Analysis and Approximations for Time Dependant Queueing Models

Walid Nasr

Dissertation submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Industrial and Systems Engineering

Michael R. Taaffe, Chair

Ebru K. Bish

Joel A. Nachlas

Barry L. Nelson

Raghu Pasupathy

February 18, 2008

Blacksburg, Virginia

Keywords: Queueing, phase-type distribution, stochastic, time-dependent queues,

Polya-Eggenberger distribution, departure process, tandem queues

Analysis and Approximations for Time Dependant Queueing Models

Walid Nasr

(ABSTRACT)

We consider $Ph_t/Ph_t/s/c$ queueing systems, which can be viewed as a matrix-geometric approximate model of the $GI_t/G_t/s/c$ system due to the denseness of the family of Ph distributions and processes. We present approximations for the time-dependent probability mass function for the number of entities in the system at time t, for a $Ph_t/Ph_t/s/c$ queueing system.

We build on the $Ph_t/M_t/s/c$ number-in-system Markovian representation (MR) and construct an MR for the $Ph_t/M_t/s/c$ departure-count process. A state in the departure-count MR is defined by the number-in-system at time $t+\tau$ and the number of departures in the time interval $[t, t+\tau)$. The corresponding time-dependent departure-count state probabilities can be approximated by truncating and then numerically solving the corresponding infinite set of Kolmogorov Forward Equations (KFEs). We then present a finite set of time-dependent departure-count partial-moment differential equations (PMDEs) for the first two moments of the departure-count process. We further reduce the NEPMDEs to a smaller set of departure-count PMDEs. We then extend our analysis to systems with Ph_t service time service distributions and analyze queues in tandem. We present accurate and computationally efficient algorithms in this dissertation.

Acknowledgments

I would like to express my deepest gratitude to my academic advisor, Dr. Michael Taaffe, for his valuable help, time, encouragement, advice and support which he continuously provided throughout my stay at Virginia Tech.

I thank all my committee members for serving on my committee and for all their helpful comments and advise. I would also like to thank the QNATS team for their valuable input and insights.

I thank my family and especially my parents for their support and encouragement as I pursued all my academic degrees.

I would also like to thank the friends and colleagues I have known during my stay at Virginia Tech.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Environment

Developing equations to compute congestion measures for the general $G/G/s/c$ queueing model and networks of such nodes has always been a challenge. One approach to analyzing such systems is to approximate the model-specified general input processes and distributions by processes and distributions from the more computationally friendly family of *phase-type* processes and distributions. Work by Johnson and Taaffe ([5], [6] and [7]) allows us to arbitrarily closely approximate the input for general models by matching moments of the input processes and distributions to moments of approximating phase-type processes and distributions. Thus the $G/G/s/c$ queueing model can be represented and closely approximated by a $Ph/Ph/s/c$ model. The main disadvantage of the $Ph/Ph/s/c$ model is state-space size.

Rueda and Taaffe [14] address the state-space size computational problem by providing approximations for the behavior of the $Ph/Ph/s/c$ model by numerically computing and approximating time-dependent probabilities and moments via *closure*-type approximations. With the closure approximations the computational burden is independent of the system capacity and thus there is no need to numerically solve the entire set of time-dependent state probabilities.

Here we further develop numerical approximation methods for analysis of general time-dependent queueing nodes and develop new approximations for networks of such nodes. Analysis of time-dependent queueing networks with general distributions has rarely been considered in previous literature. The work presented in this dissertation along with previous work by Taaffe and colleagues are all part of a broader project to construct a Queueing Network Approximator for Time-dependent Systems (QNATS). This project is being funded by NSF grant DMII-0521945.

1.2 QNATS Objective

Whitt [15] laid out the Queueing Network Analyzer (QNA) algorithms and approximations to analyze fairly general steady-state queueing networks. This work is reviewed in Section 1.3. Our objective in developing QNATS is to create approximation methods for time-dependent queueing networks. The input parameters and output performance measures are analogous to those of QNA except QNATS considers time-dependent models and model performance

measures, a special case of which is the class of stationary systems. The goal of developing QNATS is to provide better approximations than QNA does for steady state-systems and also provide analysis for transient and time-dependent systems. One computationally attractive property of QNA is that its execution time is very fast and linear in network size. Providing accurate, convenient, and fast algorithms for approximate analysis of general time-dependent queueing networks is our QNATS objective.

1.3 Background and Literature Review

In the first part of this section we briefly review literature on stationary queueing systems, and in the second part of this section we review literature on time-dependent queueing systems. We review the most relevant results for time-dependent queueing systems in more detail in Chapter 2.

1.3.1 Stationary Systems

Whitt's background work in the development of QNA is relevant to the approximations that we develop for use in QNATS.

QNA Approximation Approach

In QNA queueing systems are modeled by nodes and arcs where nodes represent service facilities and arcs represent the flow of entities in the system as well as the flow of entities to and from the outside world. At the nodes the inter-arrival-time and service-time distributions are represented by their means and coefficients of variation only. The QNA approximations do not take into account third or higher-order moments of the input distributions. QNA decomposes the network into separate nodes. Each node is then approximated individually. Traffic departing one node flows into other nodes as well as flows out of the network, therefore there is node-to-node dependence. QNA captures node-to-node dependence by adjusting parameters of the approximating inter-arrival-time/departure-time distributions.

The first step in the QNA approach to approximating a network is to approximate the *traffic flow* into a node; see Whitt [18], and for background see Daley [4]. It is important to note that QNA approximations are exact if all the arrival/service processes are Poisson/exponential. QNA approximates all traffic flow as renewal processes even though the actual traffic into a node is composed of flows that are not renewal. QNA computes/approximates the first and second moments of the arrival process to each node. This analysis/approximation can involve superposition of departure processes at other nodes since a departure process from many nodes can contribute to the arrival process of another.

The second step is to approximate the *congestion measures* for each node separately given the first two moments of the arrival and service distributions. Details of the specific QNA

congestion approximations can be found in Whitt [15] and [17]. Those approximations compute congestion measures for stationary systems while our approximations are developed for non-stationary or time-dependent systems.

The first step in the development of approximation methods for time-dependent queueing systems is similar to the first step in the development of QNA; i.e., approximations for traffic flow among nodes. We focus on developing approximations for nodal departure processes and the superposition of nodal departure processes.

The second step in the development of approximation methods for time-dependent queueing systems is not similar to the second step in the development of QNA because time-dependent congestion performance measures for stationary systems are constants whereas congestion performance measures for time-dependent systems are functions of time.

Note that in [17] Whitt developed improved approximations for stationary queueing-network congestion performance measures and suggests substituting them into QNA for improved accuracy.

QNA makes the following network assumptions: 1) the network is open; i.e., all entities eventually leave the system with probability 1, 2) each node has infinite capacity, 3) at each node there are a finite number of identical servers, 4) the queue discipline is first-come first-serve, 5) the traffic intensity at each node is less than 1, and 6) combining and creating customers is allowed at each of the nodes.

QNA allows for two types of input. The first type is the standard input where the user enters

the number of nodes, the number of servers at each node, the external arrival rates and the corresponding coefficients of variation, and the service-time means and the corresponding coefficient of variations for each node. The QNA standard input also requires the user to input the routing matrix Q, where $[q_{ij}] \equiv$ proportion of customers completing service at node i who then proceed to node j. So for a system composed of n nodes the user entered $n^2 + 4n$ values.

The second type of input allowed by QNA allows for arrivals to be categorized according to classes. Classes of entities can traverse different paths in the network, as well as having different arrival-processes/service-distribution parameters resulting in different output performance measures. For a class with n nodes, $3n + 2$ input values are needed. QNA transforms all input options into the standard input representation in order to perform the analysis.

Computations in QNA are all linear; i.e., all computed parameters and performance measures are computed by solving, at most, systems of linear equations. The arrival rate to each node is obtained by the following equation:

$$\lambda_j = \lambda_{0j} + \sum_{i=1}^{n} \lambda_i \gamma_i q_{ij} \text{ for } j = 1, 2, \cdots, n,$$

where the outside world is represented by node 0.

In matrix notation this is:

$$\Lambda = \Lambda_0 (I - \Gamma Q)^{-1},$$

where $\Lambda_0 = [\lambda_{0j}]$ is a vector containing the external arrival rates to node j, and $\Gamma = \text{Diag}\,[\gamma_i]$,

for $i = 1, 2, \cdots, n$, is a diagonal matrix, where γ_i is a multiplicative factor representing customer creation or combination at node i.

QNA computations include:

- Solving for the effective nodal arrival rates is equivalent to inverting $[I - \Gamma Q]$. Notice that if the row sums of Q are all 1 and if each $q_{ij} \geq 0$, for every i and j, then $\text{sp}(Q) = 1$, where $\text{sp}(Q)$ is the spectral radius of Q, and $[I - TQ]$ is not invertible meaning that entities cannot leave the network. Thus, because QNA considers only open networks QNA will only accept networks having $\text{sp}(Q) < 1$.

- The traffic intensity or utilization is $\rho_i = \lambda_i \tau_i / s_i$, $1 \leq i \leq n$ where τ_i is the mean service time at node i, and s_i is the number of servers at node i.

- If $\rho_i \geq 1$, then the i^{th} node is unstable.

- The offered load at node i is $\alpha_i = \lambda_i \tau_i$, $1 \leq i \leq n$.

QNA approximates traffic flow from nodes with independent renewal processes. QNA also approximates the superposition of these (possibly dependent) approximating renewal processes with a composite renewal process. Thus, the overall arrival process at a node is represented by a composite renewal process using results from Albin [1] and Whitt [16].

Computing the parameters of these approximating composite-arrival processes requires so-

lution of the set of linear equations:

$$c_{aj}^2 = a_j + \sum_{i=1}^{n} c_{ai}^2 b_{ij},$$

where a_j and b_{ij} are functions of the network-input parameters using the so-called hybrid method (Albin [1]), and c_{aj}^2 is the coefficient of variation of the approximating inter-arrival-time distribution to node j; see Whitt [15] for the specific set of computations for the coefficients of variation, c_{ai}^2.

1.3.2 Approximating Possibly Dependent Non-Renewal Processes with Renewal Processes

In this section we review the methods used in QNA to approximate superpositions of non-renewal processes with a composite renewal process. We briefly review some background material on point processes.

In a network we need to compute/approximate the composite nodal-arrival process for each node in the network. Those nodal-arrival processes are composed of the superposition of several (possibly dependent) departure processes as well as possible arrival processes from the outside world. The set of departure processes are rarely renewal and even the superposition of independent renewal processes does not produce a renewal process unless all the superposed processes are independent and Poisson, see Karlin and Taylor ([8] *p. 227*). Thus composite nodal-arrival processes are only themselves renewal in very special cases, and arrival processes are Poisson processes only rarely in Poisson/exponential-type queueing

networks at equilibrium.

Whitt, ([16] and [18]), discusses approximating non-renewal point processes in two basic ways; the stationary-interval method (SIM) and the asymptotic method (AM). He also considers a convex combination of these two basic methods and calls it the hybrid method (HM). His objective is to approximate a non-renewal process by a renewal process. He gives a brief background on point processes that is useful for understanding these two approximations. We summarize this background below.

Whitt [16] gives the three representations of a point process: $\{N(t)\}, \{S_n\}$ and $\{X_n\}$. The counting process, $\{N(t), t \geq 0\}$, records the number of point-process occurences in the interval $[0, t]$. The partial sum process, S_n, records the time of the n^{th} point occurrence from the origin ($S_0 = 0$). The interval process, X_n, records the interval lengths between successive point occurences; i.e., the interval between the n^{th} and the $(n-1)^{st}$ point occurences, so $X_n = S_n - S_{n-1}$.

We see that

$$\begin{aligned} N(t) &= \max\left(n \geq 0 : S_n \leq t\right), t \geq 0, \\ S_n &= \min\left(t \geq 0 : N(t) \geq n\right), n \geq 1, \text{ and} \\ X_n &= S_n - S_{n-1}, n = 1, 2, 3, \ldots. \end{aligned}$$

The counting process $N(t)$ is said to be stationary if the joint distribution of

$$[N(t_1 + h) - N(s_1 + h), N(t_2 + h) - N(s_2 + h), \cdots, N(t_k + h) - N(s_k + h)]$$

is independent of h for all $(s_1, \cdots, s_k)$ and $(t_1, \cdots, t_k)$, with $0 \leq s_i \leq t_i$.

The sequence of intervals $\{X_n\}$ is said to be stationary if the joint distribution of $[X_{n_1+h}, \cdots, X_{n_k+h}]$ is independent of h. Stationarity of $\{N(t)\}$ or $\{X_n\}$ does not always imply stationarity of $\{X_n\}$ or $\{N(t)\}$, and stationarity of $\{X_n\}$ does not always imply a renewal process.

We now review Whitt's SIM, AM, and HM approximation methods, see [16] for more details.

- **The Stationary-Interval Method:**

  The SIM method looks at the distribution of the size of the first interval of the point process and assumes it to be sufficient to represent any interval. So the dependence between intervals is not taken into account. The SIM method is exact if and only if the intervals of the superposition process are independent.

  Under certain conditions the superposition of n independent renewal processes tends to a Poisson process as n becomes large and the individual point-process intensities become small. Karlin and Taylor in [8] (*p. 223*) present the following theorem:

  Theorem 1 *Let* $\{N_{ni}(t)\}$ *be an infinitesimal array of renewal processes with superposition* $N_n(t)$. *Then* $\lim_{n\to\infty} \Pr\{N_n(t) = j\} = \frac{e^{-\lambda t}(\lambda t)^j}{j!}, \qquad j = 0, 1, 2, \ldots$ *if and only if* $\lim_{n\to\infty} \sum_{i=1}^{k_n} F_{ni}(t) = \lambda t$, *where* $F_{ni}(t)$ *is the inter-arrival time distribution, and the triangular array,* $\{N_{ni}(t)\}$, *is called infinitesimal if for every* $t \geq 0, \lim_{n\to\infty} \max_{1\leq i\leq k_n} F_{ni}(t) = 0$.

  So if n is the number of independent renewal processes being superposed, then the SIM

method provides good approximations as n becomes larger because as n increases the superposition process approaches a Poisson process.

- **The Asymptotic Method:**

  In the AM the moments of the number of renewals in the approximating counting process are matched with the corresponding moments of the counting process associated with the original non-renewal point process over a large time interval. In the case of superposing independent renewal processes, the AM is asymptotically exact when the traffic intensity ρ approaches 1 (Whitt [16]).

- **The Hybrid Method:**

  A convex combination of the SIM and AM is the hybrid method (HM). In QNA the weight for the SIM approximation increases as the number of superposed processes increases and the weight for the AM approximation increases as the traffic intensity approaches 1.

Whitt [16] defines a particular Palm counting process, associated with the original point process. The Palm process he defines is the same as the original point process only the Palm process is conditioned to have a point occurring at the origin.

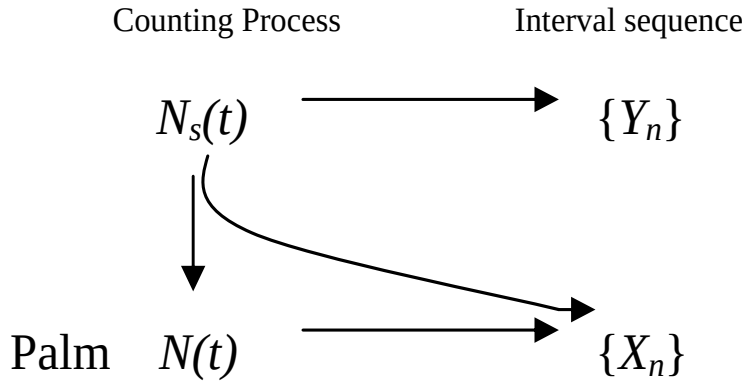
Let $\{N_s(t)\}$ be the original stationary counting process and let $\{N(t)\}$ be the associated Palm counting process.

- The Palm process, $\{N(t)\}$, is $\{N_s(t)\}$ conditioned on the occurrence of a point at the

origin.

- There is a one-to-one correspondence between $\{N_s(t)\}$ and $\{N(t)\}$.
- The interval sequence $\{X_n\}$ associated with the Palm process $\{N(t)\}$ is stationary, and so a one-to-one correspondence between $\{X_n\}$ and $\{N(t)\}$ exists.
- The interval sequence $\{Y_n\}$ associated with the original point process $\{N_s(t)\}$ may or may not be stationary.
- The Palm process identifies a one-to-one correspondence between stationary counting processes, $\{N_s(t)\}$, and the stationary interval sequences $\{X_n\}$.

Figure 1.1: Palm



The stationary point process and the associated Palm process can be related in the following manner:

$$G_{Y_1}(t) \equiv \lambda \int_0^t [1 - F(u)]du, t \geq 0 \tag{1.1}$$

$$F_{X_1}(t) \equiv 1 - \lambda^{-1} G'_{Y_1}(t), t \geq 0 \tag{1.2}$$

where X_1 is the inter-arrival time of the first entity in the Palm process $\{N(t)\}$ (conditioned on an arrival at time 0), Y_1 is the inter-arrival time of the first entity in the original process $\{N_s(t)\}$, $G_{Y_1}(t) \equiv$ c.d.f. of Y_1, $F_{X_1}(t) \equiv$ c.d.f. of X_1, and $\lambda \equiv$ intensity of the original stationary point process $\equiv \lim_{t\to\infty} N_s(t)/t$.

Let Z be the approximating renewal inter-arrival-time random variable and let $H(t)$ be its distribution function. If the SIM method is used, then Z has the same distribution as X_1. Equation 1.2 allows us to obtain $F_{X_1}(t)$ from $G_{Y_1}(t)$. It is also possible to obtain the moments of $F_{X_1}(t)$ from the moments of $G_{Y_1}(t)$: $\mu_{j+1}\left(F_{X_1}(t)\right) \equiv \mu_j\left(G_{Y_1}(t)\right)\mu_1\left(F_{X_1}(t)\right)(j+1)$. It is sometimes possible to find an approximation of $G_{Y_1}(t)$ which allows us to approximate $F_{X_1}(t)$. For example, in [18] Whitt gives can give an expression for the residual/excess time until the next departure from a $G/G/1$ queue at an arbitrary point in time when the queue is in equilibrium. The distribution of this residual time can be represented by $G_{Y_1}(t)$ from which the distribution of $F_{X_1}(t)$ can be obtained.

Whitt matches the first m moments of the n^{th} partial sum process, $\mathrm{E}\left[S_n^j\right], j = 1, 2, \cdots, m$, of the approximating renewal process with the first m moments of S_n, the partial sum process associated with the Palm process $N(t)$. Let $\beta_j(Z)$ be the j^{th} cumulant of the random variable Z (the approximating renewal inter-arrival-time random variable), where $\beta_j(Z)$ is also the

coefficient of t_j in the power series representation of $\log(\mathrm{E}\left[e^{tZ}\right])$ and $\beta_j(Z_1 + ... + Z_n) = n\beta_j(Z_1)$ for all j and n because Z_n, the sequence of approximating renewal inter-arrival times, are iid. There is a one-to-one correspondence between the first m moments and the first m cumulants (the first and second cumulants equal the mean and variance, respectively).

Whitt then uses approximations of $\beta_j(S_n), j = 1, 2, ..., m$, to set $\beta_j(H) = \beta_j(S_n)/n,\ j = 1, 2, ..., m$. The SIM method uses $n = 1$ and the AM method uses $n \to \infty$. Note that for $n \geq 2$ the cumulants $\beta_j(S_n)$ are affected by the dependence among the random variables. For example, $\beta_2(S_n) = \sum_{k=0}^{n-1} (n-k)\mathrm{Cov}(X_1, X_{1+k})$.

To make Whitt's approximation scheme more intuitive we consider the same extreme example that Whitt gives in [16] in order to clarify the difference between the SIM method and the AM method.

Consider the non-renewal interval process defined by X_i, which represents the time between the $(i-1)$ and i^{th} (arrival) point. Let $\mathrm{P}\{(X_i, X_{i+1}) = (1,2) \text{ or } (2,1)\} = 1$ for all i and $\mathrm{P}\{X_i = 1\} = \mathrm{P}\{X_i = 2\} = 1/2$. Let X be the time interval between points (arrivals) for the approximated renewal process. In general both the SIM and the AM methods give the same first moment, which in this case is $\mathrm{E}[X] = 1.5$.

The SIM method looks at the distribution of X_1 and assign it be the distribution for X. So the SIM method gives $\mathrm{P}\{X = 1\} = \mathrm{P}\{X = 2\} = 1/2$.

The AM method gives $\mathrm{Var}[X] = \lim_{n\to\infty} \mathrm{Var}[S_n]/n$, where $\mathrm{Var}[S_n] = 0$ if n is even and $\mathrm{Var}[S_n] = 1/4$ if n is odd; so this implies that $\mathrm{Var}[X] = 0$ for both the odd and even cases.

Because the variance is zero the AM method gives $\mathrm{P}(X = 1.5) = 1$.

The SIM approximation does not account for the correlation between successive intervals whereas the AM approximation does. Albin [1] states that neither method is superior to the other. She defined a hybrid convex-combination approximation of the SIM and AM approximations. This hybrid approximation, HM, is used in QNA and it is dependent on the traffic intensity and the number of processes being superposed. The HM approximation is used in the QNA when superposing processes and for approximating departure processes from nodes.

The motivation for Whitt and Albin's analysis of stationary point processes is to analyze and approximate departure processes from queueing nodes. The approximation of departure processes is an essential part in QNA. Although QNATS is not restricted to stationary systems, our approximation-design process and algorithms are influenced by Whitt et al in their development of QNA. When we analyze time-dependent processes and construct approximating distributions and processes, we compare our analysis to concepts presented in Whitt's analysis of point processes and his approximation of a point process with a renewal process.

In the next chapter we review background material for analysis and approximations for non-stationary queueing systems.

Chapter 2

Approximating System-State Probability Mass Functions for the Time-Dependent $Ph_t/Ph_t/s/c$ Queueing Models

2.1 Introduction

We present an approximation for the probability mass function for the number of entities in the system at time t, $N(t)$, for a $Ph_t/Ph_t/s/c$ queueing system. Rueda and Taaffe [14] describe a closure-approximation (CA) algorithm for this queueing system that approximates

the moments of $N(t)$ as a function of time. They also approximate the moments of the time-dependent virtual waiting times, W_{t_i}, for a user-selected set of times $t_i,\ i = 1, 2, \ldots$, as well as the corresponding virtual waiting time cdfs, $F_{W_{t_i}}(\tau)$. They then examine in detail the accuracy of their approximations for the first two time-dependent, system-size moments and conclude that the accuracy is excellent. Approximating the entire time-dependent distribution of $N(t)$ by extending their approximations is the focus of this chapter. We show that our extension of the CA method results in an accurate and efficient approximation for the entire probability mass function for the time-dependent system-size random variable.

The focus of the literature considering approximations for the number of entities in stationary, general inter-arrival-time and service-time queueing systems is usually on steady-state, system-size moments; i.e., $\lim_{t\to\infty} \mathrm{E}\left[N^k(t)\right], k = 1, 2$. Approximations for the probability that the system is idle ($\mathrm{P}\left(N(t) = 0\right)$) and the idle/busy period length are also performance measures that are also commonly considered. This chapter describes approximations for the time-dependent entire pmf of $N(t)$. The procedures presented make it possible to approximate and plot any of the system-state probabilities over a selected time intervals or plot the entire system-state pmf for selected points in time t_i.

The queueing system we consider is the $Ph_t/Ph_t/s/c$, which can be viewed as a matrix-geometric approximate model of the $GI_t/G_t/s/c$ system due to the denseness of the family of Ph distributions and processes. Johnson and Taaffe discuss the denseness and moment-matching flexibility of the family of Ph distributions, [5] and [6]. Our results are designed for steady-state, transient or time-dependent queueing models with general inter-arrival-time

and service-time distributions.

2.2 Closure Approximations

In this section we describe a *closure*-type approximation for time-dependent queueing models and the use of the Polya-Eggenberger distribution serving as a surrogate or approximating system-state distribution.

The Polya-Eggenberger (PE) distribution was first introduced as a means to approximate the system-state boundary probabilities of a queueing system by Clark [3]. Clark used the PE distribution to approximate probabilities appearing on the right hand side (RHS) of the partial-moment differential equations (PMDE) of an $M_t/M_t/s$ system, hence allowing the numerical integration of the PMDEs. By partial moment and partial-moment differential equation we mean:

$$\mathrm{E}\left[N(t), a \leq N(t) \leq b\right] \equiv \mathrm{E}\left[N(t) | a \leq N(t) \leq b\right] \cdot \mathrm{P}\left(a \leq N(t) \leq b\right),$$

and

$$d\mathrm{E}\left[N(t), a \leq N(t) \leq b\right] / dt,$$

We refer to such a solution methodology as *closure approximations* (CA). The CA approach closes the set of PMDEs by constructing approximations for the probabilities appearing on RHS of the PMDEs.

Here we extend Rueda and Taaffe's methods and use of the PE and multinomial (MN) distri-

butions to approximate the entire time-dependent, system-state pmfs for the $Ph_t/Ph_t/s/c$ system. The motivation for choosing the PE and MN distributions is presented by Rueda and Taaffe [14], and Ong and Taaffe ([12], [13]). In those papers CAs are used to solve the PMDEs for the $Ph_t/Ph_t/s/c$ and $Ph_t/M_t/s/c$ systems. Since the PE distribution provides good approximations for the system-state probabilities appearing on the RHS of the PMDEs, we here investigate whether they can also provide good approximations for the entire system-state pmf.

2.2.1 Moment Differential Equation Closure, Psuedo-Closure, and Quasi-Closure

Here we define various notions of closure for state-probability, moment, and partial-moment differential equations.

Closed set of differential equations: We consider a set of differential equations to be *closed* if all terms appearing on the RHS of the differential equations are either exogenous parameters (such as arrival or service rates, $\lambda(t)$, $\mu(t)$) or are the integral of an expression on the left hand side (LHS) of the differential equations. We define the terms appearing on the RHS that are neither exogenous parameters nor integrals of an expression appearing on the LHS of the differential equations to be the *closure terms*.

For example, the set of Kolmogorov forward differential-difference equations is closed. Another example is the k^{th} system-size moment differential equation for the $M_t/M_t/\infty$

system (because this differential equation has only exogenous parameters and integrals of j^{th} system-size moments, for $0 < j < k$, on its RHS).

Pseudo-Closed set of differential equations: We consider a non-closed set of differential equations to be *pseudo-closeable* if the closure terms (on the RHS) can be well approximated by some function of the integrals of the LHS terms. If the closure terms on the RHS are approximated then the resulting set of differential equations *and* the approximations together are now closed. We refer to this type of differential-equation approximation as a *pseudo-closure* approximation.

For example, the first two moment differential equations for the number in the system at time t for the $M_t/M_t/1$ system are

$$\mathrm{E}\left[N(t)\right]' = \lambda(t) - \mu(t)\left(1 - \mathrm{P}\left(N(t) = 0\right)\right),$$

and

$$\mathrm{E}\left[N(t)^2\right]' = \lambda(t) + 2\lambda(t)\mathrm{E}[N(t)] + \mu(t)\left(1 - \mathrm{P}\left(N(t) = 0\right)\right) - 2\mu(t)\mathrm{E}[N(t)].$$

The term $\mathrm{P}\left(N(t) = 0\right)$ is the closure term for these two moment differential equations. As will be shown, Clark [3], Taaffe and Ong ([12], [13]) close this set of differential equations via a moment-matching algorithm. They approximate the closure term, by matching two moments, $\mathrm{E}\left[N(t)\right]$ and $\mathrm{E}\left[N(t)^2\right]$, to a *surrogate* or approximation distribution, and then approximate the closure term with a probability computed from the approximation distribution.

Quasi-Closed set of differential equations: We consider a non-closed set of differential equations to be *quasi-closeable* if we can construct a state-space partitioning such that the set of corresponding partial-moment differential equations is closed.

For example, the set consisting of the first two moment differential equations for the number in the system at time t for the $Ph_t/Ph_t/\infty$ system is not closed, Nelson and Taaffe [10]. However, consider the set of partial-moment differential equations resulting from partitioning the state space based on the state of the arrival process and the service phase. The union of the resulting the set of arrival differential equations (the 0^{th}-moment differential equations for the partition consisting of the arrival process states) together with the set of partial first-and-second moment differential equations corresponding to the partitions based on the service-process phases, *is* closed. Forcing closure by this type of state-space partitioning is called *pseudo-closure*. The set of PMDEs is necessarily larger than the simple MDE; thus a modest increase in computation is necessary for quasi-closure computations.

Pseudo-Quasi-Closed set of differential equations: We consider a non-closed set of differential equations to be *pseudo-quasi closeable* if we can construct an state-space partitioning such that the resulting set of partial-moment differential equations is still non-closed (closure terms still appear on the RHS of the PMDEs) but the closure terms can be approximated. The resulting union of differential equation sets and the approximations are now closed. We refer to this type of differential-equation approximation as a *quasi-pseudo closure* approximation.

In Section 2.2.2 we consider the $M_t/M_t/s/c$ system where we partition the first and second moment differential equations and then identify and approximate the closure terms. We also explain why using *quasi-psuedo closure* on the first two moment differential equations is more efficient that solving the KFEs.

2.2.2 $M_t/M_t/s/c$ KFEs and PMDEs

As a brief review of the overall CA approach we consider a pseudo-quasi-closure approximation for the simple $M_t/M_t/s/c$ model. This example serves as an instructional building block for more complex systems.

Consider the $M_t/M_t/s/c$ model with arrival rate $\lambda(t)$ and service rate $\mu(t)$. In Section 2 we review the performance of the CA approach for the $M_t/M_t/s/c$. We can to write two sets of differential equations for this system. The first set is the set of Kolmogorov forward equations (KFE), and the second is the set of PMDEs. The KFEs are *closed* and the number of differential equations is $(c+1)$.

The KFEs for the $M_t/M_t/s/c$ system-size probabilities are:

$$\mathrm{P}'(N(t)=0) = -\lambda(t)\mathrm{P}(N(t)=0) + \mu(t)\mathrm{P}(N(t)=1), \tag{2.1}$$

$$\begin{aligned} \mathrm{P}'(N(t)=i) &= -\left(\lambda(t)+(i-1)\mu(t)\right)\mathrm{P}(N(t)=i) + \lambda(t)\mathrm{P}(N(t)=i-1) \\ &\quad +(i+1)\mu(t)\mathrm{P}(N(t)=i+1), \quad \text{for} \quad i=1,\cdots,s-1 \end{aligned} \tag{2.2}$$

$$\begin{aligned} \mathrm{P}'(N(t)=i) &= -(\lambda(t)+s\mu(t))\mathrm{P}(N(t)=i)+\lambda(t)\mathrm{P}(N(t)=i) \\ &\quad +s\mu(t)\mathrm{P}(N(t)=i+1) \quad \text{for} \quad i=s,\cdots,c-1 \end{aligned} \tag{2.3}$$

and

$$\mathrm{P}'(N(t)=c) = -s\mu(t)\mathrm{P}(N(t)=c)+\lambda(t)\mathrm{P}(N(t)=c-1). \tag{2.4}$$

The five PMDEs for the $M_t/M_t/s/c$ system-size moments are:

$$\begin{aligned} \mathrm{E}'[N(t),N(t)<s] &= -s\lambda(t)\mathrm{P}(N(t)=s-1)-\mu(t)\mathrm{E}[N(t),N(t)<s] \\ &\quad +\lambda(t)\mathrm{P}(N(t)<s)+s(s-1)\mu(t)\mathrm{P}(N(t)=s) \end{aligned} \tag{2.5}$$

$$\begin{aligned} \mathrm{E}'[N^2(t),N(t)<s] &= -2\mu(t)\mathrm{E}[N^2(t),N(t)\geq s]-s^2\lambda(t)\mathrm{P}(N(t)=s-1) \\ &\quad +s(s-1)^2\mu(t)\mathrm{P}(N(t)=s)+2\lambda(t)\mathrm{E}[N(t),N(t)<s] \\ &\quad +\mu(t)\mathrm{E}[N(t),N(t)<s]+\lambda(t)\mathrm{P}(N(t)<s) \end{aligned} \tag{2.6}$$

$$\begin{aligned} \mathrm{E}'[N(t),N(t)\geq s] &= -s(s-1)\mu(t)\mathrm{P}(N(t)=s)-\lambda(t)\mathrm{P}(N(t)=c) \\ &\quad -s\mu(t)(1-\mathrm{P}(N(t)<s))+\lambda(t)(1-\mathrm{P}(N(t)<s)) \\ &\quad +s\lambda(t)\mathrm{P}(N(t)=s-1) \end{aligned} \tag{2.7}$$

$$\begin{aligned} \mathrm{E}'[N(t)^2,N(t)\geq s] &= -(2c+1)\lambda(t)\mathrm{P}(N(t)=c)-(s-1)^2\mu(t)\mathrm{P}(N(t)=s) \\ &\quad -2\lambda(t)\mathrm{E}[N(t),N(t)<s]+\lambda(t)(1-\mathrm{P}(N(t)<s)) \\ &\quad +s\mu(t)(1-\mathrm{P}(N(t)<s)+s^2\lambda(t)(\mathrm{P}(N(t)=s-1) \\ &\quad +2\lambda(t)\mathrm{E}'[N(t),N(t)<s] \end{aligned} \tag{2.8}$$

and

$$\mathrm{P}'(N(t)<s) = -\lambda(t)\mathrm{P}(N(t)=s-1)+s\mu(t)\mathrm{P}(N(t)=s), \tag{2.9}$$

see Ong and Taaffe [14].

The PMDEs ((2.5) to (2.8)) are *not closed* since probabilities other than $\mathrm{P}(N(t < s))$ appear on the RHS; thus approximations are necessary to *close* the set of differential equations and thus making numerical computation possible. Hence the set of PMDEs together with the probability approximations are closed; i.e., we have a pseudo-closure approximation.

Observe that the set of closure terms are all subspace boundary probabilities since we partition our system state space into two subspaces (partitions); $\{N(t) < s\}$ and $\{N(t) \geq s\}$. For the $M_t/M_t/s/c$ system the set of boundary probabilities are:

$$\{\mathrm{P}\left(N(t) = s-1\right), \mathrm{P}\left(N(t) = s\right), \mathrm{P}\left(N(t) = c\right)\}.$$

Notice that the infinite-capacity $M_t/M_t/s$ system reduces the set of boundary probabilities to two.

The CA algorithm approximates these boundary probabilities by equating the two time-dependent conditional system-state random variables, $(N(t)|N(t) \leq s-1)$, and $(N(t)|N(t) > s-1)$ to a Polya-Eggenberger (PE) random variables X, having support on the set of integers $\{0, 1, \cdots, s-1\}$, and a shifted PE random variable, Y, having support on the set $\{0, 1, \cdots, c-s\}$, respectively.

The parameters of these PE random variables are obtained by matching the first two moments of X and Y, to the first two moments of $(N(t)|N(t) \leq s-1)$, and $(N(t)|N(t) > s-1)$, respectively. Once the set of moments have been matched and the parameters of the approx-

imating PE distributions of X and Y have been computed, then the time-dependent conditional system-state boundary probabilities $\{\mathrm{P}\left(N(t)=s-1\right),\mathrm{P}\left(N(t)=s\right),\mathrm{P}\left(N(t)=c\right)\}$ are set equal to the corresponding PE probabilities $\{\mathrm{P}\left(X=s-1\right),\mathrm{P}\left(Y=0\right),\mathrm{P}\left(Y=c-s\right)\}$. So the set of PMDEs and the boundary-probability approximations form a pseudo-quasi closed set of differential equations. The boundary probability approximations are done at every step of the numerical integration procedure.

The obvious advantage of this pseudo-quasi closure is that the number of differential equations is only five and *is not a function of the capacity of the system.* The KFEs (Equations 2.1 to 2.4) represent a continuous time discrete-state Markov Process having $c+1$ states (thus $c+1$ differential-difference equations).

2.2.3 $Ph_t/Ph_t/s/c$ KFEs and PMDEs

Consider the KFEs for the more general $Ph_t/Ph_t/s/c$ system (see Appendix One and Two for the associated KFEs and PMDEs, respectively). For this system the number of states (differential-difference equations) is:

$$m_B^{-1}m_A(s+m_B(c-s+1))\mathrm{C}(m_B+s-1,s) \tag{2.10}$$

where m_A and m_B are the number of arrival and service phases, respectively, s is the number of servers, c is the capacity of the system and $\mathrm{C}(i,j)=\frac{i!}{(j!)(i-j)!}$.

In Rueda and Taaffe [14] the basic $M_t/M_t/s/c$ CA algorithm is extended to the case of the

$Ph_t/Ph_t/s/c$ system. Their $Ph_t/Ph_t/s/c$ CA algorithm includes an additional multinomial-based approximation. They close the set of PMDEs by approximating the set of boundary probabilities on the RHS of the PMDEs in a manner similar to the CA for the simple $M_t/M_t/s/c$ system. This closure is accomplished by approximating the boundary probabilities as functions of systems-size moments and partial moments and by the use of the multinomial distribution to approximate partial moments. The number of differential equations that need to be computed numerically in the $Ph_t/Ph_t/s/c$ CA algorithm is $m_A(m_B^2 + 2m_B + 4)$.

In a later section of this chapter we consider several instances of the $Ph_t/Ph_t/s/c$ model and examine the performance of the CA algorithm for a variety of specific cases. By performance we mean the quality of the approximations as well its computation time. In the cases examined we vary the input parameters of the arrival and service processes as well as system capacity and number of servers. We consider a variety of first and second moment values for the interarrival-time and service-time distributions. And because the number of phases in a particular phase arrival/service process is not necessarily unique, we also observe the effect of increasing the number of phases in the representation of a particular phase arrival/service process.

We compare our CA for the time dependent system-state moments by either computing Monte Carlo simulation estimates of performance measures or by integrating numerically the full set of KFEs and obtaining all of the time-dependent state probabilities. Constructing and exercising Monte Carlo simulation models can be tedious especially since we are interested in time dependent results and not just steady-state results. If we are interested in

the system-state at a certain point in time t, then we have to run multiple independent replications of the simulation and observe the number in system at time t for each replication to provide an estimator and its corresponding (low) standard error. Further, to estimate system performance at many different values of t, we need to run many independent sets of multiple independent replications in order to provide estimators and their corresponding standard errors. Thus Monte Carlo simulation is not practical over long time intervals for time-dependent systems.

Alternately we consider solving the KFEs via numerical integration. The obvious advantage of this approach is that the numerical integration results are as accurate as we desire (to machine-level accuracy). The drawback is the computation time due to the large number of states in the Markov Process representing the system-state probabilities.

In Section 2.3 we examine the accuracy of the PE approximations for the set of boundary probabilities $\{(\mathrm{P}(N(t) = s-1), \mathrm{P}(N(t) = s) \text{ and } \mathrm{P}(N(t) = c)\}$ needed to close the $M_t/M_t/s/c$ PMDEs. We then look at using the PE closure approximation to produce an approximation for the entire pmf for the number in system at time t. In Section 2.4 we extend the examination of the accuracy of the set of boundary probability approximations to the $Ph_t/Ph_t/s/c$ system and look at the quality of the moment approximations for different arrival and service distributions. Similar to our extension of the CA for the $M_t/M_t/s/c$ pmf, we also extend the PE closure approximation to approximate the entire pmf for the number in system at time t for the $Ph_t/Ph_t/s/c$ system.

2.3 $M_t/M_t/s/c$ Closure Approximations

In this section we examine the CA of Ong and Taaffe [13] for the $M_t/M_t/s/c$ model. We outline the algorithm and refer the reader to [13] for the details. We briefly review the performance of the CA for time-dependent moments for this model and then consider augmenting the CA to approximate the entire probability mass function (pmf) of the time-dependent system-state $N(t)$. We also focus on the set of boundary-state probabilities $\{(\mathrm{P}(N(t) = s-1), \mathrm{P}(N(t) = s) \text{ and } \mathrm{P}(N(t) = c)\}$ and consider variations of the original CA to (perhaps) better approximate these boundary-state probabilities in some cases.

2.3.1 The Original CA for Approximating the Time-Dependent System-State Moments

Consider the Ong and Taaffe [13] or Rueda and Taaffe [14] CA applied to an $M_t/M_t/s/c$ system. The first and second system-size partial moment differential equations have the probabilities

$$\mathrm{P}(N(t) = s-1),\ \mathrm{P}(N(t) = s),\ \text{and } \mathrm{P}(N(t) = c)$$

on the RHS. So the differential equations are not closed and these probabilities need to be approximated.

Ong and Taaffe [13] point out that the state space for the $M/M/s/c$ model can be partitioned into two disjoint subspaces $\{0, \ldots, s-1\}$ and $\{s, \ldots, c\}$. They also observe that

the conditional steady-state distribution over states in the first subspace $\{0, \ldots, s-1\}$ is a truncated Poisson distribution. The conditional steady-state distribution over the states in the second subspace $\{s, \ldots, c\}$ is a shifted truncated geometric distribution. Their observation lead them to conclude that it was reasonable to expect that the *transient* and time-dependent distribution for the $M/M/s/c$ and the $M_t/M_t/s/c$ models also perhaps have two different forms; one form for subspace 1 and perhaps a different form for subspace 2. Thus they constructed a pseudo-quasi CA that is based on decomposing the state space for the $M_t/M_t/s/c$ into two disjoint subspaces. The purpose of this decomposition is to provide better approximations for the boundary probabilities.

2.3.2 The Performance of the Original CA for Approximating the Time-Dependent System-State Moments

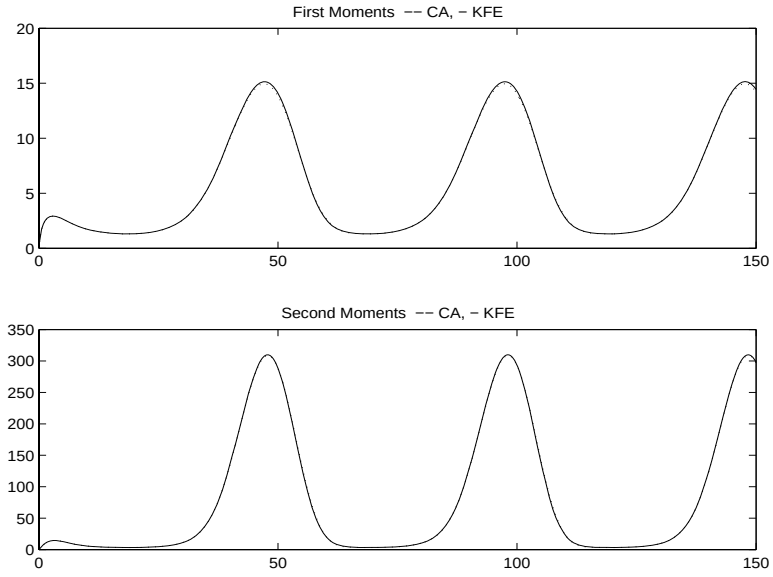
The CA usually provides good first and second-moment approximations for the $M_t/M_t/s/c$ system. The moments obtained by the CA are compared to the moments calculated by numerically solving the KFEs. It is relatively easy to implement numerical evaluation of the KFEs of the $M_t/M_t/s/c$ system and obtain the time-dependent system-state probabilities from which the first two moments are calculated. Consider Example 1 where we solve the KFEs as well as execute the CA and compare the resulting set of first and second moments.

