J. Appl. Prob. **42,** 531–549 (2005)
Printed in Israel
© *Applied Probability Trust* 2005

EFFICIENT ALGORITHMS FOR TRANSIENT ANALYSIS OF STOCHASTIC FLUID FLOW MODELS

SOOHAN AHN,* *The University of Seoul*
V. RAMASWAMI,** *AT&T Labs*

Abstract

We derive several algorithms for the busy period distribution of the canonical Markovian fluid flow model. One of them is similar to the Latouche–Ramaswami algorithm for quasi-birth–death models and is shown to be quadratically convergent. These algorithms significantly increase the efficiency of the matrix-geometric procedures developed earlier by the authors for the transient and steady-state analyses of fluid flow models.

Keywords: Stochastic fluid flow; transient analysis; matrix-geometric method; algorithm; quadratic convergence

2000 Mathematics Subject Classification: Primary 60K25
Secondary 90B05; 68M20

1. Introduction

The subject of this paper is the construction of efficient algorithms for the transient (time-dependent) analysis of the canonical Markov-modulated fluid flow (MMFF) model. This model is obtained by assuming as given an irreducible, continuous-time Markov chain $J(t)$ of 'phases' with a finite state space $S = S_1 \cup S_2 \cup S_3$ and infinitesimal generator Q such that, for a phase $i \in S_1$, the fluid level increases at rate $c_i > 0$; for $i \in S_2$, the fluid level decreases at rate $c_i > 0$; and for $i \in S_3$, the fluid level remains constant.

In [2], the authors derived the joint distribution of $(F(t), J(t))$, where $F(t)$ and $J(t)$ are respectively the fluid level and phase at time $t+$, given that $F(0) = 0$ and $J(0) = i$, $i \in S_1$; see Section 7 of [2]. For $\mathrm{Re}(s) \geq 0$, the matrix of Laplace–Stieltjes transforms (LSTs) with elements

$$\mathrm{E}_{(0,i)}[\mathrm{e}^{-sF(t)}1(J(t) = j)], \qquad i \in S_1 \text{ and } j \in S,$$

where $\mathrm{E}_{(x,i)}$ denotes conditional expectation given the initial state (x, i) and $1(\cdot)$ is an indicator function, was characterized in terms of three LST matrices $\tilde{K}(s)$, $\tilde{\Psi}(s)$, and $\tilde{\Theta}(s)$. It was shown that these matrices are readily obtained from the LST matrix $\Psi(s)$ of the busy period $\tau = \inf\{t > 0 \colon F(t) = 0\}$ of the fluid flow model, and defined by the elements

$$[\Psi(s)]_{ij} := \mathrm{E}_{(0,i)}[\mathrm{e}^{-s\tau}1(J(\tau) = j)], \qquad i \in S_1,\ j \in S_2,\ \text{and } \mathrm{Re}(s) \geq 0.$$

In the literature on stochastic fluid flow models, much attention has been focused on the steady-state distribution; see [3], [5], [7], and [16]. As for time-dependent distributions, past

Received 5 March 2004; revision received 29 October 2004.

* Postal address: Department of Statistics, The University of Seoul, 90 Jeonnong-dong, Dongdaemun-gu, Seoul, 130-743, South Korea.
** Postal address: AT&T Labs, 180 Park Avenue, E-233, Florham Park, NJ 07932, USA.
Email address: vram@research.att.com

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

approaches have been based on Wiener–Hopf factorizations or on partial differential equations. Noteworthy are the work of Sericola [17], based on an ad-hoc series expansion, and that of [9], based on spectral methods. Ramaswami [15] was the first systematic use of matrix-analytic methods in the context of fluid flow models and provided a highly efficient algorithm for computing the stationary distribution; Ahn and Ramaswami [1] demonstrated that approach to be based on stochastic coupling to a matrix-geometric queue. A continuation of that work in [2] characterized the time-dependent distributions exactly in terms of the busy period transform of the fluid model, and provided an accurate algorithm to evaluate them. The methods of [2] require repeated evaluations of the busy period transform and, therefore, the quadratically convergent algorithm of this paper improves the efficiency of those methods very significantly. Note that a characterization of the busy period distribution was obtained earlier by Asmussen [4], who also noted its importance as a fundamental quantity and provided a linearly convergent iterative scheme for its computation; being quadratically convergent, the new algorithm developed here is much faster.

The discrete-state-space analogue of the stochastic fluid flow is the quasi-birth–death (QBD) process, for which matrix-geometric methods apply; see [12] and [13]. The analogue of the transform $\Psi(s)$ in the QBD model is the matrix G of the latter, for which an efficient quadratically convergent algorithm has been obtained by Latouche and Ramaswami [11] using probabilistic arguments. The work here provides an algorithm for the fluid model that is similar in spirit to the Latouche–Ramaswami algorithm for QBDs and has quadratic convergence.

The construction of the algorithms in this paper and the determination of their properties are achieved through the consideration of a closely related queue with interarrival-dependent service times (see Section 5) and a probabilistic analysis of that queue. This coupled queue is quite different from those in [1] and [2], which involved service times distributed independently of interarrival times. Thus, this work may also be interpreted as an extension of the algorithms in [11] to queues with interarrival-dependent service times.

Throughout this paper, I will denote an identity matrix and $\mathbf{1}$ a column vector of 1s, both of whose dimensions will be determined by the context in which they appear. Where it is necessary to indicate the dimension explicitly, we will write I_n to denote the $n \times n$ identity matrix. For later use, we define the diagonal matrices

$$C_j = \operatorname{diag}\{c_i, i \in S_j\}, \qquad j = 1, 2, 3,$$

where we set $c_i = 1$ for all $i \in S_3$, and let $C = \operatorname{diag}(C_1, C_2, C_3)$. We partition the states of the Markov chain in conformity with the three sets S_i identified above and denote its infinitesimal generator, in partitioned form, as

$$Q = \begin{pmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{pmatrix}.$$

Finally, to avoid confusion between submatrices in a partitioned structure and elements of a matrix, the (i, j)th element of a matrix A will always be denoted by $[A]_{i,j}$, $[A]_{ij}$, or $A(i, j)$ instead of as A_{ij}, as is often customary.

2. Spatial uniformization

A key step in the analysis of [2] is a procedure called spatial uniformization, which we now discuss. A spatial uniformization (for the fluid flow) is effected by modeling the Markov process

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

of phases as a Markov renewal process with exponential sojourn times such that potential changes to the fluid level between epochs of the Markov renewal process are identically distributed. To that end, we let $\{(J_n, t_n), n \geq 0\}$ be such a process, with successive states $J_n \in S$, transition epochs $0 = t_0 < t_1 < t_2 < \cdots$, and with semi-Markov kernel $H(\cdot)$ defined such that $H(i, j; t)$, the (i, j)th element of $H(t)$, is given by

$$H(i, j; t) = \mathrm{P}\{J_{n+1} = j, t_{n+1} - t_n \leq t \mid J_n = i\} = (1 - \mathrm{e}^{-\lambda c_i t})[P_\lambda]_{ij},$$

where

$$P_\lambda = \lambda^{-1} C^{-1} Q + I \qquad \text{and} \qquad \lambda \geq \max_{i \in S}\{-[C^{-1}Q]_{ii}\} \quad \text{is fixed.}$$

The associated semi-Markov process $\mathcal{J} = \{J(t), t \geq 0\}$ is specified such that it takes the value J_n in the interval $t_n \leq t < t_{n+1}$. The following result shows that $\mathcal{J}$ is indeed a realization of the phase process; for a proof, we refer the reader to [2].

Theorem 1. *The process* $\mathcal{J} = \{J(t), t \geq 0\}$ *is a continuous-time Markov chain with infinitesimal generator* Q.

Since a sojourn interval of the semi-Markov process in $i \in S_1$ is distributed as $\mathrm{Exp}(\lambda c_i)$ with fluid accumulation at rate c_i per unit time, the additional fluid accumulation in that interval is distributed as $\mathrm{Exp}(\lambda)$. Similarly, for a state in S_2, given that adequate fluid exists at the start of the interval, the potential decrease of the fluid level that could be effected is distributed as $\mathrm{Exp}(\lambda)$. This underlies our reason for using the nomenclature 'spatial uniformization'. Throughout the rest of the paper, we will view the phase process, i.e. the continuous-time Markov chain $J(\cdot)$, as being specified by the above construction.

3. Busy period

For $x > 0$, $i, j \in S$, and $\mathrm{Re}(s) \geq 0$, let $[\hat{G}(s, x)]_{i,j}$ denote the LST

$$[\hat{G}(s, x)]_{i,j} := \mathrm{E}_{(x,i)}[\mathrm{e}^{-s\tau} 1(J(\tau) = j)].$$

We assume that the matrix $\hat{G}(s, x)$ of elements $[\hat{G}(s, x)]_{ij}$ is also partitioned according to the sets S_i, $i = 1, 2, 3$. Thus, for instance, the submatrix $\hat{G}_{12}(s, x)$ is the matrix of elements $[\hat{G}(s, x)]_{ij}$ for which i varies over S_1 and j varies over S_2.

In our model, fluid is depleted only in S_2. Thus, all busy periods must end in a state of S_2. From this, the following result is trivial.

