Busy Periods for a Matched Queueing Network *

Guang-Hui Hsu
Institute of Applied Mathematics Chinese Academy of Sciences
Beijing 100080, China. Asian-Pacific Operations Research Center
within CAS and APORS

Cheng-Xuan Cao
School of Management, University of Science and Technology Beijing
Beijing 100083, China

Uwe Jensen
Institute of Applied Mathematics and Statistics, University of
Stuttgart-Hohenheim, W-700 Stuttgart 70, Germany

Abstract: We consider the matched queueing network $PH/M/c \rightarrow \circ PH/PH/1$. The probability distributions of busy periods and non-idle periods for the two subsystems and the whole network are studied and their algorithms with uniform error are derived. It is proved that both the time and space complexities of the algorithms are polynomially bounded. At last, a numerical example is presented.

Key words: Matched queueing network; PH-distribution; busy period; non-idle period; algorithm; uniform error; computational complexity.

1 Introduction

Many authors studied matched queueing systems or queueing systems with paired customers, such as Chen [1], Hsu and He [2, 3], Hsu, He and Liu [4, 5], Latouche [9], Neuts [10], and Yang and Chen [11]. Such systems often appear in manufacturing, transportation, seaports, communications, etc.. Later on, Hsu and Jensen [6] studied a matched queueing network $PH/M/c \rightarrow \circ PH/PH/1$ and derived the stationary state probabilities and the distributions of virtual waiting times in the two subsystems and the total virtual waiting times in the network. But the problems of busy periods and non-idle periods

---

[*] Supported by the National Natural Science Foundation of China

for this network remain unsolved. In order to solve these computational problems, we were stimulated to study the first passage times for denumerable state Markov processes, and eventually obtained the algorithms for one-dimensional denumerable state Markov processes, see Hsu and Yuan [8]. These results seems to be able to solve the problems of busy periods and non-idle periods for the first subsystem and non-idle periods for the whole network, but obviously cannot solve the problems of busy periods and non-idle periods for the second subsystem and busy periods for the whole network. Therefore, we further studied the first passage times for multidimensional denumerable state Markov processes and finally obtained their efficient algorithms too, see Hsu and Xu [7]. On the basis of these results, we intended to solve all the problems of busy periods and non-idle periods for the matched queueing network $PH/M/c \to \circ PH/PH/1$, but unfortunately found that both the time and space complexities of these algorithms were exponential. We then tried to improve the original algorithms and derived the improved ones for this network. It is prove that both the time and space complexities of these improved algorithms are polynomially founded. The idea and approach for deriving such kinds of improved algorithms can be widely applied to the computational problems of other queueing networks and stochastic models.

To keep the appropriate length of the paper, we present only the results of busy periods for the whole network and omit those of all other indices mentioned above. In the following, we first cite the Markov process describing the matched queueing network $PH/M/c \to \circ PH/PH/1$ in Section 2. Section 3 deals with the busy period of the whole network. After the curse of dimensionality of the original algorithms for this network is explored, we derive the improved algorithm and prove that it is polynomially bounded. A numerical example is presented in the last section.

2 Matched Queueing Network $PH/M/c \to \circ PH/PH/1$

In the matched queueing network $PH/M/c \to \circ PH/PH/1$, the output of the first subsystem $PH/M/c$ is regarded as one of the input of the second subsystem and matched with another input of the PH-renewal process, and they are served by a single server with PH-service-distribution. We suppose that

1) The PH-distributions for the input of the first subsystem (called type-1 customers), the input of the second subsystem (called type-2 customers) and the service time in the second subsystem are $(\alpha,\ S)$ of order m, $(\beta,\ T)$ of order n and $(\gamma,\ U)$ of order l respectively, where $\alpha\mathbf{1}_m = 1$, $\beta\mathbf{1}_n = 1$, $\gamma\mathbf{1}_l = 1$ and $\mathbf{1}_k$ is the column unit vector of dimension k. Let $S^0 \equiv -S\mathbf{1}_m$, $T^0 \equiv -T\mathbf{1}_n$ and $U^0 \equiv -U\mathbf{1}_l$.

2) The mean service rate of each server in the first subsystem is μ.

3) The matched ratio of type-1 and type-2 customers in the second subsystem is $1 : r$.

4) The number of type-1 customers in the first subsystem is of no limitation, and

the number of type-1 and type-2 customers in the second subsystem have the maximums K_1 and K_2 respectively. Without loss of generality, we assume that $K_2 \geq 2r$.