Example 1 (E1). Let $s = 3, c = 50, \lambda = 4.1 + \cos(1/8t)$, and $\mu = 2.1 + \sin(1/8t)$.

For all the figures presented in this chapter, the solid lines represent numerical solutions to

the KFEs and the dotted lines represent the approximated solutions. The dotted and solid lines are indistinguishable for cases where the approximations and the numerically exact results are the almost the same. Figure 2.1 is a plot of the first and second system-size moments across time where the dotted and solid lines are almost indistinguishable. For this system the state-space partitioning and the associated approximating PE-distribution provide accurate *pseudo-quasi closure* approximations. Since the boundary probabilities are the only approximations required to close the PMDEs, we suspect that these boundary probability themselves are accurately approximated. If they are, in fact, well approximated, then perhaps one might conjecture that the entire pmf could be approximated well by extending use of the approximating PE distribution. In the next section we test the quality of the boundary probabilities by comparing them to the KFE solutions.

Figure 2.1: Moments vs Time



2.3.3 The Performance of the Original CA for Approximating the Time-Dependent Boundary-State Probabilities for the $M_t/M_t/s/c$

The quality of the moment approximations depends on boundary-probability approximations. Consider two methods to approximate the boundary probabilities where Method 1 utilizes the state-space partitioning (as Ong and Taaffe did [12]) whereas Method 2 is a simpler method and does not make use of any state-space partitioning.

Method 1 (M1):

The CA uses M1 to compute the key boundary probabilities. The two subspaces are $\{N(t) \in \Omega_1\}$ and $\{N(t) \in \Omega_2\}$ where $\Omega_1 \equiv \{0, 1, 2, ..., s-1\}$ indicates the system is in the lower subspace/partition and $\Omega_2 \equiv \{s, s+1, ..., c\}$ indicates the system is in the upper subspace/partition. If the system is in Ω_1 then at least one server is available and there are no entities waiting in the queue. If the system is in Ω_2 then no servers are idle and there may or may not be entities waiting in the queue.

Let X be a PE random variable having support on $\{0, 1, 2, ..., s-1\}$ with $\mathrm{E}[X]$ set to $\mathrm{E}[N(t)|N(t) \in \Omega_1]$ and $\mathrm{E}[X^2]$ set to $\mathrm{E}[N^2(t)|N(t) \in \Omega_1]$. Approximate $\mathrm{P}(N(t) = s-1|N(t) \in \Omega_1)$ with $\mathrm{P}(X = s-1)$.

$$\mathrm{P}(N(t) = s-1|N(t) \in \Omega_1) = \mathrm{P}(X = s-1)$$

Similarly for the second partition let Y be a PE random variable with support $\{0, 1, \ldots, c-s\}$

with $\mathrm{E}[Y]$ set to $\mathrm{E}[N(t)|N(t) \in \Omega_2] - s$ and $\mathrm{E}[Y^2]$ set to $\mathrm{E}[N^2(t)|N(t) \in \Omega_2] - 2s\mathrm{E}[N(t)|N(t) \in \Omega_2] + s^2$. Approximate $\mathrm{P}(N(t) = s|N(t) \in \Omega_2)$ with $\mathrm{P}(Y = 0)$ and $\mathrm{P}(N(t) = c|N(t) \in \Omega_2)$ with $\mathrm{P}(Y = s - c)$.

Since M1 conditions on being in a partition, we are also interested in the accuracy of the conditional moment approximations. We will examine the quality of the partial moments (the unconditioned conditional moments) for the $Ph_t/Ph_t/s/c$ system.

Method 2 (M2):

M2 does not condition on being in a partition. Instead let Z be a PE random variable having support on $\{0, \ldots, c\}$ with $\mathrm{E}[Z]$ set to $\mathrm{E}[N(t)]$ and $\mathrm{E}[Z^2]$ set to $\mathrm{E}[N^2(t)]$. Approximate $\mathrm{P}(N(t) = s - 1)$ with $\mathrm{P}(Z = s - 1)$, $\mathrm{P}(N(t) = s)$ with $\mathrm{P}(Z = s)$, and $\mathrm{P}(N(t) = c)$ with $\mathrm{P}(Z = c)$.

Figures 2.2 and 2.3 are plots of the boundary probabilities using M1 and M2, respectively, for the $M_t/M_t/s/c$ system in E1. Conditioning on being in a partition or subspace (Figure 2) gave better approximations especially for $\mathrm{P}(N(t) = s-1)$ and $\mathrm{P}(N(t) = s)$. M1 gives better boundary probability approximations over the plotted interval in this example. Notice that in the last plot of 2.2, the scale on the vertical axis is 10^{-4}, so the absolute error is not substantial.

In M2 there are three state probabilities to be approximated, $\mathrm{P}\left(N(t) = s - 1\right), \mathrm{P}\left(N(t) = s\right)$, and $\mathrm{P}\left(N(t) = c\right)$, having only matched two moments; whereas in M1 there are the same three state probabilities being approximated but there are four moments being matched. While

Figure 2.2: Method 1

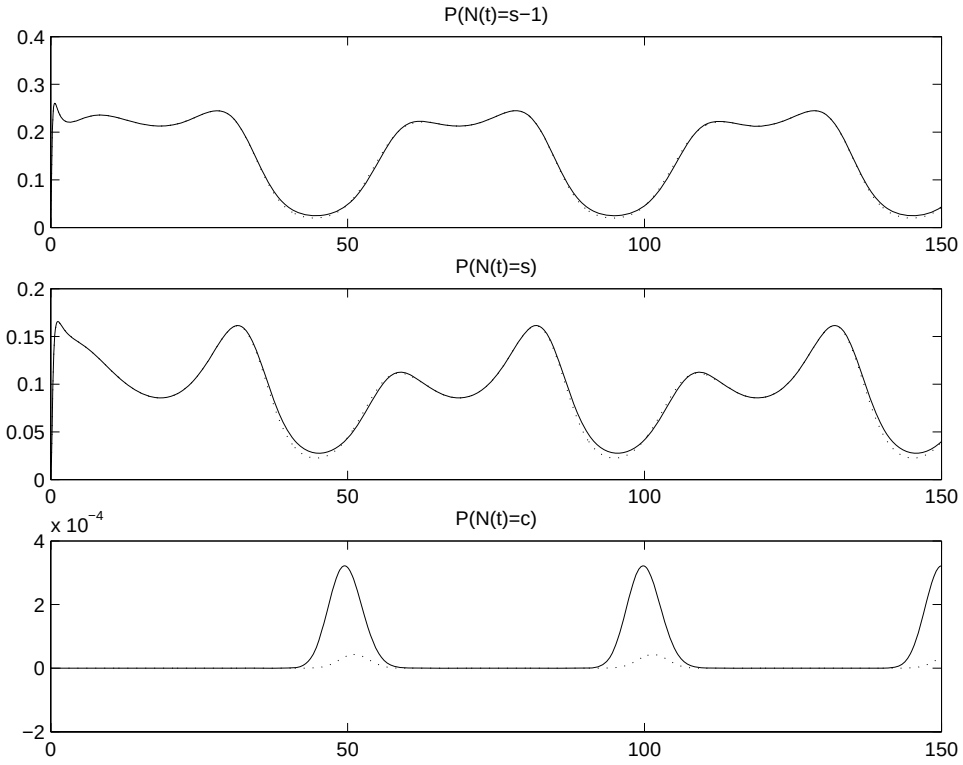
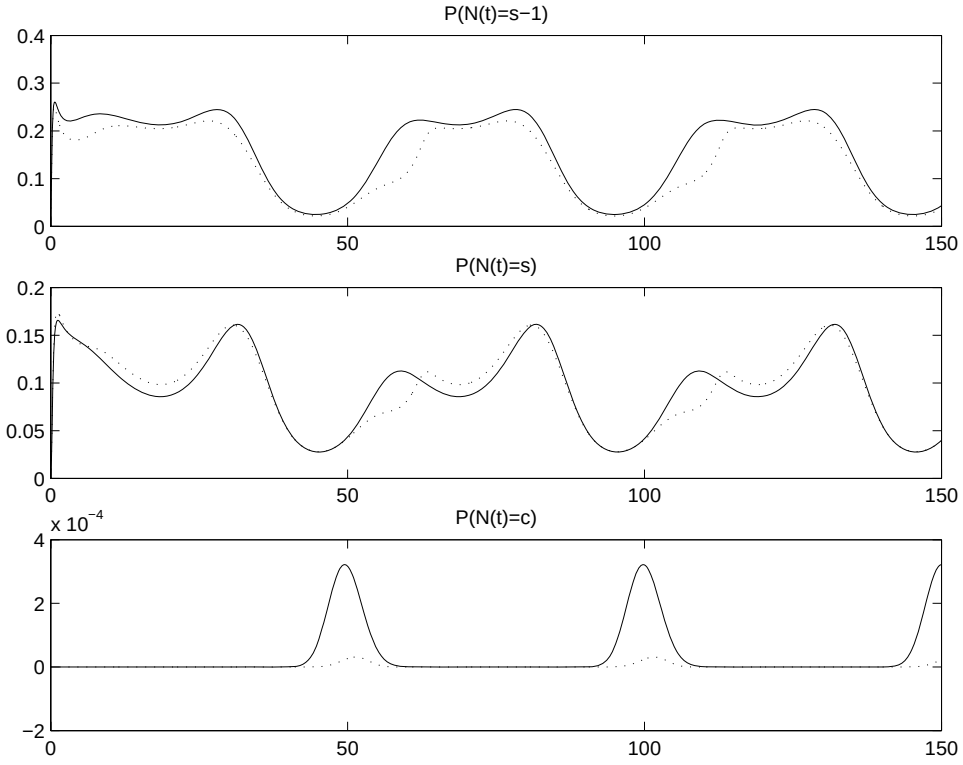


Figure 2.3: Method 2



it is natural to expect M1 to perform no worse than M2, we investigate cases to determine if the small amount of extra computation in using M1 provides noticeable improvements in accuracy.

In investigating the accuracy of M2 we quickly discovered that M2 provided poor results. We then decided to use M2 to compute the set of approximation probabilities for comparison with the approximation probabilities computed in M1. Since closing the number-in-system PMDEs by using M2 to approximate the boundary probabilities gives poor number-in-system moment approximations, we solve the (numerically) exact state probabilities as computed via solving the entire set of Kolmogorov forward equations providing numerically exact results on the first two moments of the number-in-system. Using the numerically exact number-in-system moments, we computed the three approximate boundary-probabilities via M2. This comparison amplifies any inaccuracy in the M2-approximated probabilities.

Hence we conclude that the improved accuracy of M1 over M2 is well worth the extra computation.

The preceding cases examined are typical of all the $M_t/M_t/s/c$ cases we examined. We presented E1 to clearly illustrate our conclusions and observations. Appendix 3 presents the full set of test cases in more detail.

Next we extend the M1 and M2 CAs for the $M_t/M_t/s/c$ system and now approximate *all* of the state probabilities. We examine the quality of the resulting approximate pmf, but of course we only use three of the approximated probabilities to close the RHS of the PMDEs.

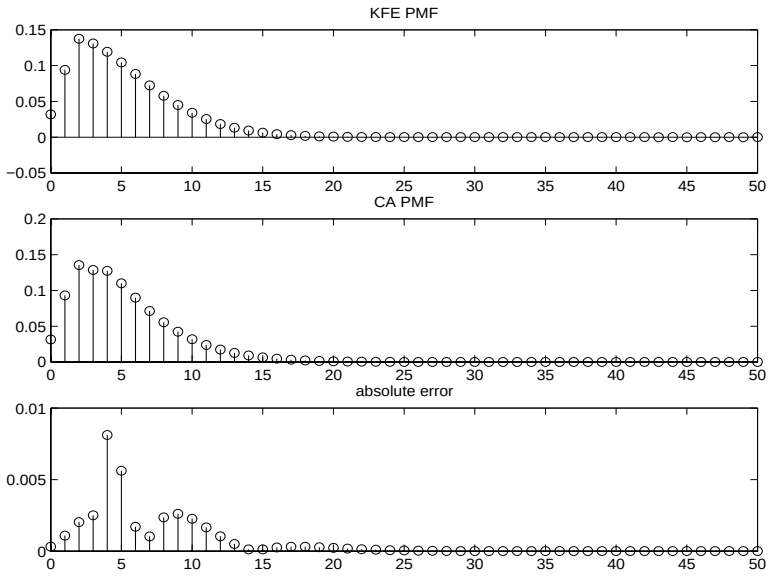
2.3.4 The Augmented CA for Approximating the Time-Dependent System-State PMF

The M1 CA provides a good approximation of the first and second moments of system state over a time interval. Using the same M1 or M2 PE-based CA approximation procedure, it is also possible to approximate the entire system-state pmf at any point in time t.

For the $M_t/M_t/s/c$ system presented in E1 and at time $t = 35$, Figure 2.4 presents the entire system-state pmf as approximated using M1; i.e., $\mathrm{P}(N(t) = i|N(t) \in \Omega_1) \approx \mathrm{P}\,(X = i)$ for $i = 1 \ldots s-1$ and $\mathrm{P}\,(N(t) = i|N(t) \in \Omega_2) \approx \mathrm{P}\,(Y = i)$ for $i = s \ldots c$), for X and Y defined as PE random variables as in the previous section. The first plot in Figure 2.4 presents the entire system-state-pmf obtained by solving the KFEs, the second plot represents the entire system-state pmf as approximated using M1 and the third plot represents the absolute error.

For M1 the maximum absolute error for any state probability across time is less than 0.01. M1 produces a good approximation for the entire pmf, not just the boundary-state probabilities. M1 outperforms M2 when computing the entire system-state pmf. So the intuition that matching more (four) conditional moments to two PE distributions provides better approximations of the system-state pmf than matching only two moments to one PE approximating distribution is supported. We reconsider the two methods in Section 2.4 for the $Ph_t/Ph_t/s/c$ system where conditioning does not uniformly provide the best approximations.

Figure 2.4: PMFs of E1 using M1



2.4 $Ph_t/Ph_t/s/c$ Closure Approximations

Asmussen [2], Johnson and Taaffe [5], [6] and [7] prove the denseness of the family of Ph distributions and illustrate the distribution's moment-matching flexibility as well as its shape-matching flexibility, and provide moment-matching algorithms for the Ph distribution. Johnson and Taaffe [6] show that with the Ph distribution it is possible to match the first k moments of any distribution with a finite-phase Ph distribution. Johnson and Taaffe also show that the first k moments can be represented by a finite mixture of common-ordered Erlangs. They also provide moment-matching algorithms for the first k moments and demonstrate that when selecting a Ph distribution to represent the arrival/service distribution, the first two moments are usually, but not always, sufficient for accurate results. So we consider that analyzing the $Ph_t/Ph_t/s/c$ system is comparable to analyzing the $G_t/G_t/s/c$ system.

2.4.1 The Original CA for Approximating the Time-Dependent System-State Moments

The KFEs for the $Ph_t/Ph_t/s/c$ are, of course, closed and have a relatively simple form. For even a moderate-capacity system, however, numerically solving the KFEs is not always practical due to the large number of states. For example, consider a stationary system with 4 phases used to represent the arrival process and 4 phases used to represent the service distribution, ($m_A = m_B = 4$), with a system capacity of 30, and with 4 servers, ($c = 30, s = 4$). The number of states (thus the number of KFEs) for this system is $3,920$, and the number of number-in-system PMDEs is $m_A\,(m_B^2 + 2m_B + 4) = 96$.

In order to evaluate the quality of CA approximations we compare the numerically obtained solution of the KFEs to the numerical solution provided by the CA approximation. To do this we obtain a numerical solution for each of the Kolmogorov forward equations. And to do this efficiently for a multi-dimensional Markov Process we first map each state into an integer. Each system state is represented by a k-tuple, and each k-tuple is mapped onto an integer in the interval $[1, n_s]$, where n_s is the number of states in the state space, as given in Equation 2.10. While we could represent the full set of Kolmogorov forward equations in matrix form, that would result in enormous matrices that were extremely sparse; i.e., the recursive representation of the Markov Process in one dimension is far more computationally efficient than the a matrix representation in many dimensions.

For the $Ph_t/Ph_t/s/c$ system the state can be defined by 3-tuple; i) the phase of the arrival

process, $A(t)$, at time t, ii) the number being served in each service phase, $S(t)$, at time t, and, iii) the number in queue, $Q(t)$, at time t. So $A(t) \in [1, m_A]$, $Q(t) \in [0, c-s]$ and $S_i(t) \in [1, s]$ where $S_i(t)$ is the number of entities in their i^{th} phase of service. Obviously $\sum_{i=1}^{m_B} S_i(t) \leq s$ and $Q(t) > 0 \Rightarrow \sum_{i=1}^{m_B} S_i(t) = s$. So the state space at time t can be represented as $\{(A(t), Q(t), S_1(t), \ldots, S_{m_B}(t)) : A(t) \in [1, m_A], Q(t) \in [0, c-s], S_i(t) \in [0, s], \sum_{i=1}^{m_B} S_i(t) \leq s\}$.

A mapping of the states of the system, as represented by a $(m_B + 2)$-tuple, onto the integers $\in [1, n_s]$ is presented in Appendix 1.

Rueda and Taaffe [14] derive the PMDEs for the $Ph_t/Ph_t/s/c$ system. The computation time required to numerically solve the PMDEs for this system for some value of time, t, is far less than the time required to solve the KFEs, of course, since there are far fewer PMDEs than there are KFEs. Specifically, the number of PMDEs is not a function of the system capacity or number of servers. The number of differential equations required for numerical evaluation of the CA for the $Ph_t/Ph_t/s/c$ system has a polynomial form in the number of phases in the arrival process and in the service distribution and is $m_A(m_B^2 + 2m_B + 4)$. Rueda and Taaffe [14] present a *quasi-pseudo closure* approximation for the PMDEs for the $Ph_t/Ph_t/s/c$. The quality of their moment approximations is examined empirically in some detail in this section.

2.4.2 Notation for the $Ph_t/Ph_t/s/c$ System

To represent time-dependent arrival processes with Ph_t processes and time-dependent service distributions with Ph_t distributions we use the following notation:

- m_A represents the number of phases in the arrival process.
- m_B represents the number of phases in the service distribution.
- For some value of time, t, let $\boldsymbol{\alpha}(t)$ be a $m_A \times 1$ vector representing the arrival-process initial- state probabilities, where $\alpha_i(t)$ is the conditional probability that an entity will start its arrival process in phase i, given that the arrival process starts at time t.
- For some value of time, t, let $\boldsymbol{\beta}(t)$ be a $(m_B \times 1)$ vector representing the service-distribution initial-state probabilities, where $\beta_i(t)$ is the conditional probability that an entity will start its service process in phase i, given that the service process starts at time t.
- Let $\mathbf{a}(t)$ be a matrix of dimension $(m_A \times (m_A + 1))$ describing the time-dependent routing probabilities of entities in the Ph arrival process. For example, $a_{ij}(t)$ represents the instantaneous time-dependent probability that at time t an entity that is completing its time in phase i will then proceed to phase j. When $j = m_A + 1$, $a_{ij}(t)$ represents the time-dependent probability that an entity completing its time in phase i will then proceed to the absorbing state; i.e. an entity arrives to the queueing node.

- Let $\mathbf{b}(t)$ be a matrix of dimension $(m_B \times (m_B + 1))$ describing the time-dependent routing probabilities of entities in the Ph service distribution. For example, $b_{ij}(t)$ represents the instantaneous time-dependent probability that at time t an entity that is completing its time in phase i will then proceed to phase j. When $j = m_B + 1$, $b_{ij}(t)$ represents the time-dependent probability that an entity completing its time in phase i will then proceed to the absorbing state; i.e. an entity completes service and leaves the queueing node.

- Let $\boldsymbol{\lambda}(t)$ be a $(m_A \times 1)$ vector representing the arrival-phase rates (i.e., the reciprocal of the arrival-phase mean resident times). Thus $\lambda_i(t)$ is the departure rate from phase i at time t.

- Let $\boldsymbol{\mu}(t)$ be a $(m_B \times 1)$ vector representing the service-distribution rates (i.e., the reciprocal of the mean time spent in service phase i). Thus $\mu_i(t)$ is the departure rate from phase i at time t.

For more information regarding the Ph distribution the reader can refer to [14] or [10].

2.4.3 Performance of the Original CA for Approximating the Time-Dependent System-Size Moments

Traffic Intensity Robustness

In examining the quality of the CA approximations for the $Ph_t/Ph_t/s/c$ system we choose to vary the traffic intensity of a $Ph_t/Ph_t/s/c$ system by increasing or decreasing the service rates (the elements in the vector $\boldsymbol{\mu}(t)$). This can easily be done by multiplying the service rate vector $\boldsymbol{\mu}(t)$ by a coefficient m. Increasing m results in faster service rates (smaller service times) and thus lower traffic intensity while decreasing m results in higher traffic intensity. We vary m for a $Ph_t/Ph_t/s/c$ system and observe the effects of the traffic intensity on the quality of our CA approximations.

We present four $Ph_t/Ph_t/s/c$ systems in Examples 1, 2, 3 and 4 and for each example we vary m. Figures 2.5 to 2.15 are plots of the zero$^{\text{th}}$, first, second, and partial moments, where the zero$^{\text{th}}$ partial moments are partial probabilities. For each of these moments we plot the time-dependent moment values versus time as well as the time-dependent moment approximation versus time.

Specifically we plot the numerically exact and the approximate zero$^{\text{th}}$, first and second moments of $N(t)$ as well as the partial moments $\mathrm{E}[N(t), N(t) < s]$ (First Partial Moment Lower, FPML), $\mathrm{E}[N(t), N(t) \geq s]$ (First Partial Moment Upper, FPMU), $\mathrm{E}[N^2(t), N(t) < s]$ (Second Partial Moment Lower, SPML), $\mathrm{E}[N^2(t), N(t) \geq s]$ (Second Partial Moment Upper,

SPMU), P($N(t) < s$) (Zeroth Partial Moment Lower, ZPML) and P($N(t) \geq s$) (Zeroth Partial Moment Upper, ZPMU).

In doing the computations to examine the accuracy of the CA approximations of course the ratio of the computer time to evaluate the full set of KFEs to approximating PMDEs is based on the ratio of the number of KFEs to the number of PMDEs, as discussed earlier. This ratio can be enormous:

$$(s + m_B\,(c - s + 1))\,\mathrm{C}\,(m_B + s - 1, s)\,m_B^{-1}\,(m_B^2 + 2m_B + 4)^{-1} \tag{2.11}$$

Example 1

$s = 3, c = 20, m_A = 3, m_B = 3$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.3, \quad 0.3, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.3 & 0.2 & 0.4 \\ 0.3 & 0.1 & 0.2 & 0.4 \\ 0.2 & 0.2 & 0.3 & 0.3 \end{pmatrix}$$

$\boldsymbol{\lambda}(t) = [3 + 3\sin(t/0.03\pi), \quad 3 + 3\sin(t/0.3\pi), \quad 2 + 2\sin(t/0.05\pi)],$

$\boldsymbol{\mu} = [0.5m, \quad 0.5m, \quad 1m]$ for $m = 1, 3, 5.$

Figure 2.5: Example 1 m=1

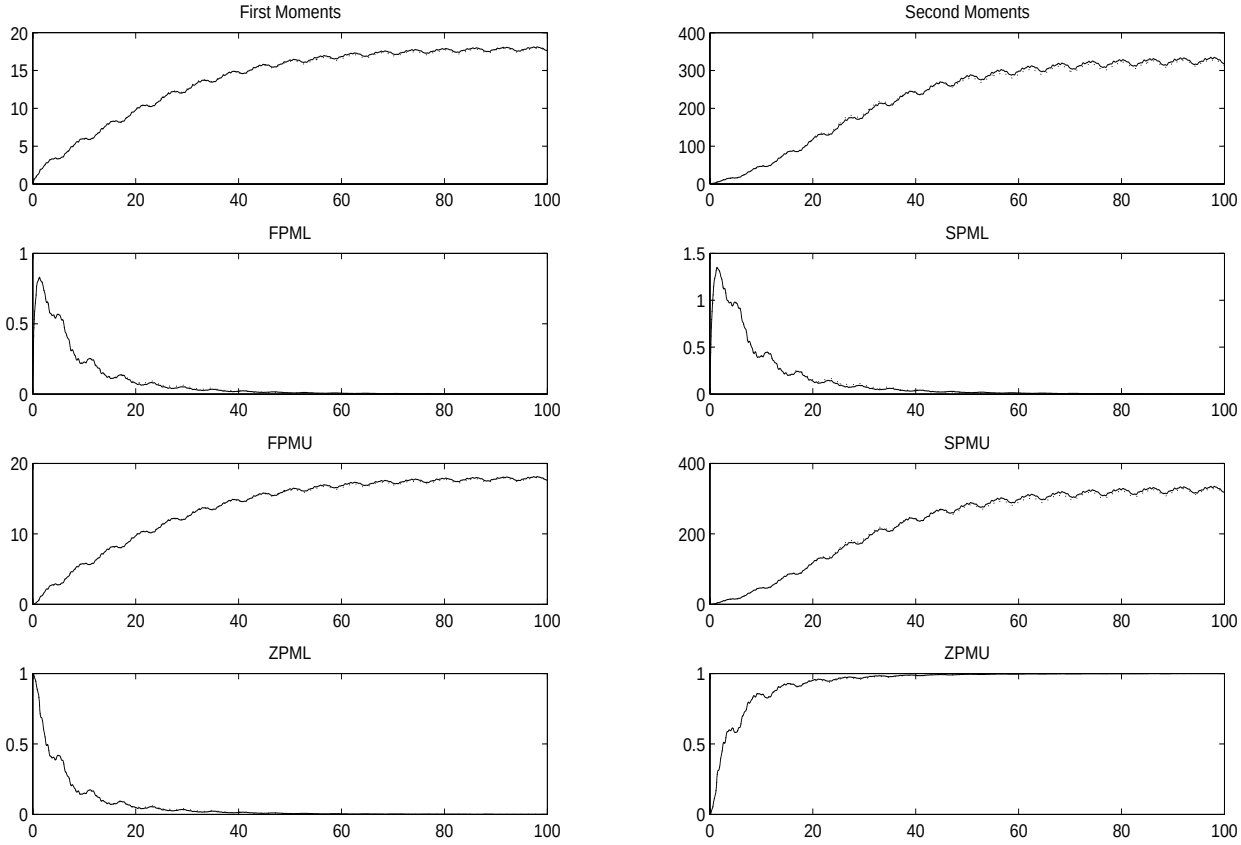


Figure 2.6: Example 1 m=3

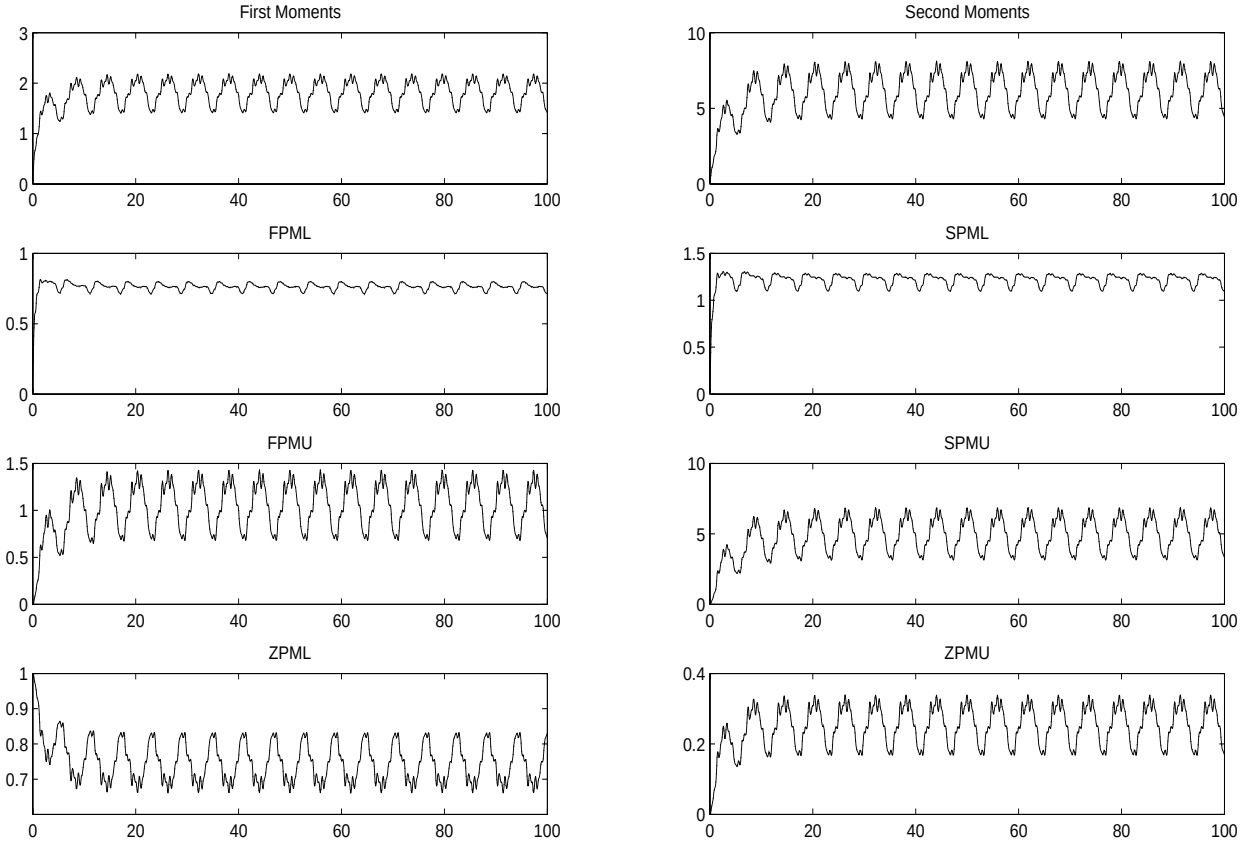
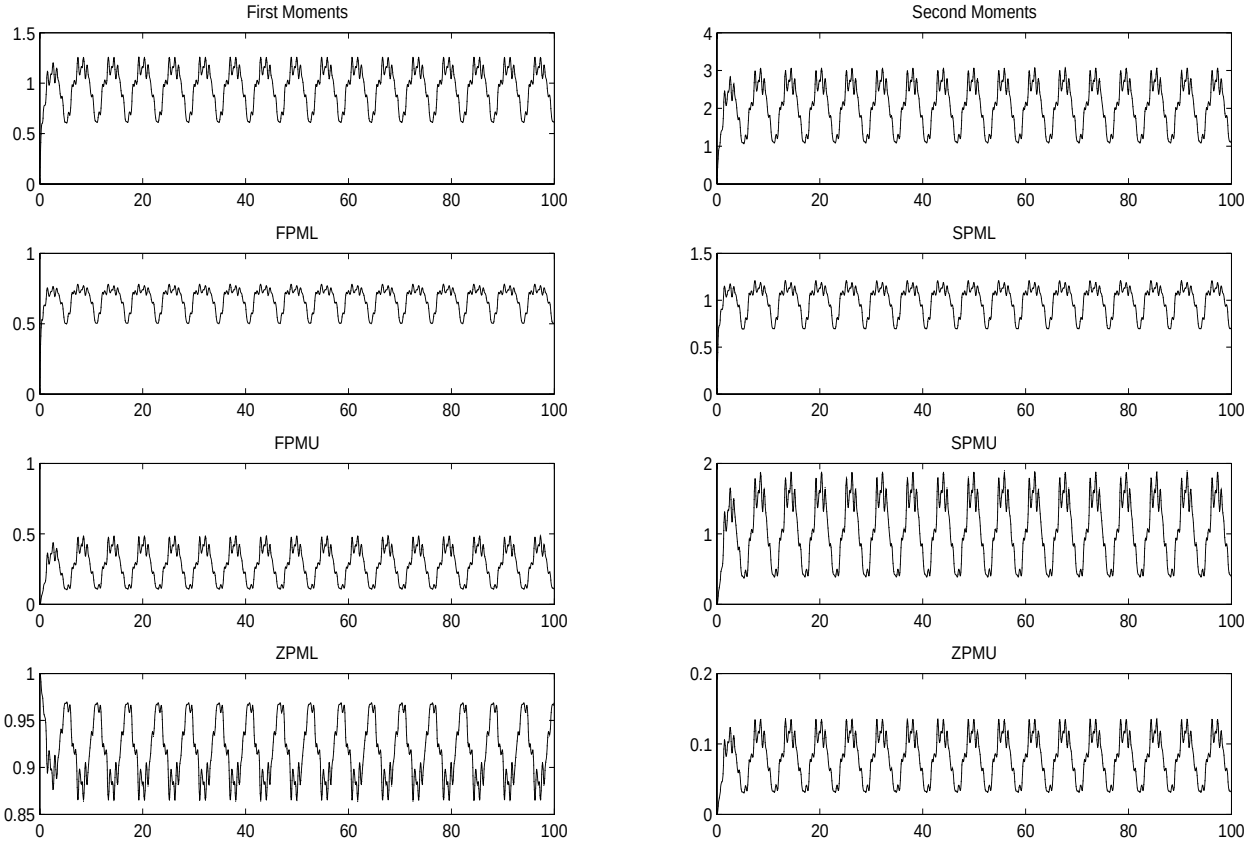


Figure 2.7: Example 1 m=5



Example 2

$s = 4, c = 20, m_A = 4, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.1, \quad 0.3]$ $\qquad$ $\boldsymbol{\beta} = [0.6, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.2 & 0.3; \\ 0.3 & 0.1 & 0.1 & 0.3 & 0.2; \\ 0.3 & 0.2 & 0.1 & 0.2 & 0.2; \\ 0.3 & 0.1 & 0.2 & 0.3 & 0.1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.4 & 0.4 & 0.2 \end{pmatrix}$$

$\boldsymbol{\lambda}(t) = [3 + 0.5\sin(t/(3\pi)), \quad 3 + 0.5\sin(t/(3\pi)), \quad 2 + 0.5\sin(t/(3\pi)), \quad 2 + 0.5\sin(t/(3\pi))]$

$\boldsymbol{\mu} = [0.5m, \quad 0.5m]$ for $m = 1, 0.2, 10.$

Figure 2.8: Example 2 m=1

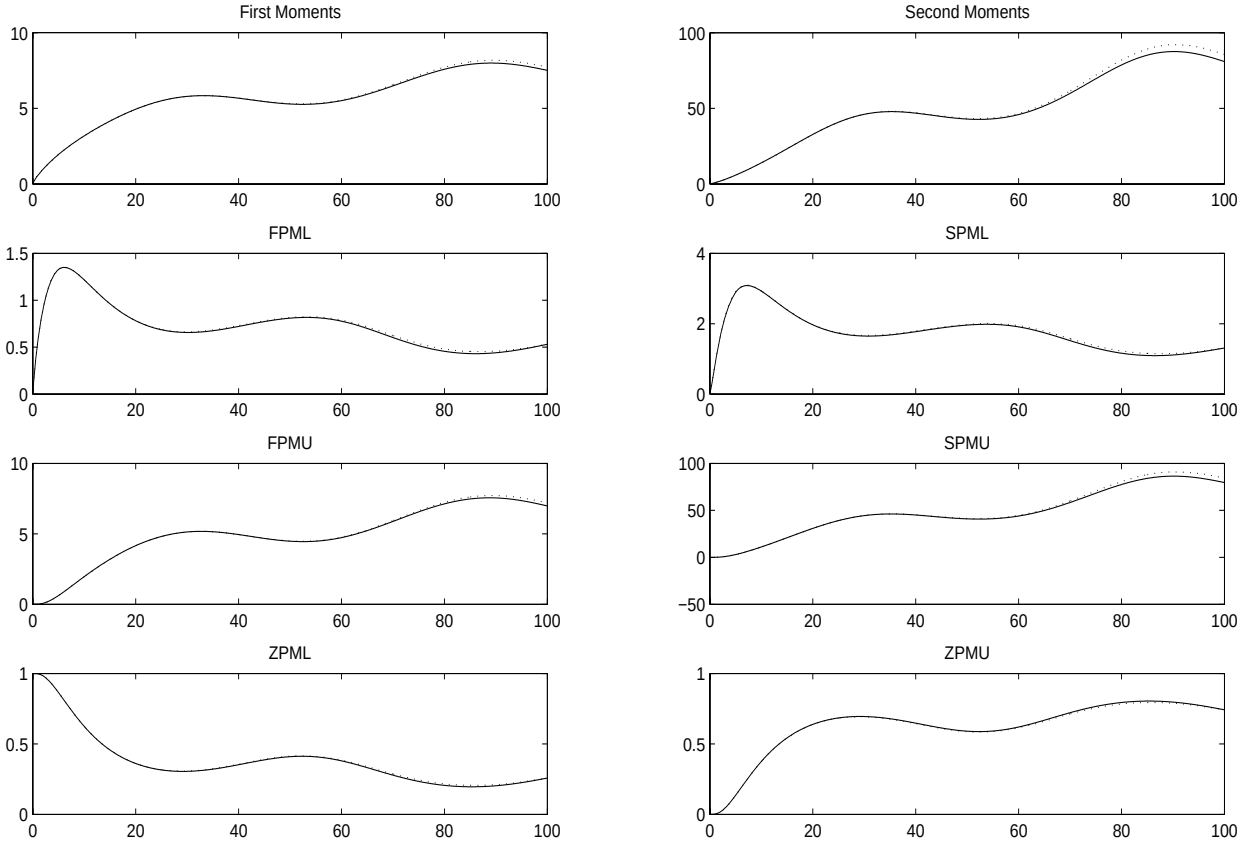


Figure 2.9: Example 2 m=0.2

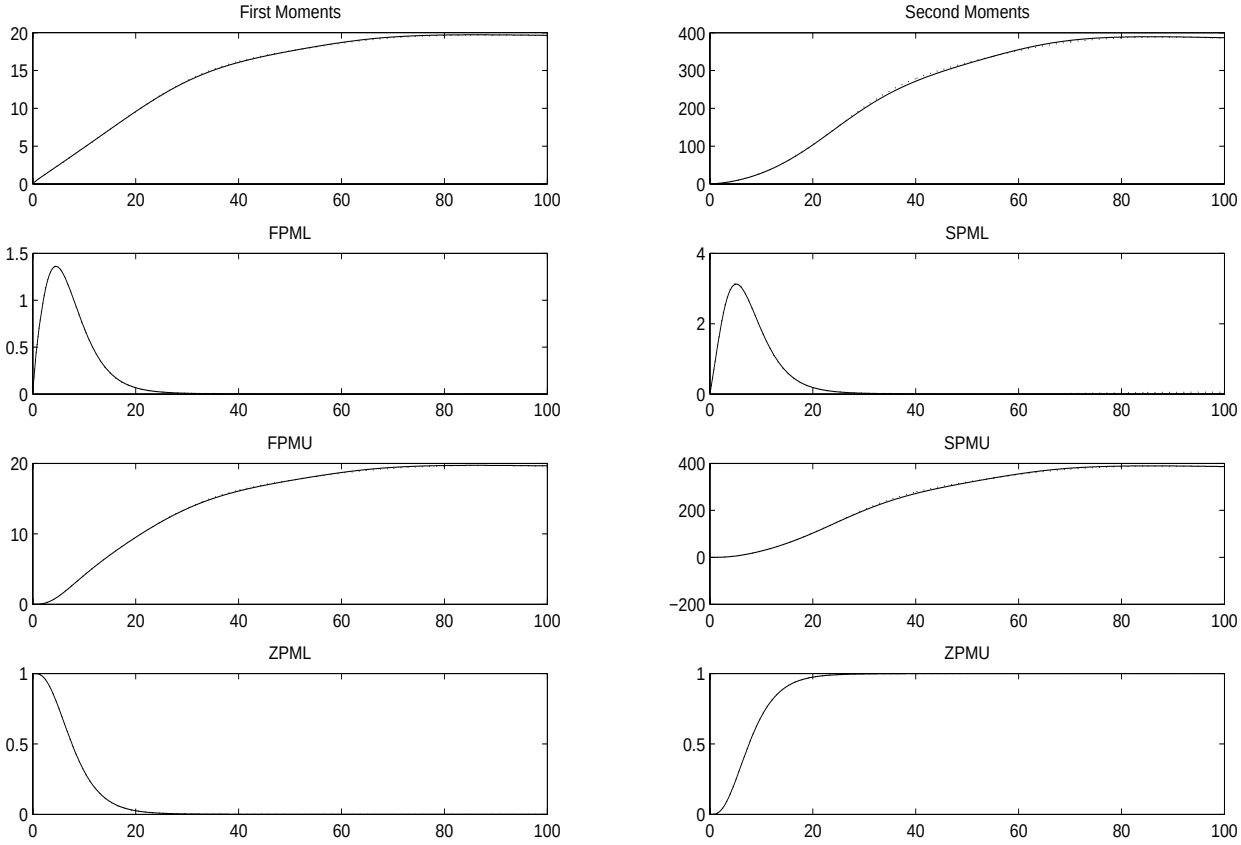
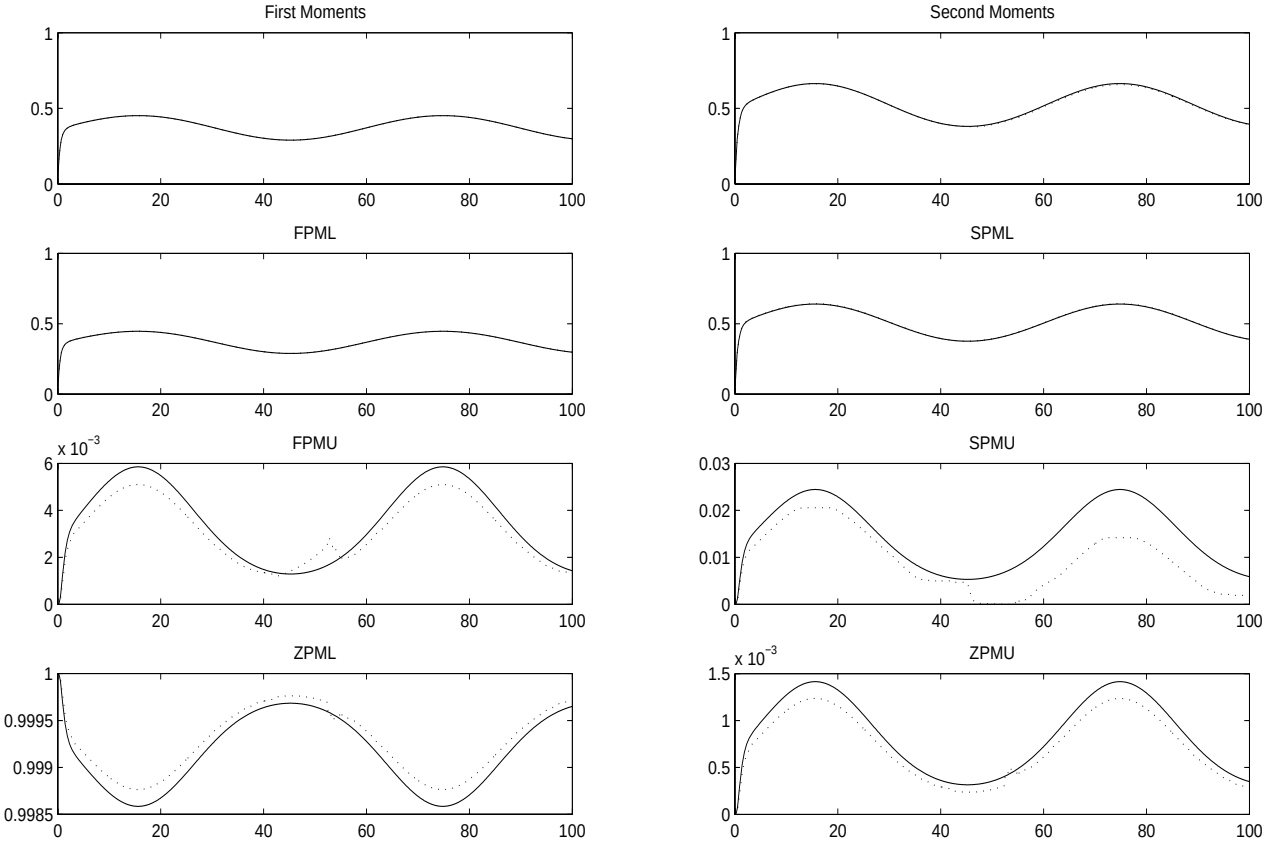


Figure 2.10: Example 2 m=10



Example 3

$s = 3, c = 20, m_A = 3, m_B = 3$

$\boldsymbol{\alpha}(t) = [0.6\,|\sin(t/0.03\pi)|\,, \quad 0.6(1 - |\sin(t/0.03\pi)|), \quad 0.4]$

$\boldsymbol{\beta}(t) = [0.4, \quad 0.6\,|\cos(t/0.03\pi)|\,, \quad 0.6(1 - |\cos(t/0.03\pi)|)]$

$$\mathbf{a}(t) = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.7\,|\sin(t/0.3\pi)| & 0.7(1 - |\sin(t/0.3\pi)|) & 0.3\,|\sin(t/0.01\pi)| & 0.3(1 - |\sin(t/0.01\pi)|) \\ 0.8\,|\sin(t/0.3\pi)| & 0.8(1 - |\sin(t/0.3\pi)) & 0.2\,|\cos(t/0.01\pi)| & 0.2(1 - |\cos(t/0.01\pi)|) \end{pmatrix}$$

$$\mathbf{b}(t) = \begin{pmatrix} 0.75\,|\sin(t/0.05\pi)| & 0.75(1 - |\sin(t/0.05\pi)|) & 0.25\,|\sin(t/0.02\pi)| & 0.25(1 - |\sin(t/0.02\pi)|) \\ 0.3 & 0.1 & 0.2 & 0.4 \\ 0.7\,|\sin(t/0.3\pi)| & 0.7(1 - |\sin(t/0.3\pi)|) & 0.3\,|\cos(t/0.01\pi)| & 0.3(1 - |\cos(t/0.01\pi)|) \end{pmatrix}$$

$\boldsymbol{\lambda}(t) = [3 + 3\,|\sin(t/0.03\pi)|\,, \quad 3 + 2\,|sin(t/0.3\pi)|\,, \quad 2 + |\sin(t/0.05\pi)|]$

$\boldsymbol{\mu}(t) = [2m(0.5 + 0.5\,|\sin(t/0.03\pi)|), \quad 2m(0.5 + 0.5\,|\sin(t/0.3\pi)|), \quad 2m(1 + |\sin(t/0.05\pi)|)]$

for $m = 1, 4$.

