Modeling a Resource Contention in the Management of Virtual Organizations

Tien Van Do

Department of Telecommunications,
Budapest University of Technology and Economics,
H-1117, Magyar tudósok körútja 2., Budapest, Hungary.
Tel: +36 1 463 2070

Abstract

A virtual organization provides a cost-efficient method allowing different autonomous entities, such as organizations, departments and individuals, to extend service offerings in a virtual marketplace. To support cost-efficient service provisioning, a suitable procedure must be applied to determine the amount of resources necessary for the operation of virtual organizations.

We propose a new mathematical model for a quantitative performance evaluation of resource management in virtual organizations. We present an efficient algorithm to determine the steady state probabilities and the performance measures of the system. A comparison with a detailed simulation model and other numerical approaches shows that the proposed algorithm is fast and accurate. This algorithm can therefore be used for resource dimensioning to support the cost-efficient operation of virtual organizations.

Keywords: virtual organization, resource management, resource contention, queueing model, performance model, fast computational algorithm.

Email address: do@hit.bme.hu (Tien Van Do).

Preprint submitted to Elsevier

1 Introduction

A virtual organization (VO) provides a cost-efficient method for different autonomous entities, such as organizations, departments and individuals, to share resources in a virtual marketplace [1–12]. This type of coalition offers a wider range of services in a more cost- efficient and competitive way than the classical service provisions that come from separate organizations without cooperation.

A number of factors, technologies and parameters (e.g., independence of autonomous organizations, adaptability, resilience, cost, quality of service, and information) are considered in the automated formation and management of VOs. Moreover, highly dynamic situations concerning the behavior of autonomous entities and customers are anticipated. Therefore, knowledge management technologies [13–20], such as collaborative technologies, extensible markup language, personal devices, wireless and fix network technology, virtual reality, portals, and multi-agents, should be applied to support the operation of VOs.

Intensive research has been conducted concerning the automated management of VOs with the use of agents [1–4, 12]. The key task of resource management (in terms of capacity versus the demands of customers) is to support the cost-efficient service provisioning of VOs. To accomplish this goal, a suitable procedure must be applied to determine the amount of resources necessary to fulfill the requirements of customers. However, the literature has not dealt with the issue of resource planning for VOs until now.

This paper presents a novel mathematical model that can be used to quantitatively evaluate the performance of resource management for service provisioning in VOs. To the best of our knowledge, this is the first attempt at modeling the dynamic behaviors that concern the interactions of customers and the resource management function of a specific VO. Since an enormous number of resource units (often of the order of several thousands or more) can be present in a single VO environment, computation can be extremely difficult because of the state explosion problem. Until now, there have been no numerically stable exact or approximate analytical solutions to this problem. We present an efficient algorithm to compute the steady state probabilities of a continuous-time discrete-state Markov (CTMC) process associated with the proposed retrial queue, and we evaluate the performance measures. We compare the results obtained by the proposed analytical model with a more detailed simulation. The comparison confirms that the proposed model can accurately estimate the amount of resources necessary to satisfy the demands of customers. A comparison with existing analytical approaches shows that the proposed algorithm is fast enough to compute accurate results. It can therefore be used for online dimensioning purposes to support the operation of VOs.

The rest of the paper is organized as follows: in Section 2, a problem description is provided; in Section 3, an approximated model is presented, and a computational algorithm is developed; in Section 4, a comparison with existing approaches and a simulation is provided to show the applicability of the proposed model; the paper is concluded in Section 5.

2 Problem Description and the Proposed Model

The concept of VOs has been introduced and discussed in several works in the last decade [21–24]. The main aim of VOs is to enhance the collaboration for the offering of services between organizations. Advances in technology (networking technologies such as the Internet, devices, and agents) have had a great impact on the application of the VO concept in different areas.

It has been pointed out in several works [1–12, 21–24] that the success of a VO largely depends on the commitment, the mutual trust and the flexibility of the participating entities. These factors should be supported by a number of knowledge management technologies [13, 17, 18] to achieve the goals of the VO.

Pioneering works [1,3,9,10] have provided proof of the concept, contributing to the framework essential to realizing VOs. Norman et al. [1] argued that agents should play an active role in VO formation. Agents autonomously and automatically make suitable decisions concerning the acquisition of information, the representation of services, the announcement of commitments regarding services, the participation in auctions and service delivery. Norman et al. [1] suggested the use of a special entity responsible for managing resources to satisfy the demands of customers. This entity is called a requirement agent (RA) in the CONOISE framework [1].

The requirement agent acts as an interface between customers and service providers. It receives information on services advertised by service providers through a yellow pages agent. The RA initiates the VO formation through the submission of a call for bids with a specific deadline to fulfill the requirements previously collected from customers. Agents representing service providers may submit their offer with certain commitments and further conditions to the RA. After the expiration of the deadline, the RA decides the rejection of some offers from a set of bidding service providers and selects suitable service providers to establish the VO.

The RA's decision is based on a number of factors, such as a specific amount and time availability of a resource offered by each service provider, the cost, the requirements of customers, the expected rate of service requests from customers and the Quality of Service (QoS). It is worth emphasizing that the necessary amount of resources in the VO is the result of a trade-off between cost-efficiency (e.g., how resources are utilized) and the QoS expectation of customers.

Norman et al. [1] focused on the problem of a combinatorial auction approach. The authors [1, 25] defined a mechanism (a compact and expressive bid representation language and efficient clearing algorithms) for combinatorial auctions. In [3], open semantic Web representations for expressing individual commitments as constraints over service descriptions, a set of commitments as a soft constraint satisfaction problem and a solution were provided. To support the cost-efficient service provisioning of VOs, a suitable procedure must be applied to determine the necessary resources for the operation of VOs.

