Operations Research Letters 45 (2017) 581–584

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

Point of queue size change analysis of the PH/PH/k system with heterogeneous servers

Attahiru Sule Alfa [a,b,*], Qi-Ming He [c]

[a] Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, Canada R3T 5V6
[b] Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Pretoria, South Africa
[c] Department of Management Sciences, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1

ARTICLE INFO

Article history:
Received 2 March 2017
Received in revised form 8 September 2017
Accepted 14 September 2017
Available online 22 September 2017

Keywords:
PH/PH/k system
Point of queue size change analysis
Continuous time and discrete time

ABSTRACT

By observing the points only when the queue size changes, we study the PH/PH/k system with heterogeneous servers. We develop Markov chain set ups that are more efficient than studying the systems at arbitrary times, by reducing the sizes of matrices that need to be computed. Specifically we present procedures for constructing the associated Markov chains so one may use matrix-analytic methods for their analysis. This work is carried out for both the continuous and discrete time cases.

1. Introduction

Multi-server queueing systems are natural systems that occur in real life. An example is in telecommunication systems where the servers are communication channels which are usually not identical. Even though such queueing systems can be set up and analyzed by observing them at arbitrary times this leads to huge matrices which are often inefficient to compute and use in the system analysis. When the servers are identical the method of studying the system by considering the number of servers in each phase makes it easier to analyze (see [3,5,9,10]). However when the servers are not identical the methods in those papers cannot be used. Trying to study the systems by the traditional approaches of studying the systems at arbitrary time points would lead to huge block matrices in the associated Markov chains. If, however, we study the systems at points of queue size changes only, a form of embedded system, then we can reduce the block sizes of the associated matrices. The performance measures at arrival (or departure) time points can be obtained from that of the points of queue size changes. Nonetheless, constructing the Markov chains and obtaining the block matrices is a challenging procedure. In this paper we show how to construct these matrices for both the continuous time and discrete time cases. Latouche and Ramaswami [6] presented a method for analyzing the continuous time PH/PH/1 system at points of queue size changes. Their work is a special case with a single server, and they showed that the method helps to reduce the matrix needed for computation to be of size $(m + n) \times (m + n)$, from $mn \times mn$, where n and m are the dimensions associated with the PH-distributions for the interarrival and service times, respectively. The discrete time case of PH/PH/1 was presented in [1].

2. The continuous time model

We consider the continuous time PH/PH/k queue in this section. In Section 2.1, we define the PH/PH/k queue and an embedded Markov chain at the points of events. In Section 2.2, we construct the block matrices in the transition matrix of the embedded Markov chain. A brief analysis of the queue length and waiting time is presented in Section 2.3.

2.1. PH/PH/k system at points of events

Let the arrival process be PH with representation $(\boldsymbol{\beta}_0, S_0)$ of order m_0. There are k servers and the service time of server r is of the PH type with representation $(\boldsymbol{\beta}_r, S_r)$ of order m_r, $r = 1, 2, \ldots, k$. If an arriving customer finds multiple servers available, the customer chooses the server with the smallest index for service. We note that models with other Markovian rules for server selection can be treated with minor modifications of the analysis in this paper.

Let $q(t)$ be the number of customers in the system at time t, $I_0(t)$ be the phase of the PH arrival process at time t, and $I_r(t)$ be the phase of the service process of server r at time t, for $r = 1, 2, \ldots, k$. We note that $I_r(t) \in \{1, 2, \ldots, m_r\}$, for $r = 0, 1, 2, \ldots, k$. For the

* Corresponding author at: Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, Canada R3T 5V6.
E-mail addresses: attahiru.alfa@umanitoba.ca (A.S. Alfa), q7he@uwaterloo.ca (Q.-M. He).

https://doi.org/10.1016/j.orl.2017.09.004
0167-6377/

PH/PH/k queue, there are $k+1$ points of queue size changes, which are denoted as $\{0, 1, 2, \ldots, k\}$. We define

- Points of arrivals (a) $\Rightarrow 0$
- Points of service (r) completions $\Rightarrow r$, for $r = 1, 2, \ldots, k$.