Theorem 2. *The matrices* $\hat{G}(s, x)$ *have the structure*

$$\hat{G}(s, x) = \begin{pmatrix} 0 & \hat{G}_{12}(s, x) & 0 \\ 0 & \hat{G}_{22}(s, x) & 0 \\ 0 & \hat{G}_{32}(s, x) & 0 \end{pmatrix}.$$

We now proceed to determine the submatrices in the second column of the partitioned structure above.

Theorem 3. *For* $x > 0$,

(a) $\hat{G}_{12}(s, x) = \Psi(s)\hat{G}_{22}(s, x)$;

(b) $\hat{G}_{32}(s, x) = (sI - Q_{33})^{-1} Q_{31} \hat{G}_{12}(s, x) + (sI - Q_{33})^{-1} Q_{32} \hat{G}_{22}(s, x)$; *and*

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

(c) *with*

$$H(s) = C_2^{-1}[Q_{22} - sI + Q_{23}(sI - Q_{33})^{-1}Q_{32} + \{Q_{21} + Q_{23}(sI - Q_{33})^{-1}Q_{31}\}\Psi(s)],$$

we have

$$\hat{G}_{22}(s, x) = \mathrm{e}^{H(s)x}.$$

Proof. Part (a) follows easily by conditioning on the first return to level x in the set S_2. Similarly, part (b) is proved by conditioning on the first epoch during which the phase process escapes from S_3. To prove (c), consider the first epoch of spatial uniformization and note that, with δ_{ij} denoting the Kronecker delta and $P \equiv P_\lambda$, we can write

$$\begin{aligned}[\hat{G}_{22}(s, x)]_{i,j} &= \delta_{ij}\mathrm{e}^{-\lambda c_i x/c_i}\mathrm{e}^{-sx/c_i} \\ &\quad + \int_0^{x/c_i} \lambda c_i \mathrm{e}^{-\lambda c_i t}\mathrm{e}^{-st}\sum_{k\in S}[P]_{i,k}[\hat{G}(s, x - c_i t)]_{k,j}\,\mathrm{d}t \\ &= \delta_{ij}\mathrm{e}^{-(\lambda+s/c_i)x} + \lambda\int_0^x \mathrm{e}^{-(\lambda+s/c_i)(x-z)}\sum_{k\in S}[P]_{i,k}[\hat{G}(s, z)]_{k,j}\,\mathrm{d}z.\end{aligned}$$

Multiplying by $\mathrm{e}^{(\lambda+s/c_i)x}$ and differentiating with respect to x, we obtain

$$\bigg(\lambda + \frac{s}{c_i}\bigg)[\hat{G}_{22}(s, x)]_{i,j} + \bigg[\frac{\partial}{\partial x}\hat{G}_{22}(s, x)\bigg]_{i,j} = \lambda\sum_{k\in S}[P]_{i,k}[\hat{G}(s, x)]_{k,j},$$

which, being written in matrix form, yields, due to parts (a) and (b), the differential equation $(\partial/\partial x)\hat{G}_{22}(s, x) = H(s)\hat{G}_{22}(s, x)$ with the initial condition $\hat{G}_{22}(s, 0) = I$. Therefore, $\hat{G}_{22}(s, x) = \mathrm{e}^{H(s)x}$ and the proof is complete.

4. Random initial fluid

We wish to appeal to matrix-geometric results that have mainly been developed for queues. So, a tool we shall employ is to view the fluid model as being derived from the work process of a suitably defined queue. To relate quantities of interest to the busy period of such a queue, it helps to consider a busy period starting with an amount of fluid X that is exponentially distributed with mean λ^{-1}; in the context of the queue, X will be the amount of work brought in by the customer starting a busy period of the queue.

We thus consider the transform matrix $\tilde{G}(s, \lambda)$ defined by the elements

$$[\tilde{G}(s, \lambda)]_{i,j} := \mathrm{E}[\mathrm{E}_{(X,i)}[\mathrm{e}^{-s\tau}1(J(\tau) = j)]],$$

where the outer expectation is with respect to X. Then, from Theorem 3 we can easily obtain

$$\tilde{G}_{22}(s, \lambda) = \int_0^\infty \lambda\mathrm{e}^{-\lambda y}\mathrm{e}^{H(s)y}\,\mathrm{d}y = \lambda(\lambda I - H(s))^{-1}, \tag{1}$$

$$\tilde{G}_{12}(s, \lambda) = \Psi(s)\tilde{G}_{22}(s, \lambda), \tag{2}$$

$$\tilde{G}_{32}(s, \lambda) = (sI - Q_{33})^{-1}Q_{31}\tilde{G}_{12}(s, \lambda) + (sI - Q_{33})^{-1}Q_{32}\tilde{G}_{22}(s, \lambda). \tag{3}$$

The following result expresses $\Psi(s)$ in terms of the submatrices of $\tilde{G}(s, \lambda)$.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

Theorem 4. *The LST matrix* $\Psi(s)$ *can be written*

$$\Psi(s) = \left(P_{11} - \frac{s}{\lambda}C_1^{-1}\right)\tilde{G}_{12}(s,\lambda) + P_{12}\tilde{G}_{22}(s,\lambda) + P_{13}\tilde{G}_{32}(s,\lambda). \tag{4}$$

Proof. If we consider the first epoch of spatial uniformization of the underlying Markov process, then, for $i \in S_1$ and $j \in S_2$,

$$\begin{aligned}[\Psi(s)]_{i,j} &= \int_0^\infty \lambda c_i \mathrm{e}^{-\lambda c_i t}\mathrm{e}^{-st}\sum_{k\in S}[P]_{i,k}[\hat{G}(s, c_i t)]_{k,j}\,\mathrm{d}t \\ &= \lambda\int_0^\infty \mathrm{e}^{-(\lambda + s/c_i)y}\sum_{k\in S}[P]_{i,k}[\hat{G}(s,y)]_{k,j}\,\mathrm{d}y,\end{aligned}$$

which we can rewrite in matrix form as

$$\Psi(s) = \lambda\int_0^\infty \mathrm{e}^{-(\lambda I + sC_1^{-1})y}[P_{11}\hat{G}_{12}(s,y) + P_{12}\hat{G}_{22}(s,y) + P_{13}\hat{G}_{32}(s,y)]\,\mathrm{d}y.$$

Using (3) and Theorem 3, we can rewrite $\Psi(s)$ as

$$\Psi(s) = \lambda\int_0^\infty \mathrm{e}^{-sC_1^{-1}y}L(s)\mathrm{e}^{-(\lambda I - H(s))y}\,\mathrm{d}y,$$

where

$$L(s) = P_{11}\Psi(s) + P_{12} + P_{13}(sI - Q_{33})^{-1}Q_{31}\Psi(s) + P_{13}(sI - Q_{33})^{-1}Q_{32}.$$

Using integration by parts, we obtain

$$\Psi(s) = L(s)\lambda(\lambda I - H(s))^{-1} - \frac{s}{\lambda}C_1^{-1}\Psi(s)\lambda(\lambda I - H(s))^{-1}.$$

Substituting for $L(s)$ and using (1), (2), and (3) immediately yields (4).

The following result is a key theorem in the development of an algorithm for computing $\Psi(s)$ via $\tilde{G}(s,\lambda)$.

Theorem 5. *The matrices* $\tilde{G}_{12}(s,\lambda)$, $\tilde{G}_{22}(s,\lambda)$, *and* $\tilde{G}_{32}(s,\lambda)$ *satisfy the following equations:*

$$\begin{aligned}\tilde{G}_{12}(s,\lambda) &= \left[P_{11} - \frac{s}{\lambda}C_1^{-1}\right]\tilde{G}_{12}(s,\lambda)\tilde{G}_{22}(s,\lambda) + P_{12}\tilde{G}_{22}^2(s,\lambda) + P_{13}\tilde{G}_{32}(s,\lambda)\tilde{G}_{22}(s,\lambda), \\ \tilde{G}_{22}(s,\lambda) &= \lambda C_2(sI + 2\lambda C_2)^{-1}[I + P_{21}\tilde{G}_{12}(s,\lambda) + P_{22}\tilde{G}_{22}(s,\lambda) + P_{23}\tilde{G}_{32}(s,\lambda)], \\ \tilde{G}_{32}(s,\lambda) &= \frac{\lambda}{s+\lambda}P_{31}\tilde{G}_{12}(s,\lambda) + \frac{\lambda}{s+\lambda}P_{32}\tilde{G}_{22}(s,\lambda) + \frac{\lambda}{s+\lambda}P_{33}\tilde{G}_{32}(s,\lambda).\end{aligned}$$

Proof. The result of the first equation is clear from (2) and (4). The third equation can be obtained easily by conditioning on the first epoch of spatial uniformization. Thus, we only need to prove the second equation.