The state process of the matched queueing network is a Markov process and its states

$$\{(i,j,k,\delta_1,\delta_2,\delta_3):\ i \geq 0, 0 \leq j \leq K_1, 0 \leq k \leq K_2, 1 \leq \delta_1 \leq m, 1 \leq \delta_2 \leq n, 0 \leq \delta_3 \leq l\}$$

are arranged in lexicographic order, where i and j are the numbers of type-1 customers in the first and second subsystems respectively, k is the number of type-2 customers in the second subsystem, δ_1 and δ_2 are the phases of type-1 and type-2 inputs respectively, and δ_3 is the service phase in the second subsystem if the server is working; otherwise $\delta_3 = 0$. The number of states for each level i $(i = 0, 1, \cdots)$ is $f + K_1 g \equiv h$, where $f \equiv (K_2+1)mn$ and $g \equiv rmn + (K_2 - r + 1)mnl$. The infinitesimal generator of this Markov process is given by (see Hsu and Jensen [6])

$$Q = \begin{pmatrix} H_0 & B_0 & & & & & \\ F_1 & H_1 & B_0 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & F_{c-1} & H_{c-1} & B_0 & & \\ & & & B_2 & B_1 & B_0 & \\ & & & & B_2 & B_1 & B_0 \\ & & & & & \ddots & \ddots & \ddots \end{pmatrix}, \tag{1.1}$$

where

$$H_d = \begin{pmatrix} C_{00} - d\mu I_f & & & & & \\ C_{10} & C_{11} - d\mu I_g & & & & \\ & C_{21} & C_{11} - d\mu I_g & & & \\ & & \ddots & \ddots & & \\ & & & C_{21} & C_{11} - d\mu I_g & \\ & & & & C_{21} & C_{11} \end{pmatrix}_{h \times h},$$

$0 \leq d \leq c$,

$$B_0 = \begin{pmatrix} C_{01} & & & \\ & C_{12} & & \\ & & \ddots & \\ & & & C_{12} \end{pmatrix}_{h \times h}, \quad F_d = \begin{pmatrix} 0_f & E_d & & & \\ & & d\mu I_g & & \\ & & & \ddots & \\ & & & & d\mu I_g \\ & & & & 0_g \end{pmatrix}_{h \times h},$$

$1 \leq d \leq c, \quad B_2 \equiv F_c, \quad B_1 \equiv H_c,$

$$C_{00} = \begin{pmatrix} I_m \otimes T + S \otimes I_n & & I_m \otimes T^0\beta & \\ & \ddots & & \ddots \\ & & I_m \otimes T + S \otimes I_n & I_m \otimes T^0\beta \\ & & & S \otimes I_n \end{pmatrix}_{f \times f},$$

$$C_{01} = \begin{pmatrix} S^0\alpha \otimes I_n & & \\ & \ddots & \\ & & S^0\alpha \otimes I_n \end{pmatrix}_{f\times f},$$

$$C_{10} = \begin{array}{r} r\left\{\vphantom{\begin{matrix}0_{mn}\\ \vdots\\ 0_{mn}\end{matrix}}\right. \\ \\ \\ \\ \end{array} \begin{pmatrix} 0_{mn} & & & \\ \vdots & & & \\ 0_{mn} & & & \\ I_{mn}\otimes U^0 & & & \\ & \ddots & & \\ & & I_{mn}\otimes U^0 & \underbrace{0_{mnl\times mn}\dots 0_{mnl\times mn}}_{r} \end{pmatrix}_{g\times f},$$

$$C_{11} = \begin{array}{r} r\left\{\vphantom{\begin{matrix}I\\ \ddots\\ \ddots\end{matrix}}\right. \\ \\ \\ \\ \\ \\ \end{array} \begin{pmatrix} I_m\otimes T+S\otimes I_n & I_m\otimes T^0\beta & & & & & \\ \ddots & \ddots & & & & & \\ & \ddots & I_m\otimes T^0\beta & & & & \\ & & I_m\otimes T+S\otimes I_n & I_m\otimes T^0\beta\otimes\gamma & & & \\ & & & G & I_m\otimes T^0\beta\otimes I_l & & \\ & & & & \ddots & \ddots & \\ & & & & & G & I_m\otimes T^0\beta\otimes I_l \\ & & & & & & I_{mn}\otimes U+S\otimes I_{nl} \end{pmatrix}_{g\times g},$$

$$G \equiv I_{mn}\otimes U + I_m\otimes T\otimes I_l + S\otimes I_{nl},$$

$$C_{12} = \begin{pmatrix} S^0\alpha\otimes I_n & & & & & \\ & \ddots & & & & \\ & & S^0\alpha\otimes I_n & & & \\ & & & S^0\alpha\otimes I_{nl} & & \\ & & & & \ddots & \\ & & & & & S^0\alpha\otimes I_{nl} \end{pmatrix}_{g\times g},$$