Let τ_n be the nth event epoch. Define $q_n = q(\tau_n+)$ and $I_{n,r} = I_r(\tau_n+)$ for $r = 0, 1, 2, \ldots, k$. It is easy to see that the embedded process $\{(q_n, I_{n,0}, \ldots, I_{n,k}),\ n = 0, 1, 2, \ldots\}$ is a discrete time Markov chain. Since the queue length can only increase or decrease by one at the event epochs, the Markov chain is a QBD process with transition probability matrix

$$P = \begin{bmatrix} \mathbf{0} & A_{0,1} & & & & & & & \\ A_{1,0} & \mathbf{0} & A_{1,2} & & & & & & \\ & \ddots & \ddots & \ddots & & & & & \\ & & A_{k-1,k-2} & \mathbf{0} & A_{k-1,k} & & & & \\ & & & A_{k,k-1} & \mathbf{0} & A_0 & & & \\ & & & & A_2 & \mathbf{0} & A_0 & & \\ & & & & & A_2 & \mathbf{0} & A_0 & \\ & & & & & & \ddots & \ddots & \ddots \end{bmatrix}. \quad (2.1)$$

Next, we construct the block matrices in P.

2.2. Computing the block matrices

We begin with matrices $\{A_0, A_2\}$ for queue length equal to k or bigger. For given $q_n \geq k$, the states of $(I_{n,0}, I_{n,1}, \ldots, I_{n,k})$ are given by $\Omega_k = \Delta_0 \cup \Delta_1 \cup \cdots \cup \Delta_k$, where

$$\Delta_r = \{(i_0, i_1, \ldots, i_r) : 1 \leq i_v \leq m_v,\ v = 0, 1, \ldots, k,\ v \neq r, i_r = 0\},$$

for $r = 0, 1, 2, \ldots, k$. Transitions in P from subset Δ_r to subset Δ_v mean that the last event was of type r and the current event is of type v. Let B_{rv} be the transition block matrix from Δ_r to Δ_v. Then we have

$$A_0 = \begin{bmatrix} B_{00} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ B_{10} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ B_{20} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ B_{k0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix},$$

$$A_2 = \begin{bmatrix} \mathbf{0} & B_{01} & B_{02} & \cdots & B_{0k} \\ \mathbf{0} & B_{11} & A_{12} & \cdots & B_{1k} \\ \mathbf{0} & B_{21} & B_{22} & \cdots & B_{2k} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \mathbf{0} & B_{k1} & B_{k2} & \cdots & B_{kk} \end{bmatrix}. \quad (2.2)$$

Let $U_r(x) = exp(S_r x)$, for $r = 0, 1, \ldots, k$. Then the block matrices in A_0 and A_2 are given as, for $r, v = 0, 1, 2, \ldots, k$,

$$B_{rv} = \int_0^\infty U_0(x) \otimes \cdots \otimes \boldsymbol{\beta}_r U_r(x) \otimes \cdots \otimes U_v(x)\mathbf{s}_v \otimes \cdots \otimes U_r(x)dx,$$

where $\mathbf{s}_v = -S_v\mathbf{1}$ and $\mathbf{1}$ is the column vector with all elements being one and $\otimes$ is the Kronecker product. The argument behind this is as follows. The block matrix B_{rv} represents a transition in which at time 0 an event r occurred and it is re-started and the next event that occurred at time interval $(x, x+\delta x)$ is v. Hence we have $\boldsymbol{\beta}_r U_r(x)$ capturing the first part, $U_v(x)\mathbf{s}_v$ capturing the second one and $U_l(x)$, $l = 0, 1, \ldots, K$; $l \neq r, l \neq v$ capturing the other events that did not occur yet by time x.

Taking advantage of the fact that $exp(A) \otimes exp(B) = exp(A \oplus B)$, we obtain a compact form of those block matrices as, for $r, v = 0, 1, 2, \ldots, k$,

$$B_{rv} = -(I \otimes \boldsymbol{\beta}_r \otimes I)(S_0 \oplus S_1 \oplus \cdots \oplus S_k)^{-1}(I \otimes \mathbf{s}_v \otimes I),$$

where $\oplus$ is the Kronecker sum.