By considering the first epoch of spatial uniformization, we can write

$$[\hat{G}_{22}(s,x)]_{i,j} = \mathrm{e}^{-\lambda c_i x/c_i}\mathrm{e}^{-sx/c_i} + \int_0^{x/c_i}\lambda c_i \mathrm{e}^{-\lambda c_i t}\mathrm{e}^{-st}\sum_{k\in S}[P]_{i,k}[\hat{G}(s, x - c_i t)]_{k,j}\,\mathrm{d}t.$$

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

Multiplying this by $\lambda e^{-\lambda x}$ and integrating over x, it is easy to obtain

$$[\tilde{G}_{22}(s,\lambda)]_{i,j} = \lambda\left(2\lambda + \frac{s}{c_i}\right)^{-1} + \lambda\left(2\lambda + \frac{s}{c_i}\right)^{-1} \sum_{k\in S}[P]_{i,k}[\tilde{G}(s,\lambda)]_{k,j}.$$

Written in matrix form, this yields the required formula for $\tilde{G}_{22}(s,\lambda)$.

Now, if we define the matrices

$$A_0(s,\lambda) = \begin{pmatrix} P_{11} - (s/\lambda)C_1^{-1} & P_{12} & P_{13} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{5}$$

$$A_1(s,\lambda) = \Lambda C(sI + \Lambda C)^{-1} \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{2}P_{21} & \frac{1}{2}P_{22} & \frac{1}{2}P_{23} \\ P_{31} & P_{32} & P_{33} \end{pmatrix}, \tag{6}$$

$$A_2(s,\lambda) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \lambda C_2(sI + 2\lambda C_2)^{-1} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{7}$$

where $\Lambda = \operatorname{diag}(\lambda I, 2\lambda I, \lambda I)$, then the set of equations given by Theorem 5 can be written simply as

$$\tilde{G}(s,\lambda) = A_2(s,\lambda) + A_1(s,\lambda)\tilde{G}(s,\lambda) + A_0(s,\lambda)(\tilde{G}(s,\lambda))^2, \tag{8}$$

reminiscent of the equation for the G-matrix of a QBD; see [12] and [13]. It also yields the following corollary.

Corollary 1. *If we define*

$$\tilde{U}(s,\lambda) = A_1(s,\lambda) + A_0(s,\lambda)\tilde{G}(s,\lambda) \tag{9}$$

then, for $\operatorname{Re}(s) \geq 0$, *we have*

$$\tilde{G}(s,\lambda) = (I - \tilde{U}(s,\lambda))^{-1}A_2(s,\lambda). \tag{10}$$

Proof. We note first of all that the blocks $\tilde{G}_{ij}(s,\lambda)$ of the matrix $\tilde{G}(s,\lambda)$ vanish except for when $j = 2$; this is because fluid is depleted only when the phase is in S_2 and, therefore, the busy period must also end in S_2. If we now let $g_2(s,\lambda) = 2\lambda C_2(sI + 2\lambda C_2)^{-1}$ and $g_3(s,\lambda) = \lambda(s+\lambda)^{-1}$, then it follows from evaluating the right-hand side of (9) in partitioned form, using Theorem 5 and (1)–(4), that

$$\tilde{U}(s,\lambda) = \begin{pmatrix} 0 & \Psi(s) & 0 \\ g_2(s,\lambda)\frac{1}{2}P_{21} & g_2(s,\lambda)\frac{1}{2}P_{22} & g_2(s,\lambda)\frac{1}{2}P_{23} \\ g_3(s,\lambda)P_{31} & g_3(s,\lambda)P_{32} & g_3(s,\lambda)P_{33} \end{pmatrix}.$$

This shows that $\tilde{U}(s,\lambda)$ is a matrix of LSTs that has eigenvalues all less than 1 in absolute value and that is strictly substochastic for all $s > 0$. For complex s with $\operatorname{Re}(s) > 0$, the matrix of absolute values of this transform matrix is bounded above by a strictly substochastic matrix and, therefore, all its eigenvalues are also less than 1 in absolute value;

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

see [14, para. 2.4.9]. Thus, the inverse in (10) exists, and we can recover that equation from (8) using (9).

Note that (9) and (10) are similar to those obtained by Latouche [10] (also see [12, Chapters 8.1 and 8.2]) for the G-matrix of the QBD, and suggest an iterative procedure. Later in the paper, we will consider such an iterative scheme (Algorithm 2) and demonstrate that it converges to the required matrices.

5. An algorithm for $\tilde{G}(s, \lambda)$ through a queue

5.1. Notation and definitions

In what follows, we assume that the MMFF process $(\mathcal{F}, \mathcal{J}) = \{(F(t), J(t)), t \geq 0\}$ operates under a last-in–first-out scheme; that is, the most recently arrived fluid is purged first. This does not affect the distribution of the busy period, the quantity of interest to us. Our approach rests on constructing a closely related queue whose work process yields the fluid flow.

To facilitate the discussion, we introduce some notation and definitions below. It is helpful to illustrate the underlying ideas with the example shown in Figure 1, which depicts a path of the MMFF process; for simplicity, we have in that illustration (alone) assumed that S_3 is empty. Note that if we replace each of the upward linear segments of the path of the MMFF (during the sojourn in an exponentially distributed interval that results in the spatial uniformization) by a jump that occurs at the termination of that segment, we could interpret the resulting path as that of the work in a last-in–first-out queue. In such a queue, at the end of each sojourn of the phase process in states of S_1, a new customer with an exponentially distributed amount of work with mean λ^{-1} joins the queue, and work is depleted only while the phase process is in S_2, and, in particular, at rate per unit time c_j when in state $j \in S_2$. The queue is quite complex, in that the amount of work brought by a customer is directly proportional to the length of the sojourn in S_1 whose end marks the arrival epoch of that customer. What is, however, noteworthy is that with this correspondence between the MMFF and the queue, the busy period of the MMFF process is identical to the corresponding busy period of the queue. When such a busy period of the MMFF starts with an exponentially distributed amount of fluid, the distribution of the busy period of the MMFF is identical to the busy period of the queue initiated by an arrival to an

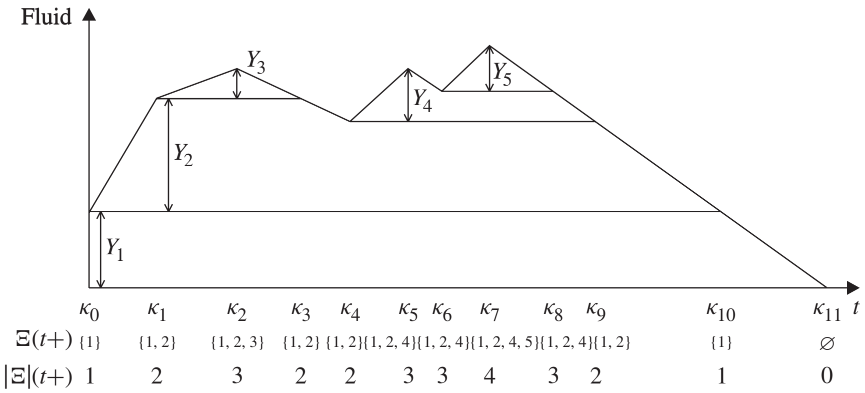
Fluid
Y_3
Y_5
Y_4
Y_2
Y_1
κ_0 κ_1 κ_2 κ_3 κ_4 κ_5 κ_6 κ_7 κ_8 κ_9 κ_{10} κ_{11} t
$\Xi(t+)$ $\{1\}$ $\{1, 2\}$ $\{1, 2, 3\}$ $\{1, 2\}$ $\{1, 2\}$ $\{1, 2, 4\}$ $\{1, 2, 4\}$ $\{1, 2, 4, 5\}$ $\{1, 2, 4\}$ $\{1, 2\}$ $\{1\}$ $\varnothing$
$|\Xi|(t+)$ 1 2 3 2 2 3 3 4 3 2 1 0

Figure 1: A path of the MMFF process (with S_3 empty).

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

empty system. Many of the terms we introduce will also be interpreted in terms of this queue, and this will allow us to draw freely from the matrix-geometric literature.

Here is some of the notation we will use.

1. Let $\{0 = \tau_0 < \tau_1 < \cdots\}$ denote the set of successive spatial uniformization epochs of $\mathcal{J}$.

2. Let Y_1 denote the initial amount of fluid (at time 0), which we assume to be exponentially distributed with mean $1/\lambda$. Let Y_{n+1}, $n \geq 1$, denote the amount of fluid coming in during the nth sojourn in S_1 of the (spatially uniformized) phase process. Note that, because of the spatial uniformization, Y_n, $n \geq 2$, are all also exponentially distributed with mean $1/\lambda$.

3. Let $n \geq 1$ and assume that the index $r(n)$ is such that $\tau_{r(n)}$ is the epoch of the nth visit to S_1. (If $J(0) \in S_1$ then we treat 0 as the epoch of the first visit to S_1.) Now, for $t \geq \tau_{r(n)+1}$, define $Y_{n+1}(t)$ to be the amount of fluid, out of Y_{n+1}, which remains in the system at time t. Note that $Y_{n+1}(\tau_{r(n)+1}) = Y_{n+1}$. Similarly, let us denote by $Y_1(t)$ the amount of fluid, out of Y_1, which remains in the system at time $t > 0$. The epoch during which a $Y_n(t)$ attains the value 0 is clearly a departure epoch for the last-in–first-out queue; in Figure 1, such epochs correspond to the points where right-hand ends the of lines drawn parallel to the t axis meet the path of the MMFF.