$$C_{21} = \begin{array}{r} r\left\{\vphantom{\begin{matrix}0_{mn}\\ \vdots\\ 0_{mn}\end{matrix}}\right. \\ r\left\{\vphantom{\begin{matrix}I\\ \ddots\\ I\end{matrix}}\right. \\ \\ \\ \end{array} \begin{pmatrix} 0_{mn} & & & & & & \\ \vdots & & & & & & \\ 0_{mn} & & & & & & \\ I_{mn}\otimes U^0 & & & & & & \\ & \ddots & & & & & \\ & & I_{mn}\otimes U^0 & & & & \\ & & & I_{mn}\otimes U^0\gamma & & & \\ & & & & \ddots & & \\ & & & & & I_{mn}\otimes U^0\gamma & \underbrace{0_{mnl}\dots 0_{mnl}}_{r} \end{pmatrix}_{g\times g},$$

$$E_d = \begin{array}{l} r\left\{\begin{matrix} \\ \\ \end{matrix}\right. \\ \\ \\ \\ \end{array} \begin{pmatrix} d\mu I_{mn} & & & & & \\ & \ddots & & & & \\ & & d\mu I_{mn} & & & \\ & & & d\mu I_{mn}\otimes\gamma & & \\ & & & & \ddots & \\ & & & & & d\mu I_{mn}\otimes\gamma \end{pmatrix}_{f\times g}, \ 1\le d\le c\ ,$$

I_d is the unit matrix of order d, $0_{d\times b}$ is the zero matrix of dimensions $d\times b$, $0_d\equiv 0_{d\times d}$, and $\otimes$ is the operation of Kronecker product.

3 Busy Period for the Network

The network is said to be in a busy period, if all the servers in the two subsystems are occupied (working or blocked); otherwise, the network is said to be in a non-busy period. In order to consider the busy period for the network, corresponding to the process Q, we define an associated Markov process $\widehat{Q}$ with state space

$\{(-1,i,j,k,\delta_1,\delta_2,\delta_3): i=c-1, 0\le j\le K_1, 0\le k\le K_2, 1\le\delta_1\le m, 1\le\delta_2\le n, 0\le\delta_3\le l\}$
$\bigcup\{(-1,i,j,k,\delta_1,\delta_2,\delta_3): i\ge c, j=0, 0\le k\le K_2, 1\le\delta_1\le m, 1\le\delta_2\le n, \delta_3=0\}$
$\bigcup\{(-1,i,j,k,\delta_1,\delta_2,\delta_3): i\ge c, 1\le j\le K_1, 0\le k\le r-1, 1\le\delta_1\le m, 1\le\delta_2\le n, \delta_3=0\}$
$\bigcup\{(i-c,i,j,k,\delta_1,\delta_2,\delta_3): i\ge c, 1\le j\le K_1, r\le k\le K_2, 1\le\delta_1\le m, 1\le\delta_2\le n, 1\le\delta_3\le l\}$.
Its infinitesimal generator is denoted by

$$\widehat{Q} = \begin{pmatrix} 0_\infty & & & & & \\ \overline{B}_0 & \widehat{B}_1 & \widehat{B}_0 & & & \\ \overline{B}_1 & \widehat{B}_2 & \widehat{B}_1 & \widehat{B}_0 & & \\ \overline{B}_2 & & \widehat{B}_2 & \widehat{B}_1 & \widehat{B}_0 & \\ \vdots & & & \ddots & \ddots & \ddots \end{pmatrix}, \tag{3.1}$$

where

$$\widehat{B}_1 = \begin{pmatrix} \widehat{C}_{11}-c\mu I_{\widehat{g}} & & & & \\ \widehat{C}_{21} & \widehat{C}_{11}-c\mu I_{\widehat{g}} & & & \\ & \ddots & \ddots & & \\ & & \widehat{C}_{21} & \widehat{C}_{11}-c\mu I_{\widehat{g}} & \\ & & & \widehat{C}_{21} & \widehat{C}_{11} \end{pmatrix}_{\widehat{h}\times\widehat{h}},$$

$$\widehat{B}_0 = \begin{pmatrix} \widehat{C}_{12} & & & \\ & \widehat{C}_{12} & & \\ & & \ddots & \\ & & & \widehat{C}_{12} \end{pmatrix}_{\widehat{h}\times\widehat{h}}, \quad \widehat{B}_2 = \begin{pmatrix} 0_{\widehat{g}} & c\mu I_{\widehat{g}} & & \\ & & \ddots & \\ & & & c\mu I_{\widehat{g}} \\ & & & 0_{\widehat{g}} \end{pmatrix}_{\widehat{h}\times\widehat{h}},$$