In this paper, we model an interaction between the service requests of customers and the resource management function of the specific VO as follows. Let integer number c denote the normalized maximum capacity of the VO. Each customer will request the usage of one capacity unit from the VO. Note that this assumption does not limit the applicability of our model. For example, if we want to determine the necessary capacity for a multiservice scenario [1], we shall apply our approach separately for each type of service. The traffic load offered to the VO is characterized by the arrival of service requests. The interarrival times of service requests are exponentially distributed with a mean interarrival time of $1/\lambda$. Holding times are exponentially distributed with a mean value of $1/\mu$. Let $1 - \alpha$ denote the probability that clients will renew the use of a service unit after the expiration of its holding time. In a renewal case, a client will request a new usage time, which is exponentially distributed with a mean value of $1/\mu$.

Random variable $I(t)$ represents the number of occupied capacity units at time t. Therefore, $0 \leq I(t) \leq c$ holds. A client who does not receive the allocation because of the temporary shortage (when $I(t) = c$) of resources sets a timer to wait for a fixed time T_r and will retry the request upon the expiration of the timer. We model this phenomenon as the client joins the "virtual orbit". Random variable $J(t)$ $(0 \leq J(t))$ denotes the number of clients in the orbit at time t.

The resource management problem can be solved with the application of a multiserver retrial queue. The evolution of the system is modeled by a stochastic process $\{I(t), J(t)\}$. However, there is no mathematically tractable solution

[1] where a complex service package of different types of services (e.g., movies, news, text messaging and phone) is offered by a specific VO [1].

for the retrial queuing model with multi-servers if the number of servers is larger than two and the retrial rate depends on the number of waiting clients in the orbit [26–28]. Constant retrial times will further contribute to the mathematical intractability of the model presented above.

In what follows, we shall show that we can compute the utilization of resources and the quantitative QoS measures (e.g., the probability that there is a temporary shortage capacity in the VO and the average number of waiting service requests) in a highly efficient way.

3 An Approximate Model and a Computational Algorithm

3.1 An Approximate Model

To have a mathematically tractable solution, we assume that the retrial rate is independent of the number of waiting clients in the orbit. That is, the retrial rate of clients waiting in the orbit is ν (i.e., the inter-repetition times are exponentially distributed with parameter $\nu = 1/T_r$). Therefore, we arrive at an approximate model that can be described by a continuous-time Markov chain $Y = \{I(t), J(t)\}$ with state space $\{0, 1, \ldots, c\} \times \{0, 1, \ldots\}$. We denote the steady state probabilities by $\pi_{i,j} = \lim_{t\to\infty} \Pr(I(t) = i, J(t) = j)$, and we introduce $\mathbf{v}_j = (\pi_{0,j}, \ldots, \pi_{c,j})$.

The evolution of CTMC Y is driven by the following dynamics.

(a) $A_j(i,k)$ denotes a transition rate from state (i,j) to state (k,j) $(0 \leq i,k \leq c; j = 0,1,\ldots)$, which is caused by the arrival of customer requests. Matrix A_j is defined as the matrix with elements $A_j(i,k)$. Because A_j is j-independent, it can be written as

$$A_j = A = \begin{bmatrix} 0 & \lambda \; 0 \ldots 0 & 0 & 0 \\ \alpha\mu & 0 \; \lambda \ldots 0 & 0 & 0 \\ \vdots & \vdots \; \vdots \; \vdots \;\; \vdots & \vdots & \vdots \\ 0 & 0 \quad \ldots \alpha(c-1)\mu & 0 & \lambda \\ 0 & 0 \quad \ldots 0 & \alpha c\mu & 0 \end{bmatrix} \quad \forall j \geq 0\,.$$

(b) $B_j(i,k)$ represents one step upward from state (i,j) to state $(k, j+1)$ $(0 \leq i,k \leq c; j = 0,1,\ldots)$. This transition is due to the arrival of requests when no free resources are available. In a similar way, matrix B_j (B) with elements $B_j(i,k)$ is defined as

$$B_j = B = \begin{bmatrix} 0\ 0\ 0 \ldots 0\ 0\ 0 \\ 0\ 0\ 0 \ldots 0\ 0\ 0 \\ \vdots\ \vdots\ \vdots\ \vdots\ \ \vdots\ \vdots\ \vdots \\ 0\ 0\ \quad \ldots 0\ 0\ 0 \\ 0\ 0\ \quad \ldots 0\ 0\ \lambda \end{bmatrix} \quad \forall j \geq 0\ .$$

(c) $C_j(i,k)$ is the transition rate from state (i,j) to state $(k,j-1)$ $(0 \leq i,k \leq c; j = 1,\ldots)$, which is due to the successful retrial of a request from the orbit. Matrix C_j $(\forall j \geq 1)$ with elements $C_j(i,k)$ is written as

$$C_j = C = \begin{bmatrix} 0\ \nu\ 0 \ldots 0\ 0\ 0 \\ 0\ 0\ \nu \ldots 0\ 0\ 0 \\ \vdots\ \vdots\ \vdots\ \vdots\ \ \vdots\ \vdots\ \vdots \\ 0\ 0\ \quad \ldots 0\ 0\ \nu \\ 0\ 0\ \quad \ldots 0\ 0\ 0 \end{bmatrix} \quad \forall j \geq 1.$$