This compact form is efficient for computations when the sizes of $m_i,\ i = 0, 1, 2, \ldots, k$ are small. Using the conventional approach, the number of states in a level is $\prod_{j=0}^k m_j$. The number of states in a level for the above approach is $\left(\prod_{j=0}^k m_j\right)\left(\sum_{j=0}^k \frac{1}{m_j}\right)$. The ratio of the two numbers is $\sum_{j=0}^k \frac{1}{m_j}$. Thus, if $\sum_{j=0}^k \frac{1}{m_j} < 1$, the above new approach has a smaller state space. However, if $\sum_{j=0}^k \frac{1}{m_j} > 1$, the conventional approach actually has a smaller state space. For sufficiently large $m_j,\ j = 0, 1, 2, \ldots, k$, there is a reasonable saving.

If all the servers are identical with same PH service distributions the system can actually be reduced to the PH/PH/1 case. We can then use the CSFP (count-server-for-phase) to reduce the state space significantly. For more details on CSFP see [3].

Now, we construct the transition block matrices for $q_n \leq k-1$. For given $q_n = m < k$, there are only m servers working, which are indexed by $(i_1, i_2, \ldots, i_m)$. Let $\Phi_m = \{(\xi_1, \ldots, \xi_m) :\ 1 \leq \xi_1 < \cdots < \xi_m \leq k\}$, which gives all the possible combinations of servers that are in service. For given servers in service $(\xi_1, \ldots, \xi_m)$, the possible states of the service process is given by

$$\Delta^{(\xi_1,\ldots,\xi_m)} = \left\{(i_{\xi_1}, \ldots, i_{\xi_m}) :\ i_{\xi_v} = 1, 2, \ldots, m_{\xi_v},\ v = 1, 2, \ldots, m\right\}.$$

We define $\Delta^{(0,\xi_1,\ldots,\xi_m)}$ similarly with $i_0 = 1, 2, \ldots, m_0$. Then the set of all states associated with $q_n = m$ can be written as, for $m = 1, 2, \ldots, k$,

$$\Omega_k(m) = \bigcup_{(\xi_1,\ldots\xi_m)\in\Phi_m} \left(\Delta^{(0,\xi_1,\ldots,\xi_m)} \cup \Delta^{(\xi_1,\ldots,\xi_m)}\right).$$

For $m = 0$, $\Omega_k(0) = \Delta^{(0)} = \{1, 2, \ldots, m_0\}$. Since the k servers can be different, the state space can be big. If servers are identical, the state space can be reduced significantly.

By their definitions, it is clear that $A_{m,m+1}$ contains the transition probabilities from $\Omega_k(m)$ to $\Omega_k(m+1)$ and $A_{m,m-1}$ from $\Omega_k(m)$ to $\Omega_k(m-1)$. Next, we construct the transition block matrices within the two matrices.

The transition block matrices within $A_{m,m-1}$ are $B_{rv}^{(\xi_1,\ldots,\xi_m)}$ for $(\xi_1, \ldots, \xi_m) \in \Phi(m)$ and $v \in \{\xi_1, \ldots, \xi_m\}$ or $v = 0$.

- First, if the last event is the completion of a service, we have, for $m < k$,

$$B_{*v}^{(\xi_1,\ldots,\xi_m)} = -\left(S_0 \oplus \bigoplus_{j=1}^m S_{\xi_j}\right)^{-1}(I \otimes \mathbf{s}_v \otimes I),$$

  which contains the transition probabilities from $\Delta^{(0,\xi_1,\ldots,\xi_m)}$ to $\Delta^{(0,\{\xi_1,\ldots,\xi_m\}\setminus\{v\})}$, where $\{\xi_1, \ldots, \xi_m\} \setminus \{v\}$ means deleting v from the set. Since there is no waiting customer as $m < k$, the last service completion has no impact on the state space and the queueing process. That is why we use "*" for the last service completion event. For $m = k$, the transition block matrix from Δ_k to $\Delta^{(0,\{1,2,\ldots,k\}\setminus\{v\})}$ is B_{rv}.
- Second, if the last event is the arrival of a customer, we have