Figure 2.11: Example 3 m=1

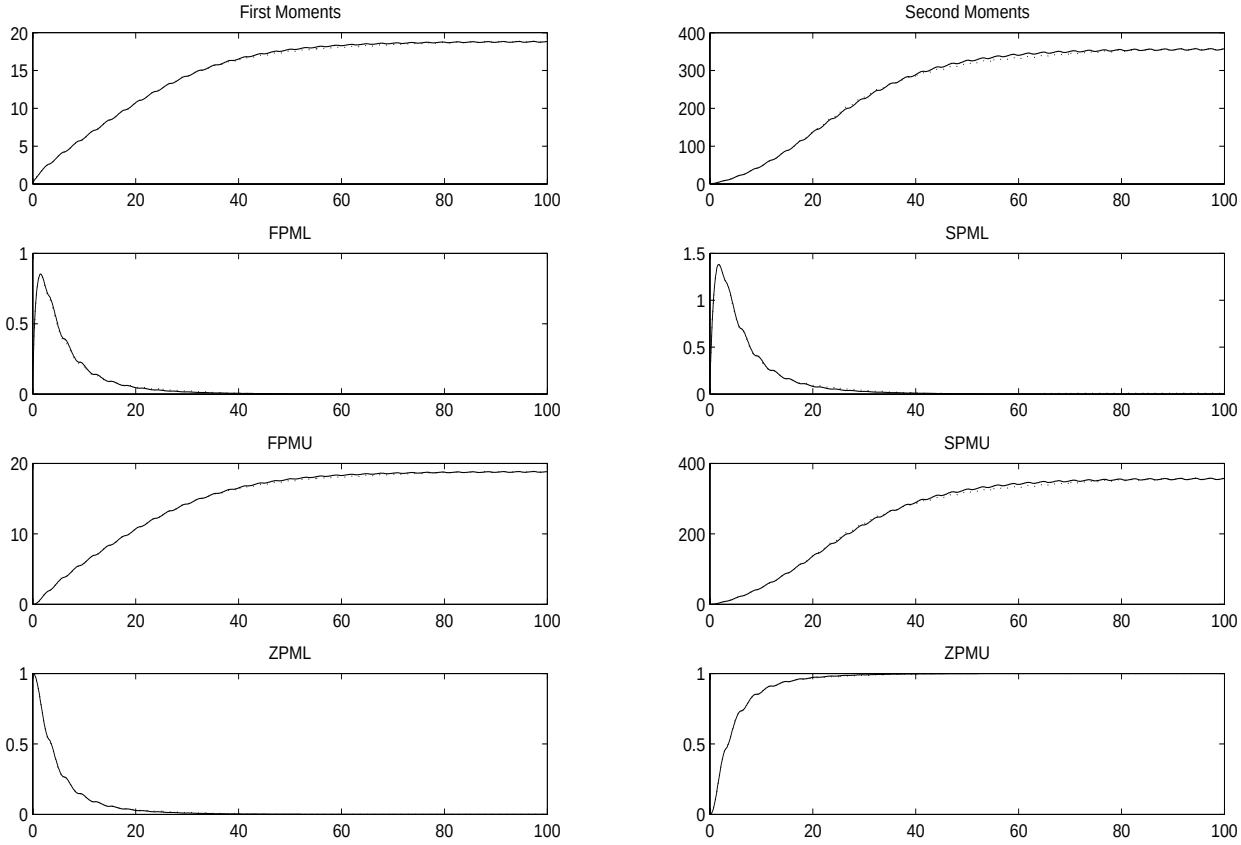
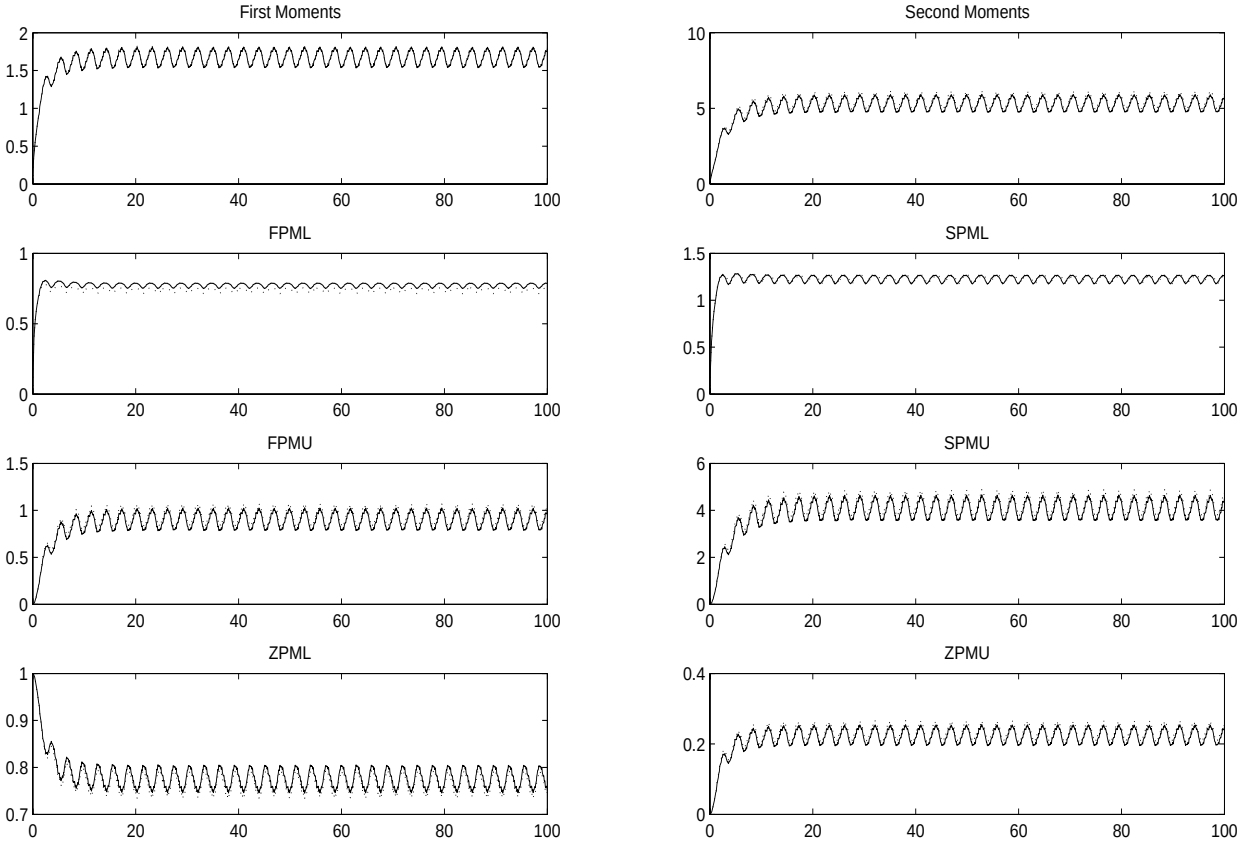


Figure 2.12: Example 3 m=4



Example 4

$s = 3, c = 20, m_A = 1, m_B = 1$

$\boldsymbol{\alpha} = [1]$ $\quad\quad$ $\boldsymbol{\beta} = [1]$

$\mathbf{a} = \begin{pmatrix} 0 & 1 \end{pmatrix}$ $\quad\quad$ $\mathbf{b} = \begin{pmatrix} 0 & 1 \end{pmatrix}$

$\boldsymbol{\lambda} = [2]$ $\quad\quad$ $\boldsymbol{\mu} = [0.8m]$

Figure 2.13: Example 4 m=1

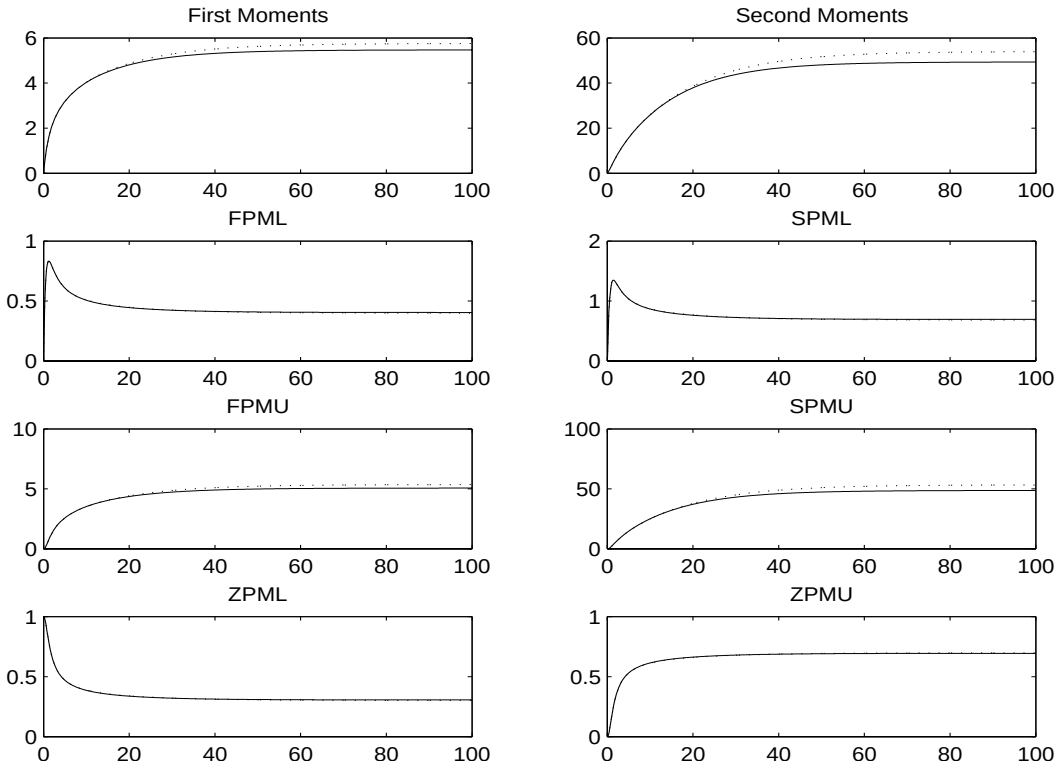


Figure 2.14: Example 4 m=2

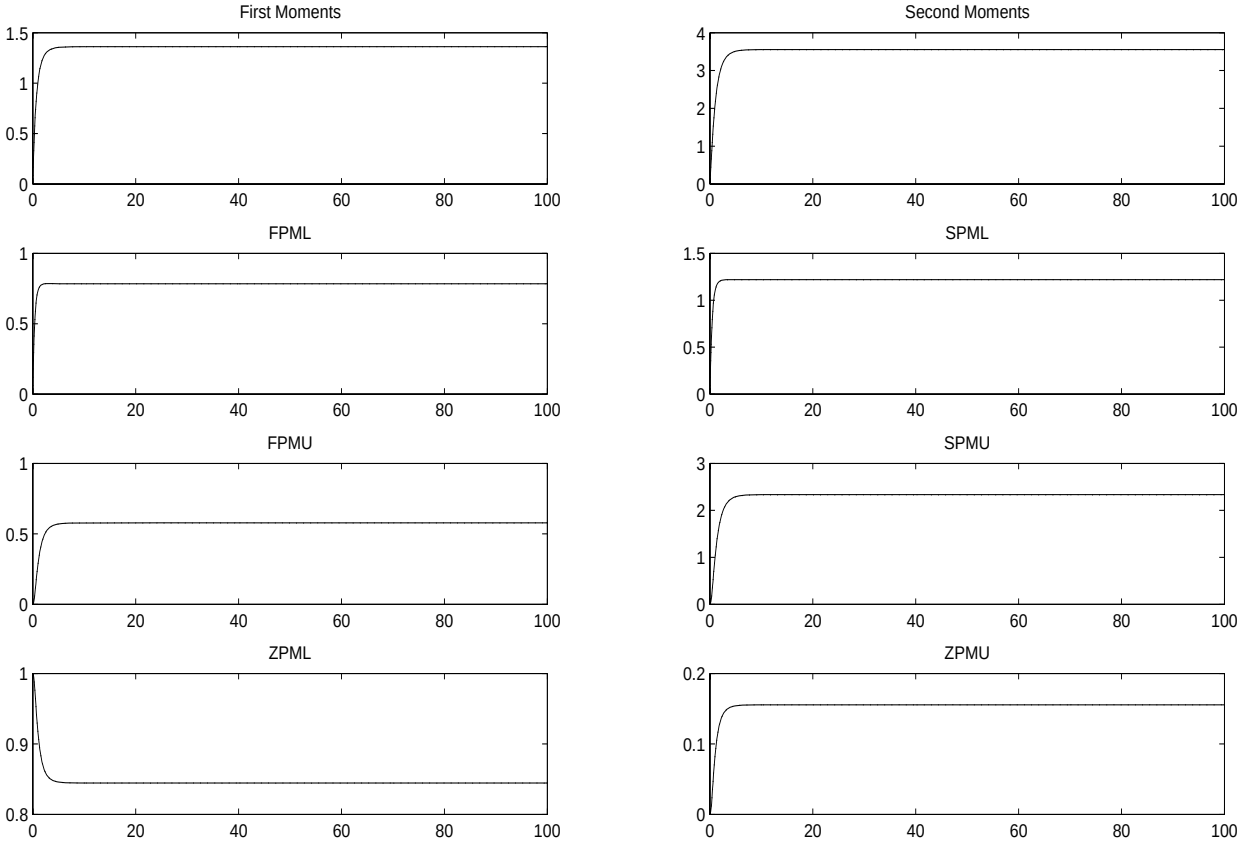
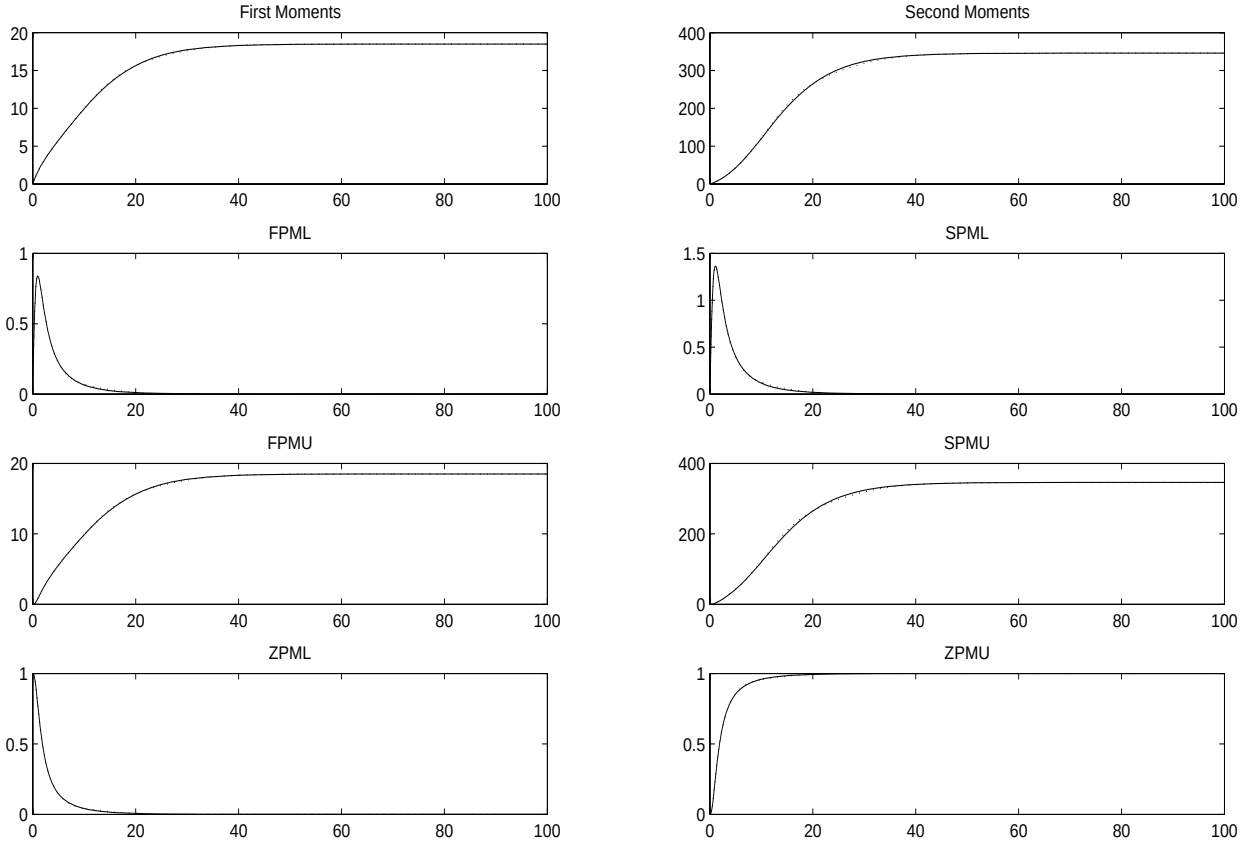


Figure 2.15: Example 4 m=0.5



In Examples 1, 2 and 3 the approximate moments obtained by solving the CAs are consistent with the numerically exact results obtained by solving the KFEs. In these examples we examined the robustness of the accuracy to traffic intensity. This can be seen in Figures 2.5 to 2.12.

We highlight Example 4 because for Example 4 the approximations are not uniformly as good for all traffic intensities. For the $m = 1$ case, the moment approximations have a larger percent error. For time $t = 90$ the percent error for the first moment approximation is 5.15% and the percent error for the second moment approximation is 9.2% (Figure 2.13). We next consider the partial moments and the boundary probabilities appearing on the RHS of the PMDEs in detail to determine the source of the error.

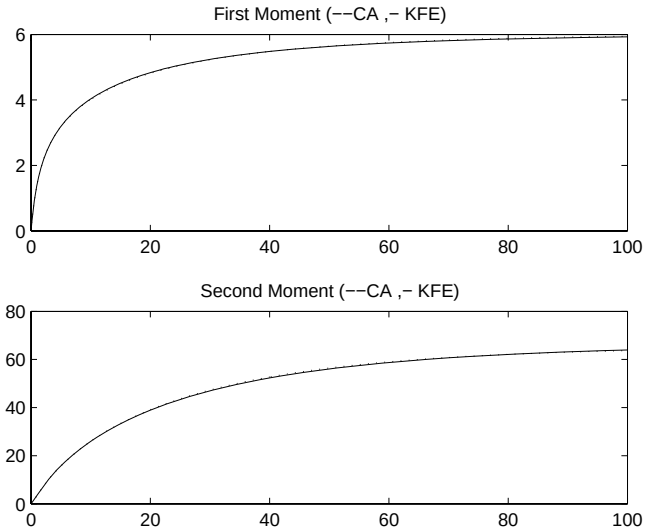
Example 4 is an $M/M/s/c$ stationary queueing node and the PMDEs are given in Equations 2.5 to 2.9. The transient partial-moment approximations in Figure 2.13 are all accurate except for FPMU and SPMU. The transient boundary probabilities $\mathrm{P}(N(t) = s - 1)$ and $\mathrm{P}(N(t) = s)$ appear on the RHS of the FPML, SPML and ZPML PMDEs (Equations 2.5, 2.6 and 2.9 respectively). It is reasonable to conclude that the boundary probability approximations for $\mathrm{P}(N(t) = s-1)$ and $\mathrm{P}(N(t) = s)$ provide accurate results over time since the FPML, SPML and ZPML approximations are good. The source of error is found in the remaining terms on the RHS of the FPMU and SPMU PMDEs (Equations 2.7 and 2.8); i.e., the boundary probability $\mathrm{P}(N(t) = c)$. Since $\mathrm{P}(N(t) = c)$ has a negative coefficient in the PMDEs and the FPMU and SPMU are over approximated, then $\mathrm{P}(N(t) = c)$ must be under approximated. For example at time 90, $\mathrm{P}(N(t) = c = 20) = 0.0054$ and the approximation

is $\widetilde{\mathrm{P}}(N(t) = 20) = 0.0025$.

We next consider an $M/M/s/c$ stationary system having the same input parameters as Example 4, except now the system capacity is set to 100 instead of 20. We examine this case too because with $C = 100$, we expect $\mathrm{P}(N(t) = c = 100)$ to be essentially 0.0, thus we expect the overall CA to be more accurate and the percent error in the moments to decrease.

The moment plots are shown in Figure 2.16. Observe that the percent error in the first and second moments has decreased significantly from the $c = 20$ case; so the approximate moments are now indistinguishable from the numerically exact moments.

Figure 2.16: Example 4 Moments (c=100)



Example 4 with $c = 20$ and $c = 100$ provide evidence that our intuition is correct. Necessarily the CA will perform well for systems for which the boundary-state probabilities, $\mathrm{P}(N(t) = s-1)$, $\mathrm{P}(N(t) = s)$, and $\mathrm{P}(N(t) = c)$ are essentially zero. For example the CA computations for systems having an infinite number of servers and infinite capacity are exact because all

boundary probabilities are zero, see Nelson and Taaffe [10]. For these cases we need do no experiments to determine accuracy.

If the approximated boundary probability is several standard deviations away from the mean number-in-system, then our approximate probability $\rightarrow 0$ for any reasonably shaped distribution such as the PE distribution. Boundary-state probabilities are close to zero for systems having the mass of probability all in the lower partition but not near state $s-1$, or for systems that have enough traffic that the mass of probability is in the upper partition but both far from state s and far from state c. We also refer to the Markov inequality as presented in [11]. Another way to look at these cases is to observe that the traffic intensity is such that neither the finiteness of the number of servers nor the finiteness of the capacity is relevant; e.g., the capacity is *infinite* relative to the traffic intensity. For these examples there is essentially no approximation occurring and only a small number of experiments are needed to verify accuracy.

We do need to examine carefully experiments for cases having significant probability in the boundary-states. For the CA to perform well in these cases the boundary-state probabilities (significantly different from zero) need to be approximated well.

We vary the traffic intensities for each case presented in Examples 1, 2, 3 and 4. Different traffic intensities results in different probability mass functions; thus for each example case the quality of the approximations is tested for different values of boundary-state probabilities (tested for zero as well as significant values of the boundary-state probabilities). As

stated earlier, the CA approximations perform well for different traffic intensities as shown in Examples 1, 2, and 3.

We further analyze Example 4 for $m = 1$ and vary the traffic intensity by varying μ. We compute the percent error at time $t = 90$ for the first and second moments of the number-in-system for different values of μ as shown in Table 2.1. The moments errors seem to peak at around $\mu = 0.8$ as can be seen in Table 2.1. So the first and second moments errors of 5.15% and 9.14% respectively are close to the maximum attainable percent error when varying μ. So the CA still provides relatively good approximations for the queueing system.

Table 2.1: Example 4, First and Second Moment % Errors ($t = 90$)

	First Moment % Error	Second Moment % Error
$\mu = 0.5$	0.38%	1.08%
$\mu = 0.6$	2.45%	4.32%
$\mu = 0.7$	3.10%	5.31%
$\mu = 0.8$	5.15%	9.20%
$\mu = 0.9$	1.24%	0.76%

Coefficients of Variation Robustness

We consider two examples (E5 and E6). The arrival and service processes have high variability in E5 and low variability in E6. We chose two Erlang distributions for the inter-arrival

time and service time distributions in E6 since the Erlang distribution can attain the lowest coefficient of variation among all Ph distributions with n phases. When the rates of the Erlang distribution are all equal then the coefficient of variation attains its minimum which is $n^{-1/2}$ and is independent of the rate. In E6 we test the performance of the CA algorithm under low variability inter-arrival time and service time distributions.

We chose two Hyper-exponential distributions for the inter-arrival time and service time distributions in E5. In E5 we tested the performance of the CA algorithm under high variability inter-arrival time and service time distributions.

Example 5 (Hyper-exponential distributions)

In E5 the inter-arrival-time distribution has a mean of 0.505 and a coefficient of variation of 1.7093. The service-time distribution has a mean of 0.674 and a coefficient of variation of 1.4037. We consider coefficients of variation to be large or small relative to the coefficient of variation for the negative exponential distribution; thus by "large" we mean $>> 1$, and by "small" we mean $<< 1$.

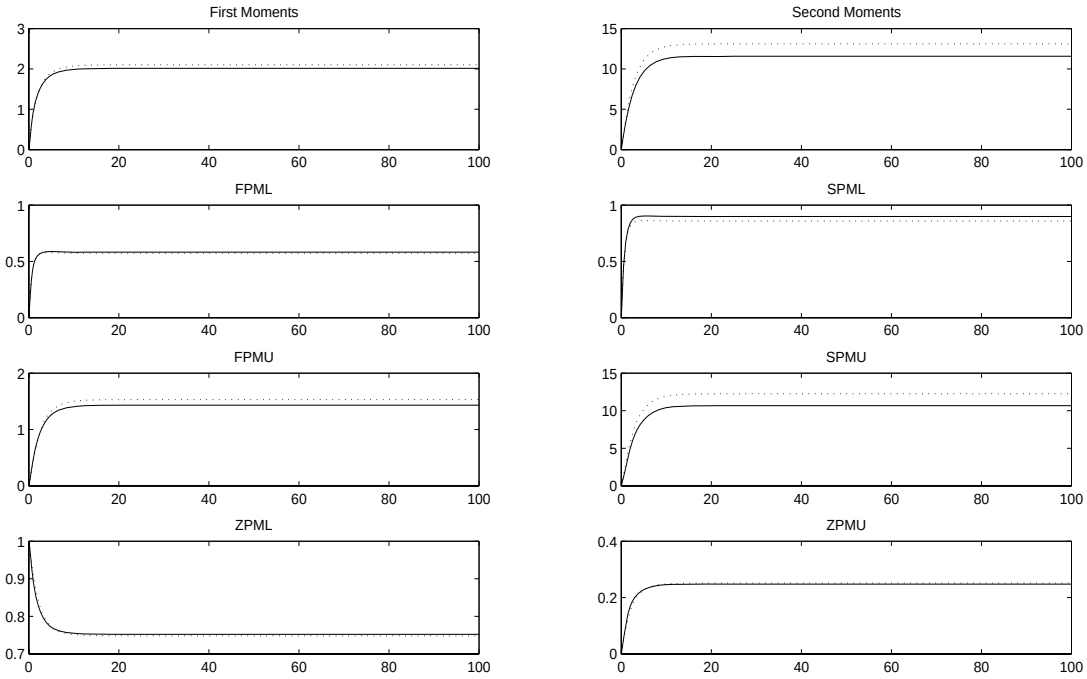
$s = 3, c = 20, m_A = 2, m_B = 2$

$\boldsymbol{\alpha} = [1/2, \quad 1/2]$ $\qquad\qquad$ $\boldsymbol{\beta} = [2/3, \quad 1/3]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 1 \\ \\ 0 & 0 & 1 \end{pmatrix} \qquad\qquad \mathbf{b} = \begin{pmatrix} 0 & 0 & 1 \\ \\ 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1, \quad 100], \qquad \boldsymbol{\mu} = [1, \quad 100]$

Figure 2.17: Example 5



The first and second moment percent errors of the number in system at times (t = 10, 20, 30, ..., 100) are computed and the averages are 4.35% and 13.31% respectively for the high variability inter-arrival and service distributions of E5. We decrease the variability of the arrival and service distributions by changing the arrival and service rates as follows,

$\boldsymbol{\lambda} = [1, \quad 10], \qquad \boldsymbol{\mu} = [1, \quad 10]$

The first and second moment percent error averages is 2.05% and 7.73% . We further decrease the variability of the arrival and service distributions without significantly changing the first moment of the inter-arrival and inter-service distributions as follows

$\boldsymbol{\lambda} = [4, \quad 4/3], \qquad\qquad \boldsymbol{\mu} = [2, \quad 1]$

The first and second moment percent error averages is 0.58% and 2.24%. The first and second moment percent error averages decreased as the variability of the inter-arrival and service distributions decreased.

Example 6 (Erlang distributions)

In E6 the inter-arrival-time distribution has a mean of 1 and a coefficient of variation of 0.5. The service-time distribution has a mean of 2.667 and a coefficient of variation of 0.5. Both distributions are hypo-exponential (Erlang) hence they are distributions with low variability. On our laptop computer the full set of KFEs required 3.29 minutes to solve only 0.045 minutes to solve the PMDEs using our CAs.

$s = 3, c = 20, m_A = 4, m_B = 4$

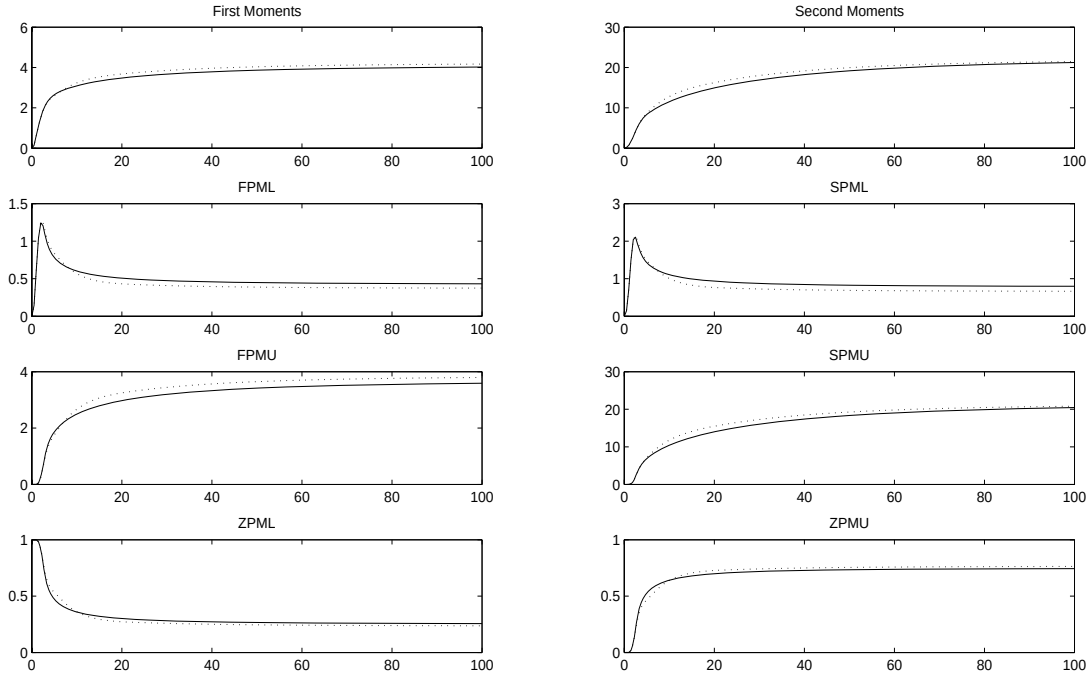
$\boldsymbol{\alpha} = [1, \quad 0, \quad 0, \quad 0] \qquad\qquad \boldsymbol{\beta} = [1, \quad 0, \quad 0, \quad 0]$

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [4, \quad 4, \quad 4, \quad 4] \qquad\qquad \boldsymbol{\mu} = [1.5, \quad 1.5, \quad 1.5, \quad 1.5]$

The closure approximations perform modestly better for the low-variability models. These

Figure 2.18: Example 6



examples demonstrate our conclusion that the CA seems to be sensitive to varying the variability of the service/arrival distribution.

Robustness to the Number of Phases

We next considered the effect that the number of service and arrival phases has on the approximation quality. In order to perform such a comparison, we represented the same distribution in different ways where each way has a different number of phases. For example, the obvious Ph representation of an exponential distribution with rate λ is to use one phase. It is also possible to represent the same distribution with a Ph distribution that has n phases.

Consider an exponential distribution with rate $\lambda = 2.1$. This exponential distribution can be represented by :

1) A one phase Ph distribution with rate $\lambda = 3$ and a feedback probability of 0.3 (geometric mix), hence an effective rate of $3(1 - 0.3) = 2.1$.

$\boldsymbol{\alpha} = [1]$

$\mathbf{a} = [0.3, 0.7]$

$\boldsymbol{\lambda} = 3$

2) An n-phase Ph distribution with each phase having rate $\lambda = 3$ and the following transition matrices:

$\boldsymbol{\alpha} = [1/n, \quad 1/n, \quad \ldots \quad 1/n]$

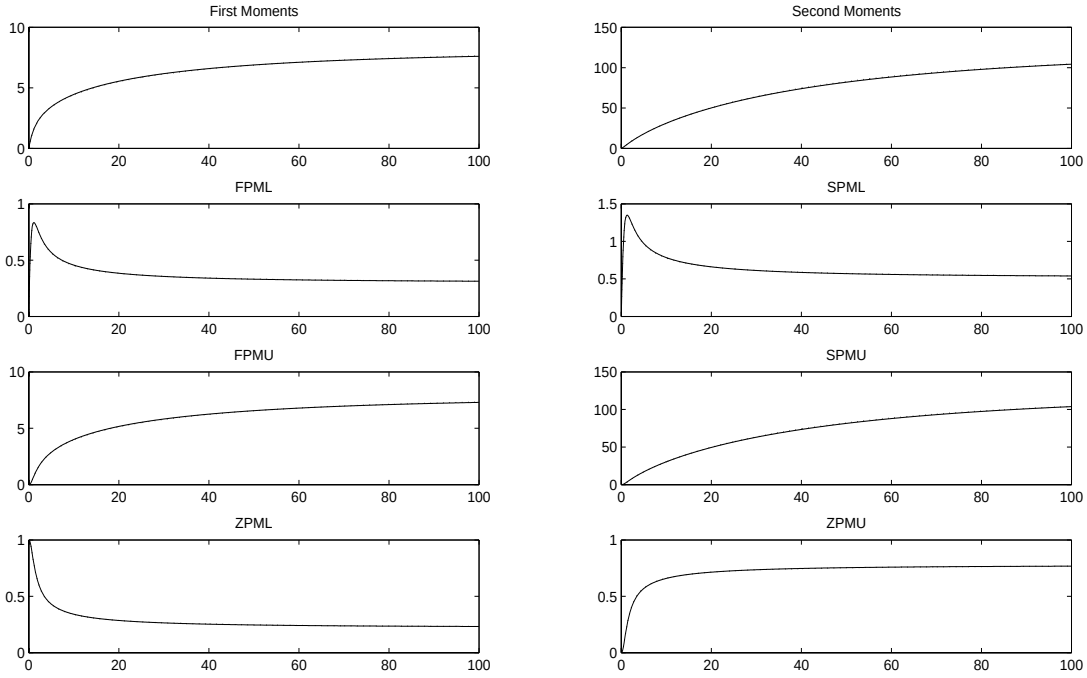
$$\mathbf{a}_{n\times(n+1)} = \begin{pmatrix} 0 & 0.3 & 0 & \cdots & 0 & 0.7 \\ 0 & 0 & 0.3 & \cdots & 0 & 0.7 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0.3 & \cdots & 0.3 & 0.7 \\ 0.3 & 0 & 0 & \cdots & 0 & 0.7 \end{pmatrix}$$ where the $(n+1)$ state is the absorbing state

We investigate the robustness of the CA to the number of phases used in representing (equivalent) arrival/service distributions because the number of PMDEs is a function of the number of phases in the arrival/service distributions. Of course so too is the number of KFEs

a function of the number of phases used in the arrival/service distribution representations, but in that case we can solve the full set of KFEs to numerical exactness. Since the number of PMDEs is a function of the number of phases used in representing the arrival/service distribution, then so too is the number of approximations a function of the number of phases used to represent the arrival/service distribution. So the number of approximations is proportional to the number of PMDEs as shown in Equation 2.11. So our intuition suggests that the quality of the overall CA approximation should improve as the ratio of approximations to the number of differential equations decreases.

Consider the $M/M/s/c$ system with $s = 3, c = 100, \lambda = 3(1-0.3) = 2.1, \mu = 1(1-0.2) = 0.8$. The arrival and service processes are exponential, so we vary their phase representation without changing the distribution. Let $(m_A, m_B) \equiv \{$The $M/M/s/c$ system represented by m_A phases and m_B phases$\}$. Using (1,1), the CA is almost exact, as can be seen in Figure 2.19. The same holds for the (3,3). Thus increasing the number of phases in the arrival and service distribution did not have an effect on the moment approximations although more approximations are required. So the ratio of the number of approximations or the ratio of the number of differential equations (Equation 2.11) is not a good predictor of the overall approximation quality. The approximation quality is robust to the number of phases we might choose to represent a fixed input distribution.

Figure 2.19: (1,1) and (3,3)



Observations Regarding the Partial Moment Approximations

The quality of the boundary probabilities depends on the quality of the first and second number-in-system partial-moment approximations. We show that in general the quality of the first and second moment approximations are better than the first and second partial-moment approximations, respectively.

While analysts may not be interested directly in the values of the partial moments, we are interested in use of the partial moments in constructing our overall CA approximation. The observation that the overall moments are more accurately approximated than the partial moments may suggest that no state-space partitioning is necessary; whereas, the opposite

is actually true. The overall moment approximations are indeed improved if the partial moments are approximated and used to compute the overall moment approximations.

We take a closer look at the PMDEs used in [14] (also presented in Appendix 2) from which the moments are obtained. Consider the differential equations for the marginal probabilities for the system, at time t, to be in subspace $\Omega_1 \equiv \{0, 1, 2, ..., s-1\}$ or $\Omega_2 \equiv \{s, s+1, ..., c\}$. The random variable $A(t) = \ell$ indicates the system is in the ℓ^{th} arrival phase at time t:

$$\begin{aligned}
\mathrm{P}'\ (N(t) \in \Omega_1,\ A(t) = \ell) \ &= \ -\lambda_\ell(t)\,\mathrm{P}\left(N(t) \in \Omega_1, A(t) = \ell\right) \\
&\quad - \sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t) = s-1, A(t) = i\right) \\
&\quad + \sum_{i=1}^{m_A} a_{i\ell}(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t) \in \Omega_1, A(t) = i\right) \\
&\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t) \in \Omega_1, A(t) = i\right) \\
&\quad + \sum_{i=1}^{m_B} b_{i,m_B+1}(t)\,\mu_i(t)\,\mathrm{E}\left[N_i(t), N(t) = s, A(t) = \ell\right]
\end{aligned} \tag{2.12}$$

$$\begin{aligned}
\mathrm{P}'\ (N(t) \in \Omega_2,\ A(t) = \ell) \ &= \ -\lambda_\ell(t)\,\mathrm{P}\left(N(t) \in \Omega_2, A(t) = \ell\right) \\
&\quad - \sum_{i=1}^{m_B} b_{i,m_B+1}(t)\,\mu_i(t)\,\mathrm{E}\left[N_i(t), N(t) = s, A(t) = \ell\right] \\
&\quad + \sum_{i=1}^{m_A} a_{i\ell}(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t) \in \Omega_2, A(t) = i\right) \\
&\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t) = s-1, A(t) = i\right) \\
&\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t) \in \Omega_2, A(t) = i\right)
\end{aligned} \tag{2.13}$$

Adding 2.12 and 2.13:

$$\begin{aligned} \mathrm{P}'(A(t)=\ell) &= -\lambda_\ell(t)\mathrm{P}(A(t)=\ell) \\ &\quad + \sum_{i=1}^{m_A} a_{i\ell}(t)\lambda_i(t)\mathrm{P}(A(t)=i) \\ &\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}(t)\alpha_\ell(t)\lambda_i(t)\mathrm{P}(A(t)=i) \end{aligned} \tag{2.14}$$

Equation 2.14 does not require any closure approximations, so $\mathrm{P}(A(t)=\ell)$ can either be accurately obtained by solving equation 2.14 numerically exactly or by evaluating 2.12 and 2.13 – which require approximations to close – and then adding the integrals of 2.12 and 2.13 to produce $\mathrm{P}(A(t)=\ell)$. Notice, however, that on the RHS of equations 2.12 and 2.13 the boundary probabilities that require approximation occur with opposite signs. Thus even if 2.12 and 2.13 are used *and* the boundary probabilities are badly approximated, it does not matter in the computation of $\mathrm{P}(A(t)=\ell)$, since the boundary-probability terms cancel each other out. Of course the quality of the overall CA approximation is affected by the quality of the boundary-probability approximations because these approximations are used to close other PMDEs.

