Available online at www.sciencedirect.com

SciVerse ScienceDirect

Mathematics and Computers in Simulation 89 (2013) 38–49

MATHEMATICS AND COMPUTERS IN SIMULATION

www.elsevier.com/locate/matcom

Original article

A hybrid method for performance analysis of *G*/*G*/*m* queueing networks

Boualem Rabta

Entreprise Institute, University of Neuchâtel, Rue A.L. Breguet 1, CH-2000 Neuchâtel, Switzerland

Received 17 June 2010; received in revised form 21 February 2013; accepted 14 March 2013
Available online 27 March 2013

Abstract

Open queueing networks are useful for the performance analysis of numerous real systems. Since exact results exist only for a limited class of networks, decomposition methods have been extensively used for approximate analysis of general networks. This procedure is based on several approximation steps. Successive approximations made in this approach can lead to a considerable error in the output. In particular, there are no general accurate formulas for computing the mean waiting time and the inter-departure variance in general multiple-server queues. This causes the results from decomposition methods when applied to *G*/*G*/*m* queueing networks to be very approximative and to significantly deviate from actual performance values. We suggest substituting some approximate formulae by low-cost simulation estimates in order to obtain more accurate results when benefiting from the speed of an analytical method. Numerical experiments are presented to show that the proposed approach provides improved performance.

2000 MSC: 60K25; 68M20; 90B22

Keywords: *G*/*G*/*m* queueing networks; Two-moments decomposition; Simulation; Hybrid method; Performance measurement

1. Introduction

Queueing networks are an extremely useful class of models that have seen use in a host of application areas. In particular, they have been successfully used to model the performance of a variety of complex systems such as computer systems, communication networks, production lines and manufacturing systems. Queueing models allow the consideration of the randomness of different components. Unfortunately, exact results exist only for a limited class of networks (product form). Decomposition methods among other approximation methods, have been extensively used to obtain approximate results. This approach has been developed by Kuehn [23], Whitt [34] among others, and is based on two-moments approximations, that is, all stochastic processes are characterized by their mean and squared coefficient of variation (SCV) (see, e.g., [26]). Whitt [34] proposed the extension of this method to multiple server nodes by suggesting approximate formulas for computing the mean waiting time and the inter-departure second moment. However, a drawback of this method is that it performs well in some situations, but not others (see, e.g. [35]). Whitt [37] proposed an enhancement to the parametric-decomposition method. Instead of using a variability parameter for

E-mail addresses: boualem.rabta@unine.ch, brabta@yahoo.fr

0378-4754/$36.00
http://dx.doi.org/10.1016/j.matcom.2013.03.003

each arrival process, he suggested using a variability function; i.e., the variability parameter should be regarded as a function of the traffic intensity of a queue to which the arrival process might flow.

The classical decomposition method assumes that all arrival processes to a station are renewal processes, ignoring therefore the possible correlation among different arrival streams. Also, it is assumed that the superposition of renewal processes is a renewal process which is generally not correct. Albin [2] remarks that if at least one component process is not Poisson, then the superposition process is not renewal and the intervals between arrival points are identically distributed, but are not independent. Ignoring this interdependence (i.e. correlation) between inter-arrival intervals might cause errors in the approximation. Kim et al. [18] proposed the innovations method as an improvement to Whitt's method by replacing relations among squared coefficients of variability with approximate regression relationships among in the underlying point processes. These relationships allow adding information on correlations between different streams. Kim [16] combined the innovations method with Whitt's variability functions to deal with the heavy traffic bottleneck phenomenon.

Van Nyen et al. [33] pointed out that the classical decomposition method for networks with multiple customer classes and aggregation as proposed by Albin [1,2] and Whitt [34] performs poorly in manufacturing context. They used simulation to show that the method generates serious approximation errors in some cases. Note that since the Albin–Whitt procedure has been proposed, several improvements followed (e.g., multi-class decomposition methods proposed by different authors (e.g., Bitran and Tirupati [6], Whitt [36], Caldentey [9])) and many papers proposed methods to capture correlations in the arrival process reported as the principal source of errors by Van Nyen et al. (see, e.g., Heindl and Telek [14], Heindl et al. [15], Kim [17,16], Kim et al. [18], Balcioglu et al. [7]).

Ignoring the correlation among arrival streams is not the only source of errors in the decomposition method. This procedure includes a number of approximation formulae that are more or less precise depending on the input values, and generates errors at each step. Previous papers focused on improving the merging step (e.g., [18,7]) and improving the estimation of inter-departure SCV (e.g., in multiple customer classes case [6,36,9]). Haverkort [13] proposes to approximate $G/G/1$ nodes by $PH/PH/1$ ones through approximating arrival and service distributions by phase-type distributions based on their first two moments and then deriving exact measures for each node using matrix-geometric techniques. This suggestion leads to improved performance but is only limited to single-server nodes. In addition, the quality of the approximation may also depend on higher moments. The inter-departure SCV computation by means of Marshall's formula in the classical decomposition method, involves the computation of the waiting time. For the $G/G/1$ queueing system only approximate formulae for the waiting time are available and even less are available for the multiple-server system $G/G/m$. These formulae seem to perform more or less accurately depending on the system's parameters.