$$B_{0v}^{(\xi_1,\ldots,\xi_m)} = -(\boldsymbol{\beta}_0 \otimes I)\left(S_0 \oplus \bigoplus_{j=1}^m S_{\xi_j}\right)^{-1}(I \otimes \mathbf{s}_v \otimes I),$$

  which contains the transition probabilities from $\Delta^{(\xi_1,\ldots,\xi_m)}$ to $\Delta^{(0,\{\xi_1,\ldots,\xi_m\}\setminus\{v\})}$.

The transition block matrices within $A_{m,m+1}$ can be given as follows. For given $\{\xi_1, \ldots, \xi_m\}$, define $\xi = \min\{r :\ 1 \leq r \leq k, r \notin \{\xi_1, \ldots, \xi_m\}\}$. Recall that we assume that arriving customer selects the available server with the smallest index for service. Thus, ξ is the index of the server chosen by an arriving customer.

- If the last event is the completion of a service, we have

$$B_{*0}^{(\xi_1,\ldots,\xi_m)} = -\left(S_0 \oplus \bigoplus_{j=1}^{m} S_{\xi_j}\right)^{-1} \left(\mathbf{s}_0 \otimes I \otimes \boldsymbol{\beta}_\xi \otimes I\right),$$

  which contains the transition probabilities from $\Delta^{(0,\xi_1,\ldots,\xi_m)}$ to $\Delta^{\overline{(\xi_1,\ldots,\xi_m,\xi)}}$, where $\overline{(\xi_1,\ldots,\xi_m,\xi)}$ means we rearrange elements in the vector in ascending order.
- If the last event is the arrival of a customer, we have

$$B_{00}^{(\xi_1,\ldots,\xi_m)} = -(\boldsymbol{\beta}_0 \otimes I)\left(S_0 \oplus \bigoplus_{j=1}^{m} S_{\xi_j}\right)^{-1} \left(\mathbf{s}_0 \otimes \boldsymbol{\beta}_\xi \otimes I\right), \quad (2.3)$$

  which contains the transition probabilities from $\Delta^{(\xi_1,\ldots,\xi_m)}$ to $\Delta^{(\overline{\xi_1,\ldots,\xi_m,\xi})}$.

Example. For the PH/PH/2 queue, we have $\Omega_2(0) = \Delta^{(0)}$, $\Omega_2(1) = \Delta^{(1)} \cup \Delta^{(0,1)} \cup \Delta^{(0,2)}$, and $\Omega_2(2) = \Omega_2 = \Delta_0 \cup \Delta_1 \cup \Delta_2$. Note that we do not have the set $\Delta^{(2)}$ since an arriving customer selects the available server with the smallest index for service. The transition block matrices are given as

$$A_{0,1} = [B_{*0}^{(0)},\ \mathbf{0},\ \mathbf{0}];\quad A_{1,2} = \begin{bmatrix} B_{00}^{(1)} & \mathbf{0} & \mathbf{0} \\ B_{*0}^{(0,1)} & \mathbf{0} & \mathbf{0} \\ B_{*0}^{(0,2)} & \mathbf{0} & \mathbf{0} \end{bmatrix};$$

$$A_{1,0} = \begin{bmatrix} B_{01}^{(1)} \\ B_{*1}^{(1)} \\ B_{*2}^{(2)} \end{bmatrix};\quad A_{2,1} = \begin{array}{c} \\ \Delta_0 \\ \Delta_1 \\ \Delta_2 \end{array} \begin{array}{c} \begin{array}{ccc} \Delta^{(1)} & \Delta^{(0,1)} & \Delta^{(0,2)} \end{array} \\ \begin{bmatrix} \mathbf{0} & B_{02}^{(1,2)} & B_{01}^{(1,2)} \\ \mathbf{0} & B_{12} & B_{11} \\ \mathbf{0} & B_{22} & B_{21} \end{bmatrix} \end{array}.$$

Transition blocks A_0 and A_2 are given in Eq. (2.2).