Now consider the first moment PMDEs presented in [14]:

$$
\begin{aligned}
\mathrm{E}'\ [N_i(t),\ A(t)=\ell, N(t)\in\Omega_1] \ &= \ -\lambda_\ell(t)\,\mathrm{E}\left[N_i(t), A(t)=\ell, N(t)\in\Omega_1\right] \\
&-\mu_i(t)\,\mathrm{E}\left[N_i(t), A(t)=\ell, N(t)\in\Omega_1\right] \\
&-\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_j(t)\,\mathrm{E}\left[N_i(t), N(t)=s-1, A(t)=j\right] \\
&-\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_j(t)\,\beta_i(t)\,\mathrm{P}\left(N(t)=s-1, A(t)=j\right) \\
&-b_{i,m_B+1}(t)\,\mu_i(t)\,\mathrm{E}\left[N_i(t), N(t)=s, A(t)=\ell\right] \\
&+\sum_{j=1}^{m_A} a_{j\ell}(t)\,\lambda_j(t)\,\mathrm{E}\left[N_i(t), A(t)=j, N(t)\in\Omega_1\right] \\
&+\sum_{j=1}^{m_B} b_{ji}(t)\,\mu_j(t)\,\mathrm{E}\left[N_j(t), A(t)=\ell, N(t)\in\Omega_1\right] \\
&+\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_j(t)\,\mathrm{E}\left[N_i(t), A(t)=j, N(t)\in\Omega_1\right] \\
&+\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_j(t)\,\beta_i(t)\,\mathrm{P}\left(N(t)\in\Omega_1, A(t)=j\right) \\
&+\sum_{j=1}^{m_B} b_{j,m_B+1}(t)\,\mu_j(t)\,\mathrm{E}\left[N_i(t)\,N_j(t), N(t)=s, A(t)=\ell\right]
\end{aligned}
\tag{2.15}
$$

$$
\begin{aligned}
\mathrm{E}'\ [N(t),\ A(t)=\ell, N(t)\in\Omega_2] \ &= \ -\lambda_\ell(t)\,\mathrm{E}\left[N(t), A(t)=\ell, N(t)\in\Omega_2\right] \\
&-\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t)=c, A(t)=i\right) \\
&-\sum_{i=1}^{m_B} b_{i,m_B+1}(t)\,\mu_i(t)\,\mathrm{E}\left[N_i(t), N(t)\in\Omega_2, A(t)=\ell\right] \\
&-\sum_{i=1}^{m_B} b_{i,m_B+1}(t)\,\mu_i(t)\,(s-1)\,\mathrm{E}\left[N_i(t), N(t)=s, A(t)=\ell\right] \\
&+\sum_{i=1}^{m_A} a_{i\ell}(t)\,\lambda_i(t)\,\mathrm{E}\left[N(t), A(t)=i, N(t)\in\Omega_2\right] \\
&+\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,s\mathrm{P}\left(N(t)=s-1, A(t)=i\right) \\
&+\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,\mathrm{E}\left[N(t), A(t)=i, N(t)\in\Omega_2\right] \\
&+\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,\mathrm{P}\left(N(t)\in\Omega_2, A(t)=i\right)
\end{aligned}
\tag{2.16}
$$

Adding 2.15 and 2.16:

$$\begin{aligned}
\mathrm{E}'[N(t), A(t)=\ell] &= \sum_{i=1}^{m_B} \mathrm{E}'[N_i(t), A(t)=\ell, N(t)\in\Omega_1] \\
&\quad +\mathrm{E}[N(t), A(t)=\ell, N(t)\in\Omega_2] \\
&= -\lambda_\ell(t)\mathrm{E}[N(t), A(t)=\ell] \\
&\quad -\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\alpha_\ell(t)\lambda_j(t)\mathrm{P}(N(t)=c, A(t)=j) \\
&\quad -\sum_{j=1}^{m_B} b_{j,m_B+1}(t)\mu_j(t)\mathrm{E}[N_j(t), A(t)=\ell] \\
&\quad +\sum_{j=1}^{m_A} a_{j\ell}(t)\lambda_j(t)\mathrm{E}[N(t), A(t)=j] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\alpha_\ell(t)\lambda_j(t)\mathrm{E}[N(t), A(t)=j] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\alpha_\ell(t)\lambda_j(t)\mathrm{P}(A(t)=j)
\end{aligned} \tag{2.17}$$

The first moment differential equation (Equation 2.17) only requires the approximation of $\mathrm{P}(N(t)=c, A(t)=j)$ as well as $\sum_{j=1}^{m_B} b_{j,m_B+1}(t)\mu_j(t)\mathrm{E}[N_j(t), A(t)=\ell]$ to obtain *pseudo closure*. Obtaining the first moment requires less approximations then the FPMU and FPML which are used to approximate the higher moments in [14]. So we expect the first and second moment approximations to exhibit smaller absolute error than the first (FPML and FPMU) and second (SPMU and SPML) partial moments, respectively.

Even though the partial moment approximations require more approximations than the full moment, we show in the following sections that partitioning still provides better boundary-probabilities and as a consequence better moment approximations.

Conclusion

Other approximation procedures such as those provided by QNA are not designed for low traffic intensity and are designed only for steady-state, infinite-capacity systems having traffic intensity < 1 at every node. The CA discussed here is designed for all traffic intensities for systems where the inter-arrival and service-time distributions are approximated by a phase-type distribution or process. The run time of the CAs is fast and efficient. In Appendix 3 we consider different $Ph_t/Ph_t/s/c$ systems and in all the considered cases the execution time of the CAs is much faster than the numerical integration of the KFEs.

As discussed in Section 2.4, PMDEs for the partial moments (FPMU, FPML, SPML and SPMU) require more approximations than the first and second moments of the number in system. In the next section we show that good partial moment approximations are essential for good approximations of the overall number in system pmf. In this section and in Appendix 3 we show that the CA approximations do provide good approximations for both the partial moments as well as the overall moments.

We note that the CA also performs well when the input processes have high-frequency components. By high-frequency components we mean individual arrival (service) phase rates or time-dependent routing probabilities might be trigonometric functions of t having a high frequency. While the computation time for (any) differential equation increases as a function of the coefficient frequency, the accuracy does not necessarily suffer. Differential equations having trigonometric coefficients become increasingly stiff as frequency increases,

and thus computation time increases and may require stiff-differential equation solvers, such as `ODE15s` in MATLAB. We consider examples E1 and E3 in Section 2.4.3. These examples have high frequency components in the input and are accurately approximated.

2.4.4 The Performance of the Original CA for Approximating the Time-Dependent Boundary-State Probabilities

Recall that the CA makes use of boundary-probability approximations and that these approximations are obtained via moment matching using the first two moments for various conditional distributions. The CA approach partitions the state space into subspaces or partitions and then makes use of PMDEs for each of those partitions as well as conditioning on the system being in a particular arrival phase. Let this method of computing the boundary probabilities be M3.

Define M4 the same as M3; but M4 does not condition on being in a partition but conditions only on being in a particular arrival phase. M1 as defined earlier only conditions on being in a particular partition and M2 does not.

Consider the following example, Example 7 (E7), where the arrival process is represented with three phases ($m_A = 3$), so 9 boundary probabilities are approximated. We provide more examples in the Appendix, but we present a typical exanple, E7, to illustrate our algorithm. Figure 2.20 is a plot of the first and second moments. Figures 2.21 and 2.22 are plots of the 9 boundary probabilities.

Example 7 input:

$s = 3, c = 50, m_A = 3, m_B = 3$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4]$ $\qquad$ $\boldsymbol{\beta} = [0.3, \quad 0.3, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.3 & 0.2 & 0.4 \\ 0.3 & 0.1 & 0.2 & 0.4 \\ 0.2 & 0.2 & 0.3 & 0.3 \end{pmatrix}$$

$\boldsymbol{\lambda}(t) = [3 + 3\sin(t/0.9\pi), \quad 3 + 3\sin(t/0.8\pi), \quad 2 + 2\sin(t/0.5\pi)]$

$\boldsymbol{\mu}(t) = [0.5, \quad 0.5, \quad 1].$

Figure 2.20: First and Second Moments

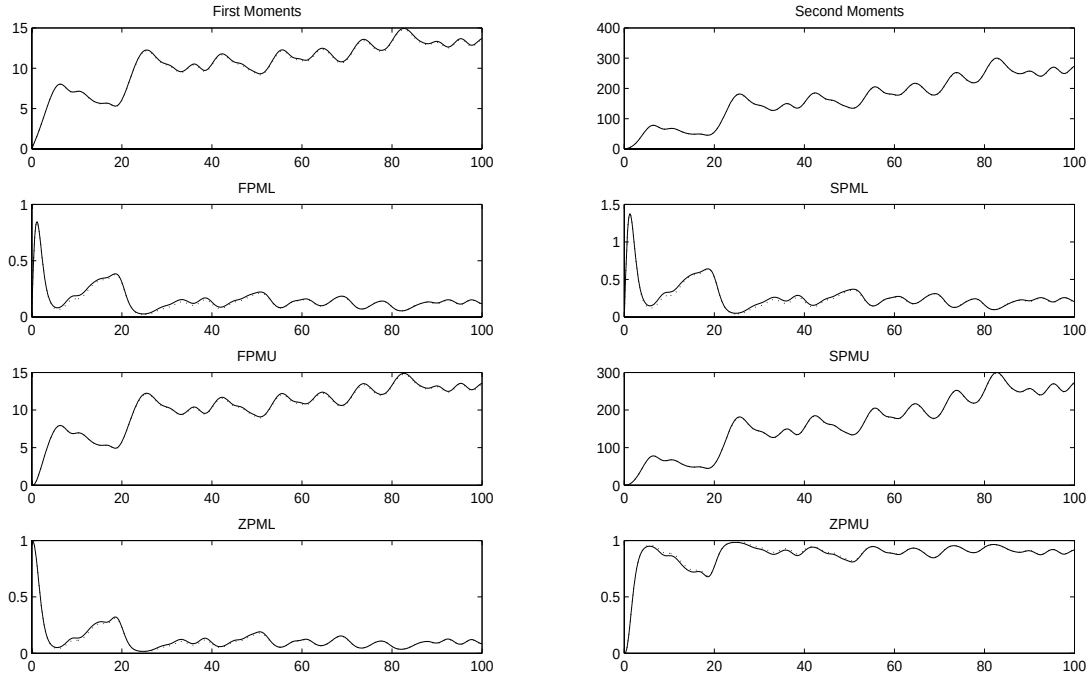


Figure 2.21: Method 3 Probabilities

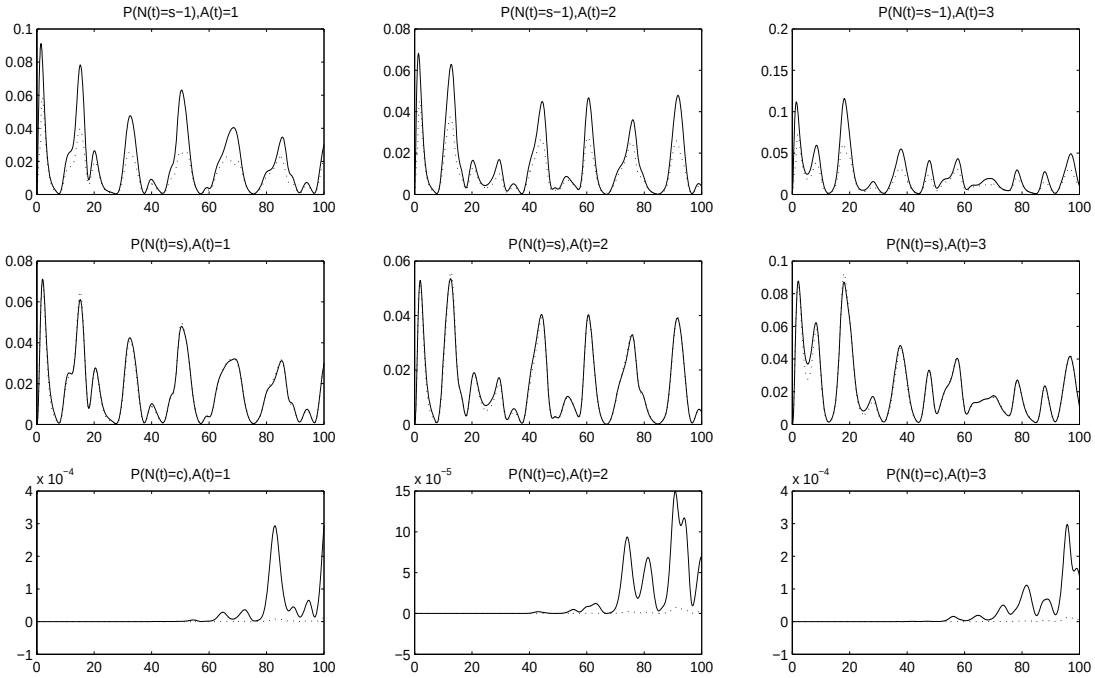
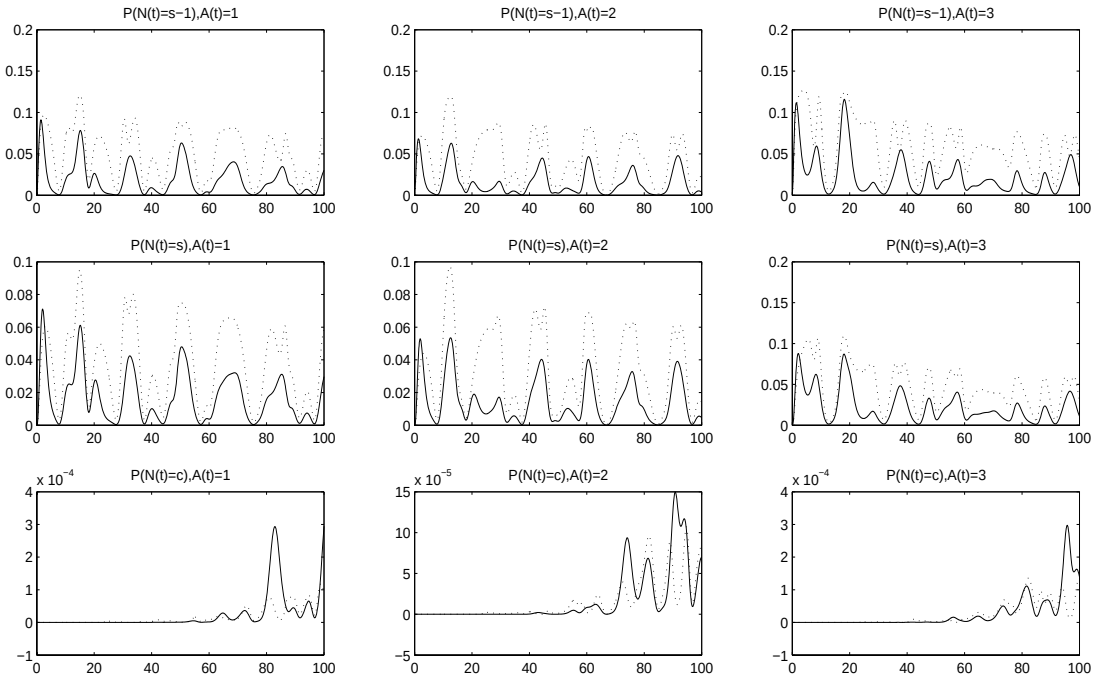


Figure 2.22: Method 4 Probabilities



For an $M/M/s/c$, M3 becomes M1 and M4 becomes M2. Since we previously showed that it is not a good idea to use M2 when computing the PMDEs at each iteration in an $M/M/s/c$ system, it is likewise not a good idea to use M4 in a $Ph/Ph/s/c$ system because M4 is M2 when applied to an $M/M/s/c$ system. The boundary probability approximations in Figure 2.21 are obviously more accurate than the approximations in Figure 2.22. Method M3 utilizes conditioning and is superior to M4. Overall M4 performs poorly.

2.4.5 The Augmented CA for Approximating the Time-Dependent System-State PMF

We now use M3 and M4 to approximate the entire system-state pmf at time t. We then evaluate the quality of the M3 and M4 approximations.

M3:

We defined M3 to condition on being in a particular subspace as well as a particular arrival phase. Let X_i be a PE random variable with first moment $\mathrm{E}[N(t)|A(t) = i, N(t) \in \Omega_1]$ set to $\mathrm{E}[Y_i]$, second moment $\mathrm{E}[N^2(t)|A(t) = i, N(t) \in \Omega_1]$ set to $\mathrm{E}[X_i^2]$ and range $\{0, \cdots, s-1\}$ for $i = 1 \ldots m_A$. Similarly let Y_i be a PE random variable with first moment $(\mathrm{E}[N(t)|A(t) = i, N(t) \in \Omega_2] - s)$ set to $\mathrm{E}[Y_i]$, second moment $(\mathrm{E}[N^2(t)|A(t) = i, N(t) \in \Omega_2] - 2s\mathrm{E}[N(t)|A(t) = i, N(t) \in \Omega_2 + s^2)$ set to $\mathrm{E}[Y_i^2]$ and range $\{0, \ldots, c-s\}$ for $i = 1 \ldots m_A$.

Approximate

$$\mathrm{P}(N(t)=j) \approx \sum_{i=1}^{m_A} \mathrm{P}(X_{i,1}=j)\mathrm{P}(A(t)=i, N(t)\in\Omega_1) + \sum_{i=1}^{m_A} \mathrm{P}(X_{i,2}=j)\mathrm{P}(A(t)=i, N(t)\in\Omega_2)$$

M4:

M4 does not condition on being in a particular subspace but conditions only on being in an arrival phase. Let Y_i be a PE random variable with first moment

$\mathrm{E}[N(t), A(t)=i, N(t)\in\Omega_1] + \mathrm{E}[N(t), A(t)=i, N(t)\in\Omega_2] = \mathrm{E}[N(t), A(t)=i]$ set to $\mathrm{E}[Y_i]$

and second moment

$\mathrm{E}[N^2(t), A(t)=i, N(t)\in\Omega_1] + \mathrm{E}[N^2(t), A(t)=i, N(t)\in\Omega_2] = \mathrm{E}[N^2(t), A(t)=i]$ set to $\mathrm{E}[Y_i^2]$ and range $\{0,\ldots,c\}$.

Approximate

$$\mathrm{P}(N(t)=j) \approx \sum_{i=1}^{m_A} \mathrm{P}(Y_i=j)\mathrm{P}(A(t)=i)$$

We need an approximation for the probability mass function at time t. Consider the four methods at time $t=35$. Method M4 consistently gives bad results, whereas the methods that condition on being in a partition (M1 and M3) give the same results. Recall that M1 only conditions on being in a subspace (Ω_1 or Ω_2) and M2 does not use any type of conditioning. Method M3 conditions on being in a subspace and arrival phase. Method M4 only conditions on being in an arrival phase. Figures 2.23, 2.24 and 2.25 contain plots of the number in system pmf at time 35 using Methods 1, 2, 3 and 4. The Figures also contain the pmf obtained by solving the KFEs as well as the absolute error. The sum of the absolute errors is given in the Figure titles.

Figure 2.23: Methods 1 and 3, total abs error=0.1264

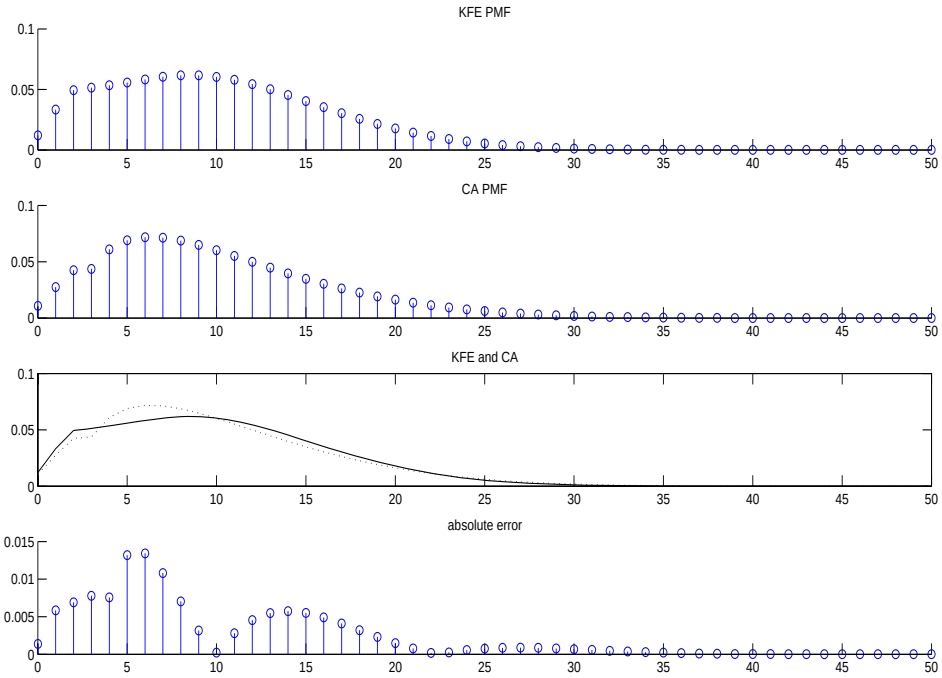


Figure 2.24: Method 2 , total abs error= 0.0848

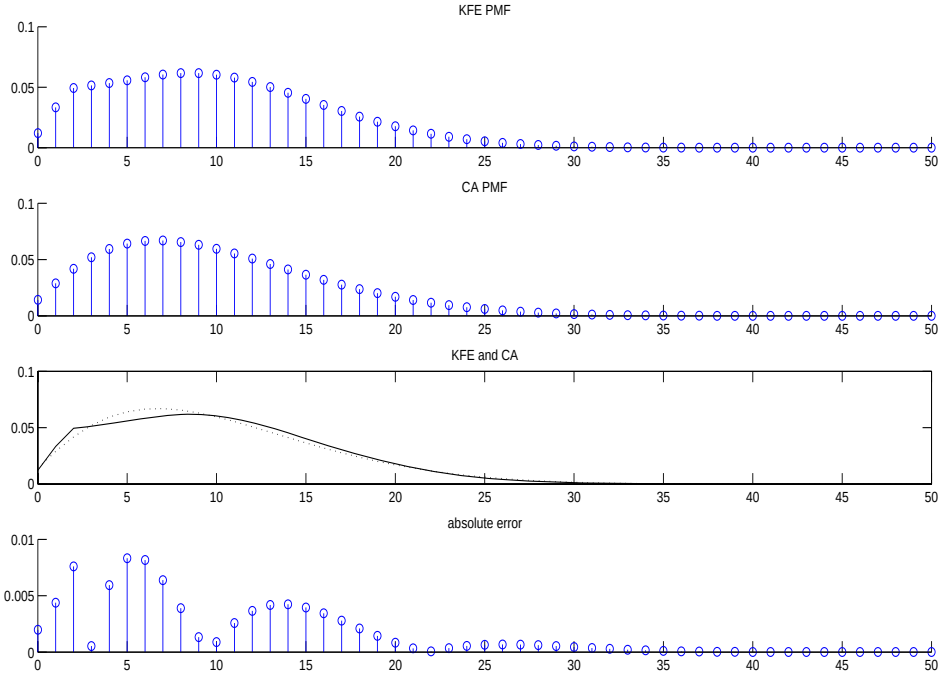
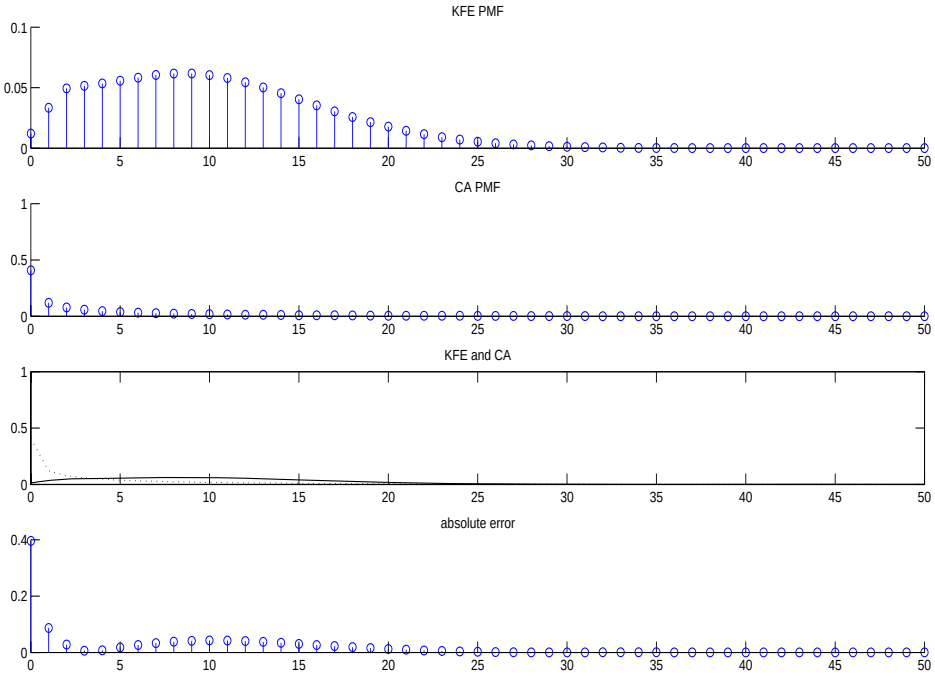


Figure 2.25: Method4 total abs error>1



In order to compare the performance of the methods over time we choose the following ten points $(10, 20, 30, \ldots, 100)$ and obtain the total absolute error for each method. We take the averages of the absolute error and present the results in Table 2.2. Table 2.2 also contains the average percent error of the first and second moments of the time-dependent number in system over the ten points in time as well as the run-time, in minutes, required to solve the CAs and the KFEs. Similar tables are constructed in Appendix 3 for different $Ph_t/Ph_t/s/c$ systems.

Referring to the E7 results displayed in Table 2.2, approximation method M2 produces more accurate approximations for the moments of the time-dependent number in system, as can

Table 2.2: Method Comparison Table

M1	M2	M3	M4	% FM	% SM	KFE time	CA time
0.0915	0.0800	0.0915	0.9330	1.0412	0.4248	1.8969	0.0247

be seen from Figure 2.24 and Table 2.2, but does not produce better boundary-probability approximations, as can be seen in Figures 2.21 and 2.20. While M2 does not perform better than M1 and M3 in general, it does for this example. Methods M3 and M1 perform much better than M2 in all cases considered in Appendix 3. Method M4 consistently gives the least accurate results.

We highlight this example E7 to show that although M3 gives better boundary probabilities and in general better pmf approximations, M2 can perform better in some cases. Since M2 does not condition on being in a partition, the jump in the pmf from $\mathrm{P}(N(t) = s - 1)$ to $\mathrm{P}(N(t) = s)$ is smoother, as can be seen in Figure 2.24 which can results in a smaller overall error. We recommend using M1 or M3 to approximate the entire pmf and especially the boundary probabilities, but in some cases conditioning on being in a partition may not always give the best pmf approximations.

2.5 Conclusion

When approximating characteristics of the number-in-system for a $GI/G/s/c$ or $GI/G/s/\infty$ system, most of the previous research approximates the steady-state expected number and coefficient of variation of the number in system. The input parameters for these models being approximated are the expected number and coefficients of variation for the inter-arrival-time distribution and the service-time distribution. The input inter-arrival-time distribution and the service-time distribution in our models are matched to a Ph_t distributions which allows for convenient matching of any number of moments. So the inter-arrival-time and service-time distribution approximations are not restricted to the first two moments. Furthermore the CA algorithm can be used for transient or time-dependent analysis and is not restricted to steady-state analysis.

We tested the accuracy of the CA approximation presented in [14] for the time-dependent number-in-system first two moments of the $Ph_t/Ph_t/s/c$ system for different inter-arrival-time and the service-time distributions. Furthermore we tested the accuracy of the time-dependent boundary probabilities approximations as well as the entire pmf of the number-in-system at any point in time.

We introduced some modifications to the CA algorithm and concluded that the intuition behind the initial CA algorithm presented in [14] is the superior approach when approximating number-in-system moments and is usually the best approach when approximating the entire pmf of the number in system at any point in time t.

Chapter 3

$Ph_t/M_t/s/c$ Time Dependent Departure Process and Queues in Tandem

3.1 Introduction and Background

The queueing systems we consider are Markovian or can be approximated arbitrarily closely by a Markovian process and thus can be represented by Kolmogorov Forward Equations (KFEs). In this chapter we develop differential equations to examine departure processes from Markovian queues. We then develop approximations for these departure processes. The approximations that we develop are very similar in structure to the closure approximations

(CA) that we developed in Chapter 2 for the time-dependent moments of the number-in-system.

Also in this chapter we present algorithms to obtain approximations for the moments of the number of departures over any time interval. We then construct approximate departure processes based on the approximated moments of the number of departures from a node in a specified time interval. Next we develop methods to superpose and split approximate departure processes. We then can use these approximated (and possibly superposed and split) departure processes as composite arrival processes at downstream nodes in a queueing network. We specifically look at tandem networks in this chapter because the effects of using approximated arrival processes should be most evident in small tandem networks.

The systems considered in this chapter are tandem queueing networks having time-dependent phase-type external arrival processes, Markovian (M_t) service times, multiple servers, and finite (infinite) capacity. We identify key characteristics of the actual $Ph_t/M_t/s/c$ departure processes and present an algorithm to match these key characteristics with a $\widetilde{Ph_t}$ process serving as an approximate departure. We then use the $\widetilde{Ph_t}$ approximating departure process (or that process split or superposed with other approximated departure processes and perhaps superposed with external Ph_t arrival processes) as the approximate composite arrival process to downstream node(s) in the network.

3.2 Departure Processes, Departure Process Approximations, and Closure Approximations for Number-in-System Moments

Consider, for example, the simplest case of the $M_t/M_t/\infty$ system. The first moment differential equation (MDE) for the time-dependent number-in-system is closed (see [12]) and has this simple form:

$$\frac{d}{dt}\mathrm{E}\left[N(t)\right] = \lambda(t) - \mu(t)\mathrm{E}\left[N(t)\right] \tag{3.1}$$

where $N(t)$ is the number in system at time at time t, $\lambda(t)$ and $\mu(t)$ are the arrival and service rates. For this special system we also know that the time-dependent distribution of the number of entities in the system is a time-dependent Poisson distribution. The departure process from the $M_t/M_t/\infty$ is also a time-dependent Poisson process. Thus we can evaluate the time-dependent number-in-the system as well as the departure process across time, by simply specifying an initial condition, and numerically integrating just one differential equation, 3.1.

The more direct but less efficient method for evaluating time-dependent behavior of this system requires solving for the entire distribution of $N(t)$ across time by evaluating the full set of KFEs. Evaluating the KFEs is not computationally practical because doing so would require the numerical integration of an infinite set of KFEs.

This infinite-server simple system is a very special case. Most often we do not know the

functional form of the time-dependent number-in-system distribution and thus we do not know the functional form of the departure process.

Differential equations describing departure processes and MDEs are related. More complex queueing models do not have closed MDEs and thus require some type of approximation to avoid evaluation of the full set of KFEs, which we would need to do to evaluate either the time-dependent distribution of the number-in-system or to evaluate the time-dependent inter-departure-time distributions. In this chapter we will develop CAs to approximate departure processes that are based on the CAs for the MDEs for the number-in-the system moments discussed in Chapter 2.

Consider the number-in-system PMDEs for the time-dependent $M_t/M_t/s/c$. We construct a CA for the departure process for the $M_t/M_t/s/c$ based on the PMDEs for an extended related state space. We extend the state space to include a count random variable for the number of departures in an interval.

For the $M_t/M_t/s/c$ system the number of KFEs is $c+1$ and the number of PMDEs required by the CA discussed in Chapter 2 to solve for the first two moments is only five. The PMDEs for this system are Equations (2.9 to 2.5).

Closure approximations for Equations (2.9 to 2.5) are provided in [14], and examined in detail in Chapter 2. The performance of related CAs for the more general $Ph_t/Ph_t/s/c$ system is examined in [14] and [9].

The $M_t/M_t/s/c$ Departure Process:

The departure random process from a $M_t/M_t/s/c$ system can be represented by a Markov process. We extend the state space description and define an auxiliary random variable by including the number of entities that depart the system in the interval $[t, t+\tau)$ in the description of this stochastic process. This enables us to represent the departure process by a Markovian process.

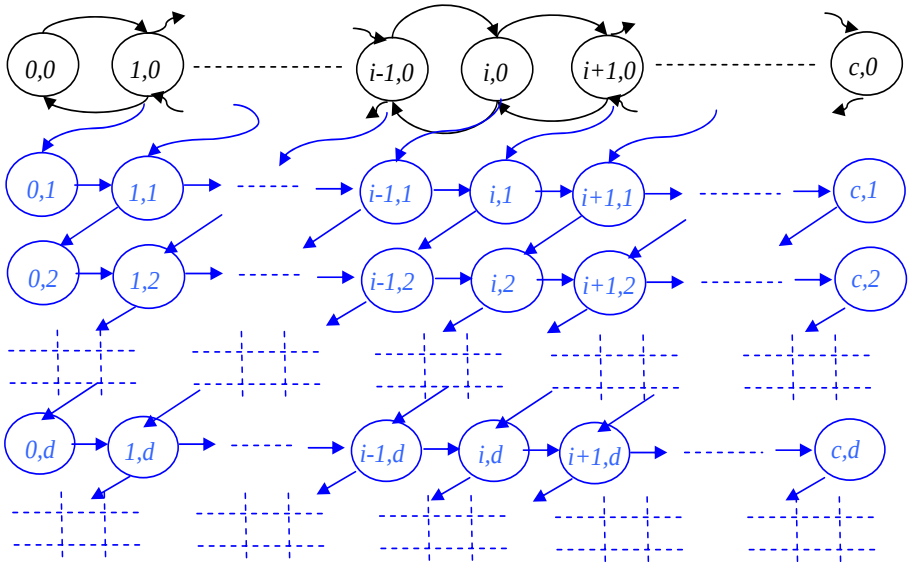
For $M_t/M_t/s/c$ system let the number in system be $\{N(t) = t \ : \ t \geq 0\}$ where $N(t) = 0, 1, 2, \ldots, c$, and let the number of departures in the interval $[t, t+\tau)$ be $\{D_t(t+\tau) \ : \ t \geq 0, \ \tau > 0\}$ where $D_t(t+\tau) = 0, 1, 2, \ldots$. Define the state probability $P_{i,d,t}(\tau) \equiv \mathrm{P}(N(t+\tau) = i, D_t(t+\tau) = d)$. Thus $P_{i,d,t}(\tau)$ is the probability that the number in system at time $(t+\tau)$ is i and the number of departures in the interval $[t, t+\tau)$ is d. States are represented by the number in system at time t+τ and the number departed in the interval $[t, t+\tau)$. This extended state-space system remains a Markovian process. Figure 3.1 is a representation of this extended state-space system. In Figure 3.1 (i, j) represents a state where the number in system at time $(t+\tau)$ is i and the number of departures in the interval $[t, t+\tau)$ is j.

The differential-difference equations for this extended state-space system are equations 3.2 to 3.8. We abbreviate $P_{i,d,t}(\tau)$ with $P_{i,d}$, and $P'_{i,d}$ is the corresponding derivative with respect to τ. The KFEs are:

$$P'_{0,0} = -\lambda P_{0,0} \quad \text{for} \quad i = 0 \quad \text{and} \quad d = 0 \tag{3.2}$$

$$P'_{i,0} = -\lambda P_{i,0} - \min(i, s)\mu P_{i,0} + \lambda P_{i-1,0}, \quad \text{for} \quad 0 < i < c \quad \text{and} \quad d = 0 \tag{3.3}$$

Figure 3.1: Number in System and Departures States $M_t/M_t/s/c$



$$P'_{c,0} = -s\mu P_{c,0} + \lambda P_{c-1,0}, \quad \text{for} \quad i = c \quad \text{and} \quad d = 0 \tag{3.4}$$

$$P'_{0,d} = -\lambda P_{0,d} + \mu P_{1,d-1}, \quad \text{for} \quad i = 0 \quad \text{and} \quad d > 0 \tag{3.5}$$

$$\begin{aligned} P'_{i,d} = {} & -\lambda P_{i,d} - \min(i,s)\mu P_{i,d} + \lambda P_{i-1,d} + \min(i+1,s)\mu P_{i,d-1}, \\ & \text{for} \quad 0 < i < c \quad \text{and} \quad d > 0 \end{aligned} \tag{3.6}$$

$$P'_{c,d} = -s\mu P_{c,d} + \lambda P_{c-1,d}, \quad \text{for} \quad i = c \quad \text{and} \quad d > 0 \tag{3.7}$$

$$P'_{i,d_m} = \lambda P_{i-1,d_m}, \quad \text{for} \quad d = d_m. \tag{3.8}$$

Theoretically, the number of states is infinite since within a time interval there is no upper limit on the number of departures; thus d can be infinite and so the number of KFEs is infinite. We define a "practical upper limit" on the number of departures, d_m, which should be large enough with respect to the time interval for accurate approximations. Thus, by

truncating the state space the number of KFEs is now $(c+1)d_m$ and is a function of the system capacity as well as d_m.

Example: We solve the departure KFEs (Equations 3.2 to 3.8) for the stationary $M/M/s/c$ system with parameters $s = 3, c = 20, \lambda = 3, \mu = 2$ and we truncate the state space by setting $d_m = 50$. We choose three time intervals $[10, 20)$, $[0, 4)$ and $[90, 93)$. In this example, $c = 20$ is large enough to have no affect on the system performance measures; i.e., the first and second moments of the number of departures over the intervals $[10, 20)$ and $[90, 93)$ are essentially the same as the first and second moments of the departure process from an equivalent $M/M/s/\infty$ where the state space is not truncated. The first and second moments of the number of departures over the intervals are:

$$t = 10, t+\tau = 20 : E[D_t(t+\tau)] = 29.9989, E[D_t^2(t+\tau)] = 929.9048$$

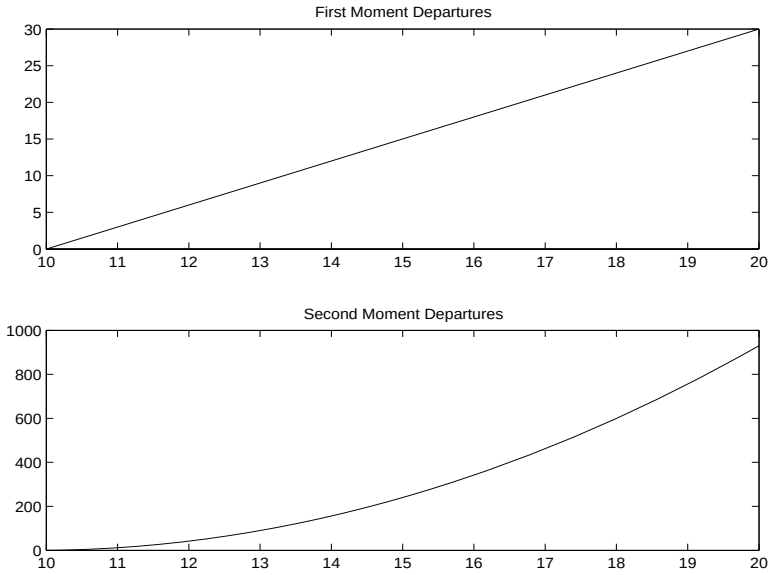
$$t = 0, t+\tau = 4 : E[D_t(t+\tau)] = 10.2872, E[D_t^2(t+\tau)] = 115.5527$$

$$t = 90, t+\tau = 93 : E[D_t(t+\tau)] = 9, E[D_t^2(t+\tau)] = 90$$

The choice of d_m could result in less accurate approximations if it is set too small. There exists positive integers d^* and n, where increasing d_m beyond d^* no longer improves the quality of the approximations up to the n^{th} decimal point. So d_m can be the largest positive integer that satisfies $\mathrm{P}(D_t(t+\tau) \geq d_m) < \epsilon$ for an "appropriate" choice of ϵ.