4. Now consider the spatial uniformization epochs $\{\tau_n\}$ along with the departure epochs identified above, and denote the resulting set of ordered epochs by κ_n; we have

$$0 = \kappa_0 < \kappa_1 < \kappa_2 < \cdots \quad \text{almost surely.}$$

5. Let $\Xi(t)$ denote the set of indices $n = 1, 2, \dots$ such that $Y_n(t) > 0$. We define $|\Xi|(t)$ to be the total number of elements in $\Xi(t)$, and we represent the set $\Xi(t)$ as

$$\Xi(t) = \{n_1(t), \dots, n_{|\Xi|(t)}(t)\},$$

where $n_j(t)$ denotes the jth-largest index among the indices in the set $\Xi(t)$. Defined thus, the set $\Xi(t)$ gives the identities of the customers still present in the queue and $|\Xi|(t)$ the total number of customers present at time t. In Figure 1, we have shown the epochs κ_n in a busy period along with the corresponding sets Ξ and their cardinalities at those epochs. Note that, almost surely, $|\Xi|(t) = 0$ if and only if $F(t) = 0$.

Armed with this notation, we are now ready to state the following important result, which is a direct consequence of the memoryless property of the exponential distribution and can be established by mathematical induction (we omit the proof).

Theorem 6. *Given that $|\Xi|(\kappa_n) = m$, the fluid level $F(\kappa_n)$ (or, equivalently, the total amount of work in the queue at κ_n) is distributed as the sum of m independent, identically distributed random variables with common distribution* Exp(λ). *Furthermore, the residual amounts of work for the m customers in the queue at κ_n are independent and identically distributed with distribution* Exp(λ).

Remark 1. The above result shows that if the fluid level (i.e. the amount of work in the system) at κ_0 is distributed as Exp(λ), then knowing the number of customers present at κ_n determines the law of the MMFF process in the interval $[\kappa_n, \infty)$. In fact, if N_n denotes the level at κ_n+, then (κ_n, N_n), $n \geq 0$, form a semiregenerative sequence (see [6, Chapter 10.6]) for the MMFF process as well as for the work in the queue.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

For the queue, we now introduce a set of random variables similar to those introduced by Latouche [10] in the computation of G in a QBD. Given that $|\Xi|(\kappa_m) = n \geq 1$, let κ_U denote the first epoch in $\{\kappa_j, j \geq m+1\}$ for which $|\Xi|(\kappa_j) = n$. Also, let κ_G denote the first epoch in $\{\kappa_j, j \geq m+1\}$ for which $|\Xi|(\kappa_j) = n-1$. Thus, if by 'level' n we denote the set of states with queue length n, then $\kappa_U - \kappa_m$ is a return time to level n avoiding lower levels, and $\kappa_G - \kappa_m$ is the first passage time to the (immediately lower) level $n-1$, given that we start in level n. Finally, we denote the amount of remaining work of the customer in service at κ_m by X_m.

Now, let $\hat{U}(k, s, x)$ denote the transform matrix such that

$$\begin{aligned}[]
&[\hat{U}(k,s,x)]_{i,j} \\
&\quad = \mathrm{E}[\mathrm{e}^{-s(\kappa_U-\kappa_m)} 1(J(\kappa_U) = j, n \leq |\Xi|(t) < n+k \text{ for all } t \in [\kappa_m, \kappa_U)) \mid |\Xi|(\kappa_m) = n, \\
&\qquad\qquad J(\kappa_m) = i, X_m = x].
\end{aligned}$$

Also, let

$$\begin{aligned}[]
&[\hat{G}(k,s,x)]_{i,j} \\
&\quad = \mathrm{E}[\mathrm{e}^{-s(\kappa_G-\kappa_m)} 1(J(\kappa_G) = j, n \leq |\Xi|(t) < n+k \text{ for all } t \in [\kappa_m, \kappa_G)) \mid |\Xi|(\kappa_m) = n, \\
&\qquad\qquad J(\kappa_m) = i, X_m = x].
\end{aligned}$$

From the structure of the process under consideration, it is clear that the above transform matrices do not depend on n. Note that these matrices, of course, depend on the uniformization parameter λ, but we have suppressed this fact to simplify our notation.

We also define the matrices $\tilde{U}(k, s, \lambda)$ and $\tilde{G}(k, s, \lambda)$ (making their dependence on λ explicit) as

$$\begin{aligned}
\tilde{U}(k,s,\lambda) &= \int_0^\infty \lambda \mathrm{e}^{-\lambda x} \hat{U}(k,s,x)\,\mathrm{d}x, \\
\tilde{G}(k,s,\lambda) &= \int_0^\infty \lambda \mathrm{e}^{-\lambda x} \hat{G}(k,s,x)\,\mathrm{d}x. \qquad (11)
\end{aligned}$$

The following results follow from these definitions.

Lemma 1. (a) *If we let* $g_2(s,\lambda) = 2\lambda C_2(sI + 2\lambda C_2)^{-1}$ *and* $g_3(s,\lambda) = \lambda(s+\lambda)^{-1}$*, then* $\tilde{U}(k,s,\lambda)$ *has the following form:*

$$\tilde{U}(k,s,\lambda) = \begin{pmatrix} 0 & \tilde{U}_{12}(k,s,\lambda) & 0 \\ g_2(s,\lambda)\frac{1}{2}P_{21} & g_2(s,\lambda)\frac{1}{2}P_{22} & g_2(s,\lambda)\frac{1}{2}P_{23} \\ g_3(s,\lambda)P_{31} & g_3(s,\lambda)P_{32} & g_3(s,\lambda)P_{33} \end{pmatrix}. \qquad (12)$$

(b) *For* $\mathrm{Re}(s) \geq 0$, $\tilde{U}_{12}(k,s)$ *converges as* k *increases, such that*

$$\lim_{k\to\infty} \tilde{U}_{12}(k,s,\lambda) = \Psi(s).$$

Furthermore, for $s \geq 0$ *the matrices* $\tilde{U}_{12}(k,s,\lambda)$ *are (entrywise) monotone nondecreasing. That is,*

$$\tilde{U}(k,s,\lambda) \to \tilde{U}(s,\lambda) \text{ as } k \to \infty,$$

and the convergence is monotone for $s \geq 0$.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

Proof. In part (a), the submatrices in the second and third rows are obtained by noting that the first return to a given level, avoiding lower levels, occurs at the first step of the spatial uniformization if and only if the time to that step is less than the amount of time taken to serve the customer in service at time 0. The zero elements appear in the first row because, when a return to a given level occurs from S_1, the phase returned to must be in S_2. All the other results follow immediately upon noting that the set of paths that go up to level $n+k-1$ in a return to level n, avoiding lower levels, form a nondecreasing set converging to the set of *all* paths returning to level n avoiding lower levels.

Lemma 2. *Let* $\mathrm{Re}(s) \geq 0$. *Then the following statements hold.*

(a) $\hat{G}_{12}(k,s,\lambda) = \tilde{U}_{12}(k,s,\lambda)\hat{G}_{22}(k,s,\lambda)$.

(b) $\hat{G}_{32}(k,s,\lambda) = (sI - Q_{33})^{-1}Q_{31}\hat{G}_{12}(k,s,\lambda) + (sI - Q_{33})^{-1}Q_{32}\hat{G}_{22}(k,s,\lambda)$.

(c) *If we let*

$$\begin{aligned} H(k,s) = C_2^{-1}[&Q_{22} - sI + Q_{23}(sI - Q_{33})^{-1}Q_{32} \\ &+ \{Q_{21} + Q_{23}(sI - Q_{33})^{-1}Q_{31}\}\tilde{U}_{12}(k,s,\lambda)], \end{aligned}$$

then

$$\hat{G}_{22}(k,s,x) = \mathrm{e}^{H(k,s)x}.$$

(d) $\tilde{G}_{22}(k,s,\lambda) = \lambda(\lambda I - H(k,s))^{-1}$.

(e) *For* $\mathrm{Re}(s) \geq 0$, $H(k,s) \to H(s)$, *and* $\tilde{G}(k,s,\lambda) \to \tilde{G}(s,\lambda)$ *as* $k \uparrow \infty$. *Furthermore, for* $s \geq 0$, $\tilde{G}(k,s,\lambda) \uparrow \tilde{G}(s,\lambda)$ *as* $k \uparrow \infty$.

Proof. Part (a) follows by conditioning on the first epoch of return to level 1 before the end of the busy period. Part (b) follows by conditioning on the first exit time from S_3. Part (c) is proven along the same along the same lines as Theorem 3(c), and immediately yields part (d). The proof of part (e) follows from the simple observation that, as $k \to \infty$, the set of paths yielding a first passage from level 1 to level 0, avoiding level $n+k$, form an increasing set converging to the set of *all* paths yielding a first passage from level 1 to level 0.

5.2. Relation between $\tilde{U}$ and $\tilde{G}$

For the following analysis, we introduce the operator vec defined [8], on matrices $A = (a_{ij})$ of order $m \times n$, by

$$\mathrm{vec}(A) = (a_{11} \cdots a_{m1} \cdots a_{1n} \cdots a_{mn})^{\top},$$

where '$^\top$' denotes the transpose operator. We then have the following lemma.