D^A and D^C are diagonal matrices whose diagonal elements are the sum of the elements in the corresponding row of matrices A and C, respectively. The infinitesimal generator matrix of Y can be written as follows:

$$\begin{bmatrix} A_{00} & \text{B} & 0 & \ldots & \ldots & \ldots & \ldots \\ \text{C} & Q_1 & \text{B} & 0 & \ldots & \ldots & \ldots \\ 0 & \text{C} & Q_1 & \text{B} & 0 & \ldots & \ldots \\ 0 & 0 & \text{C} & Q_1 & \text{B} & 0 & \ldots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \end{bmatrix}. \tag{1}$$

The balance equations are written as follows:

$$\mathbf{v}_0 A_{00} + \mathbf{v}_1 C = 0, \tag{2}$$

$$\mathbf{v}_{j-1} B + \mathbf{v}_j Q_1 + \mathbf{v}_{j+1} C = 0 \ \ (j \geq 1), \tag{3}$$

where $A_{00} = A - D^A - B$ and $Q_1 = A - D^A - B - D^C$.

One can observe that $Y = \{I(t), J(t)\}$ is a Quasi Birth-Death (QBD) process. There are two main methods for finding the steady state probabilities for QBD processes on semi-infinite strips. The matrix-geometric [29] method and its variants [30,31] are numerical approaches to recursively compute the minimal nonnegative matrix solution of the matrix quadratic equation. The spectral expansion method is based on the eigenvalues and eigenvectors of the characteristic matrix polynomial [32–34]. However, the existing approaches face the state explosion problem when c is large. Therefore, an efficient computational algorithm is needed in the designing and dimensioning process for VOs.

3.2 An Efficient Computational Algorithm

Let $Q(x) = B + Q_1 x + Cx^2$ be defined as the characteristic matrix polynomial associated with equation (3). that can be written as:

$$Q(x) = \begin{bmatrix} q_{1,1}(x) & q_{1,2}(x) & 0 & \dots & 0 & 0 & 0 \\ q_{2,1}(x) & q_{2,2}(x) & q_{2,3}(x) & \dots & 0 & 0 & 0 \\ 0 & q_{3,2}(x) & q_{3,3}(x) & q_{3,4}(x) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & & \dots & q_{c,c-1}(x) & q_{c,c}(x) & q_{c,c+1}(x) \\ 0 & 0 & & \dots & 0 & q_{c+1,c}(x) & q_{c+1,c+1}(x) \end{bmatrix},$$

where

$$\begin{aligned} q_{1,1}(x) &= -(\lambda + \nu)x, \\ q_{i,i}(x) &= -(\lambda + \nu + (i-1)\alpha\mu)x \quad (i = 2, \dots, c), \\ q_{c+1,c+1}(x) &= \lambda - (\lambda + c\mu\alpha)x, \\ q_{i,i+1}(x) &= \lambda x + \nu x^2 \quad (i = 1, \dots, c), \\ q_{i+1,i}(x) &= \alpha\ i\ \mu\ x \quad (i = 1, \dots, c). \end{aligned}$$

The steady state probabilities are closely related to the eigenvalue-eigenvector pairs $(x, \boldsymbol{\psi})$ of $Q(x)$, which satisfy $\boldsymbol{\psi} Q(x) = 0$ and $det[Q(x)] = 0$ (see [34]). It is clear that $Q(x)$ has c zero-eigenvalues. The corresponding independent eigenvectors for c zero-eigenvalues are $\boldsymbol{\psi}_1 = [1, 0, \dots, 0]$, $\boldsymbol{\psi}_2 = [0, 1, 0, \dots, 0]$,$\dots$,$\boldsymbol{\psi}_c = [0, 0, \dots, 1, 0]$.

If the system is ergodic, the number of eigenvalues inside the unit disk is

$c+1$ (see [34]). Therefore, $Q(x)$ should have a single eigenvalue x_0 inside the unit disk because $Q(x)$ has c zero-eigenvalues. Let $\boldsymbol{\psi}_0$ be the corresponding left-hand-side eigenvector of $Q(\lambda)$ for the eigenvalue x_0.

The steady state probabilities can then be expressed as follows:

$$\begin{aligned}
&\mathbf{v}_j = b_0\boldsymbol{\psi}_0 x_0^j \ (j \geq 1),\\
&\mathbf{v}_0 = \sum_{k=0}^{c} b_k \boldsymbol{\psi}_k,
\end{aligned} \tag{4}$$

where $b_0, \ldots, b_c$s are the coefficients to be determined. Because the probabilities are greater than or equal to zero, $0 < x_0 < 1$ holds. Therefore, we have to determine x_0, $\boldsymbol{\psi}_0$, $b_0, \ldots, b_c$ efficiently to compute the steady state probabilities.

Because $Q(x_0)$ is a tridiagonal matrix and $q_{i,i}(x_0) \neq 0$, the component matrices of the LU decomposition of $Q(x_0)$ are written as:

$$L(x_0) = \begin{bmatrix}
l_1(x_0) & 0 & 0 \ldots 0 & & 0 & 0\\
\alpha\mu x_0 & l_2(x_0) & 0 \ldots 0 & & 0 & 0\\
\vdots & \vdots & \vdots\ \vdots & \vdots & \vdots & \vdots\\
0 & 0 & \ldots & \alpha(c-1)\mu x_0 & l_c(x_0) & 0\\
0 & 0 & \ldots & 0 & \alpha c\mu x_0 & l_{c+1}(x_0)
\end{bmatrix}, \tag{5}$$