Simulation is perhaps the most popular approach to the performance measure of complex systems. For instance, Cruz et al. [11] discuss the use of simulation for performance evaluation of mobile communication networks and Cruz et al. [12] describe a simulation model for state-dependent finite queueing network analysis. It is identified that simulation models allow a higher level of realism and system's details but they can be cumbersome to optimize (much time to build the model and run different scenarios), and their accuracy is largely dependent on the quality of the calibration data. In particular, simulation of large systems can be expensive both in terms of CPU time and use of available resources (e.g., memory, processors).

In this paper, we propose a hybrid simulation-decomposition method for the analysis of $G/G/m$ queueing networks. We aim, to show that the improvement of the classical decomposition method is possible and to propose a more precise tool. A low cost simulation algorithm based on a set of recursive equations proposed by Krivulin [22], is used to compute the inter-departure SCV instead of the approximative formula used in the classical decomposition method. Furthermore, the same set of recursive equations allows us to simulate the waiting time for $G/G/m$ nodes. This allows us to obtain more accurate results when considering real service distributions and multiple-server nodes.

The proposed approach attempts to combine the speed of the analytical decomposition procedure with the precision of simulation. Shanthikumar and Sergent [30] pointed out that the estimators obtained from the hybrid simulation/analytic models have lower variance than the variance of the estimators of the traditional simulation models. As opposed to a pure simulation method, our method is faster and less expensive in terms of computer resources since no memory allocation is required for entities (queues, servers, etc.). Numerical experiments demonstrate that improvements are made in specific situations and that performance measures such as the waiting time at the bottleneck stations and overall cycle time obtained by means of the hybrid method, are more accurate.

2. Overview of the parametric-decomposition method

Consider an open network of $G/G/m$ stations where customers arrive to station i according to a renewal process with rate λ_{0i} and SCV ca_{0i}. They are served on a first-come-first-served basis with mean service durations S_i ($\mu_i = 1/S_i$) and SCV cs_i. After completing their service, customers are either routed to station j with probability p_{ij} or they leave the network with probability $1 - \sum_j p_{ij}$.

The classical decomposition method is based on the following three steps:

- Merging arrival streams.
- Computing departures from single stations.
- Splitting up the departure stream.

Firstly, all arrival streams are merged in one arrival stream. This is done by assuming that all arrival processes are renewal processes. The overall arrival rate to station i is computed by solving the traffic rate equation:

$$\lambda_i = \lambda_{0i} + \sum_j \lambda_j p_{ji},$$

The asymptotic method [29] and the stationary-interval method [23] may be used to determine ca_i, i.e., the merged interarrival time SCV ($ca_i = \mathbf{V}(a_i)/\mathbf{E}(a_i)^2$, $\lambda_i = 1/\mathbf{E}(a_i)$). Moreover, the asymptotic method is asymptotically correct as $\rho_i \to 1$ (heavy traffic intensity) and the stationary-interval method is asymptotically correct when the arrival process tends to a Poisson process [5].

Let ca_{ji} be the inter-arrival time SCV at station i from station j. Based on the asymptotic method, ca_i is a convex combination of ca_{ji} given by:

$$ca_i = \frac{\lambda_{0i}}{\lambda_i} ca_{0i} + \sum_{j=1}^{n} \frac{\lambda_{ji}}{\lambda_i} ca_{ji}. \tag{1}$$

Albin [1,2] suggested an approximation to ca_i based on a convex combination between the previous value and the one obtained by the stationary interval method. [35] substituted the stationary interval method by a Poisson process and obtained:

$$ca_i = w_i \sum_{j=0}^{n} \frac{\lambda_{ji}}{\lambda_i} ca_{ji} + 1 - w_i \tag{2}$$

where

$$w_i = \frac{1}{1 + 4(1-\rho_i)^2(v_i - 1)}$$

$$v_i = \frac{1}{\sum_{j=0}^{n} (\lambda_{ji}/\lambda_i)^2}$$

The squared coefficient of variation cd_i of the inter-departure time from a single-server station i is computed by Marshall's formula [25]:

$$cd_i = ca_i + 2\rho_i^2 cs_i - 2\rho_i(1-\rho_i)\frac{\mathbf{E}(W_i)}{S_i}. \tag{3}$$