Lemma 3. *Given an* $m \times m$ *matrix* A, *an* $n \times n$ *matrix* B, *and an* $m \times n$ *matrix* Y,

$$\mathrm{vec}(AYB) = (B^{\top} \otimes A)\,\mathrm{vec}(Y),$$

where $\otimes$ *denotes the Kronecker product of matrices.*

Now we can prove the following result, establishing a relationship between $\tilde{U}$ and $\tilde{G}$.

Lemma 4. $\tilde{U}$ *and* $\tilde{G}$ *are such that the following statements hold.*

(a) $\tilde{G}(k,s,\lambda) = (I - \tilde{U}(k,s,\lambda))^{-1}A_2(s,\lambda)$.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

(b) *The submatrices* $\tilde{U}_{12}(k,s,\lambda)$ *in (12) are such that* $\tilde{U}_{12}(1,s,\lambda)=0$ *and, for* $k\geq 2$*,*

$$\begin{aligned}\tilde{U}_{12}(k,s,\lambda) &= P_{11}\tilde{G}_{12}(k-1,s,\lambda)+P_{12}\tilde{G}_{22}(k-1,s,\lambda)\\ &\quad + P_{13}\tilde{G}_{32}(k-1,s,\lambda)-\frac{s}{\lambda}C_1^{-1}\tilde{U}_{12}(k,s,\lambda)\tilde{G}_{22}(k-1,s,\lambda).\end{aligned}$$

(c) $$\begin{aligned}\operatorname{vec}(\tilde{U}_{12}(k,s,\lambda)) &= \left(I+\frac{s}{\lambda}\tilde{G}_{22}^{\top}(k-1,s,\lambda)\otimes C_1^{-1}\right)^{-1}\\ &\quad\times \operatorname{vec}(P_{11}\tilde{G}_{12}(k-1,s,\lambda)+P_{12}\tilde{G}_{22}(k-1,s,\lambda)\\ &\quad + P_{13}\tilde{G}_{32}(k-1,s,\lambda)).\end{aligned}$$

Proof. Recall the definitions of κ_U and κ_G from Section 5.1. Given the state $J(\kappa_U)$, it is clear that κ_G and $\kappa_G-\kappa_U$ are independent, and that the distribution of $\kappa_G-\kappa_U$ given $J(\kappa_U)=j$ is identical to the distribution of κ_G given $J(0)=j$. Therefore,

$$\tilde{G}(k,s,\lambda)=A_2(s,\lambda)+\tilde{U}(k,s,\lambda)\tilde{G}(k,s,\lambda),$$

which completes the proof of part (a). Part (c) is a direct consequence of part (b) and Lemma 3, so we only need to prove part (b). Now, without loss of generality, by Lemma 2(b) we can assume that $S=S_1\cup S_2$ (for the purposes of the proof). By its definition, it is trivial that $\tilde{U}_{12}(1,s,\lambda)=0$ since, given an initial state in S_1, the level increases in the very first step of the spatial uniformization. Thus, part (b) holds for $k=1$. Assume, as an induction hypothesis, that it holds for some $k-1$ with $k\geq 2$. Now consider k, and assume that $i\in S_1$ and $j\in S_2$. We then have

$$\begin{aligned}[\tilde{U}_{12}(k,s,\lambda)]_{i,j} &= \int_0^\infty \lambda c_i \mathrm{e}^{-\lambda c_i t}\mathrm{e}^{-st}\sum_{l\in S}[P]_{i,l}[\hat{G}]_{l,j}(k-1,s,c_i t)\,\mathrm{d}t\\ &= \int_0^\infty \lambda \mathrm{e}^{-\lambda y}\mathrm{e}^{-s/c_i y}\sum_{l\in S}[P]_{i,l}[\hat{G}]_{l,j}(k-1,s,y)\,\mathrm{d}y,\end{aligned}$$

and it follows from the induction assumption that

$$\begin{aligned}\tilde{U}_{12}(k,s,\lambda) &= \lambda\int_0^\infty \mathrm{e}^{-\lambda y}\mathrm{e}^{-sC_1^{-1}y}P_{11}\hat{G}_{12}(k-1,s,y)\,\mathrm{d}y\\ &\quad+\lambda\int_0^\infty \mathrm{e}^{-\lambda y}\mathrm{e}^{-sC_1^{-1}y}P_{12}\hat{G}_{22}(k-1,s,y)\,\mathrm{d}y\\ &= \lambda\int_0^\infty \mathrm{e}^{-sC_1^{-1}y}[P_{11}\tilde{U}_{12}(k-1,s,\lambda)+P_{12}]\exp\{-(\lambda I-H(k-1,s))y\}\,\mathrm{d}y.\end{aligned}$$

Using integration by parts in the expression above, and the results in Lemma 2, we now find that

$$\begin{aligned}\tilde{U}_{12}(k,s,\lambda) &= P_{11}\tilde{G}_{12}(k-1,s,\lambda)+P_{12}\tilde{G}_{22}(k-1,s,\lambda)\\ &\quad-\frac{s}{\lambda}C_1^{-1}\tilde{U}_{12}(k,s,\lambda)\tilde{G}_{22}(k-1,s,\lambda),\end{aligned}$$

and the proof follows by induction.

From Lemma 4, we can now construct the following iterative scheme.

Algorithm 1. Let $\operatorname{Re}(s)\geq 0$ and fix $\varepsilon>0$, $k=1$, diff $=100$, and $\lambda>0$ such that

$$\lambda\geq\max_{i\in S}\{-[C^{-1}Q]_{ii}\}.$$

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

```
Initialize as $\tilde{U}(1, s, \lambda) = A_1(s, \lambda)$ and $\tilde{G}(1, s, \lambda) = (I - \tilde{U}(1, s, \lambda))^{-1} A_2(s, \lambda)$.
  do while (diff $> \varepsilon$)
    $k = k + 1$;
    $\tilde{U}(k, s, \lambda) = A_1(s, \lambda)$;
    $M = (I + (s/\lambda)\tilde{G}_{22}^{\top}(k, s, \lambda) \otimes C_1^{-1})^{-1}$;
    $N = \mathrm{vec}(P_{11}\tilde{G}_{12}(k, s, \lambda) + P_{12}\tilde{G}_{22}(k, s, \lambda) + P_{13}\tilde{G}_{32}(k, s, \lambda))$;
    $\mathrm{vec}(\tilde{U}_{12}(k, s, \lambda)) = MN$;
    $\tilde{G}(k, s, \lambda) = (I - \tilde{U}(k, s, \lambda))^{-1} A_2(s, \lambda)$;
    diff $= \max_{i,j \in S} |[\tilde{G}(k, s, \lambda)]_{i,j} - [\tilde{G}(k-1, s, \lambda)]_{i,j}|$;
  end
  $\Psi(s) \approxeq \tilde{U}_{12}(k, s, \lambda)$; $\tilde{G}(s, \lambda) \approxeq \tilde{G}(k, s, \lambda)$.
```

By Lemma 4, the iterates in Algorithm 1 are such that the kth iterates $\tilde{U}(k, s, \lambda)$ and $\tilde{G}(k, s, \lambda)$ are respectively the quantities in (11) defined using the forbidden paths of the MMFF. Therefore, they converge to the required matrices $\Psi(s)$ and $\tilde{G}(s, \lambda)$ as $k \uparrow \infty$, as shown in Lemmas 1 and 2. We have already shown that, for $s \geq 0$, the convergence is (entrywise) monotonic. Furthermore, the convergence is linear since each successive iteration includes paths that go up by one more level during a busy period; this is similar to the linear algorithm of Latouche [10]. Indeed, by the results of [12, Chapter 8], the difference between the limit values and the kth iterates are asymptotically $O([\eta(s)]^k)$ as $k \to \infty$, where $0 < \eta(s) < 1$ is the minimal solution in $(0, 1)$ of the equation $\mathrm{sp}(A_0(s) + \eta(s)A_1(s) + [\eta(s)]^2 A_2(s))$. (Here $\mathrm{sp}(A)$ denotes the spectral radius of the matrix A.)

Remark 2. Although, in principle, the iterates of Algorithm 1 converge as required, when implemented on a computer (a finite arithmetic machine), we have found it to misbehave due to round-offs and truncations. A thorough numerical analysis of the iterative schemes given in this paper has not been made. However, it is easy to show that if we choose λ using a more stringent criterion; namely that if, in addition to the condition $\lambda \geq \max_{i \in S}\{-[C^{-1}Q]_{ii}\}$ of spatial uniformization, we also require that

$$\max_{i \in S}\left[\frac{\mathrm{Re}(s)}{\lambda}C^{-1}\right]_{i,i} \leq \delta < 1 \quad \text{and} \quad 0 < \max_{i \in S}\left[P_\lambda - \frac{\mathrm{Re}(s)}{\lambda}C^{-1}\right]_{i,i},$$

then the iterates remain within a bounded region of the complex plane and behave well. This becomes obvious from the easily verified fact that, under these conditions, the matrices $A_i(\mathrm{Re}(s), \lambda)$, $i = 0, 1, 2$, are nonnegative and strictly substochastic, and sum to a strictly substochastic matrix. Thus, the matrices $\tilde{U}(k, \mathrm{Re}(s), \lambda)$ and $\tilde{G}(k, \mathrm{Re}(s), \lambda)$ both remain strictly substochastic. We therefore recommend implementing all the algorithms in this paper using this more stringent scheme, so that numerical stability is maintained in the presence of round-offs and truncations.