$$\overline{B}_0 = (\widetilde{B}_2\ \widetilde{B}_1\ 0\ \cdots)_{\widehat{h}\times\infty}, \quad \overline{B}_i = (\underbrace{0_{\widehat{h}\times h} 0_{\widehat{h}\times\widetilde{h}}\cdots 0_{\widehat{h}\times\widetilde{h}}\ \widetilde{B}_1}_{i+2} 0_{\widehat{h}\times\widetilde{h}}\cdots)_{\widehat{h}\times\infty}, \ i\ge 1,$$

$$\widetilde{B}_2 = \begin{pmatrix} 0_{\widehat{g}\times 2g} & 0_{\widehat{g}\times\widetilde{g}} & c\mu I_{\widehat{g}} & & & & & \\ & & & 0_{\widehat{g}\times\widetilde{g}} & c\mu I_{\widehat{g}} & & & \\ & & & & & \ddots & & \\ & & & & & & 0_{\widehat{g}\times\widetilde{g}} & c\mu I_{\widehat{g}} \\ & & & & & & 0_{\widehat{g}\times\widetilde{g}} & 0_{\widehat{g}} \end{pmatrix}_{\widehat{h}\times h},$$

$$\widetilde{B}_1 = \begin{pmatrix} \widetilde{C}_{10} & & & & \\ & \widetilde{C}_{21} & & & \\ & & \ddots & & \\ & & & \widetilde{C}_{21} & 0_{\widehat{g}\times\widetilde{g}} \end{pmatrix}_{\widehat{h}\times\widetilde{h}}, \ \widetilde{C}_{21} = \begin{pmatrix} I_{mn}\otimes U^0 & & \\ & \ddots & \\ & & I_{mn}\otimes U^0 \\ & & 0_{mnl} \\ & & \vdots \\ & & 0_{mnl} \end{pmatrix}_{\widehat{g}\times\widetilde{g}},$$

$$\widetilde{C}_{10} = \begin{pmatrix} I_{mn}\otimes U^0 & & & & & \\ & \ddots & & & & \\ & & I_{mn}\otimes U^0 & 0_{mnl\times mn} & \ldots & 0_{mnl\times mn} \end{pmatrix}_{\widehat{g}\times f},$$

$$\widehat{C}_{11} = \begin{pmatrix} G & & I_m\otimes T^0\beta\otimes I_l & & \\ & \ddots & & \ddots & \\ & & G & & I_m\otimes T^0\beta\otimes I_l \\ & & & & I_{mn}\otimes U + S\otimes I_{nl} \end{pmatrix}_{\widehat{g}\times\widehat{g}},$$

$G \equiv I_{mn}\otimes U + I_m\otimes T\otimes I_l + S\otimes I_{nl}$,

$$\widehat{C}_{12} = \begin{pmatrix} S^0\alpha\otimes I_{nl} & & \\ & \ddots & \\ & & S^0\alpha\otimes I_{nl} \end{pmatrix}_{\widehat{g}\times\widehat{g}},$$

$$\widehat{C}_{21} = \begin{pmatrix} 0_{mn} & & & & \\ \vdots & & & & \\ 0_{mn} & & & & \\ I_{mn}\otimes U^0\gamma & & & & \\ & \ddots & & & \\ & & I_{mn}\otimes U^0\gamma & 0_{mnl} & \ldots & 0_{mnl} \end{pmatrix}_{\widehat{g}\times\widehat{g}}.$$

$\widehat{g} = (K_2 - r + 1)mnl$, $\widehat{h} = K_1\widehat{g}$, $\widetilde{g} = rmn$, $\widetilde{h} = f + K_1\widetilde{g}$.

Let $(X_0^0,\ X_1^0,\ X_2^0,\ \cdots)$ be an arbitrary given initial state probability vector of the Markov process Q, where X_i^0, $i \geq 0$, are non-negative row vectors of dimension h. From the definition, we know that the busy period for the network with initial state probability $(X_0^0,\ X_1^0,\ X_2^0,\ \cdots)$ is the time until the $\widehat{Q}$ is absorbed into level -1 with initial state probability vector $(\widehat{X}_{-1}^0,\ \widehat{X}_0^0,\ \widehat{X}_1^0,\ \cdots)$, where $\widehat{X}_{-1}^0 = (\widehat{X}_{-1,0}^0,\ \widehat{X}_{-1,1}^0,\ \cdots)$, $\widehat{X}_{-1,0}^0 = \sum_{i=0}^{c-1} X_i^0$, $\widehat{X}_{-1,i}^0 = (X_{i+c-1,0}^0,\ \widetilde{X}_{i,1}^0,\ \cdots,\ \widetilde{X}_{i,K_1}^0)$ $(i \geq 1)$, $\widetilde{X}_{i,j}^0 = (X_{i+c-1,j,0}^0,\ \cdots,\ X_{i+c-1,j,r-1}^0)$ $(i \geq 1,\ 1 \leq j \leq K_1)$, $\widehat{X}_i^0 = (\overline{X}_{i,1}^0,\ \cdots,\ \overline{X}_{i,K_1}^0)$ $(i \geq 0)$, and $\overline{X}_{i,j}^0 = (X_{i+c,j,r}^0,\ \cdots,\ X_{i+c,j,K_2}^0)$ $(i \geq 0,\ 1 \leq j \leq K_1)$.