$$U(x_0) = \begin{bmatrix}
1 & u_1(x_0) & \ldots & 0 & 0 & 0 & 0\\
0 & 1 & u_2(x_0) & \ldots & 0 & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & 0 & & \ldots & 0 & 1 & u_c(x_0)\\
0 & 0 & & \ldots & 0 & 0 & 1
\end{bmatrix}, \tag{6}$$

where $l_i(x_0)$ $(i = 1, \ldots, c+1)$ and $u_i(x_0)$ $(i = 1, \ldots, c)$ are the elements of $L(x_0)$ and $U(x_0)$, respectively. After simple algebra, we get:

$$\begin{aligned}
l_1(x_0) &= q_{1,1}(x_0) = -(\lambda+\nu)x_0,\\
l_i(x_0) + \alpha(i-1)\mu x_0 u_{i-1}(x_0) &= q_{i,i}(x_0), \quad (i = 2, \ldots, c+1),\\
l_i(x_0)u_i(x_0) &= \lambda x_0 + \nu x_0^2, \quad (i = 1, \ldots, c).
\end{aligned} \tag{7}$$

The determinant of $Q(x_0)$, therefore, is expressed as

$$det[Q(x_0)] = det[L(x_0)]det[U(x_0)] = \prod_{i=1}^{c+1} l_i(x_0). \tag{8}$$

As the consequence of equation (7), $l_i(x_0) \neq 0$ $(1 < i \leq c)$. Hence, $det[Q(x_0)] = 0$ follows $l_{c+1}(x_0) = 0$, which means x_0 is the root of $l_{c+1}(x)$. To compute the root of $l_{c+1}(x)$ in interval $(0, 1)$, several algorithms, such as bisection, the false position method, Dekker's method and Brent's method, can be applied. In this paper, we use Brent's method [35] to determine x_0 in interval $(0, 1)$.

Note that the recursive relations $\psi_{0,i} = \frac{-i\alpha\mu x_0}{l_i(x_0)}\psi_{0,i+1}$ between $\psi_{0,i}$ and $\psi_{0,i+1}$ $(i = c, \ldots, 1)$ are easily derived from equation $\boldsymbol{\psi}_0 Q(x_0) = \boldsymbol{\psi}_0 L(x_0)U(x_0) = 0$. Based on the property that multiplying an eigenvector with a scalar number results in an eigenvector, we can determine $\boldsymbol{\psi}_0 = [\psi_{0,1}, \psi_{0,2}, \ldots, \psi_{0,c+1}]$ by setting $\psi_{0,c+1} = 1$ and using the recursive relations to compute $\boldsymbol{\psi}_{0,i}$ for $i = c, \ldots, 1$.

From equations (2) and (4), we obtain

$$\begin{aligned} \sum_{i=0}^{c} b_i \boldsymbol{\psi}_i \left(D^A + B - A\right) &= b_0 \boldsymbol{\psi}_0 x_0 C, \\ \mathbf{b}\left(D^A + B - A\right) &= b_0 \boldsymbol{\psi}_0 x_0 C - b_0 \boldsymbol{\psi}_0 \left(D^A + B - A\right), \end{aligned} \tag{9}$$

where $\mathbf{b} = \sum_{i=1}^{c} b_i \boldsymbol{\psi}_i = [b_1, b_2, \ldots, b_c, 0]$.

By expanding equation (9), we get

$$\begin{bmatrix} \lambda b_1 - \mu\alpha b_2 \\ -\lambda b_1 + (\lambda + \mu\alpha) b_2 - 2\mu\alpha b_3 \\ \vdots \\ -\lambda b_{i-1} + (\lambda + (i-1)\mu\alpha) b_i - i\mu\alpha b_{i+1} \\ \vdots \\ -\lambda b_{c-1} + (\lambda + (c-1)\mu\alpha) b_c \\ -\lambda b_c \end{bmatrix} =$$

$$\begin{bmatrix} \text{-}\ b_0 \lambda \psi_{0,1} + b_0 \mu\alpha \psi_{0,2} \\ b_0 (x_0 \nu \psi_{0,1} + \lambda \psi_{0,1} - (\lambda + \mu\alpha) \psi_{0,2} + 2\mu\alpha \psi_{0,3}) \\ \vdots \\ b_0 (x_0 \nu \psi_{0,i-1} + \lambda \psi_{0,i-1} - (\lambda + (i-1)\mu\alpha) \psi_{0,i} + i\mu\alpha \psi_{0,i+1}) \\ \vdots \\ b_0 (x_0 \nu \psi_{0,c-1} + \lambda \psi_{0,c-1} - (\lambda + (c-1)\mu\alpha) \psi_{0,c} + c\mu\alpha \psi_{0,c+1}) \\ b_0 (x_0 \nu \psi_{0,c} + \lambda \psi_{0,c} - (\lambda + c\mu\alpha) \psi_{0,c+1}) \end{bmatrix}, \tag{10}$$

which means that all b_i $(i = c, \ldots, 1)$ can be expressed in b_0.

Finally, the normalization equation

$$\sum_{i=0}^{c} \sum_{j=0}^{\infty} \pi_{i,j} = \sum_{k=1}^{c} b_k \boldsymbol{\psi}_k \mathbf{e} + b_0 \boldsymbol{\psi}_0 \mathbf{e}/(1 - x_0) = 1 \tag{11}$$

can be used to compute coefficient b_0 (note that $\mathbf{e}$ is a vector of size $c+1$ with all elements equal to 1). Vector $\mathbf{b}$ can then be determined.