To compute the first two moments of the number of departures over the interval $[t, t+\tau)$, we can solve a set of partial-moment differential equations where the number of equations does not dependent on d_m or ϵ. This set of differential equations is closed, hence we can obtain

Figure 3.2: Departures ($t = 10, t + \tau = 20$)



numerically exact solutions. We refer to them as the departure-count numerically exact partial-moment differential equations (NEPMDEs). We obtain the NEPMDEs by summing the derivatives of the state probabilities. The derivative is with respect to τ.

$$\mathrm{E}'[D_t(t+\tau), N(t+\tau) = i] \equiv \sum_{d=1}^{\infty} dP'_{i,d} \tag{3.9}$$

$$\mathrm{E}'[D_t^2(t+\tau), N(t+\tau) = i] \equiv \sum_{d=1}^{\infty} d^2 P'_{i,d} \tag{3.10}$$

The NEPMDEs are:

$$P'_0(t+\tau) = \mu P_1(t+\tau) - \lambda P_0(t+\tau) \quad \text{where} \quad P_i(t+\tau) \equiv \mathrm{P(N(t+\tau) = i)} \tag{3.11}$$

$$\begin{aligned} \mathrm{E}'[D_t(t+\tau), N(t+\tau) = 0] &= -\lambda \mathrm{E}[D_t(t+\tau), N(t+\tau) = 0] \\ &\quad +\mu \mathrm{E}[D_t(t+\tau), N(t+\tau) = 1] + \mu P_1(t+\tau) \end{aligned} \tag{3.12}$$

$$\begin{aligned}\mathrm{E}'[D_t^2(t+\tau), N(t+\tau)=0] &= -\lambda \mathrm{E}[D_t^2(t+\tau), N(t+\tau)=0] \\ &\quad +\mu \mathrm{E}[D_t^2(t+\tau), N(t+\tau)=1] + \mu P_1(t+\tau) \\ &\quad +2\mu \mathrm{E}[D_t(t+\tau), N(t+\tau)=1]\end{aligned} \tag{3.13}$$

$$\begin{aligned}P_i'(t+\tau) &= -(\lambda+i\mu)P_i(t+\tau) + (i+1)\mu P_{i+1}(t+\tau) \\ &\quad +\lambda P_{i-1}(t+\tau) \quad \text{for } 0<i<s.\end{aligned} \tag{3.14}$$

$$\begin{aligned}\mathrm{E}'[D_t(t+\tau), N(t+\tau)=i] &= -(\lambda+i\mu)\mathrm{E}[D_t(t+\tau), N(t+\tau)=i] \\ &\quad +\lambda \mathrm{E}[D_t(t+\tau), N(t+\tau)=i-1] \\ &\quad +(i+1)\mu \mathrm{E}[D_t(t+\tau), N(t+\tau)=i+1] \\ &\quad +(i+1)\mu P_{i+1}(t+\tau) \quad \text{for } 0<i<s.\end{aligned} \tag{3.15}$$

$$\begin{aligned}\mathrm{E}'[D_t^2(t+\tau), N(t+\tau)=i] &= -(\lambda+i\mu)\mathrm{E}[D_t^2(t+\tau), N(t+\tau)=i] \\ &\quad +\lambda \mathrm{E}[D_t^2(t+\tau), N(t+\tau)=i-1] \\ &\quad +(i+1)\mu \mathrm{E}[D_t^2(t+\tau), N(t+\tau)=i+1] \\ &\quad +2(i+1)\mu \mathrm{E}[D_t(t+\tau), N(t+\tau)=i+1] \\ &\quad +(i+1)\mu P_{i+1}(t+\tau), \quad \text{for } 0<i<s.\end{aligned} \tag{3.16}$$

$$\begin{aligned}P_i'(t+\tau) &= -(\lambda+s\mu)P_i(t+\tau) + s\mu P_{i+1}(t+\tau) + \\ &\quad \lambda P_{i-1}(t+\tau) \quad \text{for } s \le i < c.\end{aligned} \tag{3.17}$$

$$\begin{aligned}\mathrm{E}'[D_t(t+\tau), N(t+\tau)=i] &= -(\lambda+s\mu)\mathrm{E}[D_t(t+\tau), N(t+\tau)=i] \\ &\quad +\lambda \mathrm{E}[D_t(t+\tau), N(t+\tau)=i-1] \\ &\quad +s\mu \mathrm{E}[D_t(t+\tau), N(t+\tau)=i+1] \\ &\quad +s\mu P_{i+1}(t+\tau) \quad \text{for } s \le i < c.\end{aligned} \tag{3.18}$$

$$\begin{aligned}
\mathrm{E}'[D_t^2(t+\tau), N(t+\tau)=i] &= -(\lambda+s\mu)\mathrm{E}[D_t^2(t+\tau), N(t+\tau)=i] \\
&\quad +s\mu\mathrm{E}[D_t^2(t+\tau), N(t+\tau)=i+1] \\
&\quad +\lambda\mathrm{E}[D_t^2(t+\tau), N(t+\tau)=i-1] \\
&\quad +2s\mu\mathrm{E}[D_t(t+\tau), N(t+\tau)=i+1] \\
&\quad +s\mu P_{i+1}(t+\tau), \quad \text{for } s \le i < c.
\end{aligned} \tag{3.19}$$

and

$$P_c'(t+\tau) = \lambda P_{c-1}(t+\tau) - s\mu P_i(t+\tau) \quad \text{for } i=c. \tag{3.20}$$

$$\begin{aligned}
\mathrm{E}'[D_t(t+\tau), N(t+\tau)=c] &= -s\mu\mathrm{E}[D_t(t+\tau), N(t+\tau)=c] \\
&\quad +\lambda\mathrm{E}[D_t(t+\tau), N(t+\tau)=c-1] \quad \text{for } i=c.
\end{aligned} \tag{3.21}$$

$$\begin{aligned}
\mathrm{E}'[D_t^2(t+\tau), N(t+\tau)=c] &= -s\mu\mathrm{E}[D_t^2(t+\tau), N(t+\tau)=c] \\
&\quad +\lambda\mathrm{E}[D_t^2(t+\tau), N(t+\tau)=c-1], \quad \text{for } i=c.
\end{aligned} \tag{3.22}$$

Differential equations 3.11 to 3.22 are closed. While it is an advantage that the number of departure-count NEPMDEs is not a function of the number of departures, it is inconvenient that the number of departure-count NEPMDEs is a function of the system capacity, c. To compute the first two moments of the departure count random variable we need to numerically evaluate $3\,(c+1)$ departure-count NEPMDEs. We next develop a CA approximation that will result in a number of departure-count PMDEs that is *not* a function of the capacity of the system, c.

It is possible to further reduce the number of differential equations to a set that is smaller than the set of NEPMDEs by summing the NEPMDEs over all values of $N\,(t+\tau) \in \Omega_1$ and

summing them over all values of $N(t+\tau) \in \Omega_2$. This reduced set is:

$$\mathrm{E}'[D_t(t+\tau)] = \sum_{j=0}^{\infty} j \left(\sum_{i=0}^{c} P'_{i,j} \right)$$

and

$$\mathrm{E}'[D_t^2(t+\tau)] = \sum_{j=0}^{\infty} j^2 \left(\sum_{i=0}^{c} P'_{i,j} \right)$$

where

$$P_{i,j} = \mathrm{P}(N(t+\tau) = i, D_t(t+\tau) = j).$$

Thus, moment differential equation for the first moment of the departure count is:

$$\begin{aligned} \mathrm{E}'[D_t(t+\tau)] &= \sum_{d=0}^{\infty} d \left(\sum_{i=0}^{c} P'_{i,d} \right) \\ &= \mu \mathrm{E}[N(t+\tau), N(t+\tau) \in \Omega_1] + s\mu \mathrm{P}(N(t+\tau) \in \Omega_2) \end{aligned}$$

where Ω_2 is $\{s, s+1, ..., c\}$ and Ω_1 is $\{0, 1, 2, ..., s-1\}$.

If only the first moment of the departure counts are needed then one might consider combining expressions 3.23 and 3.24 then close the single differential equation via a one-moment distribution serving as the approximating distribution. We have found that one-moment CAs are not accurate and that two moment CAs are far more accurate. Thus we develop the second-moment departure-count PMDEs. We use the same state-space partitioning as used to develop the PMDEs for the CA used for the number-in-system moments.

The first- and second-moment departure-count PMDEs are:

$$\begin{aligned}
\mathrm{E}'[D_t(t+\tau), N(t+\tau) \in \Omega_1] &= -\lambda \mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1] \\
&\quad +\mu \mathrm{E}[N(t+\tau), N(t+\tau) \in \Omega_1] \\
&\quad +\mu s \mathrm{E}[D_t(t+\tau), N(t+\tau) = s] \\
&\quad +\mu s \mathrm{P}(N(t+\tau) = s)
\end{aligned} \tag{3.23}$$

$$\begin{aligned}
\mathrm{E}'[D_t(t+\tau), N(t+\tau) \in \Omega_2] &= -\mu s P_s - \mu s \mathrm{E}[D_t(t+\tau), N(t+\tau) = s] \\
&\quad +s\mu \mathrm{P}(N(t+\tau) \in \Omega_2) \\
&\quad +\lambda \mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1]
\end{aligned} \tag{3.24}$$

$$\begin{aligned}
\mathrm{E}'[D_t^2(t+\tau)] &= 2\mu \sum_{i=1}^{s-1} i \mathrm{E}[D_t(t+\tau), N(t+\tau) = i] \\
&\quad +2\mu s \mathrm{E}[D_t(t+\tau), N(t+\tau) \in \Omega_2] \\
&\quad +\mu \mathrm{E}[N(t+\tau), N(t+\tau) \in \Omega_1] \\
&\quad +s\mu \mathrm{P}(N(t+\tau) \in \Omega_2)
\end{aligned} \tag{3.25}$$

$$\begin{aligned}
\sum_{i=1}^{s-1} i \mathrm{E}'[D_t(t+\tau), N(t+\tau) = i] &= -\mu \sum_{i=1}^{s-1} i \mathrm{E}[D_t(t+\tau), N(t+\tau) = i] \\
&\quad -\mu \mathrm{E}[N(t+\tau), \Omega_1] \\
&\quad -\lambda s \mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1] \\
&\quad +\mu s(s-1) \mathrm{P}(N(t+\tau) = s) + \mu \mathrm{E}[N^2(t), \Omega_1] \\
&\quad +\lambda \mathrm{E}[D_t(t+\tau), \Omega_1] \\
&\quad +\mu s(s-1) \mathrm{E}[D_t(t+\tau), N(t+\tau) = s]
\end{aligned} \tag{3.26}$$

The set of four departure-count PMDEs required to compute the first two moments of $D_t(t+\tau)$ together with the set of five number-in-system PMDEs, 2.5 to 2.9, result in a set of differential equations that is not closed. The terms appearing on the RHS of the departure-count PMDEs that need to be approximated are:

$$\mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1],$$

$$\mathrm{E}[D_t(t+\tau), N(t+\tau) = s]$$

and

$$\mathrm{P}(N(t+\tau) = s).$$

The term $\mathrm{P}(N(t+\tau) = s)$ also appears on the RHS of the number-in-system PMDEs. We pseudo closed the set of number-in-system PMDEs via a CA using Polya Eggenberger distributions. Next we pseudo close the departure-count PMDEs via a similar CA that also makes use of Polya Eggenberger distributions.

In this chapter we develop the departure-count NEPMDEs and the departure-count PMDEs for the more general $Ph_t/M_t/s/c$ system. We then develop CAs for the $Ph_t/M_t/s/c$ departure-count PMDEs.

3.3 The $Ph_t/M_t/s/c$ Departure Process

3.3.1 Counts versus Inter-Departures

In this section we discuss the choice of using departure-count random variables instead of inter-departure-time intervals as a basis for constructing approximating departure processes.

Consider the departure process from a single $Ph_t/M_t/s/c$ node. We would like to extract some key characteristics of the departure process from the $Ph_t/M_t/s/c$ node and then use those characteristics to construct a point process to serve as an accurate approximation for the time-dependent departure process. We necessarily cannot extract a sufficient number of characteristics to fully describe the departure process and thus any point process constructed from the partial set of characteristics of the actual departure process is necessarily an approximation.

Consider two approaches to characterizing the departure process. The first approach is to identify a set of characteristics for the inter-departure-time intervals (or perhaps the intervals of time between time t and the time of the k^{th} departure after time t). The second approach is to identify a set of characteristics for the count of the number of departures within a time interval.

Let $D_t(t+\tau) \equiv$ number of departures in the interval $[t, t+\tau)$, $t \geq 0$, $\tau > 0$. For any time $t \geq 0$ and $\tau > 0$ the first two moments of $D_t(t+\tau)$ provide information about the departure process, but, of course, not full information about the $\{D_t(t+\tau) : t \geq 0, \tau > 0\}$ process.

An example of the first approach for approximating the departure process might be to obtain the first two (or more) moments of the time-until-the-n^{th} departure starting from time t, and then fitting those moments to a Ph_t process that would then serve as the approximating departure process. If the time-untill-the-n^{th} departure distribution were known, then we could compute characteristics of the time between successive departures, such as moments. Johnson and Taaffe [5], and Asmussen [2], prove the denseness of the family of Ph distributions, and [6] and [7] illustrate the use of the Ph distribution for approximating inter-departure (arrival) processes based on moments. They also provide inter-departure moment-matching algorithms for the Ph distribution. Johnson and Taaffe [6] show that with the Ph distribution it is possible to match the first k moments of any distribution with a finite-phase Ph distribution. They go on to show that the first k moments can be represented by a finite mixture of Erlangs, and provide moment-matching algorithms for the first k moments.

The second approach considers a time interval $[t, t+\tau)$ for $t \geq 0$ and $\tau > 0$ and obtains the moments of the number of departures over the interval and then constructs an approximating departure process using a Ph distribution. We choose the second approach since we believe that it is computationally more friendly. We do not describe the computational details of the first approach here, but they require many more sub-interval computations for each considered point in time.

3.4 The $Ph_t/M_t/s$ Departure-Count Process

Building on the CA algorithm and the moments of the time-dependent number-in-system PMDEs, presented in [14], we obtain approximations for the moments of the departure-count process over any chosen time interval. For example, consider the first moment of the departure-count in the more general $Ph_t/Ph_t/s$ system in $[t, t+\tau)$.

$$D_t(t+\tau) = A_t(t+\tau) + N(t^-) - N((t+\tau)^-) \tag{3.27}$$

where $A_t(t+\tau)$ is the number of arrivals in the interval $[t, t+\tau)$, and $N(t)$ is the number in system at time t. We see that

$$\mathrm{E}[D_t(t+\tau)] = \mathrm{E}[A_t(t+\tau)] + \mathrm{E}[N(t)] - \mathrm{E}[N(t+\tau)]. \tag{3.28}$$

We can approximate $\mathrm{E}[N(t)]$ and $\mathrm{E}[N(t+\tau)]$ by applying the CA algorithm presented in [14] for the time-dependent moments of the number-in-system. Approximating $\mathrm{E}[A_t(t+\tau)]$ requires an approximation of the expected number of arrivals at time t, $\mathrm{E}[A_0(t)]$, and the expected number of arrivals by time $t+\tau$, $\mathrm{E}[A_0(t+\tau)]$. So $\mathrm{E}[A_t(t+\tau)] = \mathrm{E}[A_0(t+\tau)] - \mathrm{E}[A_0(t)]$. We can present two simple and accurate methods to obtain the expected number of arrivals in the interval $(t, t+\tau)$.

1) Consider the Ph_t arrival process separately from the rest of the $Ph_t/Ph_t/s/\infty$ system and analyze the arrival process independently. The states of the arrival process are the m_A phases of the Ph_t arrival process. Evaluating the corresponding m_A KFEs gives the probabilities of the arrival process being in each of the arrival phases across time. Let $\pi_i(t)$

be the probability that the arrival process is in state i at time t and let $\lambda_i(t)$ be the transition rate from phase i at time t. Let $A_t(t+\tau)$ for $t \geq 0, \tau > 0$ be the number of arrivals in the interval $[t, t+\tau)$. So we have $\mathrm{E}[A_t(t+\tau)] = \int_t^{(t+\tau)^-} \pi(u)\lambda(u)a_{i,m_A+1}(u)du$. In Section 3.5 we introduce differential equations to obtain the entire distribution of $A_t(t+\tau)$ as well as the n^{th} moment.

2) Another simple approach to evaluate $\mathrm{E}[A_t(t+\tau)]$ is to set the service rate to 0 in the $Ph_t/M/1$ system, thus not allowing entities to leave the queueing system. The number of arrivals over the interval $[t, t+\tau)$ becomes $N(t+\tau)-N(t)$. For example, evaluate the PMDEs using the CAs presented in [14] for the $Ph_t/M/1$ system where the service distribution has a rate 0. Thus $\mathrm{E}[A_t(t+\tau)] = \mathrm{E}[N(t+\tau)] - \mathrm{E}[N(t)]$. The two methods are numerically exact for the $Ph_t/Ph_t/s$ system and are good approximations for the $Ph_t/Ph_t/s/c$ system when c is large enough such that $\mathrm{P}(N(t) = c) < \epsilon$ for every $t \geq 0$ with an "appropriate" choice of ϵ. The arrival process can not be considered independently from the rest of the system if c is not sufficiently large.

Finding the first (and second) moments of $D_t(t+\tau)$ for the $Ph_t/Ph_t/s/c$ system is more complex. We approach the problem by dividing the time line into intervals and for each time interval we find the first and second moments of the number of departures. We make use of the time intervals to approximate the first moments of the number of departures shorter than the time intervals used to approximate the second moments of the number of departures. We do this for two reasons: 1) It is computationally faster and easier to compute the first moment, and 2) except for when the inter-departure intervals are exponential the

number of departures over a sequence of intervals is autocorrelated. Obtaining the second moment over small time intervals is not recommended since the number of departures over consecutive small intervals is highly correlated. In fact it is necessary to capture the affects of this correlation in order to approximate the departure-count process well.

If it were sufficient to approximate the nodal departure processes with Poisson processes, the task would be simpler. Since it is reasonable to assume that using Poisson processes is not sufficient, in general, then computing the second moment of the number of departures in a time interval is required and the computation involves a more complex procedure; thus we derive the second moment for larger time intervals. Having the ability to find the second moment of the number of departures for any interval also enables us to find the correlation between the number of departures between intervals.

Let $D_1, D_2, \cdots, D_n$ be the number of departures in n consecutive non-overlapping intervals. Assume it is possible to find the first two moments of the departure-count in any time interval. We have $\mathrm{E}[D_i + D_{i+1} + \cdots + D_j]$ and $\mathrm{E}[(D_i + D_{i+1} + \cdots + D_j)^2]$ where $= 1, 2, \ldots, n-1$ and $h = 1, 2, ..., n - i$

$$\mathrm{Cov}[D_i, D_{i+1}] = \mathrm{E}[D_i D_{i+1}] - \mathrm{E}[D_i]\mathrm{E}[D_{i+1}], \quad \text{where}$$

$$\mathrm{E}[D_i D_{i+1}] = \frac{1}{2}(\mathrm{E}[(D_i + D_{i+1})^2] - \mathrm{E}[D_i^2] - \mathrm{E}[D_{i+1}^2])$$

$$\mathrm{Cov}[D_i, D_{i+2}] = \mathrm{E}[D_i D_{i+2}] - \mathrm{E}[D_i]\mathrm{E}[D_{i+2}], \quad \text{where}$$

$$\mathrm{E}[D_i D_{i+2}] = \frac{1}{2}(\mathrm{E}[(D_i + D_{i+1} + D_{i+2})^2] - \mathrm{E}[(D_i + D_{i+1})^2] - \mathrm{E}[D_{i+2}^2]) - \mathrm{E}[D_{i+1} D_{i+2}]$$

$$\vdots$$

$$\mathrm{Cov}[D_i, D_{i+h}] = \mathrm{E}[D_i D_{i+h}] - \mathrm{E}[D_i]\mathrm{E}[D_{i+h}], \quad \text{where}$$

$$\mathrm{E}[D_i D_{i+h}] = \frac{1}{2}(\mathrm{E}[(D_i+D_{i+1}+\cdots+D_{i+h})^2]-\mathrm{E}[(D_i+\cdots+D_{i+h-1})^2]-\mathrm{E}[D_{i+h}^2])-\sum_{j=1}^{h-1}\mathrm{E}[D_{i+j}D_{i+h}]$$

Next we develop departure-count based algorithms that are accurate and efficient for approximating first two departure-count moments over specified time intervals.

3.4.1 The $Ph_t/M_t/s/c$ Departure-Count Process

We present an algorithm that computes the first two moments of the number of departures over any chosen interval. In our algorithm the number of differential equations does not depend on the capacity of the system, and is accurate. First we present the departure process KFEs. Then we introduce a set of partial-moment differential equations that do not require closure, hence are numerically exact. The number of NEPMDEs is $3m_A(c+1)$ which is dependent on c. We reduce the number of NEPMDEs to $4m_A$ and refer to the reduced set of differential equations as the departure partial-moment differential equations DPMDE. The set of DPMDEs is not closed, so the final step is to introduce appropriate closure approximations.

3.4.2 The $Ph_t/M_t/s/c$ Departure Process Kolmogorov Forward Equations

Let $P'_{i,d;\ell} = \mathrm{P}(N(t+\tau) = i, D_\tau(t+\tau) = d; A(t+\tau) = \ell)$ and $\ell = 1, \ldots, m_A$.

$$P'_{0,0;\ell} = -\lambda_\ell P_{0,0;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{j,0;0} \quad \text{for } i = 0. \tag{3.29}$$

$$\begin{aligned} P'_{i,0;\ell} &= -\lambda_\ell P_{i,0;\ell} - \min(i,s)\mu P_{i,0;\ell} + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j P_{i-1,d;\ell} \\ &\quad + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{i,0;j}, \quad \text{for } 0 < i < c. \end{aligned} \tag{3.30}$$

$$P'_{0,d;\ell} = -\lambda_\ell P_{0,d;\ell} + \mu P_{1,d-1;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{0,d;j}, \quad \text{for } d > 0 \tag{3.31}$$

$$\begin{aligned} P'_{i,d;\ell} &= -\lambda_\ell P_{i,d;\ell} - \min(i,s)\mu P_{i,d;\ell} + \sum_{j=1}^{m_A} \lambda_j \alpha_\ell a_{j,m_A+1} P_{i-1,d;j} \\ &\quad + \min(i+1,s)\mu P_{i+1,d-1;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{i,d;j}, \\ &\quad \text{for } 0 < i < c \text{ and } d > 0. \end{aligned} \tag{3.32}$$

$$\begin{aligned} P'_{c,d;\ell} &= -\lambda_\ell P_{c,d;\ell} - s\mu P_{c,d;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{c,d;j} + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell P_{c-1,d;j} \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell P_{c,d;j}, \quad \text{for } d > 0. \end{aligned} \tag{3.33}$$

3.4.3 The $Ph_t/M_t/s/c$ Numerically Exact Departure-Count Partial-Moment Differential Equations (NEPMDEs)

Let $\mathrm{E}[D_\tau(t+\tau), N(t+\tau) = i; A(t+\tau) = \ell] = \mathrm{E}[D, i; \ell]$ and $P_{i;\ell} = \mathrm{P}(N(t+\tau) = i, A(t+\tau) = \ell)$ and $\ell = 1, \ldots, m_A$.

$$P'_{0;\ell} = -\lambda_\ell P_{0;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{0;j} + \mu P_{1;\ell} \tag{3.34}$$

$$\mathrm{E}'[D,0;\ell] = -\lambda_\ell \mathrm{E}[D,0;\ell] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D,0;j] + \mu\mathrm{E}[D,1;\ell] + \mu P_{1;\ell} \tag{3.35}$$

$$\begin{aligned}\mathrm{E}'[D^2,0;\ell] &= -\lambda_\ell \mathrm{E}[D^2,0;\ell] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D^2,0;j] \\ &\quad +\mu\mathrm{E}[D^2,1;\ell] + 2\mu\mathrm{E}[D,1;\ell] + \mu P_{1;\ell}.\end{aligned} \tag{3.36}$$

$$\begin{aligned}\mu P'_{i;\ell} &= -\lambda_\ell P_{i;\ell} - \min(i,s)\mu P_{i;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{i;j} \\ &\quad + \min(i+1,s)\mu P_{i+1;\ell} \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell P_{i-1;j} \quad \text{for } 0<i<c\end{aligned} \tag{3.37}$$

$$\begin{aligned}\mathrm{E}'[D,i;\ell] &= -\lambda_\ell \mathrm{E}[D,i;\ell] - \min(i,s)\mu\mathrm{E}[D,i;\ell] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D,i;j] \\ &\quad + \min(i+1,s)\mu\mathrm{E}[D,i+1;\ell] + \min(i+1,s)\mu P_{i+1;\ell} \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell \mathrm{E}[D,i-1;j] \quad \text{for } 0<i<c\end{aligned} \tag{3.38}$$

$$\begin{aligned}\mathrm{E}'[D^2,i;\ell] &= -\lambda_\ell \mathrm{E}[D^2,i;\ell] - \min(i,s)\mu\mathrm{E}[D^2,i;\ell] \\ &\quad + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D^2,i;j] + \min(i+1,s)\mu\mathrm{E}[D^2,i+1;\ell] \\ &\quad + 2\min((i+1),s)\mu\mathrm{E}[D,i+1;\ell] + \min(i+1,s)\mu P_{i+1;\ell} \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell \mathrm{E}[D^2,i-1;j], \quad \text{for } 0<i<c\end{aligned} \tag{3.39}$$

$$\begin{aligned}\mu P'_{c;\ell} &= -\lambda_\ell P_{c;\ell} - s\mu P_{c;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j P_{c;j} \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell P_{c-1;j} + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell P_{c;j}\end{aligned} \tag{3.40}$$

$$\begin{aligned} \mathrm{E}'[D,c;\ell] &= -\lambda_\ell \mathrm{E}[D,c;\ell] - s\mu \mathrm{E}[D,c;\ell] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D,c;j] \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell \mathrm{E}[D,c-1;j] + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell \mathrm{E}[D,c;j] \end{aligned} \tag{3.41}$$

$$\begin{aligned} \mathrm{E}'[D^2,c;\ell] &= -\lambda_\ell \mathrm{E}[D^2,c;\ell] - s\mu \mathrm{E}[D^2,c;\ell] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D^2,c;j] \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell \mathrm{E}[D^2,c-1;j] + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell \mathrm{E}[D^2,c;j] \end{aligned} \tag{3.42}$$

3.4.4 The $Ph_t/M_t/s/c$ Numerically Exact Departure-Count Moment Algorithm (NEDMA):

The $Ph_t/M_t/s/c$ NEDMA solves for the first two departure-count moments over any chosen set of n time intervals by numerically solving the $Ph_t/M_t/s/c$ departure-count. Represent the i^{th} time interval by $[t_i, t_i + \tau_i)$ for $i = 1, \ldots, n$.

The $Ph_t/M_t/s/c$ NEDMA is described by the following two steps:

Step 1: Numerically solve the number-in-system KFEs (Equations 5.1 and 5.2), up to time t_j where $t_j = \max(t_1, \ldots, t_n)$.

Step 2: Numerically solve the $Ph_t/M_t/s/c$ NEDPMEs (Equations 3.34 to 3.41) over the n time intervals. At time t_i, the initial values for the departure-count NEPMDEs is zero since $D_{t_i}(t_i) = 0$ for $i = 1, \ldots, n$. Initialize the number-in-system KFE probabilities at time t_i with the results obtained in Step 1, for $i = 1, \ldots, n$.

3.4.5 Departure Partial-Moment Differential Equations (DPMDEs)

Let $\mathrm{E}[D_\tau(t+\tau), N(t+\tau) \in \Omega_1; A(t+\tau) = \ell] = \mathrm{E}[D, \Omega_1; \ell]$, $\ell = 1, \ldots, m_A$,

$\Omega_1 = \{0, 1, \ldots, s-1\}$ and $\Omega_2 = \{s, \ldots, c\}$

$$\begin{aligned}
\mathrm{E}'[D, \Omega_1; \ell] &= -\lambda_\ell \mathrm{E}[D, \Omega_1; \ell] - \sum_{j=1}^{m_A} a_{j,m_A+1} \lambda_j \alpha_\ell \mathrm{E}[D, s-1; j] + \mu s \mathrm{E}[D, s; \ell] \\
&\quad + \sum_{j=1}^{m_A} a_{j,\ell} \lambda_j \mathrm{E}[D, \Omega_1; j] + \mu \mathrm{E}[N, \Omega_1; \ell] + \mu s P_{s;\ell} \\
&\quad + \sum_{j=1}^{m_A} a_{j,m_A+1} \lambda_j \alpha_\ell \mathrm{E}[D, \Omega_1; j]
\end{aligned} \tag{3.43}$$

$$\begin{aligned}
\mathrm{E}'[D, \Omega_2; \ell] &= -\lambda_\ell \mathrm{E}[D, \Omega_2; \ell] - \mu s \mathrm{E}[D, s; \ell] - \mu s P_{s;\ell} \\
&\quad + \sum_{j=1}^{m_A} a_{j,\ell} \lambda_j \mathrm{E}[D, \Omega_2; j] + \sum_{j=1}^{m_A} a_{j,m_A+1} \lambda_j \alpha_\ell \mathrm{E}[D, \Omega_2; j] \\
&\quad + \sum_{j=1}^{m_A} a_{j,m_A+1} \lambda_j \alpha_\ell \mathrm{E}[D, s-1; j] + \mu s P_{\Omega_2,\ell}
\end{aligned} \tag{3.44}$$

$$\begin{aligned}
\Rightarrow \mathrm{E}'[D; \ell] &= -\lambda_\ell \mathrm{E}[D; \ell] + \sum_{j=1}^{m_A} a_{j,\ell} \lambda_j \mathrm{E}[D; j] + \mu \mathrm{E}[N, \Omega_1; \ell] \\
&\quad + \sum_{j=1}^{m_A} a_{j,m_A+1} \lambda_j \alpha_\ell \mathrm{E}[D; j] + \mu s P_{\Omega_2,\ell}
\end{aligned} \tag{3.45}$$

$$\begin{aligned}
\mathrm{E}'[D^2; \ell] &= -\lambda_\ell \mathrm{E}[D^2; \ell] + \sum_{j=1}^{m_A} a_{j,\ell} \lambda_j \mathrm{E}[D^2; j] + 2\mu \sum_{i=1}^{s-1} i \mathrm{E}[D, i; \ell] \\
&\quad + 2\mu s \mathrm{E}[D, \Omega_2; \ell] + \mu \mathrm{E}[N, \Omega_1; \ell] \\
&\quad + s\mu P_{\Omega_2;\ell} + \sum_{j=1}^{m_A} a_{j,m_A+1} \lambda_j \alpha_\ell \mathrm{E}[D^2; j]
\end{aligned} \tag{3.46}$$

$$\begin{aligned}
\mu\sum_{i=1}^{s-1} i\mathrm{E}'[D,i;\ell] &= -\lambda_\ell\sum_{i=1}^{s-1} i\mathrm{E}[D,i;\ell] - \mu\sum_{i=1}^{s-1} i\mathrm{E}[D,i;\ell] \\
&\quad -s\sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j\alpha_\ell\mathrm{E}[D,s-1;j] - \mu\mathrm{E}[N,\Omega_1;\ell] \\
&\quad +\mu s(s-1)\mathrm{E}[D,s;\ell] + \sum_{j=1}^{m_A}(a_{j,\ell}\lambda_j + a_{j,m_A+1}\lambda_j\alpha_\ell)\sum_{i=1}^{s-1} i\mathrm{E}[D,i;j] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j\alpha_\ell\mathrm{E}[D,\Omega_1;j] + \mu\mathrm{E}[N^2,\Omega_1;\ell] + \mu s(s-1)P_{s;\ell}
\end{aligned} \tag{3.47}$$

The DPMDEs (3.43 to 3.47) along with the number-in-system PMDEs (3.48 to 3.53) form a set of differential equations that is not closed.

The $Ph_t/M_t/s/c$ Number-in-System Partial-Moment Differential Equations (PMDEs)

$$\begin{aligned}
P'_{\Omega_1;\ell} &= -\lambda_\ell(t)P_{\Omega_1;\ell} - \sum_{i=1}^{m_A} a_{i,m_A+1}(t)\alpha_\ell(t)\lambda_i(t)P_{s-1;i} \\
&\quad +\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\alpha_\ell(t)\lambda_i(t)P_{\Omega_1;i} \\
&\quad +\sum_{i=1}^{m_A} a_{i,\ell}(t)\lambda_i(t)P_{\Omega_1;i} + \mu s P_{s;\ell}
\end{aligned} \tag{3.48}$$

$$\begin{aligned}
P'_{\Omega_2;\ell} &= -\lambda_\ell(t)P_{\Omega_2;\ell} - \mu s P_{s;\ell} + \sum_{i=1}^{m_A} a_{i,\ell}(t)\lambda_i(t)P_{\Omega_2;i} \\
&\quad +\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\alpha_\ell(t)\lambda_i(t)P_{s-1;i} + \sum_{i=1}^{m_A} a_{i,m_A+1}(t)\alpha_\ell(t)\lambda_i(t)P_{\Omega_2;i}
\end{aligned} \tag{3.49}$$

$$\begin{aligned}
\mathrm{E}'[N(t),\Omega_1;\ell] &= -\lambda_\ell(t)\mathrm{E}[N(t),\Omega_1;\ell] - \mu\mathrm{E}[N(t),\Omega_1;\ell] \\
&\quad -s\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j P_{s-1;j} - \mu s P_{s;\ell} \\
&\quad +\sum_{j=1}^{m_A} a_{j,\ell}(t)\lambda_j(t)\mathrm{E}[N(t),\Omega_1;j] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell(t)\lambda_j(t)\mathrm{E}[N(t),\Omega_1;j] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell(t)\lambda_j(t)P_{\Omega_1;j} + \mu s^2 P_{s;\ell}
\end{aligned} \tag{3.50}$$

$$
\begin{aligned}
\mathrm{E}'[N(t),\Omega_2;\ell] &= -\lambda_\ell(t)\mathrm{E}[N(t),\Omega_2;\ell] - \sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j P_{c;j} - \mu s P_{\Omega_2;\ell} \\
&\quad -\mu s(s-1)P_{s;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}(t)\lambda_j(t)\mathrm{E}[N(t),\Omega_2;j] \\
&\quad +s\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell(t)\lambda_j(t)P_{s-1;j} \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell(t)\lambda_j(t)\mathrm{E}[N(t),\Omega_2;j] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell(t)\lambda_j(t)P_{\Omega_2;j}
\end{aligned} \tag{3.51}
$$

$$
\begin{aligned}
\mathrm{E}'[N^2(t),\Omega_1;\ell] &= -\lambda_\ell(t)\mathrm{E}[N^2(t),\Omega_1;\ell] - 2\mu\mathrm{E}[N^2(t),\Omega_1;\ell] \\
&\quad -s^2\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j P_{s-1;j} \\
&\quad +\sum_{j=1}^{m_A} a_{j,\ell}\lambda_j\mathrm{E}[N^2(t),\Omega_1;j] + \mu\mathrm{E}[N(t),\Omega_1;\ell] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j\mathrm{E}[N^2(t),\Omega_1;j] \\
&\quad +2\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j\mathrm{E}[N(t),\Omega_1;j] \\
&\quad +\sum_{j=1}^{m_A+1} a_{j,m_A}\alpha_\ell\lambda_j P_{\Omega_1;j} + \mu s(s-1)^2 P_{s;\ell}
\end{aligned} \tag{3.52}
$$

$$
\begin{aligned}
\mathrm{E}'[N^2(t),\Omega_2;\ell] \;=\; & -\lambda_\ell(t)\mathrm{E}[N^2(t),\Omega_2;\ell] \\
& -(2c+1)\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j P_{c;j} - 2\mu s\mathrm{E}[N(t),\Omega_2;\ell] \\
& -\mu(s-1)^2 s P_{s;\ell} + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[N^2(t),\Omega_2;j] \\
& +s^2\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j P_{s-1;j} \\
& +\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j \mathrm{E}[N^2(t),\Omega_2;j] \\
& +2\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j \mathrm{E}[N(t),\Omega_2;j] + \mu s P_{\Omega_2;\ell} \\
& +\sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j P_{\Omega_2;j}
\end{aligned}
\tag{3.53}
$$

3.4.6 Closure Approximations: The Truncated NEPMDEs Method

Closure approximations are constructed at each iteration of the numerical integration in order to close the set of differential equations 3.43 to 3.53. We call the following algorithm the Truncated NEPMDEs Method (TM) algorithm where at every iteration:

Step 1: The number-in-system PMDEs (3.48 to 3.53) can be closed without respect to the departure process. At each iteration Polya-Eggenberger (PE) distributions are used to approximate the probabilities appearing on the RHS of the number-in-system PMDEs; thus closing the set of differential equations. The use of the PE distribution was first introduced by Clark [3] and further examined by Rueda and Taaffe [14], and Ong and Taaffe ([12], [13]). We refer the reader to [9] for further examination on the number in system closure approximations.

The remaining terms appearing on the RHS of the set of DPMDEs, 3.4.5, that need to be approximated for closure are:

$$\mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1; A(t+\tau) = \ell] \quad \text{and}$$

$$\mathrm{E}[D_t(t+\tau), N(t+\tau) = s; A(t+\tau) = \ell] \quad \text{for} \quad \ell = 1 \ldots m_A$$

Step 2: Solve the number-in-system KFEs (Equation 3.29) from $P_{0;\ell}(t+\tau)$ to $P_{s-2;\ell}(t+\tau)$ for $\ell = 1 \ldots m_A$.

Step 3: On the RHS of the number-in-system KFE, $P'_{s-2;\ell}(t+\tau)$, the term $P_{s-1;\ell}(t+\tau)$ needs to be approximated. Set $P_{s-1;\ell(t+\tau)} = \mathrm{P}(N(t+\tau) \in \Omega_1; \ell) - \sum_{i=0}^{s-2} P_{i;\ell}(t+\tau)$.

Step 4: Solve the truncated NEPMDEs, (3.54), for the departure partial moments of

$$\mathrm{E}'\left[D_t\left(t+\tau\right), N\left(t+\tau\right) = i; A\left(t+\tau\right) = \ell\right]$$

for $i = 0, \ldots, s-2$ and $\ell = 1 \ldots m_A$, where

$$\mathrm{E}\left[D, i; \ell\right] = \mathrm{E}\left[D_t\left(t+\tau\right), N\left(t+\tau\right) = i; A\left(t+\tau\right) = \ell\right].$$

$$\begin{aligned}
\mathrm{E}'[D,0;\ell] &= -\lambda_\ell \mathrm{E}[D,0;\ell] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D,0;j] + \mu \mathrm{E}[D,1;\ell] + \mu P_{1;\ell} \\
\mathrm{E}'[D,i;\ell] &= -\lambda_\ell \mathrm{E}[D,i;\ell] - i\mu \mathrm{E}[D,i;\ell] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[D,i;j] \\
&\quad + (i+1)\mu \mathrm{E}[D,i+1;\ell] + (i+1)\mu P_{i+1;\ell} \\
&\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\lambda_j \alpha_\ell \mathrm{E}[D,i-1;j] \text{ for } i = 1 \ldots s-2
\end{aligned} \tag{3.54}$$

Step 5: In Equation 3.54, when $i = s - 2$, the term

$\mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1; A(t+\tau) = \ell]$ requires approximation. Approximate

$\mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1; A(t+\tau) = \ell]$ with

$\mathrm{E}[D_t(t+\tau), N(t+\tau) \in \Omega_1; A(t+\tau) = \ell] - \sum_{i=0}^{s-2} \mathrm{E}[D_t(t+\tau), N(t+\tau) = i; A(t+\tau) = \ell]$.

Step 6: The final step is to provide an approximation for

$\mathrm{E}[D_t(t+\tau), N(t+\tau) = s; A(t+\tau) = \ell]$. We approximate

$\mathrm{E}[D_t(t+\tau) | N(t+\tau) = s; A(t+\tau) = \ell]$ with

$\mathrm{E}[D_t(t+\tau) | N(t+\tau) = s-1; A(t+\tau) = \ell]$, then set

$\mathrm{E}[D_t(t+\tau) | N(t+\tau) = s; A(t+\tau) = \ell] = \mathrm{E}[D_t(t+\tau), N(t+\tau) = s; A(t+\tau) = \ell] P_{s;\ell}$.

Approximating

$$\mathrm{E}\left[D_t\left(t+\tau\right) | N\left(t+\tau\right) = s; A\left(t+\tau\right) = \ell\right]$$

with

$$\mathrm{E}\left[D_t\left(t+\tau\right) | N\left(t+\tau\right) = s-1; A\left(t+\tau\right) = \ell\right]$$

provides good results. It is also possible to extrapolate an approximation of

$$\mathrm{E}[D_t(t+\tau) | N(t+\tau) = s; A(t+\tau) = \ell]$$

using the set

$$\mathrm{E}\left[D_t\left(t+\tau\right) | N\left(t+\tau\right) = s-1; A\left(t+\tau\right) = \ell\right],$$

$$\mathrm{E}\left[D_t\left(t+\tau\right) | N\left(t+\tau\right) = s-2; A\left(t+\tau\right) = \ell\right].$$

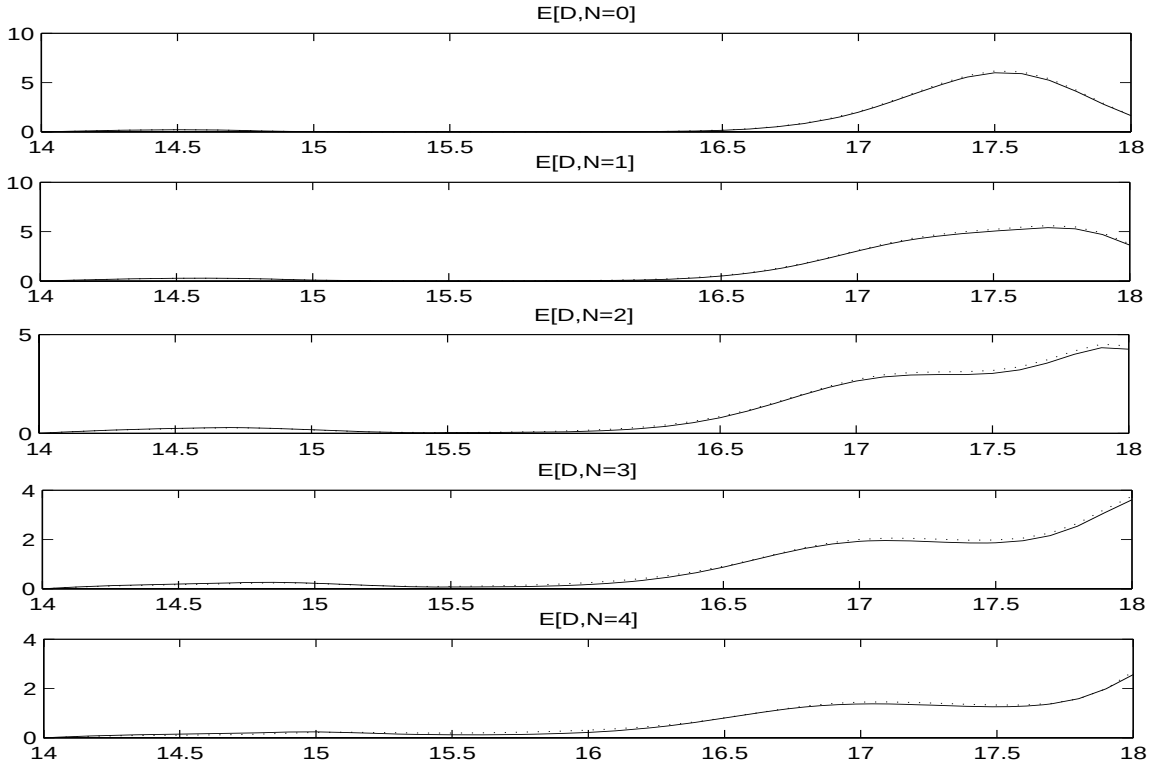
$$\vdots$$

Numerical Example

We test our closure approximations for the departure PMDEs with the results obtained from solving the numerically exact differential equations. We consider an $M_t/M_t/s/c$ node and test our approximation of $\mathrm{E}\left[D|i\right] = \mathrm{E}\left[D_t\left(t+\tau\right), N\left(t+\tau\right) = i\right]$ with and without the extrapolation of Step 6 in the TM algorithm presented in Section 3.4.6.

Consider the $M_t/M_t/s/c$ example with $s = 4, c = 20, \lambda\left(t\right) = 7(1+\cos 2t)$ and $\mu\left(t\right) = 2(1+\sin 2t)$. Figure 3.3 is a plot of the partial moments of $\mathrm{E}\left[D_t\left(t+\tau\right), N\left(t+\tau\right) = i\right]$ from $i = 0, \ldots, s$ using our closure approximations and the numerically exact solutions.