Let $\widehat{X}^0 = (\widehat{X}_0^0,\ \widehat{X}_1^0,\ \widehat{X}_2^0,\ \cdots)$. Let φ be the maximum of the absolute value of the diagonal elements of the matrix $\widehat{Q}$. Then

$$\varphi = c\mu + \max_{1\le i\le m}\{-S_{ii}\} + \max_{1\le i\le n}\{-T_{ii}\} + \max_{1\le i\le l}\{-U_{ii}\}. \tag{3.2}$$

Let

$$T^0 = \begin{pmatrix} \overline{B}_0 \\ \overline{B}_1 \\ \overline{B}_2 \\ \vdots \end{pmatrix}, \qquad T = \begin{pmatrix} \widehat{B}_1 & \widehat{B}_0 & & & \\ \widehat{B}_2 & \widehat{B}_1 & \widehat{B}_0 & & \\ & \widehat{B}_2 & \widehat{B}_1 & \widehat{B}_0 & \\ & & \ddots & \ddots & \ddots \end{pmatrix},$$

$$\overline{D} = \frac{1}{\varphi}T^0 \equiv \begin{pmatrix} \overline{D}_0 \\ \overline{D}_1 \\ \overline{D}_2 \\ \vdots \end{pmatrix}, \quad D = I + \frac{1}{\varphi}T \equiv \begin{pmatrix} D_1 & D_0 & & & \\ D_2 & D_1 & D_0 & & \\ & D_2 & D_1 & D_0 & \\ & & \ddots & \ddots & \ddots \end{pmatrix}, \tag{3.3}$$

$$F'(x) = \varphi e^{-\varphi x}\widehat{X}^0 \sum_{n=0}^{\infty} D^n \overline{D}\frac{(\varphi x)^n}{n!}, \quad x > 0. \tag{3.4}$$

Then the density function of the busy period of the network is $F'(x)\mathbf{1}$, where $\mathbf{1}$ is the column unit vector of infinite dimensions (see Hsu and Xu [7]). Write $F'(x)$ componentwise as $F'(x) = (F_1'(x),\ F_2'(x), \cdots)$, where $F_i'(x)$, $i \ge 1$, are all functions.

It now seems that the results of Hsu and Xu [7] can be used directly. For example, the increasing sequence of positive integers $\{N_0, N_1, N_2, \cdots\}$ can be generated by

$$\begin{cases} \widehat{X}_{-1}^0(\mathbf{1}_{N_0}^{\mathbf{T}}, 0, 0, \cdots)^{\mathbf{T}} + \sum\limits_{i=0}^{N_0} \widehat{X}_i^0 \mathbf{1}_{\widehat{h}} > 1 - \varepsilon, \\ \sum\limits_{j=1}^{N_n} \overline{D}_k^{(lj)} > 1 - \varepsilon,\ 0 \le k \le N_{n-1},\ 1 \le l \le \widehat{h},\ n \ge 1, \end{cases}$$

where $\mathbf{T}$ is the transpose operation of a matrix. However, we immediately find that in such a case the computational complexity may be exponential. In fact, since the most right $\widehat{h} \times \widetilde{h}$ dimensional nonzero submatrix $\frac{1}{\varphi}\widetilde{B}_1$ of $\overline{D}_i$ moves $\widetilde{h}$ columns to the right when i becomes $i+1$, we may take that $N_n = h + \widetilde{h}(N_{n-1}+1)$. It then follows by induction that

$$N_n = \widetilde{h}^n N_0 + (h + \widetilde{h})(\widetilde{h}^{n-1} + \widetilde{h}^{n-2} + \cdots + \widetilde{h} + 1) = O(\widetilde{h}^n).$$

Therefore, we will readily find that both the time and space complexities of the computations in the algorithm are exponential.

To avoid the curse of dimensionality, we will improve the algorithm of Hsu and Xu [7] as follows.