In summary, the proposed algorithm consists of the following steps:

- the application of Brent's method [35] to find root x_0 of $l_{c+1}(x)$ in interval $(0, 1)$,
- the computation of $\boldsymbol{\psi}_0$ based on $\psi_{0,i} = \frac{-i\alpha\mu x_0}{l_i(x_0)} \psi_{0,i+1}$ and $\psi_{0,c+1} = 1$,
- the calculation of coefficient b_0 based on equations (10) and (11),
- the computation of $\mathbf{b}$ based on equation (10).

The average number of occupied resource units characterizing the utilization of resources can be written as follows:

$$N_{occ} = \sum_{i=1}^{c} i \sum_{j=0}^{\infty} \pi_{i,j} = \sum_{i=1}^{c} i(\pi_{i,0} + \sum_{j=1}^{\infty} b_0 \boldsymbol{\psi}_{0,i+1} x_0^j) = \sum_{i=1}^{c} i(\pi_{i,0} + \frac{b_0 \boldsymbol{\psi}_{0,i+1} x_0}{1 - x_0}). \quad (12)$$

Other performance measures that quantitatively characterize QoS are expressed as follows:

- the average number of clients waiting in the orbit

$$N_{orbit} = \sum_{j=1}^{\infty} j \sum_{i=0}^{c} \pi_{i,j} = \sum_{j=1}^{\infty} j b_0 \sum_{i=0}^{c} \boldsymbol{\psi}_{0,i+1} x_0^j = \frac{b_0 x_0}{(1 - x_0)^2} \sum_{i=0}^{c} \boldsymbol{\psi}_{0,i+1}, \quad (13)$$

- the probability that all resources are occupied

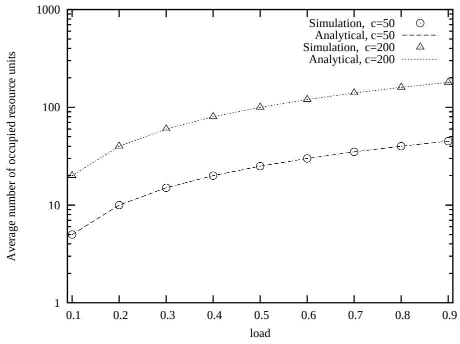
$$\sum_{j=0}^{\infty} \pi_{c,j}. \quad (14)$$

4 Numerical Results

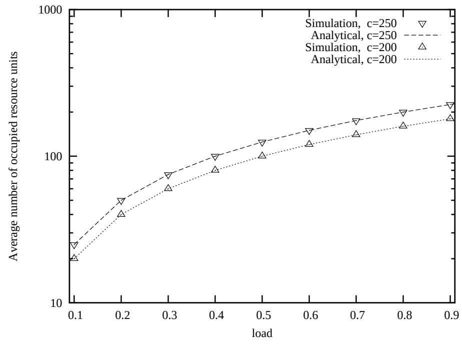
To validate our method, we built our own simulation program based on the SimPack toolkit [2] . The interactions (see Section 2) between the management of resources and the clients are implemented in the simulation model. In Figure 1, we plot the performance measures versus c and ρ for $T_r = 30$, $1/\mu = 30$ and 120 min, and the load $\rho = \lambda/(c\mu\alpha)$ in interval $(0, 1)$. Simulation results below 10^{-5} are not presented because reliable results could not be obtained within a reasonable amount of time. Note that the simulation is performed with a confidence level of 99%. The simulation is stopped when the accuracy (i.e., the ratio of the half-width of the confidence interval and the mean of the collected observations) of the average number of occupied resource units N_{occ} reaches 0.099%. The process takes several minutes with low loads and tens of minutes with moderate loads; it takes several hours to obtain results for a high value of c on a machine with an Intel® Xeon® E5410 2.33GHz processor.

The curves show excellent agreement between the analytical and simulated results concerning the average number of occupied resource units and waiting customers. The maximum difference between the simulation and analytical results concerning $Pr(I(t) = c)$ is only 8%. A similar agreement can also be observed in a wide range of other parameter values.

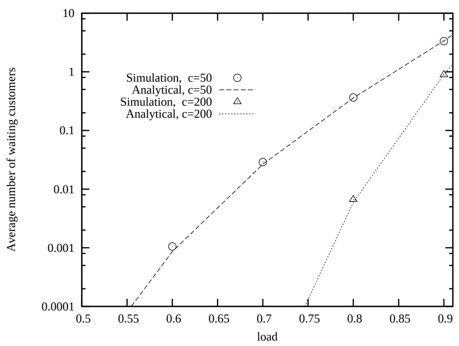
[2] `http://www.cise.ufl.edu/~fishwick/simpack.html`