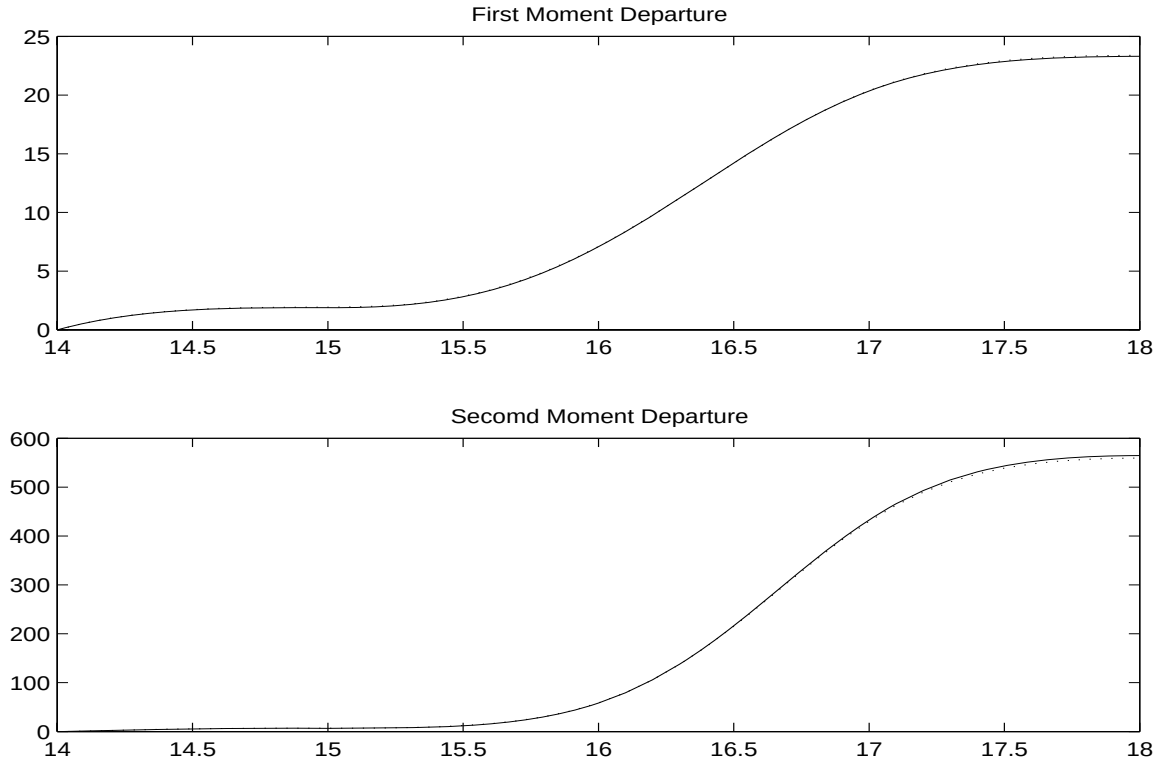
Figure 3.3: Partial Moments



Recall that $\mathrm{E}[D_t(t+\tau), N(t+\tau) = s-1]$ and $\mathrm{E}[D_t(t+\tau), N(t+\tau) = s]$ are the terms needed

to close the DPMDEs for the first two moments. The plot of the first two moments of the departure count, $D_t(t+\tau)$ is shown in Figure 3.4.

Figure 3.4: Departure Moments



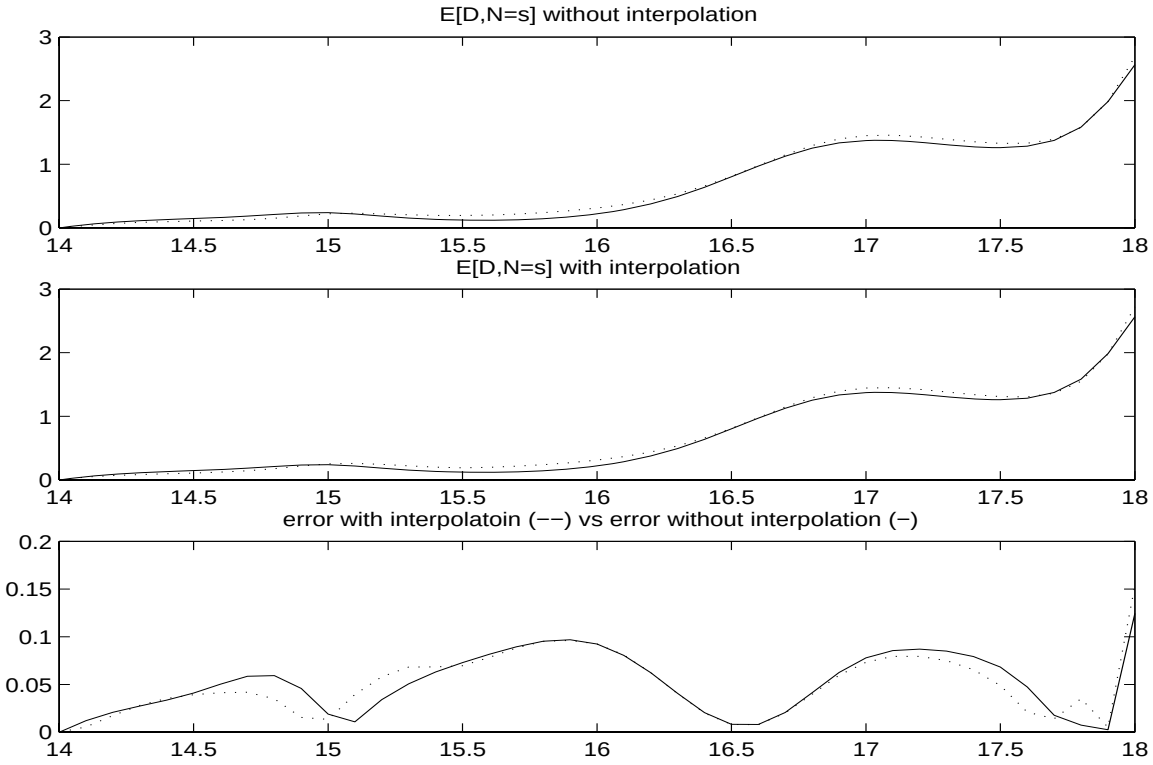
Here we use the simple approximation

$$\mathrm{E}[D_t(t+\tau)|N(t+\tau)=s] \approx \mathrm{E}[D_t(t+\tau)|N(t+\tau)=s-1].$$

We compare this simple approach for approximating $\mathrm{E}[D_t(t+\tau)|N(t+\tau)=s]$ with a polynomial interpolation/extrapolation approach using the values of $\mathrm{E}[D_t(t+\tau), N(t+\tau)=i]$ from $i=0\cdots s-1$. This interpolation does consistently improve the quality of the approximation. Figure 3.5 is a plot of the $\mathrm{E}[D_t(t+\tau)|N(t+\tau)=s]$ approximation both using the interpolation and without using the interpolation, as well as the numerically exact

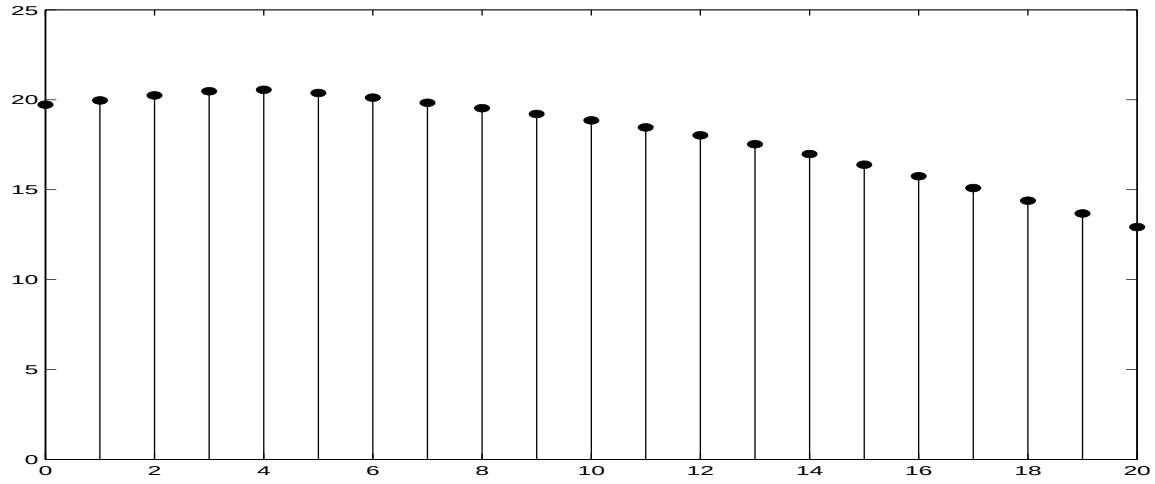
solution. The third plot in Figure 3.5 is the absolute errors across time for each method where interpolation error is plotted with the dotted lines.

Figure 3.5: E$[D, N = s]$



Consider Figure 3.6 – a plot of $\mathrm{E}[D_t(t+\tau)|N(t+\tau) = i]$ for $i = 0, \ldots, c$ at time $t + \tau = 17$. The reasoning behind our approximating $\mathrm{E}[D_t(t+\tau)|N(t+\tau) = s]$ with $\mathrm{E}[D_t(t+\tau)|N(t+\tau) = s-1]$ is clear; the two values are very close. The slight difference in values is diminished further since we use the approximated conditional-moment to approximate the partial-moment $\mathrm{E}[D_t(t+\tau), N(t+\tau) = s]$ which is then multiplied by $\mathrm{P}(N(t) = s)$; thus diminishing the effects of the approximation error.

Figure 3.6: $\mathrm{E}[D|N=i]$ for $i=1\ldots c$ and $s=4$



3.4.7 Closure Approximations: The Independence Method

Clearly the correlation between the number of departures over the time interval $[t, t+\tau)$ and the number in system at time t and/or $(t+\tau)$ decreases as τ increases. In our approximations we assume that the expected number of departures in an interval $[t, t+\tau)$ conditioned on the number-in-system being equal to s at time $t+\tau$ is equal to the expected number of departures conditioned on the number in system being $s-1$. As τ increases, it is also reasonable to use $\mathrm{E}[D_t(t+\tau)|N(t+\tau) \in \Omega_1]$ as an approximation for $\mathrm{E}[D_t(t+\tau)|N(t+\tau) = s-1]$ at each iteration of the numerical integration. Although such an approximation assumes less dependence between the number of departures and the number in system, it is computationally efficient to use this approximation because using it eliminates the need to solve the truncated NEPMDEs and it provides better approximations as τ increases.

Since solving the NEPMDEs is superior for smaller values of τ and inferior for larger values, we suggest finding a limit value γ where if $\tau > \gamma$ use the independence method otherwise use

the truncated NEPMDEs method. Both methods effect the second moment approximation and the partial first moments of $D_t(t+\tau)$, but not the first moment as can be seen in Equation 3.45. In the next section we present examples and we compute the second moment of the number of departures using both methods separately and then combining the methods by approximating γ.

$$\mathrm{E}[D_t(t+\tau)|N(t+\tau)=s-1] \approx \mathrm{E}[D_t(t+\tau)|N(t+\tau)\in\Omega_1]$$

3.4.8 Examples of the Number-of-Departures Moments over Time Intervals

Consider $n(n-1)/2$ consecutive time intervals defined by choosing any 2 of the n time points $p_1, p_2, \ldots, p_n$. We find the first two moments of the number of departures in an interval for the set of intervals defined by points $p_1, p_2, \ldots, p_n$. We find the numerically exact solutions and the closure approximations. We consider four examples. The first example is a time dependent $M_t/M_t/s/c$ system where only the first two moments are computed. The second example is a stationary $M/M/s/c$ system with large capacity. The reason for choosing the second example is that we know that the departure process for this system is Poisson. The third and fourth examples are $Ph_t/M_t/s/c$ systems.

Example 1:

In this example we set $s=4, c=200, \lambda(t)=7(1+\cos 2t)$ and $\mu(t)=2(1+\sin 2t)$. Consider the four points $p_1=12$, $p_2=16$, $p_3=20$ and $p_4=24$. Tables 3.1, 3.2 and 3.3 contain the

first two departure moments over the intervals defined by the four points, using CAs and the numerically exact (NE) method.

Table 3.1: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	26.7665	60.5182	86.7886	26.7961	60.5149	86.8521	-0.1102	0.0054	-0.0730
p_2	0	33.7516	60.0221	0	33.7188	60.0560	0	0.0974	-0.0564
p_3	0	0	26.2705	0	0	26.3372	0	0	-0.2533

Table 3.2: Second Moments (Truncated NEPMDEs Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	726.2	3673.4	7522.9	741.1	3713.7	7618.9	-2.0072	-1.0849	-1.2601
p_2	0	1172.9	3650.9	0	1169.9	3665.7	0	0.2492	-0.4036
p_3	0	0	716.0	0	0	724.6	0	0	-1.1881

Table 3.3: Second Moments (Independence Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	749.2	3736.3	7644.4	741.1	3713.7	7618.9	1.0868	0.6103	0.3353
p_2	0	1179.8	3681.3	0	1169.9	3665.7	0	0.8460	0.4267
p_3	0	0	726.2	0	0	724.6	0	0	0.2283

Example 2:

In this example $s = 4, c = 200, \lambda = 5$ and $\mu = 2$. Consider the four points $p_1 = 64, p_2 = 68, p_3 = 72$ and $p_4 = 76$. Table 3.4 displays the departure-count first moments which are the same for the CA and NE methods. Tables 3.5 and 3.6 display the second moments of the departure counts over the intervals defined by the four points.

Table 3.4: First Moments CA and NE

	p_2	p_3	p_4
p_1	20	40	60
p_2	0	20	40
p_3	0	0	20

Table 3.5: Second Moments (Truncated NEPMDEs Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	421.0	1648.9	3691.1	420.0	1640.0	3660.0	0.2348	0.5400	0.8485
p_2	0	421.0	1648.9	0	420.0	1640.0	0	0.2347	0.5399
p_3	0	0	421.0	0	0	420.0	0	0	0.2348

Table 3.6: Second Moments (Independence Method)

	p_2	p_3	p_4
p_1	420.0	1640.0	3660.0
p_2	0	420.0	1640.0
p_3	0	0	420.0

Example 3: In this example $[s = 3, c = 200, \mu = 1/2, m_A = 5,$

$$\alpha = [0.2, \quad 0.1, \quad 0.1, \quad 0.2, \quad 0.4],$$

$$a = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.2 & 0.3; \\ 0.3 & 0.1 & 0.1 & 0.3 & 0.2; \\ 0.3 & 0.2 & 0.1 & 0.2 & 0.2; \\ 0.3 & 0.1 & 0.2 & 0.3 & 0.1 \end{pmatrix},$$

and $\lambda(t) =$

$$(5 + 0.5\sin(t/(3\pi)), 3 + 0.5\sin(t/(3\pi)), 4 + 0.5\sin(t/(3\pi)), 2.5 + \sin(t/(3\pi)), 4 + \cos(t/(3\pi)))$$

Consider the four time points $p_1 = 16, p_2 = 20, p_3 = 24$ and $p_4 = 28$. Tables 3.7, 3.8 and 3.9 contain the first two moments of the number of departures over the intervals defined by the four points, using CAs and the numerically exact (NE) method.

Table 3.7: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	5.6628	11.3380	16.9798	5.6194	11.2504	16.8513	0.7729	0.7791	0.7629
p_2	0	5.6752	11.3170	0	5.6310	11.2319	0	0.7853	0.7578
p_3	0	0	5.6418	0	0	5.6009	0	0	0.7302

Table 3.8: Second Moments (Truncated NEPMDEs Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	37.4913	139.0129	303.4057	36.9316	136.9371	299.0743	1.5154	1.5159	1.4483
p_2	0	37.6982	138.6922	0	37.1321	136.6741	0	1.5244	1.4766
p_3	0	0	37.3393	0	0	36.8259	0	0	1.3942

Table 3.9: Second Moments (Independence Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	37.6029	139.5939	304.8667	36.9316	136.9371	299.0743	1.8176	1.9402	1.9368
p_2	0	37.7865	139.1715	0	37.1321	136.6741	0	1.7622	1.8273
p_3	0	0	37.4044	0	0	36.8259	0	0	1.5708

Example 4: In this example $s = 3, c = 30, \mu = 1, m_A = 4$,

$$\alpha = [1, \quad 0, \quad 0, \quad 0], \qquad \lambda = 3[4, \quad 3, \quad 2, \quad 6],$$

and

$$a = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Consider the four points $p_1 = 0, p_2 = 8, p_3 = 16$ and $p_4 = 24$. Tables 3.10, 3.11 and 3.12 contain the first two moments of the number of departures over the intervals defined by the four points, using CAs and the numerically exact (NE) method.

Table 3.10: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	15.6489	34.4739	53.5326	15.6445	34.4763	53.5409	0.0286	-0.0070	-0.0155
p_2	0	18.8250	37.8836	0	18.8319	37.8964	0	-0.0366	-0.0337
p_3	0	0	19.0587	0	0	19.0645	0	0	-0.0308

Table 3.11: Second Moments (Truncated NEPMDEs Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	234.5	1119.2	2689.1	251.1	1200.7	2884.3	-6.5742	-6.7926	-6.7669
p_2	0	345.5	1366.8	0	364.3	1452.1	0	-5.1547	-5.8749
p_3	0	0	356.5	0	0	374.0	0	0	-4.6949

Table 3.12: Second Moments (Independence Method))

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	253.4	1209.0	2899.1	251.1	1200.7	2884.3	0.9335	0.6922	0.5132
p_2	0	367.3	1460.8	0	364.3	1452.1	0	-0.8189	0.6020
p_3	0	0	376.7	0	0	374.0	0	0	0.7265

In the preceding examples, the Independence Method seems to perform better than Truncated NEPMDEs Method. This can be clearly seen in Example 2 where the departure process is Poisson and the Independence Method gives exact values for the second moments. In Example 3 the Truncated NEPMDEs Method performs slightly better than the other methods but performs poorly in Example 4.

3.4.9 Closure Approximations: The Combination Method

Consider a time, γ, such that the Independence Method performs better when $\tau > \gamma$. We estimate a value for γ that results in improved accuracy for the second moment of the number of departures in an interval. If the departure process is Poisson then it is best to use the Independence Method; i.e., $\gamma = 0$. As the squared coefficient of variation (scv) of the number of departures in the time interval $[t, t+\tau)$, $\text{scv}_t(t+\tau)$, approaches 1, we use the Independence Method. Also as $D_t(t+\tau)$ increases, the correlation between $D_t(t+\tau)$ and the number in system at time $t+\tau$ decreases. So we use the Independence Method whenever $\text{E}[D_t(t+\tau)] > 5$. Consider the following $Ph_t/M_t/s/c$ Combination Method (CM) algorithm to determine γ:

Step 1: Set the method to the $Ph_t/M_t/s/c$ Truncated NEPMDEs Method.

Step 2: At every iteration of the numerical integration compute the coefficient of variation

$$\text{scv}_t(t+\tau) = (\text{E}[D_t^2(t+\tau)] - (\text{E}[D_t(t+\tau)])^2)/(\text{E}[D_t(t+\tau)])^2).$$

Step 3: If $\mathrm{E}[D_t(t+\tau)] > 2$ and

$\Big($ $0.95 \leq \mathrm{scv}_t(t+\tau) \leq 1.05$

or ($\mathrm{scv}_t(t+\tau) > 1.05$ and $\mathrm{E}[D_t(t+\tau)] > 5\mathrm{scv}_t(t+\tau)$)

or ($\mathrm{scv}_t(t+\tau) < 0.95$ and $\mathrm{E}[D_t(t+\tau)] > 5 - 5\mathrm{scv}_t(t+\tau)$) $\Big)$,

then $\gamma = \tau$, start using the $Ph_t/M_t/s/c$ Independence Method.

Results from using the CM Algorithm on Examples 1,2,3 and 4 produced the departure-count second moments shown in Tables 3.13 to 3.16.

Table 3.13: Second Moments Example 1 (Combination Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	748.9	3736.1	7644.2	741.1	3713.7	7618.9	1.0552	0.6040	0.3322
p_2	0	1179.5	3681.0	0	1169.9	3665.7	0	0.8181	0.4177
p_3	0	0	725.6	0	0	724.6	0	0	0.1393

Table 3.14: Second Moments Example 2 (Combination Method)

	p_2	p_3	p_4
p_1	420.0	1640.0	3660.0
p_2	0	420.0	1640.0
p_3	0	0	420.0

Table 3.15: Second Moments Example 3 (Combination Method)

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	37.5000	139.4625	304.7302	36.9316	136.9371	299.0743	1.5391	1.8442	1.8911
p_2	0	37.7050	139.0345	0	37.1321	136.6741	0	1.5428	1.7270
p_3	0	0	37.3414	0	0	36.8259	0	0	1.3999

Table 3.16: Second Moments Example 4 (Combination Method))

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	251.0	1206.6	2896.6	251.1	1200.7	2884.3	-0.0373	0.4888	0.4286
p_2	0	365.6	1459.2	0	364.3	1452.1	0	-0.3545	0.4854
p_3	0	0	375.2	0	0	374.0	0	0	0.3199

Conclusion

The departure-count first-moment approximations are consistently good regardless of the method used to compute them. The CM algorithm does not always perform better than the best solution between the Independence Method and the Truncated NEPMDEs, but the CM algorithm consistently provides good approximations and in each case is better than at least one of the other two methods, if not both. In the cases where the CM does not provide the best approximations, it is very close to the best approximation. We conclude that the CM is a good choice for obtaining approximations for the first two moments of $D_t(t+\tau)$.

Note that there might exist a better algorithm than the algorithm we presented to obtain approximations for the value of γ (which can explain why the CM algorithm does not always give the best approximations). Since the departure PMDEs depend on the CA of the number in system PMDEs, we refer to [9] for details of our investigation of the performance of the CAs for different models.

3.5 Fitting a Ph_t Distribution to the Departure Count Moments

In the previous section we presented approximations for the time-dependent departure-count moments for a $Ph_t/M_t/s/c$ node. In this section we assume that the set of first-two moments for the departure counts across a set of intervals and subintervals is known. Given this set of

moments we now describe construction of a point process to approximate the true departure process as a first step in constructing an approximating arrival process for a downstream node in a network.

For clarity in this section we re-specify the initial-state probability vector, $\boldsymbol{\alpha}$, for the Ph distribution or Ph renewal process and $\boldsymbol{\alpha}_t$ for the time-dependent Ph_t process as the *process initial-state* probability vector. We do this because we will consider several time intervals and will refer to the *interval* initial-state probability vector. These two probability mass function are not, in general, the same. The process initial-state probability vector is part of the structure of the Ph or Ph_t distribution or process. This vector contains probabilities for the initial state of the arrival process immediately after an arrival event. An interval initial-state probability vector describes the arrival-process state probabilities at the moment of the beginning of a selected time interval. In the discussion below we will define several time intervals. The starting moment of time, the length and thus the ending moment of time of these time intervals are selected constants and are not random variables. Since the time intervals start at specified (non-event) times, we must specify the state of the arrival process at the beginning of these time intervals. As described below we will choose our time-interval initial-state probability vectors to be the associated equilibrium arrival-process state probabilities.

Define $\widetilde{Ph_t}$ as the approximating process for the true departure process, and of course $\widetilde{Ph_t}$ can also serve as an approximating arrival process at the downstream node. Using the departure-count moments described in the last section we construct a $\widetilde{Ph_t}$ process for use as

the approximating arrival process at the downstream node during the next time interval.

It is important to understand the various types of time intervals in this discussion. There are two types of random intervals whose characteristics we neither explicitly model nor do we approximate. Those two types of random intervals are 1) the *time-between-departure intervals* (these intervals start and stop at successive departure epics), and 2) the intervals describing the amount of time from time t until the next departure.

We do use two types of deterministically sized intervals; 1) time intervals over which a particular $\widetilde{Ph_t}$ may serve as an approximating arrival process, and 2) time intervals that are subintervals of the time interval over which a particular $\widetilde{Ph_t}$ may serve as an approximating arrival process.

For any type of interval we can define a *departure-count* random variable. These departure-count random variables record the number of departures over the span of the interval. We make use of two types of departure-count random variables. The first is a random variable that counts the number of departures over a relatively large interval, and the second is a departure-count random variable over each of a set of subintervals of the larger interval. We use the larger intervals to approximate departure-count second moments and use the smaller intervals to approximate the departure-count first moments.

Define an interval of size T to be a Second-Moment Interval (*SMI*). These SMIs are intervals during which a $\widetilde{Ph_t}$ of a particular structure will serve as our approximating arrival process. Within an SMI the approximating $\widetilde{Ph_t}$ process can be time-dependent, but its structure

(number of phases) is not. Also the second moment of the number of departures occurring within an SMI is fixed.

Define an interval of size Δ, where $n\Delta = T$ for some positive integer n, to be a First-Moment Interval (*FMI*). FMIs are subintervals of the SMIs. These FMIs are intervals during which the approximating $\widetilde{Ph_t}$ process is stationary (neither the structure nor the parameters change). Thus the first moment of the number of departures occurring within an FMI is fixed (as is the second moment). Notice that the first moment of the number of departures within an SMI does change as time progresses through the SMI.

In our algorithm for constructing approximating $\widetilde{Ph_t}$ processes we keep the structure and the second moment of the number of departures of the approximating $\widetilde{Ph_t}$ process constant over the length of an SMI interval, but we allow changes in the parameter values and departure-count first moments of the approximating $\widetilde{Ph_t}$ process across the sequence of FMIs comprising the associated SMI.

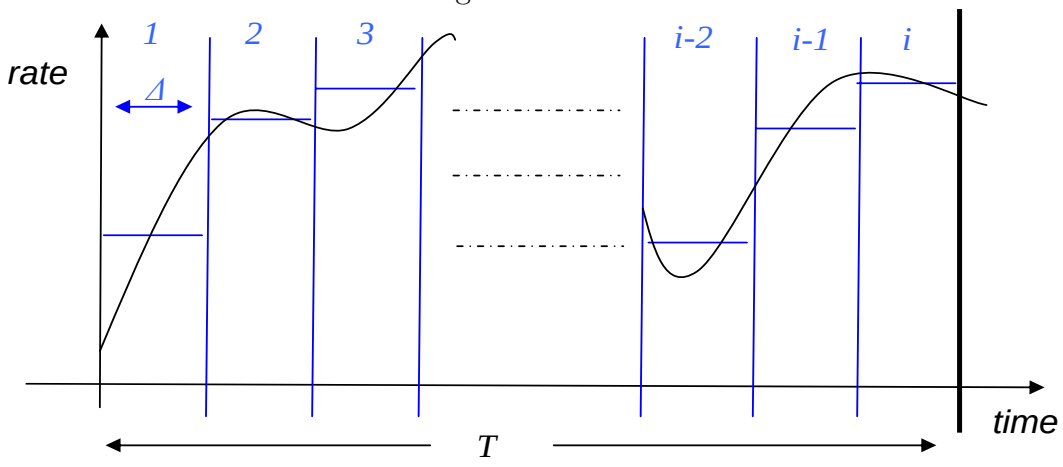
The reason that our approximation construction requires that the structure of the approximating $\widetilde{Ph_t}$ be constant across an entire SMI interval is one of efficiency and a way to avoid computational difficulties. We do allow flexibility in the approximating $\widetilde{Ph_t}$ across the sequence of FMIs comprising a particular SMI interval because we allow the parameters and first moments of the number of departures of the approximating phase-type process to change across FMIs. The manner in which we change the parameter values across FMIs within a particular SMI is such that the $\widetilde{Ph_t}$ state probabilities are held constant. We accomplish this

by specifying the initial condition that all $\Pr(A(j\Delta) = i) \Leftarrow \pi_i(j\Delta)$, where $\Pr(A(j\Delta) = i)$ are the arrivel phase-state probabilities and $\pi_i(j\Delta)$ are the steady-state arrival process probabilities; i.e., the state probabilities of the approximating $\widetilde{Ph_t}$ process. Further we require that $\Pr(A(j\Delta) = i) = \Pr(A(k\Delta) = i)$ for all i and all j and k where $j \neq k$ within the same SMI. Thus the distribution of the state of the arrival-process at the beginning of the SMI is the same throughout the entire SMI and is the equilibrium arrival-state probability distribution. With this algorithm it is easy to smoothly patch the FMIs together to form one easy-to-compute SMI interval. Obviously the count process produced by the fitted $\widetilde{Ph_t}$ distribution should closely match the approximated moments of the true departure process.

Note that in our algorithm described above we used the equilibrium arrival-process distribution as our *interval* initial-state distribution *at the beginning of both the FMIs and the SMIs.* This is *not* the same as the *process* initial-state distribution as discussed earlier.

Consider an SMI time interval of length T. The count process associated with the approximating arrival process, $\widetilde{Ph_t}$, for the interval of length T, should have its first two moments match the first two moments of the departure count over this SMI interval. Since we consider time dependent systems where the departure rate changes frequently, we match the first moment of the departure count for intervals of length Δ. Let $\Delta = T/n$ where n is a positive integer. Figure 3.7 shows a choice of T and the corresponding Δs. For example if we know that the departure process we are approximating in Figure 3.7 is Poisson, then we represent it by one phase with transition rate $\lambda(t) = \mathrm{E}[D_{(j-1)\Delta}(j\Delta)]/\Delta$ for $t \in [(j-1)\Delta, j\Delta)$.

Figure 3.7: Intervals



Over the SMI interval T we choose a hyper-exponential, Erlang or exponential distribution to approximate the departure process. Our choice depends on $\mathrm{E}[D_t(t+T)]$ and $\mathrm{E}[D_t^2(t+T)]$. Within the SMI time interval T, the approximating $\widetilde{Ph_t}$ processes (one for each FMI) are independent. Further, their second moments are all the same within a SMI but the set of first moments may all be different within the SMI. Let $\pi_i(t)$ for $i = 1, \ldots, m_A$ be the probability that the approximating $\widetilde{Ph_t}$ process is in phase i at time t. When choosing an approximating $\widetilde{Ph_t}$ process over a FMI interval of length Δ, we construct a computationally efficient and accurate algorithm to determine the parameters of the $\widetilde{Ph_t}$ process that match the departure-count first moment over the FMI interval. We require the following properties when choosing the approximating $\widetilde{Ph_t}$ process over the time interval of length T.

1. For the approximating arrival process, $\widetilde{Ph_t}$, let the random phase that the process is in when the current interval ends be a random variable called the *interval-terminating* phase. The interval initial phase in the approximating arrival process for the next

interval is set to the interval terminating phase in the previous interval. In other words, we set $\pi_i(j\Delta) \Leftarrow \pi_i(j\Delta^-)$ for $j = 1, 2, \ldots$

2. Find a family of $\widetilde{Ph}$ processes where computing the values of the equilibrium arrival-phase state probabilities is not a function of the resulting departure-count process. This property simplifies the moment-matching algorithm because closed-form analytical solution to the departure-count moments are available.

3. Match the departure-count first moments over the FMIs,

   $\lambda(t) = \mathrm{E}[D_{(j-1)\Delta}(j\Delta)]/\Delta$ for $t \in [(j-1)\Delta, j\Delta)$.

4. Match the departure-count second moment over the SMIs.

So the approximating $\widetilde{Ph_t}$ has time-dependent phase-transition rates across an SMI but constant phase-state probabilities over an SMI. The algorithm eliminates the need to consider conditioning on an arrival at the start of the SMI as one would do in considering the associated Palm process.

3.5.1 Ph_t Count Process

The Kolmogorov forward and moment differential equations for the joint arrival and count processes for a Ph_t arrival process over the interval $[t, t+\tau)$ are:

$$P'_{\ell,d} = -\lambda_\ell P_{\ell,d} + \sum_{i=1}^{m_1} \lambda_i a_{i,\ell} P_{i,d} + \sum_{i=1}^{m_1} \lambda_i a_{i,m_A+1} \alpha_\ell P_{i,d-1} \tag{3.55}$$

$$
\begin{aligned}
\mathrm{E}'[D_t(t+\tau), A(t)=\ell] &= \sum_{d=1}^{\infty} dP'_{d,\ell} \\
&= -\lambda_\ell \mathrm{E}[D_t(t+\tau), A(t)=\ell] \\
&\quad + \sum_{i=1}^{m_A} \lambda_i a_{i,\ell} \mathrm{E}[D_t(t+\tau), A(t)=\ell] \\
&\quad + \sum_{i=1}^{m_A} \lambda_i a_{i,m_A+1} \alpha_\ell \mathrm{E}[D_t(t+\tau), A(t)=\ell] \\
&\quad + \sum_{i=1}^{m_A} \lambda_i a_{i,m_A+1} \alpha_\ell \mathrm{P}(A(t)=i),
\end{aligned} \tag{3.56}
$$

and

$$
\begin{aligned}
\mathrm{E}'[D_t^n(t+\tau), A(t)=\ell] &= \sum_{d=1}^{\infty} d^n P'_{d,\ell} \\
&= -\lambda_\ell \mathrm{E}[D_t^n(t+\tau), A(t)=\ell] \\
&\quad + \sum_{i=1}^{m_A} \lambda_i a_{i,\ell} \mathrm{E}[D_t^n(t+\tau), A(t)=\ell] \\
&\quad + \sum_{i=1}^{m_A} \lambda_i a_{i,m_A+1} \alpha_\ell \\
&\quad \left(\sum_{k=1}^{n} \binom{n}{k} \mathrm{E}[D_t^k(t+\tau), A(t)=\ell] + \mathrm{P}(A(t)=i) \right),
\end{aligned} \tag{3.57}
$$

for $\ell = 1, \ldots, m_A$ and $n = 2, 3, \ldots$.

The probability distribution as well as the moments of the arrival-count process produced by a Ph_t arrival process can be numerically obtained by solving the differential equations (3.55, 3.56 and 3.57). We use those differential equations to obtain the first two moments of the $\widetilde{Ph_t}$ arrival-count process. A desirable property is an analytic form for the solution of the differential equations for the arrival-count processes, $\widetilde{Ph_t}$, we are trying to fit. The analytic closed-form solution should be over the length of the FMI. So for each FMI, in addition to the four properties described above we also require the additional property: 5) $\widetilde{Ph_t} \equiv \widetilde{Ph}$, meaning constant rates $\lambda_i(t) = \lambda$ for $i = 1, \ldots, m$ where m is the number of

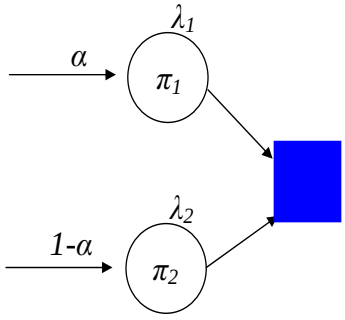
phases. Properties 1,2 and 5 enables us to analytically solve the differential equations over each FMI.

3.5.2 Approximating a Count Process with High Variability

Consider a count process over an SMI to have *high variability* if $\mathrm{E}[D_t^2(t+T)] > \mathrm{E}[D_t(t+T)]^2 + \mathrm{E}[D_t(t+T)]$. For high-variability SMIs we choose a two-phase hyper-exponential distribution to serve as the approximate interval-size distribution, as shown in Figure 3.8. The black box represents the absorbing state and α represents the probability of starting in phase 1. Let λ_1 and λ_2 represent the transition rates. Let π_1 and π_2 represent the equilibrium phase-state probabilities. We consider the balanced-means hyper-exponential distribution, described by Whitt in [16], where the rate of departure from both phases is the same. This implies $(1-\alpha)\lambda_1 = \alpha\lambda_2$, or $\lambda_1 = \lambda$ and $\lambda_2 = \lambda(1-\alpha)/\alpha$. This results in $\pi_1 = \pi_2 = 1/2$ for any λ and α which provides a smooth transition between FMIs. An important characteristic of this count process is that the rate depends only on λ. So we adjust λ to satisfy the required first moment of the departure count over the FMI. We adjust α to satisfy the second moment of the departure count over the SMI.

The first two moments of the count process produced by this inter-arrival process can be obtained by solving the differential equations 3.58, where $\mathrm{E}[D; i] = \mathrm{E}[D_t(t+\tau); N(t+\tau) = i]$ and $i = 1, 2$. The derivative is with respect to τ.

Figure 3.8: Hyper-Exponential



$$
\begin{aligned}
\mathrm{E}'[D;1] &= -\lambda_1 \mathrm{E}[D;1] + \alpha(\lambda_1 \mathrm{E}[D;1] + \lambda_2 \mathrm{E}[D_t;2] + 1/2(\lambda_1+\lambda_2)) \\
\mathrm{E}'[D;2] &= -\lambda_2 \mathrm{E}[D;2] + (1-\alpha)(\lambda_1 \mathrm{E}[D;1] + \lambda_2 \mathrm{E}[D;2] + 1/2(\lambda_1+\lambda_2)) \\
\mathrm{E}'[D^2;1] &= -\lambda_1 \mathrm{E}[D^2;1] + \alpha(\lambda_1(\mathrm{E}[D^2;1] + 2\mathrm{E}[D;1] + 1/2) \\
&\quad +\lambda_2(\mathrm{E}[D^2;2] + 2\mathrm{E}[D;2] + 1/2)) \\
\mathrm{E}'[D^2;2] &= -\lambda_1 \mathrm{E}[D^2;2] + (1-\alpha)(\lambda_1(\mathrm{E}[D^2;1] + 2\mathrm{E}[D;1] + 1/2) \\
&\quad +\lambda_2(\mathrm{E}[D^2;2] + 2\mathrm{E}[D;2] + 1/2)) \\
\Rightarrow \mathrm{E}'[D^2] &= (2\mathrm{E}[D;1] + 1/2)\lambda_1 + (2\mathrm{E}[D;2] + 1/2)\lambda_2
\end{aligned}
\tag{3.58}
$$

Only three differential equations are required to obtain the first two moments, and they simplify to (3.59).

Let $x_1(\tau) = \mathrm{E}[D;1]$, $x_2(\tau) = \mathrm{E}[D;2]$ and $y(\tau) = \mathrm{E}[D^2]$

$$
\begin{aligned}
x_1(\tau)' &= -\lambda x_1(\tau) + \alpha\lambda x_1(\tau) + (1-\alpha)\lambda^{-1} x_2(\tau) + 2^{-1}\lambda \\
x_2(\tau)' &= -\lambda x_2(\tau) + \alpha\lambda x_1(\tau) + (1-\alpha)^2 \lambda \alpha^{-1} x_2(\tau) + \lambda(1-\alpha)(2\alpha)^{-1} \\
y(\tau)' &= \lambda(2x_1(\tau) + 2^{-1}) + \lambda(1-\alpha)(2x_2(\tau) + 2^{-1})\alpha^{-1}
\end{aligned}
\tag{3.59}
$$

Let $X_1(s), X_2(s)$ and $Y(s)$ be the Laplace transforms of $x_1(\tau), x_2(\tau)$ and $y(\tau)$. So

$$X_1(s) = \int_0^\infty x_1(\tau)e^{-s\tau}d\tau, X_2(s) = \int_0^\infty x_2(\tau)e^{-s\tau}d\tau$$

and

$$Y(s) = \int_0^\infty y(\tau)e^{-s\tau}d\tau,$$

and we obtain

$$\begin{aligned} sX_1(s) - x_1(0) &= \lambda((\alpha-1)X_1(s) + (1-\alpha)X_2(s) + (2s)^{-1}) \\ sX_2(s) - x_2(0) &= \lambda(-X_2(s) + (1-\alpha)(X_1(s) + 1 - \alpha X_2(s)\alpha^{-1} + \lambda(2s)^{-1} \\ sY(s) - y(0) &= \lambda(2X_1(s) + (2s)^{-1} + (1-\alpha)\alpha^{-1}(2X_2(s) + (2s)^{-1})). \end{aligned} \tag{3.60}$$

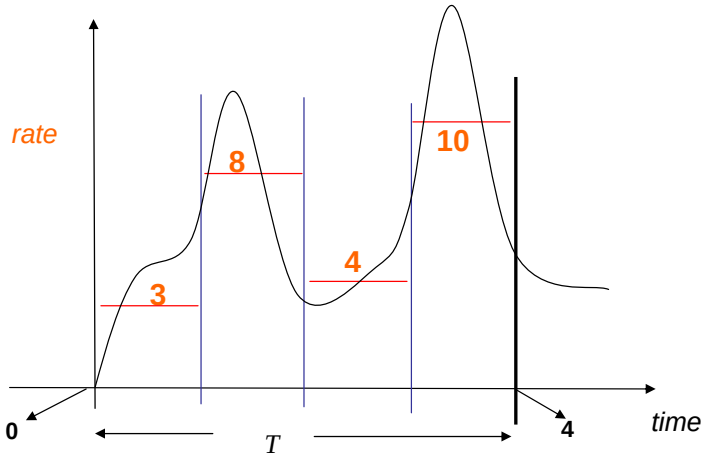
Solving for $X_1(s), X_2(s)$ and $Y(s)$ in 3.60 and then taking the Laplace inverse, we obtain the analytical representation on the first two moments of the count process as a function of the initial conditions.

$$\begin{aligned} x_1(\tau) &= (\lambda\Delta 2^{-1} - e^{-(1-\alpha)\lambda\Delta}(2\alpha x_1(0)\cosh((1-\alpha)\Delta\lambda) - f(x_2(0)))(2\alpha)^{-1} \\ x_2(\tau) &= (\lambda\Delta 2^{-1} - e^{-(1-\alpha)\lambda\Delta}(2\alpha x_2(0)\cosh((1-\alpha)\Delta\lambda) - f(x_1(0)))(2\alpha)^{-1} \\ y(\tau) &= -(2\alpha-1)(4x_2(0)\alpha^2 - 4\alpha^2 x_1(0) - 4x_2(0)\alpha - 2\alpha + 4\alpha x_1(0) + 1) \\ &\quad e^{-(1-\alpha)\lambda\Delta}\sinh((\lambda\alpha-\lambda)\Delta)(4\alpha^2(1-\alpha)^2)^{-1} \\ &\quad ((4^{-1}\lambda^2\Delta^2(-1+\alpha) + \lambda(-4\alpha x_1(0) + 4\alpha^2 x_1(0) - 1 + 2\alpha - 4x_2(0)\alpha \\ &\quad +4x_2(0)\alpha^2 - 2\alpha^2)4^{-1}\Delta + y(0)\alpha^2(1-\alpha))(\alpha^2(1-\alpha))^{-1} \end{aligned} \tag{3.61}$$

where $f(x) = (4x\alpha^2 - 4x\alpha - 2\alpha + 1)\sinh((\lambda\alpha-\lambda)\Delta)/(2(1-\alpha))^{-1}$.

Example 3.5.2 : Consider the hyper-exponential model with $\alpha = 0.8$, $\Delta = 1$ and $T = 4 \Rightarrow$ four intervals. The first moment of the departure count over the four intervals is given by $D = [3,\ 8,\ 4,\ 10]$ as shown in Figure 3.9.

Figure 3.9: Example 3.5.2



Using equations (3.59) we obtain the moments over the intervals shown in Table 3.17

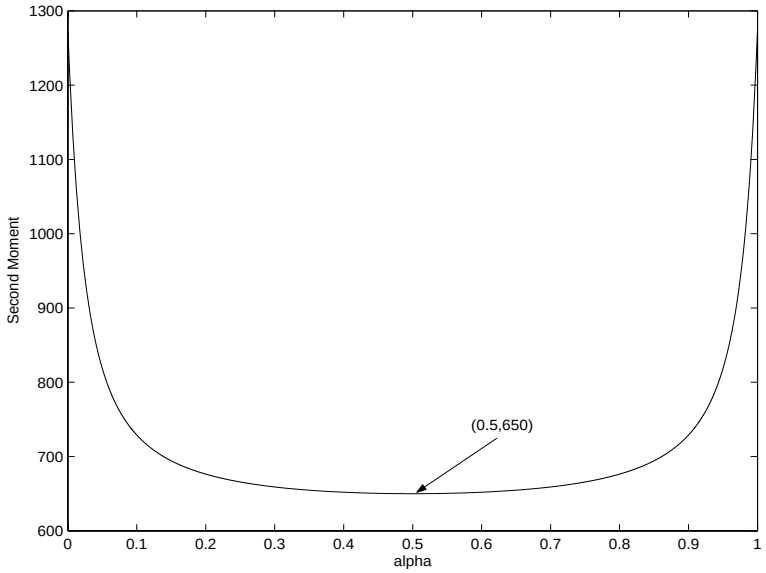
Table 3.17: Moments

	x_1	x_2	$x_1 + x_2$	y
$[0, \Delta)$	1.9000	1.1000	3	13.8749
$[0, 2\Delta)$	5.9683	5.0317	11	142.6187
$[0, 3\Delta)$	7.9687	7.0313	15	255.1173
$[0, 4\Delta)$	12.9687	12.0313	25	676.3672

The second moment of the departure count produced for $\alpha = 0.8$ is 676.3672. In our fitted Ph_t we need to find α given the set of departure-count first moments over the four FMIs

and the departure-count second moment over the SMI. Figure 3.10 is a plot of α versus the departure-count second moment.