Algorithm 3.1:
Step 1. For a given error $\Delta > 0$, $T > 0$ and $\varphi > 0$, compute $L = \varphi^2 T + 3\varphi$;

Step 2. Set $\varepsilon = \Delta/L$ and $M = \max([2\varphi e T], [\log_2(1/\varepsilon)])$;
Step 3. Generate an increasing sequence of positive integers $N_0, N_1, N_2, \cdots, N_{M+1}$ by

$$\begin{cases} \widehat{X}^0_{-1,0}\mathbf{1}_h + \sum\limits_{i=0}^{N_0} \widehat{X}^0_{-1,i}\mathbf{1}_{\widetilde{h}} + \sum\limits_{i=0}^{N_0} \widehat{X}^0_i \mathbf{1}_{\widehat{h}} > 1-\varepsilon, \\ \sum\limits_{j=1}^{h+\widetilde{h}(N_n-1)} \overline{D}^{(lj)}_k > 1-\varepsilon,\ 0 \le k \le N_{n-1},\ 1 \le l \le \widehat{h},\ n \ge 1; \end{cases} \tag{3.5}$$

Step 4. Compute ${}_{N_n}\sigma_j(n)$ by

$${}_{N_n}\sigma_j(n) = \begin{cases} \sum\limits_{i=0}^{N_n} \sum\limits_{l=1}^{\widehat{h}} {}_{N_n}\pi_{il}(n) \overline{D}^{(lj)}_i, & 1 \le j \le h + \widetilde{h}(N_{n+1}-1),\ 0 \le n \le M, \\ \\ 0, & j > h + \widetilde{h}(N_{n+1}-1),\ 0 \le n \le M, \end{cases} \tag{3.6}$$

where

$$\begin{cases} {}_{N_0}\pi_i(0) &=& \widehat{X}^0_i,\ \ 0 \le i \le N_0, \\ \\ {}_{N_n}\pi_0(n) &=& {}_{N_{n-1}}\pi_0(n-1)D_1 + {}_{N_{n-1}}\pi_1(n-1)D_2,\ 1 \le n \le M, \\ \\ {}_{N_n}\pi_i(n) &=& \sum\limits_{j=i-1}^{N_{n-1}\wedge(i+1)} {}_{N_{n-1}}\pi_j(n-1)D_{j-i+1},\ 1 \le i \le N_n,\ 1 \le n \le M; \end{cases} \tag{3.7}$$

Step 5. Compute ${}_{M,N_0}F'_j(x),\ x \in [0,T]$, by

$${}_{M,N_0}F'_j(x) = \varphi e^{-\varphi x} \sum_{n=0}^{M} \frac{(\varphi x)^n}{n!} \cdot {}_{N_n}\sigma_j(n). \tag{3.8}$$

Then ${}_{M,N_0}F'_j(x)$ approximates $F'_j(x)$ uniformly on the interval $[0,T]$ and for $j \ge 1$ with uniform error Δ, and $\sum_{j=1}^{h+\widetilde{h}(N_{M+1}-1)} {}_{M,N_0}F'_j(x)$ uniformly approximates $\sum_{j=1}^{\infty} F'_j(x)$ on the interval $[0,T]$ with uniform error Δ.

We note that after the above modification of Steps 3-5, the corresponding results in [7] can be proved without much difficulty, and thus the conclusions of Algorithm 3.1 hold. We now analyze the computational complexity of the improved algorithm. In Step 3, based on the structure of $\overline{D}_k$'s, we can now take that $N_n = N_{n-1}+1$ or $N_n = N_0 + n$. So the number of operations needed is polynomial. For (3.6) in Step 4, the complexity is $O(\widehat{h}^2 M^2)$, and for (3.7), the complexity is less than $O(h^2 M^3)$. Hence, the time complexity of the algorithm is polynomial. On the other hand, owing to $N_n = N_0 + n$, it is easy to see that the space complexity of the algorithm is also polynomial.

4 A Numerical Example

For the matched queueing network $M/M/c \to \circ M/M/1$ with $K_1 = K_2 = 3$, we consider the busy period for the whole network. The state process of the matched queueing network is a Markov process with state space
$\{(i, j, k, \delta_1, \delta_2, \delta_3) : \ i \geq 0, 0 \leq j \leq 3, 0 \leq k \leq 3, \delta_1 = \delta_2 = 1, \delta_3 = 0, 1\}$.
We set $c = 2$, $r = 2$. Then the infinitesimal generator of this Markov process is given by

$$Q = \begin{pmatrix} H_0 & B_0 & & & \\ F_1 & H_1 & B_0 & & \\ & B_2 & B_1 & B_0 & \\ & & B_2 & B_1 & B_0 \\ & & & \ddots & \ddots & \ddots \end{pmatrix},$$