2.3. Queueing analysis

Analysis of the above Markov chain can lead to results on the queue length at arrival (or departure) time points. Let $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots)$ be the limiting probabilities of P. Let $\mathbf{p}_a = (p_{a,0}, p_{a,1}, p_{a,2}, \ldots)$ be the limiting probabilities of the queue length just before the arrivals of customers. Then we have

$$p_{a,n} = \frac{1}{\lambda_0}\boldsymbol{\pi}_n \begin{bmatrix} B_{00} \\ B_{10} \\ \vdots \\ B_{k0} \end{bmatrix},\quad \text{for } n = k, k+1, k+2, \ldots,$$

where λ_0 is the arrival rate of customers. Similarly, we can find the limiting probabilities of the queue length right after the departures of customers.

Note that the limiting probabilities can be computed using the matrix-analytic methods. For details see [2,7,8].

In order to obtain the waiting time distribution we can employ the well known absorbing Markov chain approach presented in several matrix analytic methods literature. The absorbing Markov chain can be created from the transition matrix equation (2.1). The limiting probabilities are then used to initiate the absorbing Markov chain.

3. The discrete time system model

3.1. The discrete time PH/PH/k

Let us now consider the case of the discrete PH/PH/k. This is more involved than its continuous time counterpart. Let us start by considering the number of possible event occurrences. With k servers the number of event occurrences, N_k can be written as

$$N_k = \sum_{j=1}^{k+1} \binom{k+1}{j}.$$

This can be written as

$$N_k = N_{k-1} + 2^k,\ k \geq 1,\ \text{ with } N_0 = 1.$$

The number of block matrices that have to be determined is $M_k = (N_k)^2$ for a k server problem. For example for the case of $k = 2$ we have $N_2 = 7$ and $M_2 = 49$. The seven points of events are given as

1. at arrivals: (a)
2. at service completions by server 1: (s_1)
3. at service completions by server 2: (s_2)
4. at service completions by server 1 and server 2: (s_1, s_2)
5. at points of joint arrival and service completion by server 1: (a, s_1)
6. at points of joint arrival and service completion by server 2: (a, s_2)
7. at points of joint arrival and service completion by server 1 and by server 2: (a, s_1, s_2).

The resulting Markov chain for the PH/PH/k is a GI/M/1 type with k sub-diagonals. Now we consider the block matrices of this GI/M/1 type Markov chain. Once again, let the PH-representations of the k services be $(\boldsymbol{\beta}_i, S_i),\ i = 1, 2, \ldots, k$. Further let $(\boldsymbol{\beta}_0, S_0)$ be the PH-representation of the interarrival times. Let

$$\Omega = \{\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_k) :\ \sigma_i = 0 \text{ or } 1, i = 0, 1, 2, \ldots, k\}.$$

Define $\mathbf{c}_\sigma = (c(0, \sigma_0), c(1, \sigma_1), \ldots, c(k, \sigma_k))$, where

$$c(i, \sigma_i) = I,\ \text{if } \sigma_i = 0;\ \boldsymbol{\beta}_i,\ \text{if } \sigma_i = 1.$$

Define $\mathbf{d}_\sigma = (d(0, \sigma_0), d(1, \sigma_1), \ldots, d(k, \sigma_k))$, where

$$d(i, \sigma_i) = S_i,\ \text{if } \sigma_i = 0;\ \mathbf{s}_i^0,\ \text{if } \sigma_i = 1.$$

Define $A = (A_{\sigma^{(1)},\sigma^{(2)}})$, where $\sigma^{(1)}, \sigma^{(2)} \in \Omega$, and

$$A_{\sigma^{(1)},\sigma^{(2)}} = \mathbf{c}_{\sigma^{(1)}}(I - S_0 \otimes S_1 \otimes \cdots \otimes S_k)^{-1}\mathbf{d}_{\sigma^{(2)}}.$$

Decompose A as $A = A_0 + A_1 + A_2 + \cdots + A_k$ such that A_i contains all the blocks $A_{\sigma^{(1)},\sigma^{(2)}}$ satisfying $\sum_{j=1}^{k}\sigma_j^{(2)} - \sigma_0^{(2)} + 1 = i$, for $i = 0, 1, 2, \ldots, k$. In this way, we find all the $\{A_0, A_1, A_2, \ldots, A_k\}$. Transition block matrices for boundary states can be found in a similar way.