Figure 3.10: α vs SM



When $\alpha = 0.5$ the hyper-exponential distribution becomes exponential. As expected α is symmetric about 0.5 and is increasing as it moves away from 0.5 towards 0 or 1. So if we are looking for an α that corresponds to a departure-count second moment, it is sufficient to perform a simple search on the plot shown in Figure 3.10 on the interval $[0.5, 1)$. The search we implemented halves the search interval until α is within an interval of length ϵ. For example, the value of α that satisfies the departure-count first moments $D = [3,\ 8,\ 4,\ 10]$ over the FMIs and a departure-count second moment of 700 for the SMI is 0.8612.

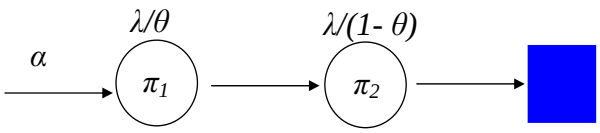
3.5.3 Approximating a Count Process with Low Variability

Consider a departure-count process to have *low variability* over an SMI if $\mathrm{E}[D_t^2(t+T)] < \mathrm{E}[D_t(t+T)]^2 + \mathrm{E}[D_t(t+T)]$. For low variability SMIs we choose three Erlang distributions of orders 2, 3, and 4 in Section 3.5.3. We select the Erlang distributions to approximate the interval-size distribution because the Erlang distribution attain the lowest coefficient of variation among all Ph distributions having n phases. When the rates of the Erlang distribution are all equal then the coefficient of variation attains its minimum which is $n^{-1/2}$ and is not a function of the rate. The balanced-means hyperexponential distribution discussed in Section 3.5.2 can achieve a second moment larger than what would be achieved if the process were to be Poisson process. In Section 3.5.3 we achieve a second moment which is less than that of a Poisson process. The Erlang-4 distribution covers the second moment range that can be achieved by the Erlang-3 and -2 distributions. The more the phases used in representing an Erlang distribution the larger the second-moment range. The tradeoff is that more phases used means more computation.

Erlang-2

The Erlang distribution we consider is represented by Figure 3.11.

Figure 3.11: Erlang-2



Only three differential equations are required to obtain the first two moments, and they simplify to (3.62). An important characteristic of this count process is that the rate depends only on λ. So we adjust λ to fit the required first moment of the departure count over the FMIs, ($\lambda(t) = \mathrm{E}[D_{(j-1)\Delta}(j\Delta)]/\Delta$ for $t \in [(j-1)\Delta, j\Delta)$). Then we adjust θ to satisfy the second moment of the departure count over the SMIs by performing a search. The phase-state probabilities are a function of θ and are given by $\pi_1 = \theta$ and $\pi_2 = 1 - \theta$.

Let $x_1(\tau) = \mathrm{E}[D; 1]$, $x_2(\tau) = \mathrm{E}[D; 2]$ and $y(\tau) = \mathrm{E}[D^2]$

$$\begin{aligned} x_1(\tau)' &= -\lambda\theta^{-1}x_1(\tau) + \lambda(1-\theta)^{-1}x_2(\tau) + \lambda \\ x_2(\tau)' &= -\lambda(1-\theta)^{-1}x_2(\tau) + \lambda\theta^{-1}x_1(\tau) \\ y(\tau)' &= 2\lambda(1-\theta)^{-1}x_2(\tau) + \lambda \end{aligned} \tag{3.62}$$

Let $X_1(s), X_2(s)$ and $Y(s)$ be the Laplace transforms of $x_1(\tau), x_2(\tau)$ and $y(\tau)$, so

$$X_1(s) = \int_0^\infty x_1(\tau)e^{-s\tau}d\tau, X_2(s) = \int_0^\infty x_2(\tau)e^{-s\tau}d\tau$$

and

$$Y(s) = \int_0^\infty y(\tau)e^{-s\tau}d\tau,$$

thus we obtain:

$$\begin{aligned} sX_1(s) - x_1(0) &= -\lambda\theta^{-1}X_1(s) + \lambda(1-\theta)^{-1}X_2(s) + \lambda s^{-1} \\ sX_2(s) - x_2(0) &= -\lambda(1-\theta)^{-1}X_2(s) + \lambda\theta^{-1}X_1(s) \\ sY(s) - y(0) &= 2\lambda(1-\theta)^{-1}X_2(s) + \lambda s^{-1} \end{aligned} \tag{3.63}$$

Solving for $X_1(s), X_2(s)$ and $Y(s)$ in 3.63 and then taking the Laplace inverse, we obtain an analytic representation of the first two moments of the count process as a function of the

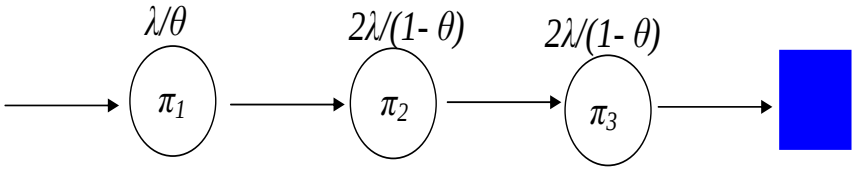
initial conditions.

$$\begin{aligned}
x_1(\tau) &= (-\theta^3+2\theta^2-\theta(1+x_2(0)+x_1(0))+x_1(0))e^{\frac{\lambda\Delta}{(\theta^2-\theta)}} \\
&\quad +\theta\lambda\Delta+\theta(x_2(0)-2\theta+x_1(0)+1+\theta^2) \\
x_2(\tau) &= -(-\theta^3+2\theta^2-\theta(1+x_2(0)+x_1(0))+x_1(0))e^{\frac{\lambda\Delta}{(\theta^2-\theta)}} \\
&\quad -\lambda(\theta-1)\Delta-(\theta-1)(\theta^2-\theta+x_2(0)+x_1(0)) \\
y(\tau) &= -2(\theta^3-2\theta^2+\theta+x_2(0)\theta+x_1(0)\theta-x_1(0))\theta e^{\frac{\lambda\Delta}{\theta^2-\theta}} \\
&\quad \lambda^2\Delta^2+\lambda(1+2x_1(0)+2x_2(0)+2\theta^2-2\theta)\Delta+2\theta^2 \\
&\quad +y(0)+2x_1(0)\theta^2-2x_1(0)\theta+2x_2(0)\theta^2+2\theta^4-4\theta^3
\end{aligned} \tag{3.64}$$

Erlang-3

We obtain the same departure moment differential for the Erlang-3:

Figure 3.12: Erlang-3



$\pi_1=\theta$, $\pi_2=\pi_3=(1-\theta)/2$.

Let $x_1(\tau)=\mathrm{E}[D;1]$, $x_2(\tau)=\mathrm{E}[D;2]$, $x_3(\tau)=\mathrm{E}[D;3]$ and $y(\tau)=\mathrm{E}[D^2]$

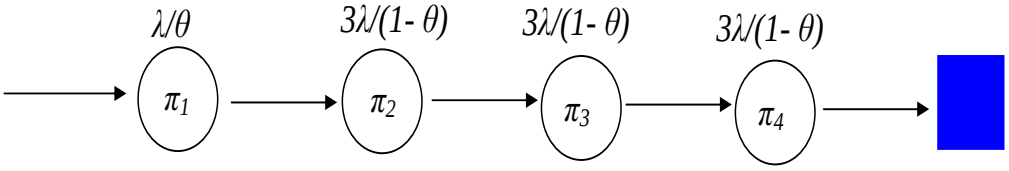
$$\begin{aligned}
x_1(\tau)' &= -\lambda(\theta)^{-1}x_1(\tau)+2\lambda(1-\theta)^{-1}x_3(\tau)+\lambda \\
x_2(\tau)' &= -2\lambda(1-\theta)^{-1}x_2(\tau)+\lambda\theta^{-1}x_1(\tau) \\
x_3(\tau)' &= -2\lambda(1-\theta)^{-1}x_3(\tau)+2\lambda(1-\theta)^{-1}x_2(\tau) \\
y(\tau)' &= 4\lambda(1-\theta)^{-1}x_3(\tau)+\lambda
\end{aligned} \tag{3.65}$$

$$
\begin{aligned}
sX_1(s) - x_1(0) &= -\lambda\theta^{-1}X_1(s) + 2\lambda(1-\theta)^{-1}X_3(s) + \lambda s^{-1} \\
sX_2(s) - x_2(0) &= -\lambda(1-\theta)^{-1}X_2(s) + \lambda\theta^{-1}X_1(s) \\
sX_3(s) - x_3(0) &= -2\lambda(1-\theta)^{-1}X_3(s) + -2\lambda(1-\theta)^{-1}X_2(s) \\
sY(s) - y(0) &= 4\lambda(1-\theta)^{-1}X_3 + \lambda s^{-1}
\end{aligned}
\tag{3.66}
$$

Erlang-4

We obtain the same departure moment differential for the Erlang-4:

Figure 3.13: Erlang-4



$\pi_1 = \theta$, $\pi_2 = \pi_3 = \pi_4 = (1-\theta)/3$.

Let $x_1(\tau) = \mathrm{E}[D;1]$, $x_2(\tau) = \mathrm{E}[D;2]$, $x_3(\tau) = \mathrm{E}[D;3]$, $x_4(\tau) = \mathrm{E}[D;4]$ and $y(\tau) = \mathrm{E}[D^2]$

$$
\begin{aligned}
x_1(\tau)' &= -\lambda\theta^{-1}x_1(\tau) + 3\lambda(1-\theta)^{-1}x_4(\tau) + \lambda \\
x_2(\tau)' &= -3\lambda(1-\theta)^{-1}x_2(\tau) + \lambda\theta^{-1}x_1(\tau) \\
x_3(\tau)' &= 3\lambda(1-\theta)^{-1}(x_2(\tau) - x_3(\tau)) \\
x_4(\tau)' &= 3\lambda(1-\theta)^{-1}(x_3(\tau) - x_4(\tau)) \\
y(\tau)' &= 6\lambda(1-\theta)^{-1}x_4(\tau) + \lambda
\end{aligned}
\tag{3.67}
$$

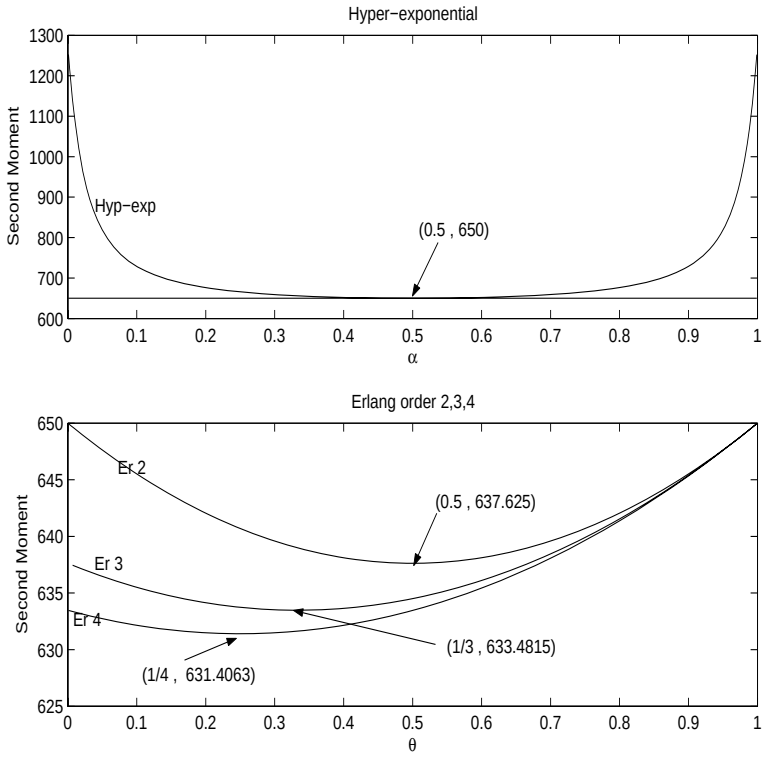
$$\begin{aligned}
sX_1(s) - x_1(0) &= -\lambda\theta^{-1}X_1(s) + 3\lambda(1-\theta)^{-1}X_2(s) + \lambda s^{-1} \\
sX_2(s) - x_2(0) &= -3\lambda(1-\theta)^{-1}X_2(s) + \lambda\theta^{-1}X_1(s) \\
sX_3(s) - x_3(0) &= -3\lambda(1-\theta)^{-1}(X_2(s) - X_3(s)) \\
sX_4(s) - x_4(0) &= -3\lambda(1-\theta)^{-1}(X_3(s) - X_4(s)) \\
sY(s) - y(0) &= 6\lambda(1-\theta)^{-1}X_4(s) + \lambda s^{-1}
\end{aligned} \tag{3.68}$$

Conclusion

Given the departure-count first moments over the FMIs, we require our approximation to cover a large range for the departure-count second moment over the SMI. Reconsider Example 3.5.2. Consider the Erlang distributions of orders 2, 3 and 4 for $\theta \in (0,1)$, and the hyper-exponential distribution for $\alpha \in (0,1)$. Figure 3.14 represents the range of second moments covered by each of these distributions.

The second-moment range of the Erlang-4 distribution is $(631.4063, 650)$ and it covers the second-moment range of the Erlang-3 and -2 distributions. The range covered by the Erlang-4 distribution is obtained for $\theta \in [1/4, 1)$. Similarly the second-moments ranges covered by the Erlang-3 and -2 distributions are obtained when $\theta \in [1/3, 1)$and $\theta \in [1/2, 1)$. So the disadvantage of using a higher order Erlang is that the search area over θ becomes larger, and the first and second moment equations require more computation. The advantage of using a higher order Erlang is that the second-moment range is larger. Reconsidering the Example 3.5.2 and using the Erlang-4 distribution along with the hyper-exponential

Figure 3.14: Second Moment Range



distribution, the second-moment range covered becomes [631.4063,1275]. Since the first moment of the number of departures over the SMI is 25, the minimum attainable second moment is 625 which occurs when the process is deterministic. So using an Erlang-4 leaves the range $[625, 631.4063)$ uncovered. Using a higher order Erlang significantly increases the computations and the range does not increase significantly.

3.6 Tandem Queues with M_t Service and Ph_t Arrival

In order to test the quality of our approximations we use the approximations to approximate the behavior of two queues in tandem with time-dependent exponential service distributions

and finite capacity. The arrival process to the second queue is approximated by a $\widetilde{Ph_t}$ distribution and the arrival process to the second node is the departure process from the first node. We present a Fitting Algorithm (FA) to obtain the approximating $\widetilde{Ph_t}$ distribution:

3.6.1 The Fitting Algorithm

Let Θ be the length of the time interval on which we perform the analysis. Let n_1 be the number of SMIs; thus $T = \Theta/n_1$. Let n_2 be the number of FMIs within an SMI; thus $\Delta = T/n_2$.

Node 1:

Step 1: Apply the closure CM algorithm or the NEDMA (Sections 3.4.9 and 3.4.4, respectively) to obtain the first and second departure-count moments over the designated intervals. At this stage the algorithm returns s_i, $f_{i,j}$ for $i = 1, \ldots, n_1$, and $j = 1, \ldots, n_2$, where s_i is the second moment of the departure count over the interval i^{th} SMI and $f_{i,j}$ is the first moment of the departure count the j^{th} FMI of the i^{th} SMI. The s and f arrays can be approximated using the closure CM algorithm or the using the NEDMA (numerically exact).

Step 2: Determine the interval distribution to be used over all the SMIs. For the i^{th} SMI:

- If $\mathrm{E}[D^2_{(i-1)T}(iT)] > 1.01\left(\mathrm{E}[D_{(i-1)T}(iT)]^2 + \mathrm{E}[D_{(i-1)T}(iT)]\right)$ then use the hyper-exponential distribution,

- if $\mathrm{E}[D^2_{(i-1)T}(iT)] < 0.99\left(\mathrm{E}[D_{(i-1)T}(iT)]^2 + \mathrm{E}[D_{(i-1)T}(iT)]\right)$, then use the Erlang-4 distribution,
- otherwise use the exponential distribution.

At this stage the algorithm returns h_i for $i = 1 \ldots n_1$, where $h_i = 1$ if over the i^{th} SMI the fitted $\widetilde{Ph_t}$ distribution is an Erlang, $h_i = 2$ if the distribution is exponential and $h_i = 3$ if the distribution is hyper-exponential.

Step 3: From s, f and h, determine the values for λ's and either α's or θ's, for each interval. The values of λ's and α's/θ's are obtained as described in Section 3.5 over the SMI.

Node 2:

The $\widetilde{Ph}$ processes constructed from Node 1 are the arrival processes to Node 2. The transition between the FMIs is smooth because the phase-state probabilities remains the same across the FMIs. The transition between SMIs is smooth when the transition is from a hyper-exponential to another hyper-exponential or from an exponential to an exponential. When the transition is from/to an Erlang or the transition involves two different distributions then the consecutive SMIs have to be *patched*, meaning the initial conditions of the number-in-system differential equations (3.48 to 3.53) have to be approximated/adjusted. To insure a smooth transition we construct the following patch:

- Transitioning from an Erlang-4 to an Erlang-4 in the i^{th} to $(i+1)^{st}$ SMI:

  Let θ_i be the parameter used if the distribution in the i^{th} SMI is Erlang for $i = 1, \ldots, n_1$.

  Let α_i be the parameter used if the distribution in the i^{th} SMI is hyper-exponential for $i = 1, \ldots, n_1$.

  Let λ_{ij} be the parameter used in the j^{th} FMI of the i^{th} SMI for $i = 1, \ldots, n_1$ and $j = 1, \ldots, n_2$.

  Assume the transition takes place between the i^{th} interval and the $(i+1)^{st}$ interval as shown in Figure 3.15. The number-in-system probability equations (3.48 and 3.49) change as follows:

  $\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = 1) = \theta_{i+1}\mathrm{P}(N(iT^-) \in \Omega_r)$ for $r = 1, 2$ and $i = 1, \ldots, n_1$.

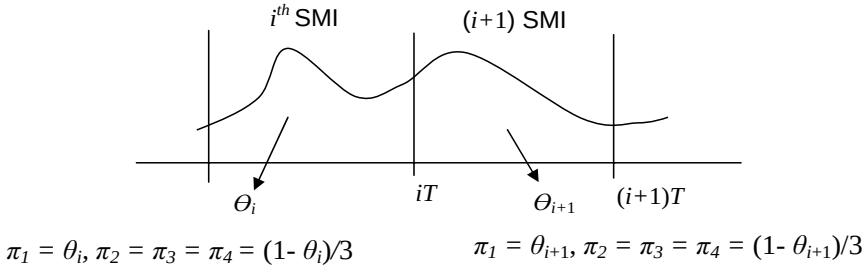
  $\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = k) = (1 - \theta_{i+1})\mathrm{P}(N(iT^-) \in \Omega_r)/3$ for $r = 1, 2,\ k = 2, 3, 4$

and $i = 1, \ldots, n_1$.

The initial conditions for the equations 3.50 to 3.53 do not change . This means that the conditional moments change, but the partial moments remain the same as we cross intervals.

$\mathrm{E}[N^j(iT^+), \Omega_r; \ell] = \mathrm{E}[N^j(iT^-), \Omega_r; \ell]$, but $\mathrm{E}[N^j(iT^+)|\Omega_r; \ell] \neq \mathrm{E}[N^j(iT^-)|\Omega_r; \ell]$ for $r = 1, 2$, $k = 1, 2, 3, 4$ and $j = 1, 2$.

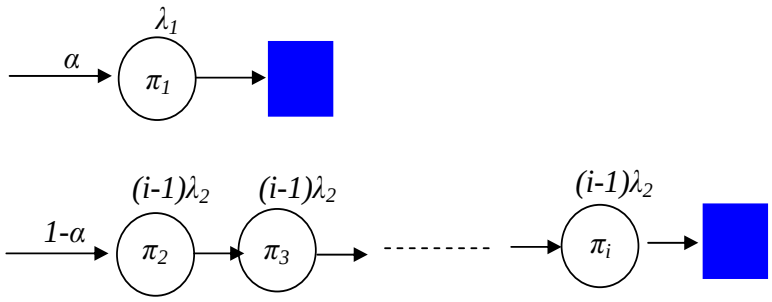
Figure 3.15: Patching the SMIs



Consider an alternate Ph distribution in Figure 3.16 for matching the first two moments of the departure counts over the SMIs for low-variability distributions. The purpose of introducing this family of Ph distributions is that they do not require patching of SMIs. The disadvantage is that it requires $n + 1$ phases to attain the second moment range attained by an Erlang with n phases.

We use the balanced-mean concept, presented in Figure 3.8, where the probability of starting in the first phase multiplied by the expected time spent before being absorbed (arrival occurs) is equal to the probability of starting in the second phase multiplied by the expected time until absorption. The rates of each phase is given in Figure 3.16,

Figure 3.16: Balanced Means for Low Variability Distributions



so it follows that the following stationary state-phase probabilities are equal ($\pi_2 = \pi_3 = \ldots = \pi_i$). The balanced-mean approach implies $\alpha\lambda_2 = (1-\alpha)\lambda_1$. The resulting stationary phase probabilities become $\pi_1 = 1/2$ and $\pi_2 = \pi_3 = \ldots = \pi_i = 2^{-1}(i-1)^{-1}$.

So the stationary phase-state probabilities are not functions of α, and transitions between SMIs does not require patching. We use the Erlang-4 distribution because patching the SMIs provides good approximations, but we present the balanced-mean distributions as an alternative to patching.

- Transition from an exponential/hyper-exponential to an Erlang-4:

  $\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = 1) \approx \theta_{i+1}\mathrm{P}(N(iT^-) \in \Omega_r)$ for $r = 1, 2$ and $i = 1, \ldots, n_1$.

  $\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = k) \approx (1-\theta_{i+1})\mathrm{P}(N(iT^-) \in \Omega_r)/3$ for $r = 1, 2,\ k = 2, 3, 4$ and $i = 1, \ldots, n_1$.

  $\mathrm{E}[N(iT^+), N(iT^+) \in \Omega_r; A(iT^+) = k] = \mathrm{E}[N(iT^-), N(iT^-) \in \Omega_r]\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = k)$ for $r = 1, 2,\ k = 1, 2, 3, 4$ and $i = 1, \ldots, n_1$.

  $\mathrm{E}[N^2(iT^+), N(iT^+) \in \Omega_r; A(iT^+) = k] = \mathrm{E}[N^2(iT^-), N(iT^-) \in \Omega_r]\mathrm{P}(N(iT^+) \in$

$\Omega_r; A(iT^+) = k)$ for $r = 1, 2,\ k = 1, 2, 3, 4$ and $i = 1, \ldots, n_1$.

- Transition from an exponential/Erlang-4 to an hyper-exponential:

  $\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = k) = \frac{1}{2}$ for $r = 1, 2$ and $i = 1, \ldots, n_1$.

  $\mathrm{E}[N(iT^+), N(iT^+) \in \Omega_r; A(iT^+) = k] = \frac{1}{2}\mathrm{E}[N(iT^-), N(iT^-) \in \Omega_r]$ for $r = 1, 2,$ $k = 1, 2$ and $i = 1, \ldots, n_1$.

  $\mathrm{E}[N^2(iT^+), N(iT^+) \in \Omega_r; A(iT^+) = k] = \frac{1}{2}\mathrm{E}[N^2(iT^-), N(iT^-) \in \Omega_r]$ for $r = 1, 2,$ $k = 1, 2$ and $i = 1, \ldots, n_1$.

- Transition from a hyper-exponential/Erlang-4 to an exponential:

  $\mathrm{E}[N(iT^+), N(iT^+) \in \Omega_r] = \frac{1}{2}\mathrm{E}[N(iT^-), N(iT^-) \in \Omega_r]$for $r = 1, 2$ and $i = 1, \ldots, n_1$.

  $\mathrm{E}[N^2(iT^+), N(iT^+) \in \Omega_r] = \frac{1}{2}\mathrm{E}[N^2(iT^-), N(iT^-) \in \Omega_r]$ for $r = 1, 2$ and $i = 1, \ldots, n_1$.

Notice that an important characteristic of patching is the creation of independent count intervals. When patching is performed, the count process produced by the $\widetilde{Ph_t}$ distribution in the i^{th} time interval is independent of the count process produced in the $i+1$ time interval. Patching is equivalent to resetting the state of the $\widetilde{Ph_t}$ distribution at time iT^+. The existing dependence between the arrival distribution $\widetilde{Ph_t}$ and the number-in-system at time iT^- is eliminated. So the arrival phase of the $\widetilde{Ph_t}$ process at time iT^+, is independent of the phase the $\widetilde{Ph_t}$ process at time iT^+. This independence property can be useful and will be utilized in Section 3.6.4.

3.6.2 Example 1

$$\Theta = 70, n_1 = 8, n_2 = 30$$

$$s_1 = 3, c_1 = 30, \mu_1 = 1$$

$$s_2 = 5, c_2 = 20, \mu_2 = 0.4$$

$$m_A = 4$$

$$a = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \qquad \begin{aligned} \alpha &= [1, \quad 0, \quad 0, \quad 0] \\ \lambda &= 3\,[4, \quad 3, \quad 2, \quad 6] \end{aligned}$$

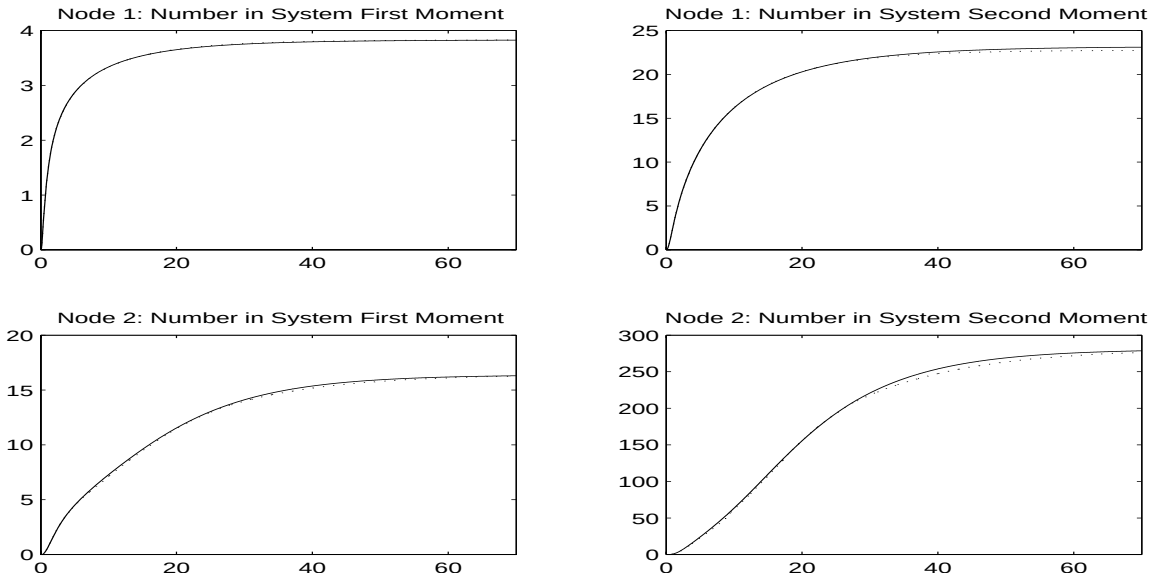
Table 3.18: Departure Second Moments

	1	2	3	4	5	6	7	8
Aprx	302.8692	433.6285	444.0490	447.9633	449.3336	449.9249	450.1582	450.2981
NE	309.2142	437.0014	447.2453	450.4879	451.7902	452.3764	452.6586	452.8006
% err	-2.0520	-0.7718	-0.7147	-0.5604	-0.5438	-0.5419	-0.5524	-0.5527
θ_i	0.2500	0.5269	0.5825	0.6210	0.6257	0.6283	0.6294	0.6298

In this example the departure process is approximated with an Erlang-4 distribution. Across the eight SMIs the algorithm produces different θ's and are presented in the last row of Table

3.18. Even though the error in approximating the second moment is relatively high (-2.05 % for the first interval), the number-in-system moment approximations are still good as can be seen in Figure 3.17 where the connected plot is obtained by solving the KFEs for the two queues.

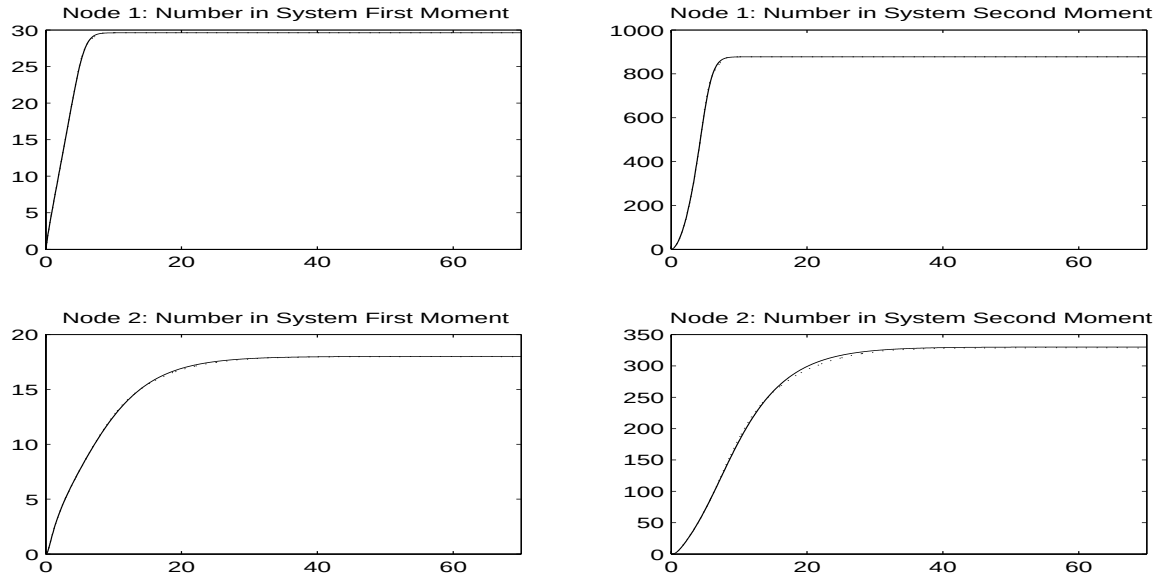
Figure 3.17: Number-in-System Moments



Increasing the arrival rate to the first node makes the departure process from the first node exponential. As expected the algorithm uses an exponential distribution to represent the arrival process to the second node. Figure 3.18 shows the moments of the number-in-system with the increased rate.

$$\lambda = 10[4, \quad 3, \quad 2, \quad 6]$$

Figure 3.18: Increased Arrival Rate



3.6.3 Example 2

$$\Theta = 70, n_1 = 8, n_2 = 30$$

$$s_1 = 3, c_1 = 30, \mu_1 = 1$$

$$s_2 = 3, c_2 = 35, \mu_2 = 1$$

$$m_A = 3$$

$$a = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \begin{aligned} \alpha &= [0.5, \quad 0.3, \quad 0.2] \\ \lambda &= [1, \quad 3, \quad 2] \end{aligned}$$

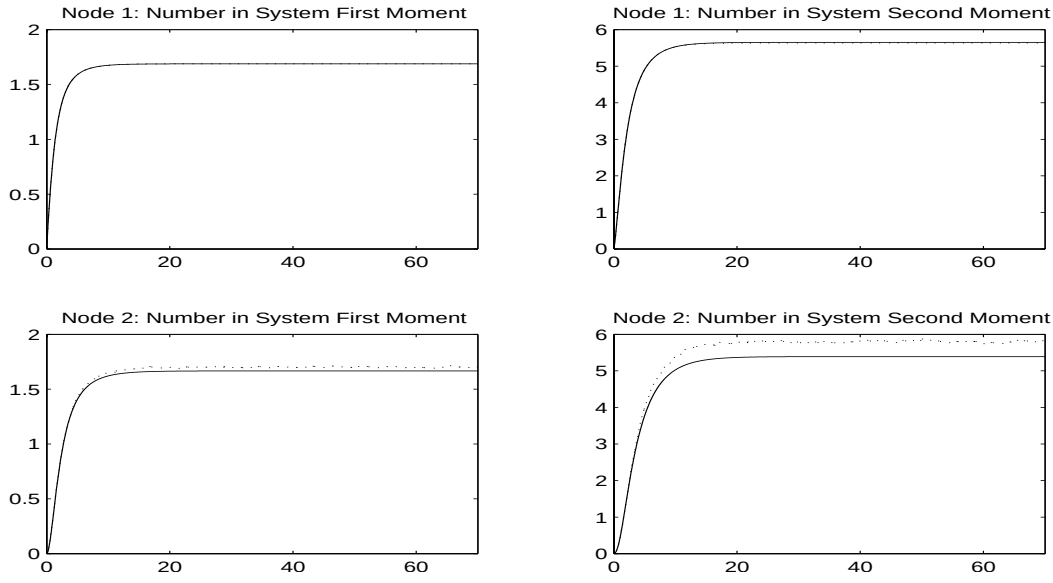
In this example the algorithm provides good approximations for the second moments of the departure counts as well as the number-in-system moments. The fitted distribution used

Table 3.19: Departure Second Moments

	1	2	3	4	5	6	7	8
Aprx	127.3514	174.5727	175.2609	174.9781	175.2951	175.3491	175.2716	175.2388
NE	125.8881	172.0192	172.5878	172.6160	172.6179	172.6180	172.6180	172.6180
% err	1.1624	1.4844	1.5488	1.3684	1.5510	1.5821	1.5372	1.5183
α	0.7203	0.7351	0.7374	0.7332	0.7375	0.7384	0.7378	0.7371

over the eight SMIs is hyper-exponential and the α's are given in the last row of Table 3.19. The plot of the moments of the number-in-system for each node are given in Figure 3.19.

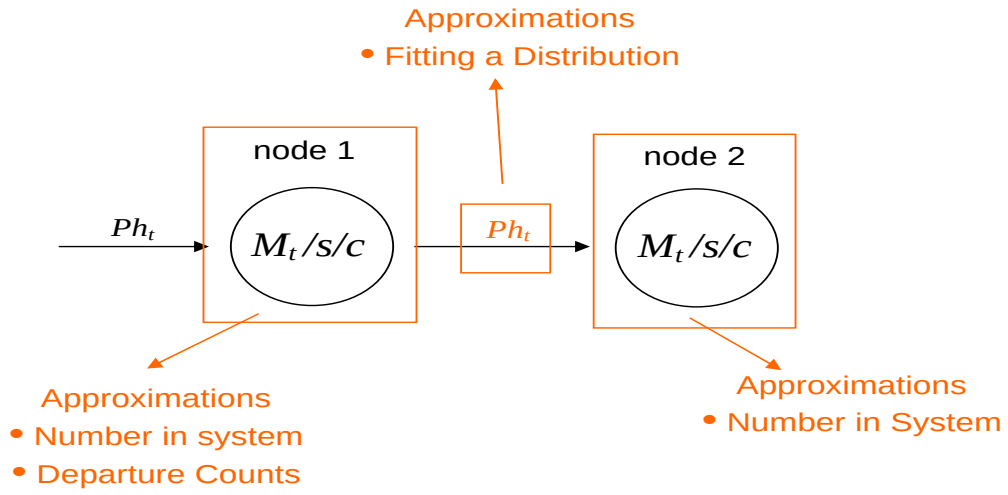
Figure 3.19: Number in System Moments



When obtaining the moments of the number-in-system at the second node, the algorithm goes through three stages as shown in Figure 3.20. Each stage involves approximations,

hence contributes to the error of the final approximation. The first stage is approximating the first and second departure-count moments from the first node over the FMIs and SMIs by using CAs on the departure process as well as the number-in-system PMDEs. The second stage is fitting a $\widetilde{Ph_t}$ across the FMIs and SMIs. The final stage is to use CAs to evaluate the number-in-system PMDEs for the second node.

Figure 3.20: Three stages of the algorithm



3.6.4 Length of the SMI

The smaller the length of the FMI, the more accurate the approximations are. So when choosing the length of the FMI, we look at the increase in computation time as the FMI length decreases. Choosing an appropriate length for the SMI is not as straight forward. Whitt in [16] introduces two methods that approximate a point process by a renewal process are the Stationary-Interval Method and the Asymptotic Method. See Chapter 1 for a review

of these methods and for some background on point processes.

In our approximations we do not deal with the time until the n^{th} departure but instead we look at the number of departures within an interval. When we increase the length of the SMI we might not be capturing the correlation between consecutive inter-departure-time intervals, which is the case in Whitt's asymptotic method. As we decrease the SMI length, we do not take into account the correlation between consecutive inter-departure-time intervals in an SMI and the inter-departure-time intervals in the next SMI. So decreasing the SMI length is comparable to using the stationary-interval method and increasing the SMI length is comparable to using the asymptotic method.

We combine both methods in our algorithm to obtain an algorithm comparable to Whitt's hybrid algorithm. Consider the following algorithm:

1. Choose T to be the largest time interval where the correlation between the number of departures within consecutive time intervals of length T is less than ϵ.

2. Find a distribution that matches the departure-count first moments over the FMIs and the departure-count second moments over the interval $[0, T/2)$.

3. Find a distribution over the interval $[T/2, T)$ that matches the departure-count first moments over the FMIs and the departure-count second moment over the interval $[0, T)$.

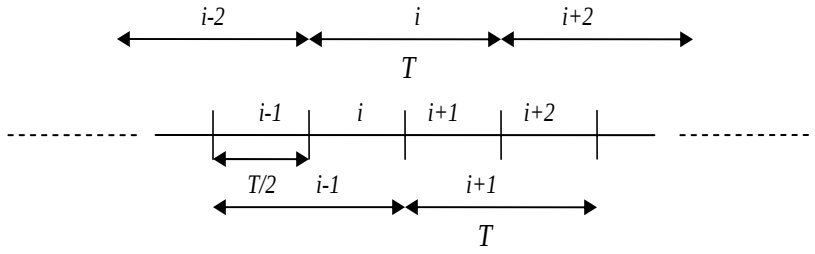
4. Find a distribution over the interval $[T, 3T/2)$ that matches the departure-count first

moments over the FMIs and the departure-count second moment over the interval $[T/2, 3T/2)$.

After i additional steps: find a distribution over the interval $[iT, iT + T/2)$ that matches the departure-count first moments over the FMIs and the departure-count second moment over the interval $[iT - T/2, iT + T/2)$.

The second moment of the departure counts is matched for over-lapping SMIs as shown in Figure 3.21 while capturing the correlation between consecutive time intervals of length $T/2$. So the algorithm moves in steps of $T/2$, but takes into consideration the second moment of the departure-count process over the previous interval of length T.

Figure 3.21: OSMIs and SMIs



Using the NEDMA or the HM closure algorithm, we obtain numerically exact results or approximations on the departure-count second moments over the overlapping second-moment intervals (OSMI) of length T. The objective now is to match the departure-count second moments over the OSMI instead of the SMIs. Applying this new hybrid method requires few changes in the algorithm presented in Section 3.5. The main difference is that the length of

the SMIs is now $T/2$ instead of T. The moments of the departure counts for the SMI over the interval $i+1$ is computed as a function of the departure-count second moment over the i^{th} OSMI as obtained by the NEDMA or the HM algorithm, and the departure-count second moment of the fitted $\widetilde{Ph_t}$ process over the i^{th} SMI.

The desired $\widetilde{Ph_t}$ process second count-moment over the $i+1$ interval is computed as follows:

Let $\widetilde{m}_i^j$ be the j^{th} moment of the number of counts obtained from the $\widetilde{Ph_t}$ distribution over the i^{th} interval.

Let $\widetilde{m}_{i,i+1}^j$ be the j^{th} moment of the number of counts obtained from the $\widetilde{Ph_t}$ distribution over the interval composed of the i^{th} and $i+1$ interval (the i^{th} OSMI).

Let $m_{i,i+1}^j$ be the desired j^{th} moment of the number of counts obtained by the NEPMDES or CAs over the i^{th} OSMI.

At the $i+1$ SMI, the objective is to match $m_{i,i+1}^2$ given $\widetilde{m}_i^2$. If $\widetilde{m}_i^2$ and $\widetilde{m}_{i+1}^2$ are independent, then :

$$\widetilde{m}_{i,i+1}^2 = m_i^2 - \widetilde{m}_i^2 - 2\widetilde{m}_i^1\widetilde{m}_{i+1}^1$$

In general $\widetilde{m}_i^2$ and $\widetilde{m}_{i+1}^2$ are not independent unless $\widetilde{Ph_t}$ is exponential or patching is performed during the transition between the i^{th} and $i+1$ SMI. For example if a hyper-exponential/Erlang-4 distribution is used in both SMIs then the count process for both SMIs will be negatively/positively correlated. This correlation is due to the residual time of the first arrival process in the $i+1$ SMI which is dependent on the number of arrivals in the i^{th} SMI. When patching is performed, the number in system is no longer dependent on

the state of the arrival phase in the previous SMI. This is equivalent to resetting the arrival process; thus the arrival process at time $(i+1)T^+$ is independent of the phase the arrival process is in at time $(i+1)T^-$.

So the changes to the algorithm in Section 3.5 are in 1) Obtaining the second moment which is used to obtain a $\widetilde{Ph_t}$ over the SMIs of length $T/2$, 2) Perform patching at the end of every SMI.

It is also possible to move in smaller steps of T/i where $i = 3, 4, 5, \ldots$ Note that if it might not always be possible to determine the smallest interval where the number of departures in consecutive intervals can be assumed independent. So a suggestion would be to choose a T which is large enough to ensure independence between the consecutive intervals and then increase i.

3.6.5 Superposing Independent Processes

Superposing independent departure-count processes is straight forward. For example let D_1 and D_2 be the departures-count of two independent processes over an interval. Let D be the superposition of the two processes $\Rightarrow D = D_1 + D_2$, $\mathrm{E}[D] = \mathrm{E}[D_1] + \mathrm{E}[D_2]$ and $\mathrm{E}[D^2] = \mathrm{E}[D_1^2] + \mathrm{E}[D_2^2] + 2\mathrm{E}[D_1]\mathrm{E}[D_2]$.

Consider a process where the arrivals/departure are routed to node i with probability p and node j with probability $1-p$. Let A be the number of arrivals over a time interval and let A_i and A_j be the number of arrivals to nodes i and j, respectively.