Using the Kraemer Langenbach–Belz (KLB) [21] approximation for the expected waiting time $\mathbf{E}(W_i)$ at $G/G/1$ nodes,

$$\mathbf{E}(W_i) = \frac{\rho_i(ca_i + cs_i)}{2\mu_i(1-\rho_i)} g,$$

where

$$g = \begin{cases} \exp\left\{\dfrac{-2(1-\rho_i)(1-ca_i)^2}{3\rho_i(ca_i+cs_i)}\right\} & \text{if } ca_i < 1, \\ \exp\left\{\dfrac{-(1-\rho_i)(ca_i-1)}{ca_i+4cs_i}\right\} & \text{if } ca_i \geq 1. \end{cases}$$

Whitt [34] ignores the parameter g and proposes:

$$cd_i = \rho_i^2 cs_i + (1-\rho_j^2)ca_i.$$

For multiple-server nodes $G/G/m$, Whitt [34] proposes the following approximation:

$$cd_i = 1 + (1-\rho_i^2)(ca_i - 1) + \frac{\rho_i^2}{\sqrt{m}}(\max\{cs_i, 0.2\} - 1).$$

The departure stream is then split up according to the routing matrix. Under markovian routing:

$$cd_{ij} = p_{ij}cd_i + 1 - p_{ij},$$

where cd_{ij} is the SCV of the departure process from station i to station j.

Finally, since the departure steam from a station is the arrival stream to the next station, we have:

$$ca_{ij} = cd_{ij}.$$

The previous set of equations allows us to determine individual parameters to each station (i.e., overall arrival rate and SCV). Performance measures are obtained by known formulae for single $G/G/1$ queues. For instance, the mean waiting time is obtained by the KLB formula and the mean jobs in the queue is given by Little's formula. The overall cycle time (i.e., the total time spent by a job in the network) is given by:

$$CT = \sum_i V_i(E(W_i) + S_i),$$

where

$$V_i = \frac{\lambda_i}{\sum_j \lambda_{0j}}$$

is the visit ratio to station i (see, e.g., [34]).

Although this approach is elegant and simple, it is based on many fairly loose approximations [18]. Namely:

(1) All arrival streams are assumed to be renewal and they are approximated by only considering their first two moments.
(2) Combination of renewal processes is assumed to be a renewal process.
(3) Merging formula is approximative.
(4) Service distributions are approximated by considering only their first two moments.
(5) Inter-departures SCV is computed using Marshall's formula (exact only for single-server nodes) but uses an approximative formula for the mean waiting time.
(6) Performance measures of the system (mean waiting time, etc.) are computed using approximate formulae.

Our proposed approach aims to eliminate some of those approximation steps to improve the output of the decomposition method. In particular, we suggest the use of a low-cost simulation algorithm to estimate inter-departures SCV and mean waiting time for each node (elimination of steps (5) and (6)). Furthermore, the simulation algorithm allows the use of the real service distribution instead of the two-moments approximation (elimination of step (4)).

Table 1
Mean waiting time in *G*/*G*/1 queue.

Arrival rate	Arrival SCV	Service rate	Service SCV	W_{KLB}	W_{Sim}	E
0.95	1.2	1	0.2	13.234	13.41	1%
	4			38.672	38.948	1%
	10			92.945	94.787	2%
	20			183.332	187.33	2%
0.2	1.2	1	0.2	0.162	0.169	4%
	4			0.274	0.318	13%
	10			0.655	0.336	95%
	20			1.216	0.365	233%

3. The waiting time approximation in *G*/*G*/*m* queues

The KLB formula for approximating the mean waiting time in *G*/*G*/1 queueing systems is a reasonable approximation in a high utilization context but it dramatically fails in some situations. This may be shown by means of experiments (Table 1). Consider a single *G*/*G*/1 queue under high arrival variability condition. Let W_{KLB} be the waiting time computed by the KLB formula, W_{Sim} be the simulated waiting time and E be the relative error, i.e.:

$$E = \left| \frac{W_{KLB} - W_{Sim}}{W_{Sim}} \right| \times 100\%.$$

Table 1 shows the results for different values of the inter-arrival SCV under high and low utilization. It is observed that the result is less precise or even poor in cases of low traffic intensity and high arrival variability.

High variability is known to be a real condition in several systems (e.g., the world wide web, [10]).

Note that, other formulae for the waiting time exist (see, e.g. [8, §6.3.4]). However, it seems that the KLB formula is the most used one and that it outperforms the other expressions in most cases.

The decomposition method might lead to considerable errors when stations have different utilization levels and variability values. Since, the departure stream from a station is the arrival stream to another one, errors in estimating the waiting time for station i might also influence the result of other stations, creating a somewhat bullwhip effect.