where

$$H_d = \begin{pmatrix} C_{00} - d\mu I_4 & & & \\ C_{10} & C_{11} - d\mu I_4 & & \\ & C_{21} & C_{11} - d\mu I_4 & \\ & & C_{21} & C_{11} \end{pmatrix}_{16\times16}, \ 0 \leq d \leq 2,$$

$$B_0 = \begin{pmatrix} C_{01} & & & \\ & C_{12} & & \\ & & C_{12} & \\ & & & C_{12} \end{pmatrix}_{16\times16}, \ F_d = \begin{pmatrix} 0_4 & E_d & & \\ & & d\mu I_4 & \\ & & & d\mu I_4 \\ & & & 0_4 \end{pmatrix}_{16\times16}, \ 1 \leq d \leq 2,$$

$B_2 \equiv F_2, \quad B_1 \equiv H_2,$

$$C_{0\,0} = \begin{pmatrix} -\lambda_1 - \lambda & \lambda_1 & 0 & 0 \\ 0 & -\lambda_1 - \lambda & \lambda_1 & 0 \\ 0 & 0 & -\lambda_1 - \lambda & \lambda_1 \\ 0 & 0 & 0 & -\lambda \end{pmatrix}_{4\times4},$$

$$C_{11} = \begin{pmatrix} -\lambda_1 - \lambda & \lambda_1 & 0 & 0 \\ 0 & -\lambda_1 - \lambda & \lambda_1 & 0 \\ 0 & 0 & -\lambda_1 - \lambda - \mu_1 & \lambda_1 \\ 0 & 0 & 0 & -\lambda - \mu_1 \end{pmatrix}_{4\times4},$$

$$C_{10} = C_{21} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \mu_1 & 0 & 0 & 0 \\ 0 & \mu_1 & 0 & 0 \end{pmatrix}_{4\times4},$$

$C_{01} = C_{12} = \lambda I_4,\ E_d = d\mu I_4,\ 1 \le d \le 2$, where λ and λ_1 are the mean arrival rates of type-1 and type-2 inputs respectively and μ and μ_1 are the mean service rates of the first and second subsystems respectively. By the definition of φ, it is easy to see that $\varphi = \lambda+\lambda_1+2\mu+\mu_1$. Denote the initial probability distribution of $\widehat{Q}$ by $(\widehat{X}^0_{-1}, \widehat{X}^0_0, \widehat{X}^0_1, \cdots)$, where $\widehat{X}^0_{-1} = (\widehat{x}^0_{-1,1}, \widehat{x}^0_{-1,2}, \cdots)$ and $\widehat{X}^0_i = (\widehat{x}^0_{i,1}, \widehat{x}^0_{i,2}, \cdots, \widehat{x}^0_{i,6})$, $i \ge 0$, and suppose that

$$\widehat{x}^0_{-1,i} = 0,\ i \ge 1,$$

$$\widehat{x}^0_{i,j} = (1-\beta)\beta^i \frac{1-\gamma}{\gamma-\gamma^7}\gamma^j,\ i \ge 0,\ 1 \le j \le 6.$$

From Section 3, we know that the busy period for the network is the time until the $\widehat{Q}$ is absorbed into level -1. For $\Delta = 0.001$, $\lambda = 0.02$, $\lambda_1 = 0.01$, $\mu = 0.05$, $\mu_1 = 0.06$, $\beta = 0.3$, $\gamma = 0.4$, $T = 10$, and $\theta = 0.3$, we obtain from Algorithm 3.1 that

$$F'_j(x) \approx \begin{cases} 0.19e^{-0.19x} \sum\limits_{n=0}^{10} a_n^{(j)} x^n, & x \in [0,10],\ 1 \le j \le 166; \\ 0, & \text{otherwise}, \end{cases}$$

where $a_n^{(j)} = 0$ for $1 \le j \le 10$ and $0 \le n \le 10$, $a_n^{(j)}$ $(11 \le j \le 166,\ 0 \le n \le 10)$ are presented in Table 1. Moreover, we have

$$\begin{aligned} F'(x)\mathbf{1} \approx\ & 0.19e^{-0.19x}(6.755 \times 10^{-1} + 3.889 \times 10^{-2}x^1 + 2.379 \times 10^{-3}x^2 \\ & +9.771 \times 10^{-5}x^3 + 3.234 \times 10^{-6}x^4 + 9.021 \times 10^{-8}x^5 + 2.127 \times 10^{-9}x^6 \\ & +4.306 \times 10^{-11}x^7 + 7.602 \times 10^{-13}x^8 + 1.187 \times 10^{-14}x^9 \\ & +1.657 \times 10^{-18}x^{10}), \end{aligned}$$

for $x \in [0, 10]$, $Re(s) > 0$ and $|s| \ge 0.3$.