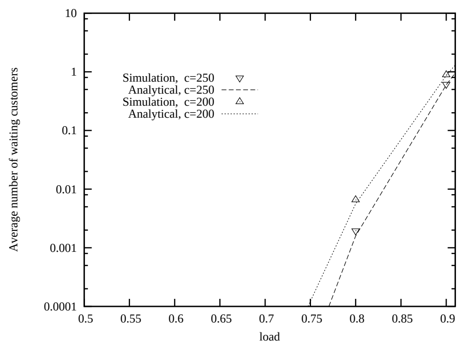
(a) $1/\mu = 30$



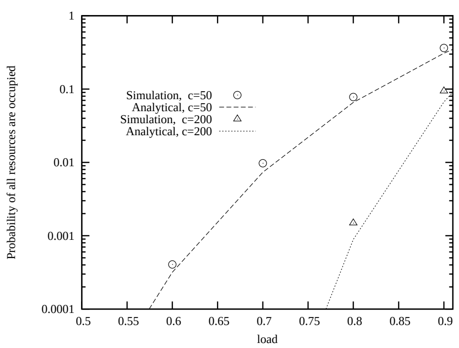
(b) $1/\mu = 120$



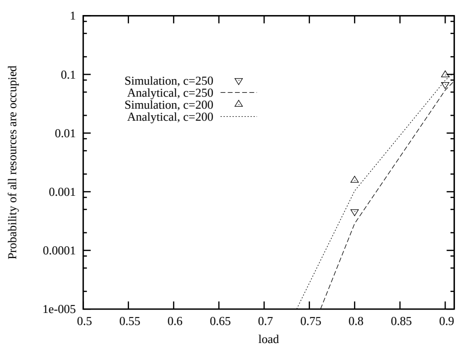
(c) $1/\mu = 30$



(d) $1/\mu = 120$



(e) $1/\mu = 30$



(f) $1/\mu = 120$

Figure 1. Comparison with simulation versus load ρ and c

We next compare the running time needed by different methods to get the steady state probabilities of the QBD process described in Section 3. Four methods are considered in the numerical comparison: the original spectral expansion method [36], the classical matrix-geometric method based on the successive substitution (SS) procedure [29], the variant of the matrix-geometric method [31] and the proposed procedure. Note that both the Brent algorithm applied in the proposed procedure and the original spectral expansion method use the machine precision (10^{-16}) to compute the necessary eigenvalues, while the stopping criterion for the iterative computation of the rate matrix by the successive substitution (SS) procedure [29] and the method of Naoumov et al. [31] is 10^{-10}.

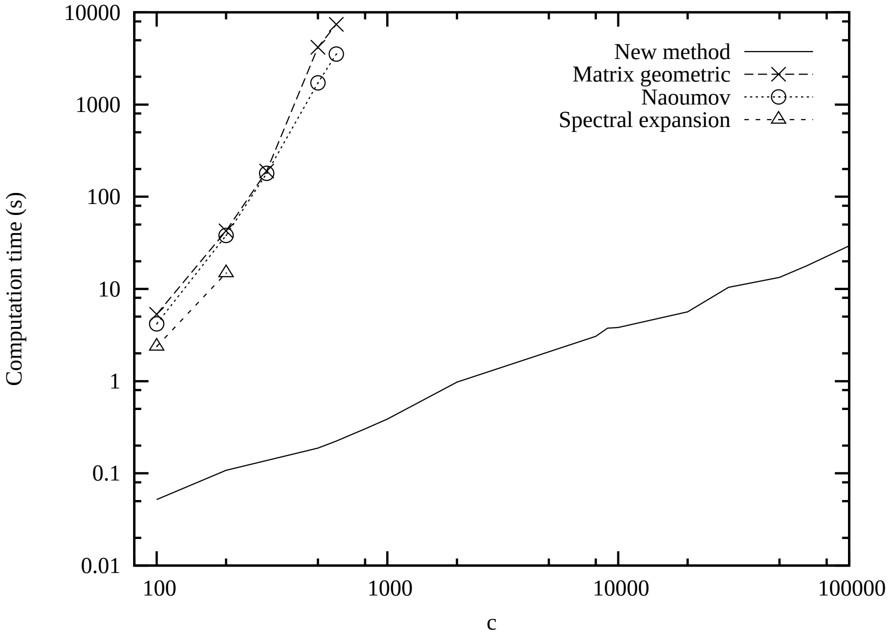
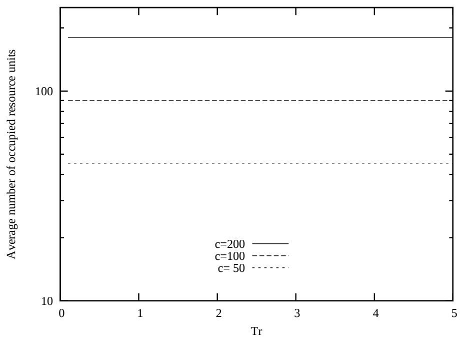


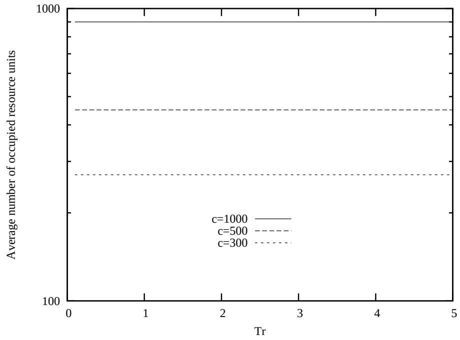
Figure 2. Computational time versus c ($\alpha = 0.5$, $1/\mu$=120 min, $1/\nu$=0.5 min, $\rho = \lambda/(c\mu\alpha) = 0.9$)

In Figure 2, we plot the computational time of the methods (implemented in Mathematica) versus c on a machine with an Intel® Xeon® E5410 2.33GHz processor for parameters $\alpha = 0.5$, $1/\mu$=120 min, $1/\nu$=0.5 min and $\rho = \lambda/(c\mu\alpha) = 0.9$. We can observe from Figure 2 that the proposed method has the smallest computation time among the methods. Moreover, the original spectral expansion method, the classical matrix-geometric method and the Naoumov et al. method fail with large values of c (e.g., $c > 600$) due to the state explosion problem. The classical matrix-geometric method needs more than 2 hours, and the Naoumov et al. method takes approximately 2 hours to get results for $c = 600$. The classical spectral expansion method could not give reliable numerical results for $c > 300$ because a large value of c gives rise to an ill-conditioned linear system of algebraic equations, while the proposed method takes a remarkably short time (approximate *30* s) to compute results for a much higher c value ($c = 100000$). Our approach is also much faster than the simulation. Several minutes are needed to obtain simulation results when $c = 200$, and a long running time is needed to get simulation results for a large value of c.