For *G*/*G*/*m* queues, no formula that allows to approximate the waiting time with good accuracy in general situations is available. Some authors suggested approximations on the basis of numerical examinations, statistical tests or interpolation. Sakasegawa [28] proposed the following approximation:

$$\mathbf{E}(W_{G/G/m}) = \frac{ca + cs}{2} \frac{\rho^{\sqrt{2(m+1)}-1}}{m\mu(1-\rho)}.$$

Similarly, Whitt [34] suggested a modified version of KLB formula:

$$\mathbf{E}(W_{G/G/m}) = \frac{ca + cs}{2} \mathbf{E}(W_{M/M/m}).$$

Kimura [19] proposed two distribution-dependent approximations for the mean waiting time in a *GI*/*G*/*m* queue based on weighted combinations of the exact mean waiting times for the *GI*/*M*/*m* and *M*/*D*/*m* queues. In addition, Kimura's method has been successfully applied to the estimation of blocking probabilities and throughput in networks of finite queues (see e.g., [31]).

4. Hybrid method

Our proposed approach suggests the use of a low-cost simulation algorithm to estimate inter-departures SCV and mean waiting time for each node. Furthermore, the simulation algorithm allows the use of the real service distribution instead of the two-moments approximation. We build our algorithm based on a set of recursive equations.

Let w_n be the waiting time of the nth customer, a_{n+1} the inter-arrival time between the nth and the $(n+1)$th customer and s_n the service time of the nth customer. A low cost simulation procedure can be derived from recursive formulas

of queueing systems. For multiple-server queues we may derive a simulation procedure based on the set of recursive equations proposed by Krivulin [22].

Let A_n be the nth customer arrival epoch to the queue and C_n be the moment when it starts the service. Given the sequence $(a_n)_n$ of interarrival times, it follows that: $A_n = A_{n-1} + a_n$. If at A_n, one or more of the m servers is free then the nth customer starts its service immediately, i.e., $C_n = A_n$. If not, the customer will wait until the $(n-m)$th departure occurs at date D_{n-m}, i.e., $C_n = D_{n-m}$. Whence,

$$C_n = \max(A_n, D_{n-m}). \tag{4}$$

The vector D of departure dates corresponds to the sequence of completion dates $(C_k + s_k)$ arranged in ascending order. The waiting time for the nth customer will be

$$w_n = C_n - A_n.$$

The following algorithm allows us to simulate $G/G/m$ systems. The output is a series of observations of the inter-departure times (resp. the waiting times) from which we can compute an estimator for the variance of the inter-departure time (resp. the mean waiting time).

$A_1 = a_1$;
For i=2,...,m, $A_i = A_{i-1} + a_i$;
For i=1,...,m do
$C_i = A_i$;
$w_i = 0$;
Generate service time s_i;
insert $d_i = (C_i + s_i)$ in the sorted vector D;
End for
$n = m$;
Repeat
Increment n;
Generate inter-arrival time a_n;
$A_n = A_{n-1} + a_n$;
$C_n = \max(A_n, D_{n-m})$;
$w_n = C_n - A_n$;
Generate service time s_n;
insert $d_n = (C_n + s_n)$ in the sorted vector D;
Until $n > N + m$;

The simulation's output is a sample $Y_i = D_i - D_{i-1}$, $i = 1, \ldots, N$ (with $D_0 = 0$) of inter-departure times. We may use the non-overlapping batch means (NBM) method (see, e.g., [3,24]) to compute an unbiased estimator $\overline{V}_i$ of the variance of inter-departure times at station i. The estimator of the mean waiting time $\overline{W}$ along with a confidence interval are also computed from the simulation output. The first l observations might be deleted in order to reduce the initial condition's bias. The required estimates are computed progressively along the simulation run. It is then sufficient to only keep the m last components of the vector D in memory. At each step, the new element replaces the smallest component.

The simulation algorithm allows the use of different kinds of distributions for service durations. In the decomposition method, the arrival process to one station is the combination of several processes representing the arrivals from outside and from other stations as well. The distribution of the merged process is unknown and considering that only the two first moments are available, it is not possible to compute it. Therefore, we fit unknown arrival distributions using hyper-exponential and hypo-exponential distributions by matching their first two moments.

Now, the classical decomposition method is modified by replacing the inter-departure SCV formula (3) by the estimated values $\overline{V}_i$. Thus, overall inter-arrival SCV for each station is computed using the following steps.

Merging arrival processes (as above).

Estimating inter-departure times SCV from simulation

$$cd_i = \lambda_i^2 \overline{V}_i.$$

Splitting up departures (as above).

$ca_{ij} = cd_{ij}$.

Table 2
Mean waiting time at the bottleneck node in the 9-nodes network.

Variability	Sim	QNA	HM
High ($ca_{0,1} = 8$)	30 ± 5.1	8.9	27.08
Low ($ca_{0,1} = 0$)	5.03 ± 0.22	8.0	3.88

The previous set of equations is solved iteratively until the changes in ca_i become small. That is, starting from initial values $(ca_i^{(0)})_i$, we compute successively $(ca_i^{(n)})_i$ until

$$\max_i |ca_i^{(n+1)} - ca_i^{(n)}| < \varepsilon,$$

for some fixed error threshold $\varepsilon > 0$.