Remark 4.1: The above computation results are not listed completely because of too many data.

5 Concluding Remarks

1) For queueing networks with absorbing state set of infinite states (level -1) and initial probability vector of infinite dimensions, even for the simplest matched queueing network with Poisson inputs and exponential service distributions showns as the numerical example, the distributions of busy periods cannot be obtained by any existing methods and algorithms in literature.

2) By means of our improved algorithm, the approximation of the distribution of the busy period for the matched queueing network can be computed with uniform error, and the curse of dimensionality are overcome. For the given numerical example, the approximation with uniform error $\Delta = 0.001$ can be computed by PC586 less than three

seconds.

3) The method in this paper is available for computing the distributions of busy periods for other queueing networks, but it needs to design different improved algorithms according to the structures of networks such that the computational complexities are polynomially bounded.

4) The method in this paper can also be applied to compute the moments of the indices such as busy periods, waiting times, etc.. Their algorithms can be derived from Algorithm 3.1 without much difficulty.

References

[1] Chen BQ (1988) Ergodicity for the queueing model of a single-berth seaport. J. Engin. Math. 5:1-7 (in Chinese)

[2] Hsu GH, He QM (1993) Matched queueing system $M \circ PH/G/1$. Acta Math. Appl. Sinica, English Series 9:104-114

[3] Hsu GH, He QM (1994) The matched queueing system $G \circ PH/PH/1$. Acta Math. Appl. Sinica, English Series 10:34-47

[4] Hsu GH, He QM, Liu XS (1990) The stationary behavior of the matched queueing system. Acta Math. Appl. Sinica (in Chinese) 13:39-48

[5] Hsu GH, He QM, Liu XS (1993) The matched queueing system with a double input. Acta Math. Appl. Sinica, English Series 9:50-62

[6] Hsu GH, Jensen U (1993) The matched queueing network $PH/M/c \rightarrow \circ PH/PH/1$. Queueing system 13:315-333

[7] Hsu GH, Xu DJ (1998) First passage times for multidimensional denumerable state Markov processes. Chinese Sci. Bulletin (to appear)

[8] Hsu GH, Yuan XM (1995) The first passage times and their algorithms for Markov processes. Stoch. Models 11:195-210

[9] Latouche G (1981) Queues with paired customers. J. Appl. Prob. 18:684-696

[10] Neuts MF (1981) Matrix-Geometric Solutions in Stochastic Models. Johns Hopkins Univ. Press, Baltimore

[11] Yang XY, Chen BQ (1984) An application of stochastic simulation to the design of the coal loading and unloading system in a seaport. Theory and Practice of syst. Engin. 3:41-47 (in Chinese)

Table 1. The coefficients $a_n^{(j)},\ 0 \le n \le 10,\ 11 \le j \le 166$

j	11	12	13	14	15
$a_0^{(j)}$	2.220×10^{-1}	8.878×10^{-2}	0.000	0.000	3.551×10^{-2}
$a_1^{(j)}$	0.000	3.106×10^{-3}	0.000	0.000	1.020×10^{-2}
$a_2^{(j)}$	0.000	1.552×10^{-5}	0.000	0.000	7.317×10^{-4}
$a_3^{(j)}$	0.000	5.170×10^{-8}	0.000	0.000	2.438×10^{-5}
$a_4^{(j)}$	0.000	1.292×10^{-10}	0.000	0.000	6.090×10^{-7}
$a_5^{(j)}$	0.000	2.582×10^{-13}	0.000	0.000	1.217×10^{-8}
$a_6^{(j)}$	0.000	4.300×10^{-16}	0.000	0.000	2.028×10^{-10}
$a_7^{(j)}$	0.000	6.140×10^{-19}	0.000	0.000	2.895×10^{-12}
$a_8^{(j)}$	0.000	7.670×10^{-22}	0.000	0.000	3.617×10^{-14}
$a_9^{(j)}$	0.000	8.517×10^{-25}	0.000	0.000	4.016×10^{-16}
$a_{10}^{(j)}$	0.000	8.512×10^{-28}	0.000	0.000	4.014×10^{-18}

j	$\cdots$	163	164	165	166
$a_0^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_1^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_2^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_3^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_4^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_5^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_6^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_7^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_8^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_9^{(j)}$	$\cdots$	0.000	0.000	0.000	0.000
$a_{10}^{(j)}$	$\cdots$	1.246×10^{-27}	6.624×10^{-28}	0.000	0.000