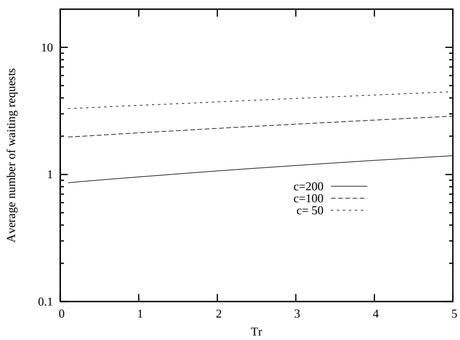
For $1/\mu = 30$ and 120 min and $\alpha = 0.1$ and load $\rho = \lambda/(c\mu\alpha) = 0.9$, we plot the performance measures versus T_r in Figure 3. Note that the practical range of T_r is chosen at around 30 s to take into account the "impatience of customers. It is observed that the increase of T_r results in the decrease of $Pr(I(t) = c)$ and the increase of N_{orbit}, while T_r in the investigated range does not have an impact on N_{occ}.



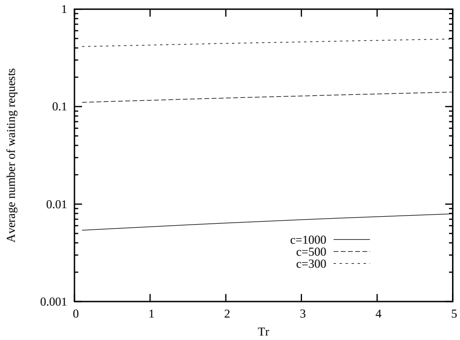
(a) $1/\mu = 30$



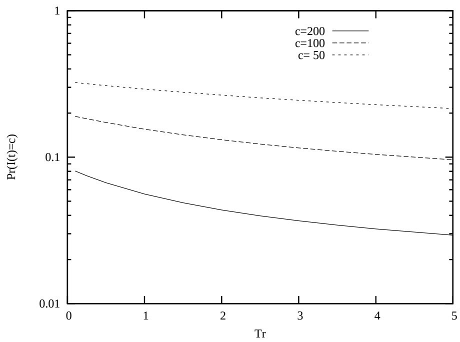
(b) $1/\mu = 120$



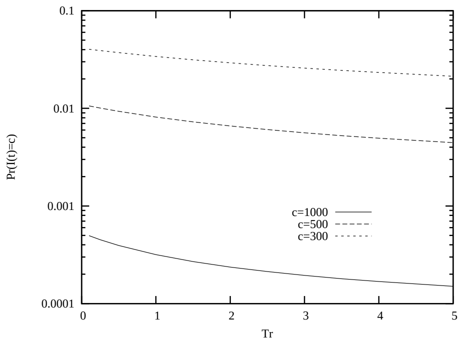
(c) $1/\mu = 30$



(d) $1/\mu = 120$



(e) $1/\mu = 30$



(f) $1/\mu = 120$

Figure 3. Performance measures versus T_r (in min) by the analytical method

5 Conclusions

We have examined the role and impact of resource dimensioning in VO formation and operation. We have also presented a queueing model for the performance evaluation of resource provisioning in VOs. To obtain a tractable and efficient solution, we have proposed an approximated model and a fast computational algorithm. The simulated and analytical results are in close agreement, which confirms that our model can accurately compute the performability measures. The comparison with existing numerical approaches shows the clear advantage of the proposed method. In particular, our method can rapidly compute the steady state probabilities of the resource contention problem in a VO environment. Therefore, it can be a useful tool for online resource dimensioning to support the formation and operation of VOs.

Acknowledgements

The author thanks the anonymous reviewers and editors for their constructive comments, which greatly helped to improve the presentation of this paper. Comments on language usage by Balázs Kádár are appreciated.

References

[1] T. J. Norman, A. D. Preece, S. Chalmers, N. R. Jennings, M. Luck, V. D. Dang, T. D. Nguyen, V. Deora, J. Shao, W. A. Gray, N. J. Fiddian, Agent-based formation of virtual organisations., Knowl.-Based Syst. 17 (2-4) (2004) 103–111.

[2] J. Patel, W. T. L. Teacy, N. R. Jennings, M. Luck, S. Chalmers, N. Oren, T. J. Norman, A. Preece, P. M. D. Gray, G. Shercliff, P. J. Stockreisser, J. Shao, W. A. Gray, N. J. Fiddian, S. Thompson, Agent-based virtual organisations for the grid, in: AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, ACM, New York, NY, USA, 2005, pp. 1151–1152.

[3] A. Preece, S. Chalmers, C. McKenzie, A reusable commitment management service using semantic web technology, Know.-Based Syst. 20 (2) (2007) 143–151.

[4] D. Romero, A. Molina, VO breeding environments & virtual organizations integral business process management framework, Information Systems Frontiers 11 (5) (2009) 569–597.

[5] H. Afsarmanesh, L. M. Camarinha-Matos, On the classification and management of virtual organisation breeding environments, Int. J. Inf. Technol. Manage. 8 (3) (2009) 234–259.

[6] J. Hodik, J. Vokrinek, P. Becvar, Support for virtual organisation creation – partners' profiles and competency management, Int. J. Agent-Oriented Softw. Eng. 3 (2/3) (2009) 230–251.