We use common random numbers to compare $ca_i^{(n+1)}$ and $ca_i^{(n)}$. This can be done by initializing the random number generator with seed x_i at each iteration for station i.

5. Numerical examples

We will test our hybrid method (HM) on several examples and compare the results to simulation (SIM) and to the classical decomposition method. In these examples, queueing network analyzer (QNA) refers to the implementation of the decomposition algorithm for *G/G/m* networks as described in Whitt's papers [34,35] (see above).

The simulation estimates of the expected waiting times at each queue were obtained from 64 independent replications of 500,000 time units long, discarding the first 10,000 in each case to allow the system to approach steady state. (Longer simulation have subsequently been executed that also support the results here.) The estimated mean steady-state waiting times in all cases are displayed together with estimates of 95% confidence intervals.

Decomposition algorithms are very fast given that they are base on analytical formulae. QNA-like software (see, e.g., [27]) are adequate for tactical decision making and initial (rough-cut) systems' analysis. With the examples below ran on a modern computer (Intel Core 2 Duo CPU at 2.33 GHz with 2 Gb of RAM), QNA finishes only in few milliseconds while the hybrid algorithm takes around half a second and pure simulation several minutes. (Notice that simulation time depends on the length and the number of replications and that it may be necessary to run longer simulation in order to reach stationary regime in big networks.) On a larger scale, the difference could be more emphasized and the trade-off between speed and accuracy may be a key-element in choosing the right tool for practical use.

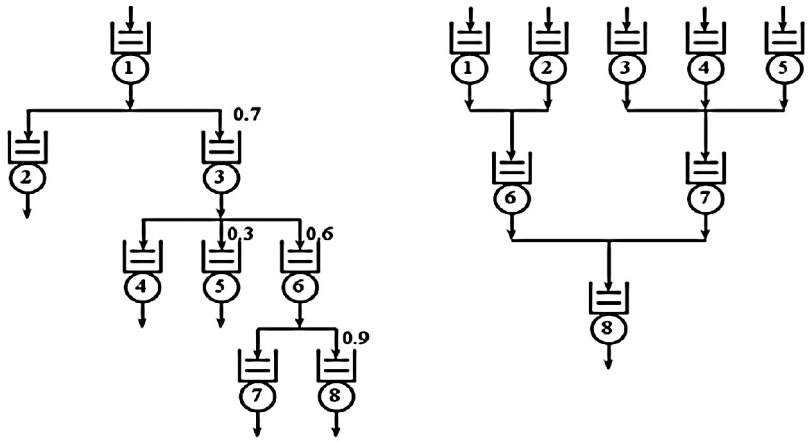
5.1. A network with 9 nodes in tandem

Firstly, we borrow an example from [32]. This is a rather simple network with 9 queues in tandem and a bottleneck at the last queue. Queues 1–8 have utilization equal to 0.6 whereas the utilization of queue 9 is equal to 0.9. All service times are exponential. In their paper the authors used this example to illustrate "the heavy-traffic bottleneck phenomenon" and to point out the limitations of the classical decomposition method in this context. Indeed, as we can observe in Table 2 (simulation and QNA results are copied from [32]), QNA fails dramatically in estimating the mean waiting time at the bottleneck station. The hybrid method performs well and is able to estimate the mean waiting time within the confidence interval.

Whitt later proposed in [36] the variability method to overcome the limitations of the classical decomposition procedure. This method performs quite well on the example above and it compares positively to our hybrid method.

5.2. A divergent feed-forward network

Now, we will consider the effect of splitting on the performance on both the classical method and the hybrid method compared to simulation. Consider the divergent feedforward network given in Fig. 1(a). The external arrival rate to station 1 is $\lambda_{01} = 0.8$. Table 3 shows the estimated mean waiting time under different combinations of arrival and service variability levels (6 scenarios).



(a) A feedforward divergent network. (b) A feedforward convergent network.

Fig. 1. Two feedforward networks.

Table 3
Mean waiting time at the bottleneck station for the convergent feedforward network.

Scenario	ca_{01}	$cs_1, \ldots, cs_8$	QNA	HM	SIM	CI-hw
1	0.8	0.64	54.76	59.72	58.91	1.46
2	0.8	4	246.54	193.82	194.99	8.97
3	4	0.64	66.80	95.41	98.87	3.10
4	4	1	87.32	104.54	122.67	5.50
5	4	4	258.422	237.40	246.67	10.27
6	12	1	116.87	228.96	231.91	10.03

The classical decomposition (QNA) performs well in many of the considered scenarios but it fails in others. In particular, in the case of highly variable arrivals combined to low or moderate variability in service. On the other hand, the hybrid method performs well in most situations and gives estimates for mean waiting time at the bottleneck which are validated by simulation. The splitting operation does not seem to have a significant impact on the performance of the proposed hybrid method.