Henceforth, we will assume that, for each s, an appropriate $\lambda(s)$ meeting the stringent criteria established above is being used. However, to simplify our notation, we will simply write λ, suppressing the dependence of λ on s.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

6. Quadratically convergent algorithm

An iterative procedure is said to have linear convergence if the error in the kth iteration is $O(\eta^k)$ as $k \to \infty$, and to have quadratic convergence if that error is $O(\eta^{2^k})$, for some $0 < \eta < 1$. Having obtained a linear algorithm for $\Psi(s)$, we now examine an algorithm resulting from Corollary 1, and then a faster version thereof. The faster version will be shown to have quadratic convergence.

The following is an iterative scheme obtained by bootstrapping in the equations of Corollary 1.

6.1. Linear algorithm

Algorithm 2. Fix $\varepsilon > 0$ and set diff $= 100$. Initialize as $U^*(1,s,\lambda) = A_1(s,\lambda)$ and $G^*(1,s,\lambda) = (I - U^*(1,s,\lambda))^{-1}A_2(s,\lambda)$.

```
do while (diff $> \varepsilon$)
    $k = k + 1;$
    $U^*(k,s,\lambda) = A_1(s,\lambda) + A_0(s,\lambda)G^*(k-1,s,\lambda);$
    $G^*(k,s,\lambda) = (I - U^*(k,s,\lambda))^{-1}A_2(s,\lambda);$
    diff $= \max_{i,j\in S} |[G^*(k,s,\lambda)]_{i,j} - [G^*(k-1,s,\lambda)]_{i,j}|;$
end
$\Psi(s) \approxeq U^*_{12}(k,s,\lambda);\ \tilde{G}(s,\lambda) \approxeq G^*(k,s,\lambda).$
```

Comparing the matrices $U^*(k,s,\lambda)$ and $G^*(k,s,\lambda)$ of Algorithm 2 with $\tilde{U}(k,s,\lambda)$ and $\tilde{G}(k,s,\lambda)$ of Algorithm 1, we can see that

(a) $U^*(1,s,\lambda) = \tilde{U}(1,s,\lambda)$ and $G^*(1,s,\lambda) = \tilde{G}(1,s,\lambda)$;

(b) $U^*_{lm}(k,s,\lambda) = \tilde{U}_{lm}(k,s,\lambda)$ for $l = 2, 3$, $m = 1, 2, 3$, and $k = 1, 2, \dots$; and,

(c) for all $k = 1, 2, \dots,$

$$\begin{aligned} G^*(k,s,\lambda) &= (I - U^*(k,s,\lambda))^{-1}A_2(s,\lambda), \\ \tilde{G}(k,s,\lambda) &= (I - \tilde{U}(k,s,\lambda))^{-1}A_2(s,\lambda). \end{aligned}$$

Thus, the difference in the two algorithms arises from the difference in the iterates $U^*_{12}(k,s,\lambda)$ and $\tilde{U}_{12}(k,s,\lambda)$. As we can see in Lemma 4(b), $\tilde{U}_{12}(k,s,\lambda)$ satisfies

$$\begin{aligned} \tilde{U}_{12}(k,s,\lambda) = {} & P_{11}\tilde{G}_{12}(k-1,s,\lambda) + P_{12}\tilde{G}_{22}(k-1,s,\lambda) + P_{13}\tilde{G}_{32}(k-1,s,\lambda) \\ & - \frac{s}{\lambda}C_1^{-1}\tilde{U}_{12}(k,s,\lambda)\tilde{G}_{22}(k-1,s,\lambda), \end{aligned} \tag{13}$$

while $U^*_{12}(k,s,\lambda)$ in Algorithm 2 satisfies

$$\begin{aligned} & U^*_{12}(k,s,\lambda) \\ & \quad = \left(P_{11} - \frac{s}{\lambda}C_1^{-1}\right)G^*_{12}(k-1,s,\lambda) + P_{12}G^*_{22}(k-1,s,\lambda) + P_{13}G^*_{32}(k-1,s,\lambda) \\ & \quad = P_{11}G^*_{12}(k-1,s,\lambda) + P_{12}G^*_{22}(k-1,s,\lambda) + P_{13}G^*_{32}(k-1,s,\lambda) \\ & \quad\quad - \frac{s}{\lambda}C_1^{-1}U^*_{12}(k-1,s,\lambda)G^*_{22}(k-1,s,\lambda). \end{aligned} \tag{14}$$

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

We draw particular attention to the indices of the U matrices arising in (13) and (14). Note that (13) yields a linear equation for the unknown that needs to be solved, while (14) is a true recursion. A result we will establish soon is that, despite these differences, the iterates in both algorithms converge to the same matrices.

We begin with the following result, whose proof by mathematical induction is quite straightforward and, therefore, omitted.

Lemma 5. (a) *For $s \geq 0$, the matrices $U^*(k, s, \lambda)$ and $G^*(k, s, \lambda)$ are monotone nondecreasing as k increases.*

(b) *For all $k \geq 1$ and $s \geq 0$, the matrices $U^*(k, s, \lambda)$ and $G^*(k, s, \lambda)$ are nonnegative and strictly substochastic.*

Let $U^*(s, \lambda) = \lim_{k\to\infty} U^*(k, s, \lambda)$ and $G^*(s, \lambda) = \lim_{k\to\infty} G^*(k, s, \lambda)$, for s such that $\mathrm{Re}(s) > 0$. Our next result shows that, for $s > 0$, $U^*(s, \lambda) = \tilde{U}(s, \lambda)$ and $G^*(s, \lambda) = \tilde{G}(s, \lambda)$, so that the iterative schemes in Algorithm 1 and Algorithm 2 yield the same results for $s > 0$.

Lemma 6. *Let $s \geq 0$. Then the following statements hold.*

(a) *For all $k \geq 1$, $U^*(k, s, \lambda) \geq \tilde{U}(k, s, \lambda)$ and $G^*(k, s, \lambda) \geq \tilde{G}(k, s, \lambda)$.*

(b) $U^*_{12}(s, \lambda) = \Psi(s)$.

(c) $U^*(s, \lambda) = \tilde{U}(s, \lambda)$ *and* $G^*(s, \lambda) = \tilde{G}(s, \lambda)$.

Proof. (a) We will prove this part by induction. Note that $U^*(1, s, \lambda) = \tilde{U}(1, s, \lambda)$ and $G^*(1, s, \lambda) = \tilde{G}(1, s, \lambda)$. From (13) and (14), we can see that

$$\begin{aligned}\tilde{U}_{12}(k, s, \lambda) = {} & P_{11}\tilde{G}_{12}(k-1, s, \lambda) + P_{12}\tilde{G}_{22}(k-1, s, \lambda) + P_{13}\tilde{G}_{32}(k-1, s, \lambda)\\ & - \frac{s}{\lambda} C_1^{-1}\tilde{U}_{12}(k-1, s, \lambda)\tilde{G}_{22}(k-1, s, \lambda)\\ & - \frac{s}{\lambda} C_1^{-1}[\tilde{U}_{12}(k, s, \lambda) - \tilde{U}_{12}(k-1, s, \lambda)]\tilde{G}_{22}(k-1, s, \lambda)\end{aligned}$$

and

$$\begin{aligned}& U^*(2, s, \lambda) - \tilde{U}(2, s, \lambda)\\ &\quad = [A_1(s, \lambda) + A_0(s, \lambda)G^*(1, s, \lambda)]\\ &\qquad - \left[A_1(s, \lambda) + A_0(s, \lambda)\tilde{G}(1, s, \lambda) - \frac{s}{\lambda}[\tilde{U}(2, s, \lambda) - \tilde{U}(1, s, \lambda)]\tilde{G}(1, s, \lambda)\right]\\ &\quad = \frac{s}{\lambda}[\tilde{U}(2, s, \lambda) - \tilde{U}(1, s, \lambda)]\tilde{G}(1, s, \lambda)\\ &\quad \geq 0.\end{aligned}$$

It then follows that

$$\begin{aligned}& G^*(2, s, \lambda) - \tilde{G}(2, s, \lambda)\\ &\quad = (I - U^*(2, s, \lambda))^{-1}A_2(s, \lambda) - (I - \tilde{U}(2, s, \lambda))^{-1}A_2(s, \lambda)\\ &\quad = \sum_{i=1}^{\infty}[\{U^*(2, s, \lambda)\}^i - \{\tilde{U}(2, s, \lambda)\}^i]A_2(s, \lambda)\\ &\quad \geq 0,\end{aligned}$$