[7] J. J. Jung, C.-M. Koo, G.-S. Jo, A divide-and-conquer approach to detecting latent community of practice from virtual organizations, in: ICCIT '07: Proceedings of the 2007 International Conference on Convergence Information Technology, IEEE Computer Society, Los Alamitos, CA, USA, 2007, pp. 129–134.

[8] J. J. Jung, Taxonomy alignment for interoperability between heterogeneous virtual organizations, Expert Syst. Appl. 34 (4) (2008) 2721–2731.

[9] J. J. Jung, Y.-S. Han, Conceptual modeling of semantic service chain management for building service networks: A preliminary result, in: KES-AMSTA '09: Proceedings of the Third KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 481–489.

[10] J. J. Jung, Semantic business process integration based on ontology alignment, Expert Systems with Applications 36 (8) (2009) 11013 – 11020.

[11] L. Li, B. Wu, Y. Yang, Agent-based approach for dynamic ontology management, in: R. Khosla, R. J. Howlett, L. C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, 9th International Conference, KES 2005, Melbourne, Australia, September 14-16, 2005, Proceedings, Part III, Vol. 3683 of Lecture Notes in Computer Science, Springer, 2005, pp. 1–7.

[12] N. P. Verdeza, N. Galeano, D. Romero, R. Mejia, A. Molina, Collaborative engineering environments for virtual organisations, Int. J. Inf. Technol. Manage. 8 (3) (2009) 298–320.

[13] S. Barnes, Knowledge management systems: theory and practice, Cengage Learning EMEA, 2002.

[14] T.-H. Chang, T.-C. Wang, Using the fuzzy multi-criteria decision making approach for measuring the possibility of successful knowledge management, Information Sciences 179 (4) (2009) 355–370.

[15] Q. He, J. Yan, R. Kowalczyk, H. Jin, Y. Yang, Lifetime service level agreement management with autonomous agents for services provision, Information Sciences 179 (15) (2009) 2591–2605.

[16] F. Li, Y. Yin, Approaches to knowledge reduction of covering decision systems based on information theory, Information Sciences 179 (11) (2009) 1694–1704.

[17] N. T. Nguyen, Consensus system for solving conflicts in distributed systems, Information Sciences 147 (1-4) (2002) 91–122.

[18] N. T. Nguyen, Advanced Methods for Inconsistent Knowledge Management, Springer, 2008.

[19] D. Richards, A social software/web 2.0 approach to collaborative knowledge engineering, Information Sciences 179 (15) (2009) 2515–2523.

[20] V. V. Rybakov, Logic of knowledge and discovery via interacting agents - decision algorithm for true and satisfiable statements, Information Sciences 179 (11) (2009) 1608–1614.

[21] H. W. Davidow, M. Malone, The Virtual Corporation: Structuring and revitalizing the corporation for the 21st century, Harper Collins, 1992.

[22] A. Mowshowitz, Virtual organization, Commun. ACM 40 (9) (1997) 30–37.

[23] A. Mowshowitz, Virtual Organization: Toward a Theory of Societal Transformation Stimulated by Information Technology, Greenwood Publishing Group, 2002.

[24] M. Rittenbruch, H. Kahler, A. B. Cremers, Supporting cooperation in a virtual organization, in: ICIS '98: Proceedings of the international conference on Information systems, Association for Information Systems, Atlanta, GA, USA, 1998, pp. 30–38.

[25] V. D. Dang, N. R. Jennings, Optimal clearing algorithms for multi-unit single-item and multi-unit combinatorial auctions with demand/supply function bidding, in: ICEC '03: Proceedings of the 5th international conference on Electronic commerce, ACM, New York, NY, USA, 2003, pp. 25–30.

[26] J. R. Artalejo, A. Gómez-Corral, Retrial Queueing Systems, Springer, 2008.

[27] G. I. Falin, J. G. C. Templeton, Retrial Queues, Chapman & Hall, London, 1997.

[28] J. Roszik, J. Sztrik, Performance analysis of finite-source retrial queues with non-reliable heterogeneous servers, Journal of Mathematical Sciences (146) (2007) 6033–6038.

[29] M. F. Neuts, Matrix Geometric Soluctions in Stochastic Model, Johns Hopkins University Press, Baltimore, 1981.

[30] G. Latouche, V. Ramaswami, Introduction to Matrix Analytic Methods in Stochastic Modeling, ASA-SIAM Series on Statistics and Applied Probability, 1999.

[31] V. Naoumov, U. R. Krieger, D. Warner, Analysis of a Multi-Server Delay-Loss System With a General Markovian Arrival Process, in: S. R. Chakravarthy, A. S. Alfa (Eds.), Matrix-analytic methods in stochastic models, Vol. 183 of Lecture Notes in Pure and Applied Mathematics, Marcel Dekker, 1996.

[32] R. Chakka, T. V. Do, The MM $\sum_{k=1}^{K} CPP_k/GE/c/L$ G-Queue with Heterogeneous Servers: Steady state solution and an application to performance evaluation, Performance Evaluation 64 (2007) 191–209.

[33] T. V. Do, Comments on multi-server system with single working vacation, Applied Mathematical Modelling 33 (12) (2009) 4435–4437.

[34] I. Mitrani, R. Chakka, Spectral expansion solution for a class of Markov models: Application and comparison with the matrix-geometric method, Performance Evaluation 23 (1995) 241–260.

[35] R. Brent, Algorithms for Minimization without Derivatives, Prentice-Hall, Englewood Cliffs, NJ, 1972.

[36] R. Chakka, Performance and Reliability Modelling of Computing Systems Using Spectral Expansion, Ph.D. thesis, University of Newcastle upon Tyne (Newcastle upon Tyne) (1995).