5.3. A convergent feed-forward network

In this example (Fig. 1(b)), we test the effect of the merging on the performance of the hybrid method. The external arrival rate to stations 1–5 is set to 0.16. The utilization of each station is 0.6 except the last station which is the bottleneck ($\rho_8 = 0.96$). Table 4 shows the mean waiting time at the bottleneck station under different arrival and service variability levels.

Table 4
Mean waiting time at the bottleneck station for the convergent feedforward network.

Scenario	$ca_{01}, \ldots, ca_{05}$	$cs_1, \ldots, cs_8$	QNA	HM	SIM	CI-hw
1	0.8	0.64	20.57	20.28	20.37	0.27
2	0.8	4	91.61	69.73	74.31	1.94
3	4	0.64	29.94	61.06	59.34	1.72
4	4	1	37.55	57.52	66.42	2.09
5	4	4	100.96	114.50	113.52	3.33
6	12	1	60.74	129.57	161.62	6.28

Table 5
Input parameters for the 9 nodes network with single-server nodes.

Node	Arrival rate	Mean service time	Service SCV	ρ_i
1	0.4	1	0.6	0.40
2	0.25	1	0.2	0.25
3	0.45	0.8	8	0.36
4	0	1	0.5	0.36
5	0	1	0.8	0.11
6	0	0.8	0.8	0.50
7	0	0.8	0.2	0.45
8	0	0.9	0.4	0.46
9	0	2.5	1.8	0.95

Again, we observe that QNA fails in high arrival variability situations and that our hybrid method gives the best estimates. As before we conclude that the combination of arrival processes does not affect significantly the performance of our method.

5.4. A general topology network

In the following examples, we test the hybrid method with a general topology network.

5.4.1. Single-server nodes

Consider the 9 nodes queueing network of $G/G/1$ queues shown in Fig. 2 (adapted from [23]). Input parameters of this network are listed in Table 5. Table 6 shows the values for the mean waiting time estimated by means of the classical decomposition method (QNA), our hybrid method (HM) and (pure) simulation (SIM) along with a 95% confidence interval (CI-hw is the confidence interval half width). We test different arrival variability levels and consequently report the results.

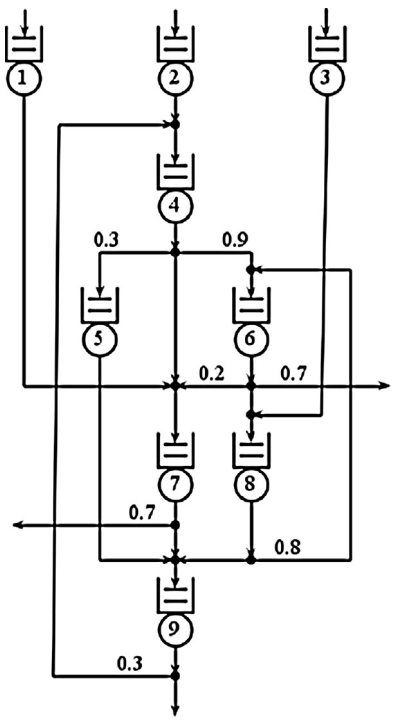


Fig. 2. The 9 nodes queueing network.

Table 6
Comparison of the mean waiting time in the 9 single-server nodes queueing network.

Node	External interarrival SCV	QNA	HM	SIM	CI-hw
1	1	0.533	0.537	0.532	0.001
2	1	0.200	0.201	0.199	< 0.001
3	1	2.025	2.023	2.021	0.011
4		0.434	0.440	0.403	0.001
5		0.109	0.111	0.093	0.001
6		0.767	0.736	0.790	0.002
7		0.385	0.374	0.341	< 0.001
8		0.773	0.547	1.165	0.005
9		67.933	66.542	66.175	2.191
1	6.6	2.400	1.433	1.420	0.005
2	4.5	0.783	0.389	0.395	0.001
3	7.3	3.443	3.434	3.511	0.026
4		0.725	0.860	0.602	0.002
5		0.126	0.171	0.118	0.001
6		1.142	1.655	1.278	0.005
7		0.952	1.046	0.636	0.002
8		2.184	1.683	2.706	0.012
9		77.846	95.525	105.444	4.088
1	10.6	2.397	1.734	1.722	0.005
2	7	0.674	0.441	0.447	0.001
3	11.8	3.805	4.022	4.143	0.026
4		0.737	0.988	0.664	0.002
5		0.126	0.241	0.124	0.001
6		1.223	2.065	1.484	0.007
7		0.964	1.337	0.739	0.002
8		2.160	2.067	3.453	0.014
9		84.607	113.473	127.225	5.263

We observe that for moderate arrival variability, the performance of our approach is comparable to that of the classical decomposition method. Both provide satisfactory estimates of the performance measures compared to (pure) simulation. Experiments show that improvements to the classical decomposition method are possible using our approach in many situations, particularly, under high arrival variability conditions. In the example above, node 9 is highly utilized and is the bottleneck. The hybrid procedure gives better approximation of the mean waiting time at this node as well as the overall cycle time.