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

because $U^*(2,s,\lambda) \geq \tilde{U}(2,s,\lambda)$. For $k \geq 2$, if we assume that $U^*(k,s,\lambda) - \tilde{U}(k,s,\lambda) \geq 0$ and $G^*(k,s,\lambda) - \tilde{G}(k,s,\lambda) \geq 0$, then

$$\begin{aligned} U^*(k+1,s,\lambda) &- \tilde{U}(k+1,s,\lambda) \\ &= [A_1(s,\lambda) + A_0(s,\lambda)G^*(k,s,\lambda)] \\ &\quad - \left[A_1(s,\lambda) + A_0(s,\lambda)\tilde{G}(k,s,\lambda) - \frac{s}{\lambda}[\tilde{U}(k+1,s,\lambda) - \tilde{U}(k,s,\lambda)]\tilde{G}(k,s,\lambda)\right] \\ &= A_0(s)[G^*(k,s,\lambda) - \tilde{G}(k,s,\lambda)] + \frac{s}{\lambda}[\tilde{U}(k+1,s,\lambda) - \tilde{U}(k,s,\lambda)]\tilde{G}(k,s,\lambda) \\ &\geq 0, \end{aligned}$$

and

$$\begin{aligned} G^*(k+1,s,\lambda) &- \tilde{G}(k+1,s,\lambda) \\ &= (I - U^*(k+1,s,\lambda))^{-1}A_2(s,\lambda) - (I - \tilde{U}(k+1,s,\lambda))^{-1}A_2(s,\lambda) \\ &= \sum_{i=1}^{\infty}[U^*(k+1,s,\lambda)^i - \tilde{U}(k+1,s,\lambda)^i]A_2(s,\lambda) \\ &\geq 0. \end{aligned}$$

This completes the proof by induction of part (a). Note that part (c) is a direct consequence of part (b) and Corollary 1, so we only need to prove part (b). From part (a) of this lemma and part (b) of Lemma 1, it is enough to show that

$$U^*_{12}(s,\lambda) \leq \Psi(s) = \tilde{U}_{12}(s,\lambda),$$

and we will prove this by induction. First, we can easily see from Corollary 1 that

$$U^*(1,s,\lambda) = A_1 \leq \tilde{U}(s,\lambda),$$

which implies that

$$G^*(1,s,\lambda) = (I - U^*(1,s,\lambda))^{-1}A_2 \leq \tilde{U}(s,\lambda))^{-1}A_2.$$

Now, for $k \geq 1$, assume that

$$U^*(k,s,\lambda) \leq \tilde{U}(s,\lambda) \quad \text{and} \quad G^*(k,s,\lambda) \leq \tilde{G}(s,\lambda).$$

Then, from Algorithm 2,

$$U^*(k+1,s,\lambda) = A_1 + A_0G^*(k,s,\lambda) \leq A_1 + A_0\tilde{G}(s,\lambda) = \tilde{U}(s,\lambda),$$

which implies that

$$G^*(k+1,s,\lambda) = (I - U^*(k+1,s,\lambda))^{-1}A_2 \leq (I - \tilde{U}(s,\lambda))^{-1}A_2 = \tilde{G}(s,\lambda).$$

Therefore,

$$\lim_{k\to\infty} U^*(k,s,\lambda) \leq \tilde{U}(s,\lambda) \quad \text{and} \quad \lim_{k\to\infty} G^*(k,s,\lambda) \leq \tilde{G}(s,\lambda).$$

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

Having established the above lemma, it is now a trivial matter to see that, for all s such that $\mathrm{Re}(s) > 0$, the two iterative schemes converge and yield the same transforms. We can prove this by appealing to an analytic continuation argument or directly, using the dominated convergence theorem, by comparing the matrices of absolute values of the iterates. We omit the details but, in light of its importance, we state the result as a theorem.

Theorem 7. *Let* $\mathrm{Re}(s) \geq 0$. *Then the iterative schemes in Algorithms 1 and 2 yield the required matrices* $\tilde{U}$ *and* $\tilde{G}$ *in the limit as* $k \to \infty$.

6.2. A quadratically convergent algorithm

We now present a faster version of Algorithm 2.

Algorithm 3. Fix $\varepsilon > 0$ and set diff $= 100$. Initialize as $H^{**}(1, s, \lambda) = (I - A_1(s, \lambda))^{-1} \times A_0(s, \lambda)$, $L^{**}(1, s, \lambda) = (I - A_1(s, \lambda))^{-1} A_2(s, \lambda)$, $G^{**}(1, s, \lambda) = L^{**}(1, s, \lambda)$, and $T(1) = H^{**}(1, s, \lambda)$.

```
do while (diff $> \varepsilon$)
    $k = k + 1$;
    $U^{**}(k, s, \lambda) = H^{**}(k-1, s, \lambda) L^{**}(k-1, s, \lambda) + L^{**}(k-1, s, \lambda) H^{**}(k-1, s, \lambda)$;
    $M = (H^{**}(k-1, s, \lambda))^2$;
    $H^{**}(k, s, \lambda) = (I - U^{**}(k, s, \lambda))^{-1} M$;
    $M = (L^{**}(k-1, s, \lambda))^2$;
    $L^{**}(k, s, \lambda) = (I - U^{**}(k, s, \lambda))^{-1} M$;
    $G^{**}(k, s, \lambda) = G^{**}(k-1, s, \lambda) + T(k-1) L^{**}(k, s, \lambda)$;
    $T(k) = T(k-1) H^{**}(k, s, \lambda)$;
    diff $= \max_{i,j \in S} |[G^{**}(k, s, \lambda)]_{i,j} - [G^{**}(k-1, s, \lambda)]_{i,j}|$;
end
$\Psi(s) \approxeq G^{**}_{12}(k, s, \lambda)[G^{**}_{22}(k, s, \lambda)]^{-1}$; $\tilde{G}(s, \lambda) \approxeq G^{**}(k, s, \lambda)$.
```

Theorem 8. *As* $k \to \infty$, $G^{**}(k, s, \lambda)$ *converges quadratically and, furthermore,*

$$\lim_{k \to \infty} G^{**}(k, s, \lambda) = G^{*}(s, \lambda) = \tilde{G}(s, \lambda).$$

Proof. As before, it suffices to prove the result for $s \geq 0$. In this case, the iterative scheme of Algorithm 2 is precisely the linear iteration scheme of Latouche ([12, p. 170]) and Algorithm 3 is the Latouche–Ramaswami algorithm (see [12, p. 193]) for the QBD defined by the nonnegative matrices $A_i(s, \lambda)$, $i = 0, 1, 2$. Thus, the result follows from [11]; see also Chapter 8 of [12] since the kth iterate of Algorithm 3 is in fact the (2^k)th iterate of Algorithm 2. A direct proof can be given in terms of forbidden paths that go up by at most 2^k levels and by identifying the kth iteration here as resulting from the restriction to such paths; however, given that these details can be found in the cited references, we omit them here. Incidentally, the results in [12, Chapter 8] also show that the error in the kth iterate is asymptotically $O([\eta(s)]^{2^k})$.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

Remarks 3. (a) Note that a comparison of Algorithm 2 with the linear scheme of Latouche at best only shows that the former converges. The fact that it converges to $\Psi(s)$ has been established by us by comparing its iterates to those of Algorithm 1. Unfortunately, we have not been able to develop an argument leading to Algorithm 3 directly.

(b) With $s = 0$, Algorithm 3 provides a powerful means to compute $\Psi(0)$, using which it is easy [1] to compute the steady-state distribution of the fluid flow, when it exists.

7. Numerical results

We have shown that all the iterative schemes developed by us converge to the quantities of interest. To facilitate their implementation, we have devised procedures that yield numerically stable schemes that do not appear to suffer from round-off and truncation errors. While a careful error analysis of the algorithms has not been possible yet, our experimentation thus far has confirmed that the procedures developed by us are sound and work well. In this section, we report on a model class studied earlier by Sericola [17] and also used by the authors in [2, Section 8] for comparison.

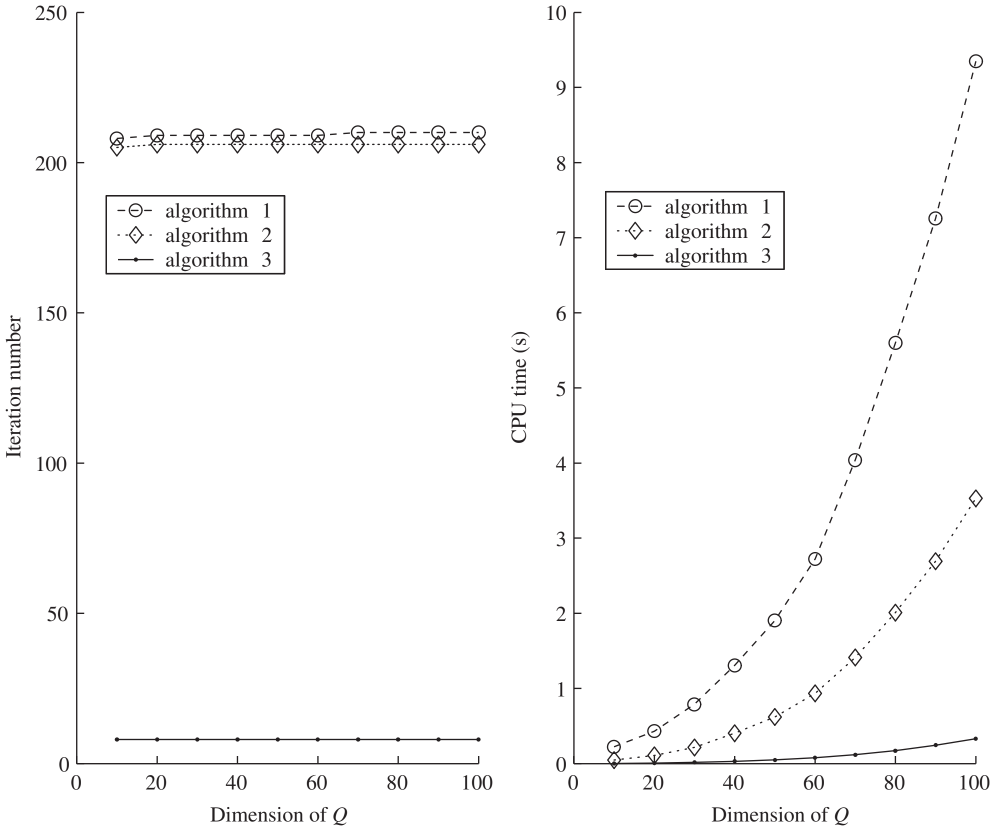

FIGURE 2: Comparison of the algorithms in the case $s = 2$ and $\rho = 0.9$.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