Remark. Consider the conventional method that keeps track of the phases, i.e. $(I_a(t), I_1(t), I_2(t), \ldots, I_k(t))$, where $I_a(t)$ refers to arrival phase at time t, and $I_j(t),\ j = 1, 2, \ldots, k$ refer to the phase of service of server j at time t. If we use that approach the number of phases is $\prod_{j=0}^{k} m_j$. The number of states of the new approach that we are proposing is $\prod_{j=0}^{k}(m_j+1) - \prod_{j=0}^{k} m_j$. Thus, the new approach has a smaller state space if $\prod_{j=0}^{k}(1 + 1/m_j) < 2$.

3.2. Feasibility of this approach for the discrete time case

For the discrete PH/PH/k the advantage of studying the system at points of events may be questionable. The growth in N_k is exponential, so is the growth in M_k. We have $N_k = N_{k-1} + 2^k,\ k \geq 1$, hence generating the transition matrix in this case could be very involved. Once all the M_k matrices are determined generating their entries $A_{i_0,i_1,\ldots,i_k;j_0,j_i,\ldots,j_k}$ is straightforward since we know that

$$A_{i_0,i_1,\ldots,i_k;j_0,j_i,\ldots,j_k} = c(I - S_0 \otimes S_1 \otimes S_2 \otimes \cdots \otimes S_k)^{-1}d,$$

and the values of c and d can be determined easily.

The reduction in matrix sizes only goes from $O(n^{k+1})$ to $O(n^k)$ when $m_j = n,\ \forall j$, if we use this approach.

4. Conclusion

Whereas there is a considerable saving in computational effort by studying the PH/PH/k system with heterogeneous servers in continuous time according to the points of events, there may not be much savings, if any at all, for its discrete time counterparts. This is because the growth in the number of events occurring increases considerably as k increases as well as the number of matrices, while the reduction of the block matrix sizes goes from $O(n^{k+1})$ to $O(n^k)$. We might be better off using the traditional approach. And if the servers are identical then, of course, one may use the CSFP approach, presented in [4], to reduce the state space in both cases.

Acknowledgments

The authors acknowledge funding from the NSERC (Canada) and the NRF (South Africa) SARChI Chair programme cohosted by UP and CSIR.

References

[1] A.S. Alfa, Applied Discrete-Time Queues, second ed., Springer, 2015.
[2] Q.-M. He, Fundamentals of Matrix-Analytic Methods, Springer, New York, 2013.
[3] Q.-M. He, A.S. Alfa, Construction of Markov chains for discrete time MAP/PH/K queues, Perform. Eval. 93 (2015) 17–26.
[4] Q.-M. He, A.S. Alfa, Space reduction for a class of multidimensional Markov chains: A summary and some applications, INFORMS J. Comput. (2017) in press.
[5] Q.-M. He, H. Zhang, Q.Q. Ye, An M/PH/K queue with constant impatient time, Math. Methods Oper. Res. (2017) accepted.
[6] G. Latouche, V. Ramaswami, The PH/PH/1 queue at epochs of queue size changes, Queueing Syst. 25 (1997) 97–114.
[7] G. Latouche, V. Ramaswami, Introduction to Matrix Analytic Methods in Stochastic Modelling, in: SIAM-ASA Series on Statistics and Applied Probability, SIAM, 1999.
[8] M.F. Neuts, Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach, The Johns Hopkins University Press, Baltimore, MD, 1981.
[9] V. Ramaswami, Independent Markov proces in parallel, Stoch. Models 1 (1985) 419–430.
[10] V. Ramaswami, D.M. Lucantoni, Algorithms for the multi-server queue with phase type service, Stoch. Models 1 (1985) 393–417.