5.4.2. Multiple-server nodes

Let's modify the previous network by changing the number of servers and arrivals as well as the service parameters. Table 7 shows the new input values. Results from simulation (SIM), Hybrid method (HM) and decomposition (QNA) are presented in Table 8.

Table 7
Input parameters for the 9 nodes network with multiple-server nodes.

Node	Arrival rate	Number of servers	Mean service time	Service SCV	ρ_i
1	0.52	3	3	4	0.520
2	0.195	4	4	4	0.195
3	0.585	3	2.8	8	0.546
4		4	10	2	0.816
5		3	2.5	1.5	0.082
6		2	2	1.5	0.722
7		3	2	2	0.465
8		3	2	2.5	0.438
9		2	4.4	2	0.965

Table 8
Comparison of the mean waiting time in the 9 multiple-server nodes queueing network.

Node	External interarrival SCV	QNA	HM	SIM	CI-hw
1	1	1.348	1.120	1.115	0.006
2	1	0.027	0.016	0.016	0.001
3	1	2.669	1.989	2.068	0.027
4		13.195	12.786	13.078	0.147
5		0.002	0.002	0.004	< 0.001
6		3.155	2.598	3.189	0.018
7		0.396	0.359	0.408	0.002
8		0.431	0.357	0.470	0.003
9		91.882	79.549	83.385	3.200
1	5.68	2.131	2.585	2.579	0.016
2	2.755	0.033	0.090	0.090	0.001
3	6.265	3.682	3.859	3.985	0.037
4		15.929	17.625	17.635	0.175
5		0.002	0.003	0.004	< 0.001
6		3.901	5.901	5.269	0.024
7		0.504	0.905	0.764	0.003
8		0.580	0.930	0.990	0.005
9		98.679	113.320	109.875	4.740
1	13.48	3.309	4.607	4.596	0.038
2	5.68	0.041	0.066	0.067	0.001
3	15.04	5.249	6.409	6.665	0.075
4		20.401	23.393	23.548	0.360
5		0.003	0.005	0.005	< 0.001
6		5.105	8.931	8.248	0.041
7		0.673	1.593	1.143	0.007
8		0.808	1.609	1.580	0.008
9		109.990	152.450	161.412	9.029

Again, it appears that the hybrid method performs better in estimating performance measures at congestion nodes.

6. Conclusion

The parametric decomposition method is a good tool for estimating the performance measures of non-product form open queueing networks. Although this approach has several attractive features, it is based on several fairly loose approximations and its output can significantly deviate from actual performance values. The approach suggested in this paper, attempts to improve the inter-departure SCV computation by replacing the analytical approximative relations by simulation estimates. This idea is motivated by the observation that existing (approximate) formulae for $G/G/m$ systems can fail in numerous situations. In addition, simulation of the individual stations allows us to consider real service distributions and multiple-server nodes rather than two-moments approximation. Numerical results show that our method compares positively to the classical decomposition method under moderate variability conditions and that improvements are made in other situations (in particular, under high arrival variability conditions). In addition, this procedure is faster than pure simulation of queueing networks and requires less computer resources.

Acknowledgement

The author thanks Gerald Reiner, Reinhold Schodl and anonymous referees for helpful comments and suggestions on earlier drafts of this paper.