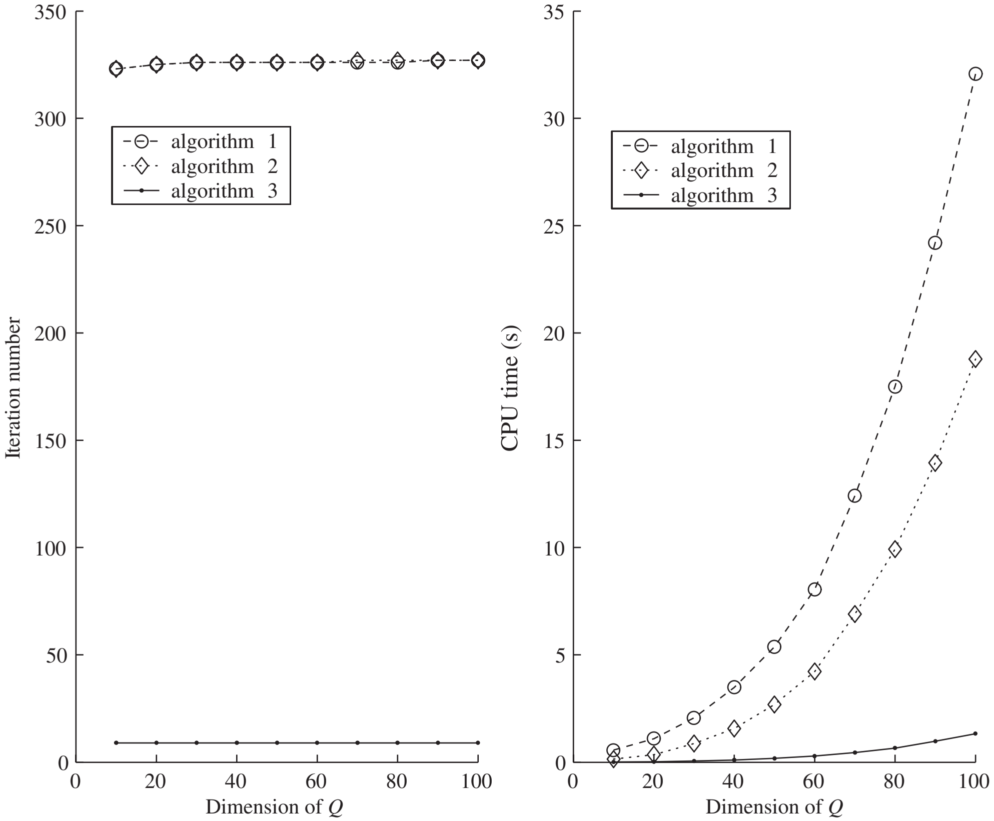

FIGURE 3: Comparison of the algorithms in the case $s = 2 + 3\mathrm{i}$ and $\rho = 0.9$.

The model class comprises of a fluid flow model wherein each of a set of m on–off sources provides fluid input at rate 1 while it is on, and the combined fluid is drained at a constant rate 0.8 per unit time. The means of the on and off periods are respectively 1 and $1/\gamma$, and the traffic intensity of the model is given by $\rho = m\gamma/0.8(1 + \gamma)$. For a large number of cases, we used our algorithms to compute the transient distribution of the fluid flow and compared the results with those of Sericola [17] (where the latter is available), obtaining extremely favorable comparisons for our methods.

The procedure we adopted for numerical experimentation with the algorithms is as follows. For each problem, we considered several values of s, both real and complex. For each fixed s value, Algorithm 3 was used to compute $\Psi(s)$ with $\varepsilon = 10^{-15}$. The resulting value of $\Psi(s)$ was used as the target value, and other algorithms were then used and iterative processes continued until the iterates differed from $\Psi(s)$ by at most 10^{-10} in absolute value. Given, in Figures 2 and 3, are the results we obtained; our emphasis here is only on providing a glimpse of the relative speeds of the various algorithms in computing the key transform $\Psi(s)$. A detailed complexity analysis has not been performed on them. For brevity, we have shown only two cases: $s = 2$ and $s = 2 + 3\mathrm{i}$. For each, we plot the number of iterations and the CPU time (in seconds) taken by each of the algorithms to reach the same level of accuracy for Ψ. All computations reported here were performed using MATLAB®. Each example was run 20 times, and what is reported is the average CPU use per run.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press

Based on these examples and the many others we have worked out, we can assert that we have an excellent algorithm for computing the transient results for stochastic fluid flow models.

Acknowledgements

This work was completed at the Institute for Mathematics and its Applications (IMA), University of Minnesota, and both authors thank the IMA for its hospitality. Soohan Ahn was supported partially by grant no. R01-2004-000-10284-0 from the Basic Science Research Program of the Korea Science and Engineering Foundation. He also thanks AT&T, where this work was initiated during his tenure as a post-doctoral researcher.

References

[1] Ahn, S. and Ramaswami, V. (2003). Fluid flow models and queues—a connection by stochastic coupling. *Stoch. Models* **19,** 325–348.

[2] Ahn, S. and Ramaswami, V. (2004). Transient analysis of fluid flow models via stochastic coupling to a queue. *Stoch. Models* **20,** 71–101.

[3] Anick, D., Mitra, D. and Sondhi, M. M. (1982). Stochastic theory of a data-handling system with multiple sources. *Bell System Tech. J.* **61,** 1871–1894.

[4] Asmussen, S. (1994). Busy period analysis, rare events and transient behavior in fluid flow models. *J. Appl. Math. Stoch. Anal.* **7,** 269–299.

[5] Asmussen, S. (1995). Stationary distributions for fluid flow models with or without Brownian noise. *Stoch. Models* **11,** 1–20.

[6] Çinlar, E. (1975). *Introduction to Stochastic Processes*. Prentice Hall, Englewood Cliffs, NJ.

[7] Gaver, D. P. and Lehoczky, J. P. (1982). Channels that cooperatively service a data stream and voice messages. *IEEE Trans. Commun.* **30,** 1153–1161.

[8] Graham, A. (1981). *Kronecker Products and Matrix Calculus: with Applications*. John Wiley, New York.

[9] Kobayashi, H. and Ren, Q. (1992). A mathematical theory for transient analysis of communication networks. *IEICE Trans. Commun.* **12,** 1266–1276.

[10] Latouche, G. (1993). Algorithms for infinite Markov chains with repeating columns. In *Linear Algebra, Markov Chains and Queueing Models* (IMA Vol. Math. Appl. **48**), eds C. D. Meyer and R. J. Plemmons, Springer, New York, pp. 231–265.

[11] Latouche, G. and Ramaswami, V. (1993). A logarithmic reduction algorithm for quasi-birth-and-death processes. *J. Appl. Prob.* **30,** 650–674.

[12] Latouche, G. and Ramaswami, V. (1999). *Introduction to Matrix Analytic Methods in Stochastic Modeling*. Society for Industrial and Applied Mathematics, Philadelphia, PA.

[13] Neuts, M. F. (1981). *Matrix-Geometric Solutions in Stochastic Models. An Algorithmic Approach*. John Hopkins University Press, Baltimore, MD.

[14] Ortega, J. M. and Rheinboldt, W. C. (1970). *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York.

[15] Ramaswami, V. (1999). Matrix analytic methods for stochastic fluid flows. In *Teletraffic Engineering in a Competitive World*. (Proc. 16th Internat. Teletraffic Congress), eds D. Smith and P. Key, Elsevier, New York, pp. 1019–1030.

[16] Rogers, L. C. G. (1994). Fluid models in queueing theory and Wiener–Hopf factorization of Markov chains. *Ann. Appl. Prob.* **4,** 390–413.

[17] Sericola, B. (1998). Transient analysis of stochastic fluid models. *Performance Evaluation* **32,** 245–263.

https://doi.org/10.1239/jap/1118777186 Published online by Cambridge University Press