual machines. *International Conference on Recent Advances in Information Technology (RAIT)*, 357–362, 2012.

[7] Iyer, R.; Illikkal, R.; Tickoo, O.; Zhao, L.; Apparao, P.; Newell, D. (2009). VM3: Measuring, modeling and managing VM shared resources. *Computer Networks*, 53(17), 2873–2887, 2009.

[8] Kim, D. S.; Hong, J. B.; Nguyen, T. A.; Machida, F.; Park, J. S.; Trivedi, K. S. (2016). Availability modeling and analysis of a virtualized system using stochastic reward nets. *In IEEE International Conference on Computer and Information Technology (CIT)*, 210–218, 2016.

[9] Kim, D. S.; Machida, F.; Trivedi, K. S. (2009). Availability modeling and analysis of a virtualized system. *In IEEE Pacific Rim International Symposium on Dependable Computing*,365–371, 2009.

[10] Kirsal, Y. (2016). Analytical modelling of a new handover algorithm for improve allocation of resources in highly mobile environments. *International Journal of Computers Communications & Control*, 11(6), 789–803, 2016.

[11] Kirsal, Y.; Paranthaman, V. V.; Mapp, G. (2018). Exploring Analytical Models for Proactive Resource Management in Highly Mobile Environments. *International Journal of Computers Communications & Control*, 13(5), 837–852, 2018.

[12] Liu, N.; Li, X.; Wang, Q. (2011). A resource and capability virtualization method for cloud manufacturing systems, *IEEE Int. Conf. on Systems, Man, and Cybernetics*, 1003–1008, 2011.

[13] Magalhaes, D.; Calheiros, R. N.; Buyya, R.; Gomes, D. G. (2015). Workload modeling for resource usage analysis and simulation in cloud computing, *Computers and Electrical Engineering*, 47, 69–81, 2015.

[14] Mitrani. I. (2001). *Queues with Breakdowns, Performability Modelling:Techniques and Tools*, Wiley, Chichester, 2001.

[15] Odun-Ayo, I.; Ajayi, O.; Falade, A. (2018). Cloud Computing and Quality of Service: Issues and Developments,*In International Multi-Conference of Engineers and Computer Scientists*, 2018.

[16] Oliveira, D.; Brinkmann, A.; Rosa, N.; Maciel, P. (2019). Performability evaluation and optimization of workflow applications in cloud environments, *Journal of Grid Computing*, 1–22, 2019.

[17] Peng, C. H.; Chong, L.S. (2010). A queueing-based model for performance management on cloud. *International Conference on Advanced Information Management and Service (IMS)*, 83–88, 2010.

[18] Sotomayor, B.; Montero, R.S.; Llorente, I.M. (2009). Virtual infrastructure management in private and hybrid clouds, *IEEE Internet Comput.*, 13(5), 14–22, 2009.

[19] Tian, W.; He, M.; Guo, W.; Huang, W.; Shi, X.; Shang, M.; Buyya, R. (2018). On minimizing total energy consumption in the scheduling of virtual machine reservations. *Journal of Network and Computer Applications*, 113, 64–74, 2018.

[20] Wu, Y.; Zhao, M. (2011). Performance modeling of virtual machine live migration. *In IEEE 4th International Conference on Cloud Computing*, 492–499, 2011.

[21] Zhang, X.; Wu, T.; Chen, M.; Wei, T.; Zhou, J.; Hu, S.; Buyya, R. (2019). Energy-aware virtual machine allocation for cloud with resource reservation. *Journal of Systems and Software*, 147, 147–161, 2019.