References

[1] S.L. Albin, On Poisson approximations for superposition arrival processes in queues, Management Science 28 (1982) 126–137.
[2] S.L. Albin, Approximating a point process by a renewal process. II. Superposition arrival processes to queues, Operations Research 32 (1984) 1133–1162.
[3] C. Alexopoulos, D. Goldsman, R.F. Serfozo, Stationary processes: Statistical estimation, in: N. Balakrishnan, C. Read, B. Vidakovic (Eds.), Encyclopedia of Statistical Sciences, 2nd Edition, John Wiley & Sons, New York, 2006, pp. 7991–8006.
[5] G. Bitran, R. Morabito, Open queueing networks: Optimization and performance evaluation models for discrete manufacturing systems, Production and Operations Management 51 (1996) 163–193.
[6] G. Bitran, D. Tirupati, Multiproduct queueing networks with deterministic routing: Decomposition approach and the notion of interference, Management Science 34 (1988) 75–100.
[7] B. Blacio glu, D. Jagerman, T. Altiok, Merging and splitting autocorrelated arrival processes and impact on queueing performance, Performance Evaluation 65 (2008) 653–669.
[8] G. Bolch, S. Greiner, H. de Meer, K.S. Trivedi, Queueing networks and Markov chains, 2nd Ed., John Wiley & Sons, New Jersey, 2006.
[9] R. Caldentey, Approximations for multi-class departure processes, Queueing Systems 38 (2001) 205–212.
[10] M. Corvella, Performance characteristics of the World Wide Web, in: G. Haring, C. Lindemann, M. Reiser (Eds.), Performance evaluation: Origins and Directions, Springer-Verlag, London, 2000, pp. 219–232.
[11] F.R.B. Cruz, P.C. Oliveira, L. Duczmal, State-dependent stochastic mobility model in mobile communication networks, Simulation Modelling Practice and Theory 18 (2010) 348–365.
[12] F.R.B. Cruz, J.M. Smith, R.O. Medeiros, An *M*/*G*/*C*/*C* state-dependent network simulation model, Computers and Operations Research 32 (2005) 919–941.
[13] B.R. Haverkort, Approximate analysis of networks of *PH*/*PH*/1/*K* queues: theory & tool support, in: MMB'95: Proceedings of the 8th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Springer-Verlag, London, 1995, pp. 239–253.
[14] A. Heindl, M. Telek, Output models of *MAP*/*PH*/1(/*K*) queues for an efficient network decomposition, Performance Evaluation 49 (2002) 321–339.
[15] A. Heindl, K. Mitchell, A. van de Liefvoort, Correlation bounds for second-order maps with application to queueing network decomposition, Performance Evaluation 63 (2006) 553–577.
[16] S. Kim, The heavy-traffic bottleneck phenomenon under splitting and superposition, European Journal of Operational Research 157 (2004) 736–745.
[17] S. Kim, Approximation of multiclass queueing networks with highly variable arrivals under deterministic routing, Naval Research Logistics 52 (2005) 399–408.
[18] S. Kim, R. Muralidharan, C.A. O'Cinneide, Taking account of correlation between streams in queueing network approximations, Queueing Systems 49 (2005) 261–281.
[19] T. Kimura, Approximating the mean waiting time in the *GI*/*G*/*s* queue, Journal of the Operational Research Society 42 (1991) 959–970.
[21] W. Kraemer, M. Langenbach-Belz, Approximate formulae for the delay in the queueing system *GI*/*G*/1, in: Proceedings of the 8th International Teletraffic Congress, Melbourne, 1976, pp. 1–8.
[22] N.K. Krivulin, A recursive equations based representation for the *G*/*G*/*m* queue, Applied Mathematics Letters 7 (1994) 73–78.
[23] P.J. Kuehn, Approximate analysis of general networks by decomposition, IEEE Transactions on Communications 27 (1979) 113–126.
[24] A.M. Law, W.D. Kelton, Simulation Modeling and Analysis, 3rd ed., McGraw-Hill, New York, 2000.
[25] K.T. Marshall, Some inequalities in queueing, Operations Research 16 (1968) 651–665.
[26] B. Rabta, A review of decomposition methods for open queueing networks, in: G. Reiner (Ed.), Rapid Modeling for Integrated Demand and Supply Management, Springer-Verlag, London, 2009, pp. 25–42.
[27] B. Rabta, A. Alp, G. Reiner, Queueing networks modelling software for manufacturing, in: G. Reiner (Ed.), Rapid Modeling for Integrated Demand and Supply Management, Springer-Verlag, London, 2009, pp. 15–23.
[28] H. Sakasegawa, An approximation formula $L_q \approx \alpha\rho^{\beta}/(l-\rho)$, Annals of the Institute of Statistical Mathematics A 29 (1977) 67–75.
[29] K.C. Sevcik, A.I. Levy, S.K. Tripathi, J.L. Zahorjan, Improving approximations of aggregated queueing network systems, in: K. Chandy, M. Reiser (Eds.), Computer Performance, North-Holland, New York, 1977, pp. 1–22.
[30] J.G. Shanthikumar, R.G. Sargent, A unifying view of hybrid simulation/analytic models and modeling, Operations Research 31 (1983) 1030–1052.
[31] J.M. Smith, F.R.B. Cruz, T. van Woensel, Topological network design of general, finite, multi-server queueing networks, European Journal of Operational Research 201 (2010) 427–441.
[32] S. Suresh, W. Whitt, The heavy-traffic bottleneck phenomenon in open queueing networks, Operations Research Letters 9 (1990) 355–362.
[33] P.L.M. Van Nyen, H.P.G. Van Ooijen, J.W.M. Bertrand, Simulation results on the performance of Albin and Whitt's estimation method for waiting times in integrated production-inventory systems, International Journal of Production Economics 90 (2004) 237–249.
[34] W. Whitt, The queueing network analyzer, Bell System Technical Journal 62 (1983) 2779–2815.
[35] W. Whitt, Performance of the queueing network analyzer, Bell System Technical Journal 62 (1983) 2817–2843.
[36] W. Whitt, Towards better multi-class parametric-decomposition approximations for open queueing networks, Annals of Operations Research 48 (1994) 221–248.
[37] W. Whitt, Variability functions for parametric-decomposition approximations of queueing networks, Management Science 41 (1995) 1704–1715.