$\mathrm{E}[A_i] = p\mathrm{E}[A]$

$\mathrm{E}[A_j] = (1-p)\mathrm{E}[A]$

$\mathrm{E}[A_i^2] = \sum_{n=1}^{\infty} np(np-p+1)\mathrm{P}(A=n) = p^2\mathrm{E}[A^2] - (1-p)p\mathrm{E}[A].$

$\mathrm{E}[A_j^2] = (1-p)^2\mathrm{E}[A^2] - (1-p)p\mathrm{E}[A].$

3.6.6 Conclusion

Consider the three stages of the algorithm shown in Figure 3.20. Each stage contributes an error to the final number in system approximations at node 2. It is possible to eliminate the first and third sources of error by running the number-in-system KFEs and the departure-count NEPMDEs instead of using CAs on the PMDEs. For example consider an n node network with M_t service time and M_t arrival times from the outside world. The capacity of each of the nodes is c. The number of states for the entire network is $(c+1)^n$ which is the number of number-in-system KFEs. If we consider each node individually by only eliminating the first and third stages, then the number of number-in-system KFEs and departure-count NEPMDEs needed is less than or equal to $12(c+1)n$ (depending whether the approximating $\widetilde{Ph_t}$ at each node is Erlang, hyper-exponential or exponential). If the user still can not afford to run $12(c+1)n$ differential equations then CAs are recommended and the number of differential equations becomes $\leq (9+12s)n$ where s is the number of servers.

The algorithms presented in this chapter could be applied on several queues in tandem or on a network. If the network has a tree structure then it is possible to solve each node

independently, otherwise each node can be analyzed separately across the SMIs.

Chapter 4

$Ph_t/Ph_t/s/c$ Departure Count Process and Queues in Tandem

4.1 Introduction

In this Chapter we consider the departure process from a queueing system with general service and interarrival times, multiple servers and finite capacity. Our analysis focuses on transient analysis and time-dependent systems. We consider moments of the departure-counts over any chosen time interval. We choose to analyze the departure-count process and obtain its key characteristics and then fit an approximating distribution to those characteristics. See Chapter 3 for a discussion of our choice of focus on the departure-count process versus the inter-departure-time intervals in developing our approximations.

Our analysis of queueing nodes is based on representing the departure-count process and the number-in-system with a Markovian Process which is achieved by approximating or representing a queueing system with a corresponding $Ph_t/Ph_t/s/c$ system. See Chapters 2 and 3 for a discussion on the advantages and motivations behind using a Markovian representation (MR) to represent a queueing node. The $Ph_t/Ph_t/s/c$ number-in-system KFEs are presented in Appendix 1 in Equations 5.1 and 5.2. We use the $Ph_t/Ph_t/s/c$ system notation presented in Chapter 2 and we also refer the reader to [14] and [10].

In considering and constructing approximations for the departure process from a $Ph_t/Ph_t/s/c$ queueing node we augment the state space to include representation of the departure count in the interval $[t, t+\tau)$, analogous to what was done in Section 3.4.1 for the $Ph_t/M_t/s/c$ node.

4.1.1 Closure Approximations on the Number-in-System PMDEs of the $Ph_t/Ph_t/s/c$ System

In the next section we present the departure-count PMDEs in Equations 4.8 to 4.13 and provide closure approximations in Section 4.4.2. On the RHS of the departure-count PMDEs are number-in-system moments and probabilities that can be solved for numerically by closing the number-in-system PMDEs.

As a prelude, in this section we present a detailed re-statement of the number-in-system PMDEs closure approximation as presented in [14] since the number-in-system closure ap-

proximations are used in constructing our closure approximation for $Ph_t/Ph_t/s/c$ departure-count PMDEs.

The number-in-system PMDEs are derived from the number-in-system KFEs. Define

$$\mathrm{P}_{n_1 \cdots n_{m_B};\ell,q}(t) \equiv \mathrm{P}\left(N_1(t) = n_1, N_2(t) = n_2, \ldots, N_{m_B}(t) = n_{m_B}, A(t) = \ell, Q(t) = q\right)$$

for $\ell = 1, 2, \ldots, m_A$, m_A is the number of phases in the arrival process, and m_B is the number of service phases.

Let the total number-in-system be $N(t) \equiv \sum_{i=0}^{m_B} N_i(t) + Q(t)$. Let Ω be the state space for $\{N(t) : t > 0\}$ and partition Ω into two disjoint subspaces $\Omega \equiv \Omega_1 \cup \Omega_2$ where $\Omega_1 \equiv \{0, 1, \ldots, s-1\}$ and $\Omega_2 \equiv \{s, s+1, \ldots, c\}$.

The number-in-system PMDEs are presented in Appendix 2 in Equations 6.1 to 6.8. The p^{th} moments are derived from the number-in-system KFEs by summing the number-in-system probabilities as follows:

Subspace Ω_1:

$$\mathrm{E}'[N_i^p(t), \ell, 1] \equiv \mathrm{E}'[N_i^p(t), A(t) = \ell, N(t) \in \Omega_1]$$
$$= \sum_{n_1=0}^{s-1} \cdots \sum_{n_{m_B}=0}^{s-1-n_1-\ldots-n_{m_B-1}} n_i^p P'_{n_1 \ldots n_{m_B};\ell,0}(t)$$

Subspace Ω_2:

$$\mathrm{E}'[N^p(t), \ell, 2] \equiv \mathrm{E}'[N^p(t), A(t) = \ell, N(t) \in \Omega_2]$$
$$= \sum_{n_1=0}^{s} \cdots \sum_{n_{m_B-1}=0}^{s-n_1-\ldots-n_{m_B-2}} \sum_{q=0}^{c-s} (s+q)^p P'_{n_1 \ldots n_{m_B-1}, n_{m_B}=s-n_1-\ldots-n_{m_B-1};\ell,q}(t)$$

From the summations, Equations 6.1 to 6.8 can be obtained, see [14] for the details of the derivation and proof.

The terms appearing on the RHS of the number-in-system PMDEs that need to be approximated at each iteration of the numerical integration to obtain closure are the number-in-system probabilities and the partial moments. These terms are:

$$P\left(N\left(t\right)=s-1,A\left(t\right)=\ell\right),\quad P\left(N\left(t\right)=s,A\left(t\right)=\ell\right),\quad P\left(N\left(t\right)=c,A\left(t\right)=\ell\right),$$

$$E\left[N_i^p\left(t\right),N\left(t\right)=k,A\left(t\right)=\ell\right],\quad E\left[N_i^p\left(t\right)N_j\left(t\right),N\left(t\right)=k,A\left(t\right)=\ell\right],$$

$$E\left[N_i^p\left(t\right),I\left(t\right)=2,A\left(t\right)=\ell\right],\quad E\left[N_i\left(t\right)N_j\left(t\right),I\left(t\right)=2,A\left(t\right)=\ell\right]$$

for $i=1,\cdots,m_B$, $j=1,\cdots,m_B$, $\ell=1,\cdots,m_A$, $p=1,2,3$ and $k=s-1,s,c$.

The probability terms are approximated with PE probabilities where the parameters of the PE distributions are dependent on the conditional moments of the number-in-system, conditioned on the system being in subspace Ω_1 or Ω_2. Chapter 2 as well as [3] and [14] contain a detailed description of how PE distributions are used to close the number-in-system PMDEs.

The partial moment terms that need to be approximated require another type of approximation in order to fully close the entire set of number-in-system PMDEs. Rueda and Taaffe [14] use the multinomial (MN) distribution as a means to approximate partial moment terms appearing on the RHS of the number-in-system PMDEs . The need for the MN approximation arises when approximating $\mathrm{E}[N_i(t), N(t) \in \Omega_2]$ for $i = 1 \ldots m_B$ where $N_i(t)$ is the

number in the i^{th} service phase at time t. Notice that for $N(t) \in \Omega_2$ it is necessarily the case that $\sum_{i=1}^{m_B} N_i(t) = s$.

Rueda and Taaffe approximate the probability of any particular arrangement of the s entities being served across the m_B service phases with a MN distribution having s trials and m_B outcomes. Let $p_1, \ldots, p_{m_B}$ be the MN probabilities. Rueda and Taaffe [14] approximate

$$p_i \approx \frac{\mathrm{E}[N_i(t), N(t) \in \Omega_1]}{\mathrm{E}[N(t), N(t) \in \Omega_1]}. \tag{4.1}$$

In Chapter 2 we provide a detailed analysis of the performance of the number-in-system closure approximations for different $Ph_t/Ph_t/s/c$ systems.

4.2 Departure-Count Kolmogorov Forward Equations

In this section we augment the state space for the $Ph_t/Ph_t/s/c$ to include consideration of the departure-count process in a manner similar to what was done in the previous section for the $Ph_t/M_t/s/c$ system augmented to include the system departure-count process.

We use notation similar to the number-in-system KFEs notation used by Rueda and Taaffe [14] to represent the departure-count and system-state probabilities. Let $\mathrm{P}_{n_1 \cdots n_{m_B};\ell,q,d}(t^*)$ be defined as

$$\mathrm{P}\left(N_1(t^*) = n_1, N_2(t^*) = n_2, \ldots, N_{m_B}(t^*) = n_{m_B}, A(t^*) = \ell, Q(t^*) = q, D_t(t^*) = d\right)$$

for $\ell = 1, 2, \ldots, m_A$, m_A is the number of phases in the arrival process, and m_B is the number

of service phases, and $D_t(t^*)$ is the number of departures in the interval $[t, t^*)$, assuming the number of departures at time t is 0, where $t^* = t + \tau$, and $\tau > 0$.

The derivatives with respect to τ of the number-in-system augmented by the departure-count state probabilities are the departure-count KFEs and are as follows: become (derivative with respect to τ):

$$
\begin{aligned}
P_{n_1 \cdots n_{m_B};\ell,0,d,t}(t^*)' \;=\; & -\lambda_\ell(t^*) P_{n_1 \cdots n_{m_B};\ell,0,d,t}(t^*) \\
& - \sum_{i=1}^{m_B} n_i \mu_i(t^*)[1 - b_{ii}](t^*) P_{n_1 \cdots n_{m_B};\ell,0,d,t}(t^*) \\
& + \sum_{i=1}^{m_A} a_{i\ell} \lambda_i(t^*) P_{n_1 \cdots n_{m_B};\ell,0,d,t}(t^*) \\
& + \sum_{i=1}^{m_B} \delta_{[n_i>0]} \sum_{\substack{j=1 \\ j \neq 1}}^{m_B} b_{ji}(t^*)[n_j + 1] \mu_j(t^*) P_{n_1 \cdots n_{i-1} \cdots n_{j+1} \cdots n_{m_B};\ell,0,d,t}(t^*) \\
& + \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*) \alpha_\ell(t^*) \lambda_i(t^*) \left\{ \sum_{j=1}^{m_B} \delta_{[n_j>0]} \beta_j(t^*) P_{n_1 \cdots n_{j-1} \cdots n_{m_B};i,0,d,t}(t^*) \right\} \\
& + \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)[n_i + 1] \mu_i(t) P_{n_1 \cdots n_{i+1} \cdots n_{m_B};\ell,0,d-1,t}(t^*)
\end{aligned}
\tag{4.2}
$$

for $\sum_{i=1}^{m_B} n_i + q < s$, and

$$
\begin{aligned}
P_{n_1\cdots n_{m_B};\ell,q,d,t} \quad (t^*)' &= -\lambda_\ell(t^*)P_{n_1\cdots n_{m_B};\ell,q,d,t}(t^*) \\
&\quad - \sum_{i=1}^{m_B} n_i\mu_i(t^*)[1-b_{ii}](t^*)P_{n_1\cdots n_{m_B};\ell,q,d,t}(t^*) \\
&\quad + \sum_{i=1}^{m_A} a_{i\ell}\lambda_i(t^*)P_{n_1\cdots n_{m_B};\ell,q,d,t}(t^*) \\
&\quad + \sum_{i=1}^{m_B} \delta_{[n_i>0]} \sum_{\substack{j=1\\ j\neq 1}}^{m_B} b_{ji}(t^*)\delta_{[n_j<s]}[n_j+1]\mu_j(t^*)P_{n_1\cdots n_{i-1}\cdots n_{j+1}\cdots n_{m_B};\ell,q,d,t}(t^*) \\
&\quad + \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*)\Big\{(1-\delta_{q>0})\sum_{j=1}^{m_B}\delta_{[n_j>0]}\beta_j(t^*)P_{n_1\cdots n_{j-1}\cdots n_{m_B};i,q,d,t}(t^*) \\
&\qquad\qquad + \delta_{[q>0]}P_{n_1\cdots n_{j-1}\cdots n_{m_B};i,q-1,d,t}(t^*)\Big\} \\
&\quad + (1-\delta_{[q<c-s]})\sum_{i=1}^{m_A} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*)P_{n_1\cdots n_{m_B};i,q,d,t}(t^*) \\
&\quad + \delta_{[q<c-s]}\sum_{i=1}^{m_B}\sum_{\substack{j=1\\ j\neq i}}^{m_B}\delta_{[n_j>0]}b_{i,m_B+1}(t^*) \\
&\qquad\qquad \beta_j(t^*)\delta_{[n_i<s]}[n_i+1]\mu_i(t^*)P_{n_1\cdots n_{i-1}\cdots n_{j-1}\cdots n_{m_B};\ell,q+1,d-1,t}(t^*) \\
&\quad + \delta_{[q<c-s]}\sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)\beta_i(t^*)n_i\mu_i(t^*)P_{n_1\cdots n_{m_B};\ell,q+1,d-1,t}(t^*)
\end{aligned}
\tag{4.3}
$$

for $\sum_{i=1}^{m_B} n_i + q \geq s$.

In order to numerically solve the departure-count KFEs we need the initial conditions of the number-in-system state probabilities at time t The initial conditions for the departure-count probabilities are all 0 at time t, except for $\mathrm{P}\left(D_t(t)=0\right)$, which we initialize to 1. At time t we set $P_{n_1\cdots n_{m_B};\ell,q,d=0,t}(t) \leftarrow P_{n_1\cdots n_{m_B};\ell,q}(t)$.

We also need to set a practical upper limit on the number of departures in an interval of size τ. We set this practical upper limit to be a value d_m in a manner analogous to what was

done for the $Ph_t/M_t/s/c$ system in the previous chapter.

4.3 Departure-Count Numerically Exact Partial Moment Differential Equations

The number of departure-count KFEs is $d_m \times n_s$ which can be very large (n_s is the number of states in a $Ph/Ph/s/c$ systems and is given in Equation (2.10)). In order to eliminate the need to specify d_m, the upper limit set on the number of departures within the interval, we derive the departure-count NEPMDEs from the departure-count KFEs.

In Appendix 1 we present a counting algorithm that maps each system-state in a $Ph_t/Ph_t/s/c$ model to a number between 1 and n_s where n_s is the system state number. This algorithm saves computer storage and allows for efficient evaluation of the corresponding KFEs. The counting algorithm is also used in efficiently numerically solving the NEPMDEs.

Let $t^* \equiv t + \tau$ for $\tau > 0$.

$$
\begin{aligned}
\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B} \quad ; \ell, 0]' &= -\lambda_\ell(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, 0] \\
&- \sum_{i=1}^{m_B} n_i \mu_i(t^*)[1 - b_{ii}(t^*)]\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, 0] \\
&+ \sum_{i=1}^{m_A} a_{i\ell}\lambda_i(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, 0] \\
&+ \sum_{i=1}^{m_B} \delta_{[n_i>0]} \sum_{\substack{j=1 \\ j\neq 1}}^{m_B} b_{ji}(t^*)[n_j + 1]\mu_j(t^*) \\
&\qquad \mathrm{E}[D_t(t^*), n_1 \cdots n_{i-1} \cdots n_{j+1} \cdots n_{m_B}; \ell, 0] \\
&+ \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*) \\
&\qquad \left\{ \sum_{j=1}^{m_B} \delta_{[n_j>0]}\beta_j(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{j-1} \cdots n_{m_B}; i, 0] \right\} \\
&+ \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)[n_i + 1]\mu_i(t)(\mathrm{E}[D_t(t^*), n_1 \cdots n_{i+1} \cdots n_{m_B}; \ell, 0] \\
&+ P_{n_1 \cdots n_{m_B}; \ell, 0}(t^*)),
\end{aligned}
\tag{4.4}
$$

and

$$
\begin{aligned}
\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B} \quad ; \ell, 0]' &= -\lambda_\ell(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, 0] \\
&- \sum_{i=1}^{m_B} n_i \mu_i(t^*)[1 - b_{ii}(t^*)]\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, 0] \\
&+ \sum_{i=1}^{m_A} a_{i\ell}\lambda_i(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, 0] \\
&+ \sum_{i=1}^{m_B} \delta_{[n_i>0]} \sum_{\substack{j=1 \\ j\neq i}}^{m_B} b_{ji}(t^*)[n_j + 1]\mu_j(t^*) \\
&\qquad \mathrm{E}[D_t^2(t^*), n_1 \cdots n_{i-1} \cdots n_{j+1} \cdots n_{m_B}; \ell, 0] \\
&+ \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*) \\
&\qquad \left\{ \sum_{j=1}^{m_B} \delta_{[n_j>0]}\beta_j(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{j-1} \cdots n_{m_B}; i, 0] \right\} \\
&+ \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)[n_i + 1]\mu_i(t)(\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{i+1} \cdots n_{m_B}; \ell, 0] \\
&+ 2\sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)[n_i + 1]\mu_i(t)(\mathrm{E}[D_t(t^*), n_1 \cdots n_{i+1} \cdots n_{m_B}; \ell, 0] \\
&+ P_{n_1 \cdots n_{m_B}; \ell, 0}(t^*)
\end{aligned}
\tag{4.5}
$$

for $\sum_{i=1}^{m_B} n_i + q < s$, and

$\mathrm{E}[D_t^2(t+\tau), n_1 \cdots n_{m_B}; \ell, q]$

$$
\begin{aligned}
&\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, q]' = -\lambda_\ell(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, q] \\
&- \sum_{i=1}^{m_B} n_i \mu_i(t^*)[1 - b_{ii}(t^*)]\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, q] \\
&+ \sum_{i=1}^{m_A} a_{i\ell}\lambda_i(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, q] \\
&+ \sum_{i=1}^{m_B} \delta_{[n_i>0]} \sum_{\substack{j=1 \\ j\neq 1}}^{m_B} b_{ji}(t^*)\delta_{[n_j<s]}[n_j+1]\mu_j(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{i-1} \cdots n_{j+1} \cdots n_{m_B}; \ell, q] \\
&+ \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*) \\
&\qquad\qquad \left\{ (1-\delta_{q>0}) \sum_{j=1}^{m_B} \delta_{[n_j>0]}\beta_j(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{j-1} \cdots n_{m_B}; i, q] \right\} \\
&+ \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*) \left\{ \delta_{[q>0]}\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; i, q-1] \right\} \qquad , \\
&+ (1-\delta_{[q<c-s]}) \sum_{i=1}^{m_A} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; i, q] \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} \sum_{\substack{j=1 \\ j\neq i}}^{m_B} \delta_{[n_j>0]} b_{i,m_B+1}(t^*)\beta_j(t^*) \\
&\qquad\qquad \delta_{[n_i<s]}[n_i+1]\mu_i(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{i-1} \cdots n_{j-1} \cdots n_{m_B}; \ell, q+1] \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)\beta_i(t^*)n_i\mu_i(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, q+1] \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} \sum_{\substack{j=1 \\ j\neq i}}^{m_B} \delta_{[n_j>0]} b_{i,m_B+1}(t^*)\beta_j(t^*)\delta_{[n_i<s]}[n_i+1]\mu_i(t^*)P_{n_1 \cdots n_{i-1} \cdots n_{j-1} \cdots n_{m_B}; \ell, q+1}(t^*) \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)\beta_i(t^*)n_i\mu_i(t^*)P_{n_1 \cdots n_{m_B}; \ell, q+1}(t^*)
\end{aligned}
\tag{4.6}
$$

and

$$
\begin{aligned}
&\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, q]' = -\lambda_\ell(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, q] \\
&- \sum_{i=1}^{m_B} n_i \mu_i(t^*)[1 - b_{ii}(t^*)]\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, q] \\
&+ \sum_{i=1}^{m_A} a_{i\ell}\lambda_i(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, q] \\
&+ \sum_{i=1}^{m_B} \delta_{[n_i>0]} \sum_{\substack{j=1 \\ j\neq 1}}^{m_B} b_{ji}(t^*)\delta_{[n_j<s]}[n_j+1]\mu_j(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{i-1} \cdots n_{j+1} \cdots n_{m_B}; \ell, q] \\
&+ \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*) \\
&\qquad\qquad \left\{ (1-\delta_{q>0}) \sum_{j=1}^{m_B} \delta_{[n_j>0]}\beta_j(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{j-1} \cdots n_{m_B}; i, q] \right\} \\
&+ \sum_{i=1}^{m_B} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*) \left\{ \delta_{[q>0]}\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; i, q-1] \right\} \\
&+ (1-\delta_{[q<c-s]}) \sum_{i=1}^{m_A} a_{i,m_A+1}(t^*)\alpha_\ell(t^*)\lambda_i(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; i, q] \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} \sum_{\substack{j=1 \\ j\neq i}}^{m_B} \delta_{[n_j>0]} b_{i,m_B+1}(t^*)\beta_j(t^*) \\
&\qquad\qquad \delta_{[n_i<s]}[n_i+1]\mu_i(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{i-1} \cdots n_{j-1} \cdots n_{m_B}; \ell, q+1] \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)\beta_i(t^*)n_i\mu_i(t^*)\mathrm{E}[D_t^2(t^*), n_1 \cdots n_{m_B}; \ell, q+1] \\
&+ 2\delta_{[q<c-s]} \sum_{i=1}^{m_B} \sum_{\substack{j=1 \\ j\neq i}}^{m_B} \delta_{[n_j>0]} b_{i,m_B+1}(t^*)\beta_j(t^*) \\
&\qquad\qquad \delta_{[n_i<s]}[n_i+1]\mu_i(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{i-1} \cdots n_{j-1} \cdots n_{m_B}; \ell, q+1] \\
&+ 2\delta_{[q<c-s]} \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)\beta_i(t^*)n_i\mu_i(t^*)\mathrm{E}[D_t(t^*), n_1 \cdots n_{m_B}; \ell, q+1] \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} \sum_{\substack{j=1 \\ j\neq i}}^{m_B} \delta_{[n_j>0]} b_{i,m_B+1}(t^*)\beta_j(t^*)\delta_{[n_i<s]}[n_i+1]\mu_i(t^*)\mathrm{P}_{n_1 \cdots n_{i-1} \cdots n_{j-1} \cdots n_{m_B}; \ell, q+1}(t^*) \\
&+ \delta_{[q<c-s]} \sum_{i=1}^{m_B} b_{i,m_B+1}(t^*)\beta_i(t^*)n_i\mu_i(t^*)\mathrm{P}_{n_1 \cdots n_{m_B}; \ell, q+1}(t^*)
\end{aligned}
\tag{4.7}
$$

for $\sum_{i=1}^{m_B} n_i + q \geq s$.

The number of differential equations is $3m_B^{-1}m_A(s+m_B(c-s+1))\binom{m_B+s-1}{s}$ which is still too large for practical use but is acceptable for the one-time purpose of obtaining numerically exact solutions to use for evaluating the quality of our approximations. Notice that the number-in-system KFEs are a part of the departure-count NEPMDEs. So at time t we need to initialize the number-in-system KFE probabilities. At time t the initial values for the departure-count NEPMDEs is zero since $D_t(t) = 0$.

To numerically solve the $Ph_t/Ph_t/s/c$ departure-count NEPMDEs moments over any chosen set of n time intervals, we use the NEDMA algorithm presented in Section 3.4.4 where we substitute the $Ph_t/M_t/s/c$ NEPMEDs/KFEs with the $Ph_t/Ph_t/s/c$ NEPMDEs/KFEs. We refer to this algorithm as the $Ph_t/Ph_t/s/c$ NEDMA.

4.4 Departure Process PMDEs

Using the NEPMDEs we derive a smaller set of departure-count PMDEs. Let

$$\mathrm{E}[D_\tau(t+\tau), N(t+\tau) \in \Omega_1; A(t+\tau) = \ell] = \mathrm{E}[D, \Omega_1; \ell],$$

for $\ell = 1, \ldots, m_A$, $\Omega_1 = \{0, 1, \ldots, s-1\}$ and $\Omega_2 = \{s, \ldots, c\}$. For the two subspaces, the p^{th} departure-count PMDEs over the time interval $[t, t+\tau)$ are derived by the following summations: for Subspace Ω_1

$$\mathrm{E}'\left[D^p\left(t\right), A\left(t\right) = \ell, N(t) \in \Omega_1\right] \equiv \sum_{n_1=0}^{s-1} \ldots \sum_{n_{m_B}=0}^{s-1-n_1-\ldots-n_{m_B-1}} \mathrm{E}'[D^p, n_1 \ldots n_{m_B}; \ell, 0],$$

and for Subspace Ω_2

$$\mathrm{E}'\left[D^p\left(t\right), A\left(t\right) = \ell, N(t) \in \Omega_2\right] \equiv$$

$$\sum_{n_1=0}^{s} \ldots \sum_{n_{m_B-1}=0}^{s-n_1-\ldots-n_{m_B-2}} \sum_{q=0}^{c-s} \mathrm{E}'[D^p, n_1 \ldots, n_{m_B-1}, n_{m_B} = s - n_1 - \ldots - n_{m_B-1}; \ell, q].$$

For example for $p = 1$:

$$\begin{aligned}
\mathrm{E}'[D, \Omega_1; \ell] &= -\lambda_\ell \mathrm{E}[D, \Omega_1; \ell] - \sum_{i=1}^{m_A} a_{i,m_A+1} \lambda_i \alpha_\ell \mathrm{E}[D, s-1; i] \\
&\quad + \sum_{i=1}^{m_A} a_{i,\ell} \lambda_i \mathrm{E}[D, \Omega_1; i] + \sum_{i=1}^{m_A} a_{i,m_A+1} \lambda_i \alpha_\ell \mathrm{E}[D, \Omega_1; i] \\
&\quad + \sum_{i=1}^{m_B} b_{i,m_B+1} \mu_i \mathrm{E}[DN_i, s; \ell] + \sum_{i=1}^{m_B} b_{i,m_B+1} \mu_i \mathrm{E}[N_i, \Omega_1; \ell] \\
&\quad + \sum_{i=1}^{m_B} b_{i,m_B+1} \mu_i \mathrm{E}[N_i, s; \ell],
\end{aligned} \tag{4.8}$$

and

$$\begin{aligned}
\mathrm{E}'[D, \Omega_2; \ell] &= -\lambda_\ell \mathrm{E}[D, \Omega_2; \ell] - \sum_{i=1}^{m_B} b_{i,m_B+1} \mu_i \mathrm{E}[DN_i, s; \ell] \\
&\quad - \sum_{i=1}^{m_B} b_{i,m_B+1} \mu_i \mathrm{E}[N_i, s; \ell] + \sum_{i=1}^{m_A} a_{i,\ell} \lambda_i \mathrm{E}[D, \Omega_2; i] \\
&\quad + \sum_{i=1}^{m_A} a_{i,m_A+1} \lambda_i \alpha_\ell \mathrm{E}[D, \Omega_2; i] + \sum_{i=1}^{m_B} b_{i,m_B+1} \mu_i \mathrm{E}[N_i, \Omega_2; \ell] \\
&\quad + \sum_{i=1}^{m_A} a_{i,m_A+1} \lambda_i \alpha_\ell \mathrm{E}[D, s-1; i].
\end{aligned} \tag{4.9}$$

These PMDEs across the two subspaces can be combined to yield the intuitive expression:

$$\mathrm{E}'[D] = \sum_{i=1}^{m_B} b_{i,m_B+1} \mu_i \mathrm{E}[N_i] \tag{4.10}$$

Although Equation 4.10 provides a simple and intuitive form for the departure-count first moments, Equations 4.8 and 4.9 are needed to form a closure approximation for departure-

count second moments.

$$\begin{aligned} \mathrm{E}'[D^2;\ell] &= -\lambda_\ell \mathrm{E}[D^2;\ell] + \sum_{i=1}^{m_A} a_{i,\ell}\lambda_i \mathrm{E}[D^2;i] \\ &\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}\lambda_i\alpha_\ell \mathrm{E}[D^2;i] \\ &\quad + \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i \mathrm{E}[N_i;\ell] \\ &\quad + 2\sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i \mathrm{E}[DN_i;\ell] \end{aligned} \tag{4.11}$$

$$\Rightarrow \mathrm{E}'[D^2] = \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i \mathrm{E}[N_i] + 2\sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i \mathrm{E}[DN_i], \tag{4.12}$$

and this equation requires $\mathrm{E}[DN_i,\Omega_1;\ell]$; thus we construct the following differential equation:

$$\begin{aligned} \mathrm{E}'[DN_i,\Omega_1;\ell] &= -\lambda_\ell \mathrm{E}[DN_i,\Omega_1;\ell] \\ &\quad -\mu_i \mathrm{E}[DN_i,\Omega_1;\ell] - \sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j \mathrm{E}[DN_i,s-1;j] \\ &\quad - \sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j\beta_i \mathrm{E}[D,s-1;j] - b_{i,m_B+1}\mu_i \mathrm{E}[DN_i,s;\ell] \\ &\quad -b_{i,m_B+1}\mu_i \mathrm{E}[N_i,\Omega_1;\ell - b_{i,m_B+1}\mu_i \mathrm{E}[N_i,s;\ell] \\ &\quad + \sum_{j=1}^{m_B} b_{ji}\mu_j \mathrm{E}[DN_j,\Omega_1;\ell] + \sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j \mathrm{E}[DN_i,\Omega_1;j] \\ &\quad + \sum_{j=1}^{m_A} a_{j,m_A+1}\alpha_\ell\lambda_j\beta_i \mathrm{E}[D,\Omega_1;j] + \sum_{j=1}^{m_A} a_{j,\ell}\lambda_j \mathrm{E}[DN_i,\Omega_1;j] \\ &\quad + \sum_{j=1}^{m_B} b_{j,m_B+1}\mu_j \mathrm{E}[DN_jN_i,s;\ell] + \sum_{j=1}^{m_B} b_{j,m_B+1}\mu_j \mathrm{E}[N_jN_i,\Omega_1;\ell] \\ &\quad + \sum_{j=1}^{m_B} b_{j,m_B+1}\mu_j \mathrm{E}[N_jN_i,s;\ell]. \end{aligned} \tag{4.13}$$

4.4.1 Observations

- The departure-count PMDEs for the $Ph_t/Ph_t/\infty$ system can be derived from Equations (4.8 to 4.13). The departure-count PMDEs for the $Ph_t/Ph_t/\infty$ are closed since

the partial moments are functions of the number-in-system being s or $(s-1)$. As $s \to \infty$ these PMDEs become closed. Nelson and Taaffe [10] also show that the number-in-system PMDEs of the $Ph_t/Ph_t/\infty$ system are closed. So it is possible to obtain numerically exact solutions for the moments of $D_t(t+\tau)$ for the infinite server case since the departure-count and number-in-system PMDEs form a set of closed PDEs.

The case of finite s requires closure approximations for the number-in-system PMDEs as well as the departure-count PMDEs. We use the closure approximations presented in [14] for the number-in-system PMDEs. We present closure approximations for the departure-count PMDEs later in this section.

- Consider the service Ph_t distribution and let phase i be a *terminating phase* if the transition probability from phase i to the absorbing phase, b_{i,m_B+1}, is greater than 0. Notice that the first moment of $D_t(t+\tau)$ is obtained from Equation 4.10 and is a function of the number of entities in terminating service phases over the time interval $[t, t+\tau)$. The second moment of $D_t(t+\tau)$ can be obtained from Equation 4.12, and is also a function of the number of entities in terminating service phases, as well as $\mathrm{E}[D_t(t+\tau)N_i(t+\tau)]$, where i represents a terminating service phase. Also observe that $\mathrm{E}[D_t(t+\tau)N_i(t+\tau)] =$

$$\mathrm{E}[D_t(t+\tau)N_i(t+\tau), N(t+\tau) \in \Omega_1] + \mathrm{E}[D_t(t+\tau)N_i(t+\tau), N(t+\tau) \in \Omega_2].$$

For a system with exponential-service time (one single service phase), the one service

phase is obviously a terminating phase and when $N(t) \in \Omega_2$, the departure-count process $D_t(t+\tau)$ is independent of the number in the terminating service phase as can be seen below.

$$\mathrm{E}[D_t(t+\tau)N_i(t+\tau), N(t+\tau) \in \Omega_2] = s\mathrm{E}[D_t(t+\tau), N(t+\tau) \in \Omega_2].$$

- Given that the number-in-system is in the subspace Ω_2 (all servers are busy), we assume independence between the number in the terminating service phases and the number of departures in the time interval $[t, t+\tau)$. So

$$\mathrm{E}[D_t(t+\tau)N_i(t+\tau), N(t+\tau) \in \Omega_2] \approx \mathrm{E}[D_t(t+\tau), N(t+\tau) \in \Omega_2]\mathrm{E}[N_i(t+\tau)|N(t+\tau) \in \Omega_2].$$

For a system where $\mathrm{P}(N(t+\tau) \in \Omega_1)$ is small enough, one could use the following approximation:

$$\begin{aligned} \mathrm{E}'[D^2] &= \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i\mathrm{E}[N_i] + 2\sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i\mathrm{E}[DN_i] \\ &\approx \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i\mathrm{E}[N_i] + 2\mathrm{E}[D]\sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i\mathrm{E}[N_i] \\ &= (1+2\mathrm{E}[D])\sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i\mathrm{E}[N_i]; \end{aligned} \tag{4.14}$$

therefore if one were to consider a $Ph_t/Ph_t/s/c$ system having a small number of servers or $\mathrm{P}(N(t+\tau) \in \Omega_1) \to 0$, this approximation would be asymptotically exact.

In the next section we present closure approximations for the first two moments of $D_t(t+\tau)$ for a $Ph_t/Ph_t/s/c$ system.

4.4.2 Closing the Departure-Count PMDEs

Approximating the departure-count first moment is simple and is represented in Equation 4.10. The departure-count second moment differential equations are:

$$\begin{aligned} \mathrm{E}'[D^2;\ell] &= -\lambda_\ell \mathrm{E}[D^2;\ell] + \sum_{i=1}^{m_A} a_{i,\ell}\lambda_i \mathrm{E}[D^2;i] \\ &\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}\lambda_i \alpha_\ell \mathrm{E}[D^2;i] \\ &\quad + \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i \mathrm{E}[N_i;\ell] \\ &\quad + 2\sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i \mathrm{E}[DN_i;\ell] \end{aligned} \tag{4.15}$$

The last term on the RHS is

$$\begin{aligned} \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i \mathrm{E}[DN_i;\ell] &= \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i (\mathrm{E}[DN_i,\Omega_1;\ell] + \mathrm{E}[DN_i,\Omega_2;\ell]) \\ &\approx \sum_{i=1}^{m_B} b_{i,m_B+1}\mu_i (\mathrm{E}[DN_i,\Omega_1;\ell] + \mathrm{E}[D,\Omega_2;\ell]\mathrm{E}[N_i|\Omega_2;\ell]). \end{aligned}$$

The term $\mathrm{E}[DN_i,\Omega_1;\ell]$ is represented in Equation 4.13, and $\mathrm{E}[D,\Omega_2;\ell]$ in Equation 4.12.

Most of the terms to be approximated in the closure approximations are on the RHS of $\mathrm{E}[DN_i,\Omega_1;\ell]$, as can be seen in Equation 4.13. The terms being approximated are either number-in-system moments/probabilities at time $t+\tau$ or departure-count moments over the time interval $[t,t+\tau)$.

To numerically solve the number-in-system differential equations for the time interval $[t,t+\tau)$, the initial conditions at time t are needed. The initial conditions at time t are obtained by solving the number-in-system PMDEs from time 0 to time t. The initial conditions for the departure-count PMDEs are set to zero at time t by our definition of $D_t\,(t,t+\tau)$.

The terms appearing on the RHS of equations 4.8 to 4.13 that need to be approximated at every iteration in order to obtain closure are:

$\mathrm{E}[D, N = s-1; \ell]$, $\mathrm{E}[D, N = s; \ell]$,$\mathrm{E}[DN_i, N = s-1; \ell]$, $\mathrm{E}[DN_i, N = s; \ell]$, and $\mathrm{E}[DN_iN_j, N = s; \ell]$, for $i = 1 \ldots m_B$, and $j = 1 \ldots m_B$.

We approximate :

$$\mathrm{E}\left[DN_i, N = s-1; \ell\right] \approx \mathrm{E}\left[N_i, N = s-1; \ell\right] \mathrm{E}\left[D | N = s-1; \ell\right],$$

$$\mathrm{E}\left[DN_i, N = s; \ell\right] \approx \mathrm{E}\left[N_i, N = s; \ell\right] \mathrm{E}\left[D | N = s; \ell\right],$$

$$\mathrm{E}\left[DN_iN_j, N = s; \ell\right] \approx \mathrm{E}\left[N_iN_j, N = s; \ell\right] \mathrm{E}\left[D | N = s; \ell\right].$$

The approximated departure-count terms above, appearing on the RHS of the departure-count PMDEs, are multiplied by the transition probability from the current service phase to the absorbing service phase. So the approximations above assume that the number of departures in the time interval $[t, t+\tau)$, conditioned on the number-in-system being s or $s-1$, is independent of number in the terminating service phases.

The remaining terms that need to be approximated are $\mathrm{E}\left[D, N = s-1; \ell\right]$ and $\mathrm{E}\left[D, N = s; \ell\right]$.

Consider the term $\mathrm{E}[D, N = s-1; \ell]$. It is possible to obtain $\mathrm{E}\left[D, N = s-1; \ell\right]$ at time $t+\tau$ from the NEPMDEs by summing all $\mathrm{E}[D, n_1 \ldots n_{m_B}; \ell, 0]$ having $N = s-1$ at time $t+\tau$. The first step in approximating $\mathrm{E}[D, N = s-1; \ell]$ is to compute the NEPMDEs for all departure-count partial-moment terms having $N < s-1$. In the RHS of every departure-count partial-moment NEPMDE, having $N = s-2$, is a departure-count partial-moment

term and a system-state probability term having $N = s - 1$. But at each iteration of the numerical integration we have an approximation of $\mathrm{E}[D, \Omega_1; \ell]$ and $\mathrm{P}(N \in \Omega_1; \ell)$ as well as the all the NEPMDEs departure-count partial-moment terms, having $N < s-2$. This allows us to approximate $\mathrm{E}[D, N = s - 1; \ell]$ by computing the difference

$$\mathrm{E}[D, \Omega_1; \ell] - \mathrm{E}[D, n_1 \dots n_{m_B}, \ell, 0],$$

for all $N < s - 1$.

The next step is to approximate $\mathrm{E}[D, n_1 \dots n_{m_B}, \ell, 0]$, for all $N = s - 1$, using $\mathrm{E}[D, N = s - 1; \ell]$.

The algorithm is summarized in the following steps:

Approximating departure-count partial moments and system-state partial probabilities having $\mathbf{N = s - 1}$: The Truncated Method

Step 1: Compute

$$h_1 = \mathrm{E}[D, N < s - 1; \ell] = \sum_{N<s-1} = \mathrm{E}[D, n_1 \cdots n_{m_B}; \ell, 0].$$

$$h_2 = \mathrm{P}(N < s - 1; \ell) = \sum_{N<s-1} \mathrm{P}_{n_1 \cdots n_{m_B}; \ell, 0}.$$

Step 2: Compute

$$\mathrm{E}[D, N = s - 1; \ell] = \mathrm{E}[D, N < s - 1; \ell] - h_1.$$

$$\mathrm{P}(N = s - 1; \ell) = \mathrm{P}(N < s; \ell) - h_2.$$

Step 3: Compute $M_{n_1 \cdots n_{m_B}} = (s-1)! \prod_{i=1}^{mb} (p_i^{n_i}/n_i!)$, where $M_{n_1 \ldots n_{m_B}}$ is the probability mass function for a multinomial random variable having $\sum_{i=1}^{m_B} n_i = s$, and $(p_1, \ldots, p_{m_B})$. The probabilities $(p_1, \ldots, p_{m_B})$ are obtained by closing the number-in-system PMDEs as shown in Equation 4.1.

Then for every term in the NEPMDEs (Equations 4.4 and 4.5) with $N = s-1$, approximate $\mathrm{P}_{n_1 \cdots n_{m_B};\ell,0} \approx \mathrm{P}_{N=s-1;\ell} M_{n_1 \cdots n_{m_B}}$

and $\mathrm{E}[D, n_1 \cdots n_{m_B}; \ell, 0] \approx \mathrm{E}[D, N = s-1; \ell] M_{n_1 \cdots n_{m_B}}$.

Step 4: Solve the Ω_1 NEPMDEs (Equation 4.4) for each moment and probability term having $N = 0 \ldots s-2$. When the system is in a state having $N = s-2$, use the multinomial approximation in Step 3 to approximate the RHS term having $N = s-1$.

Now that we have an approximation for the departure-count partial moments and system-state partial probabilities having $N = s-1$, we next approximate $\mathrm{E}[D, N = s; \ell]$ by computing

$$\mathrm{E}[D, N = s; \ell] \approx \mathrm{E}[D, N = s-1; \ell] \, \mathrm{P}(N = s) / \mathrm{P}(N = s-1).$$

This method is called the Truncated Method.

A variant of the Truncated Method, called the Independence Method, is a simpler approximation. Notice that, as stated in Section 3.4.6, as τ increases, it is also reasonable to approximate

$$\mathrm{E}[D_t(t+\tau) \,|N(t+\tau) = s-1] \approx \mathrm{E}[D_t(t+\tau) \,|N(t+\tau) \in \Omega_1],$$

$$\mathrm{E}\left[D|N = s-1;\ell\right] \approx \mathrm{E}\left[D|N \in \Omega_1;\ell\right]$$

and approximate

$$\mathrm{E}[D_t(t+\tau)|N(t+\tau) = s] \approx \mathrm{E}[D_t(t+\tau)|N(t+\tau) \in \Omega_2].$$

In our algorithm, under certain conditions, we use these approximations and we call this approach the *Independence Method*, see Chapter 3. Although the Independence Method assumes independence between the number of departures and the number-in-system, it is computationally efficient because there is no need to solve the truncated NEPMDEs. This method provides increasingly accurate approximations as τ increases and if the departure process is actually Poisson, the method gives exact results. In practice this method provided good approximations even for modest sized τ's; see Chapter 3 for a description of the accuracy of the Independence Method.

In constructing our algorithm we combine the Truncated Method and the Independence Method. We compute an indicator, γ, and when $\tau \geq \gamma$ we use the Independence Method, and when $\tau < \gamma$ we use the Truncated Method. See Chapter 3, Section 3.4.8 to see the details of determining a good value for γ. We call this approach the Combination Method algorithm. We use the Combination Method in all the remaining numerical examples in this Chapter.

4.5 $Ph_t/Ph_t/s/c$ Examples

We present five examples in this section to illustrate how well the algorithm works. In all five examples we numerically solve the NEPMDEs as well as close the departure-count PMDEs. The numerical results obtained from the NEPMDEs are plotted using solid lines and the closure approximations are plotted on the same plot using dotted lines. We test the quality of the CM algorithm for different $Ph_t/Ph_t/s/c$ and conclude that our algorithm provides good approximations on the first two departure-count moments. We present more $Ph_t/Ph_t/s/c$ systems in Appendix 4.

The time interval for which we plot the first two departure-count moments is $[15, 30)$. The number of departures within an interval is obviously dependent on the state of the number-in-system at the start of the interval. So we solve for the state of the number-in-system at time 15 by numerically solving the number-in-system KFEs and closing the number-in-system PMDEs from time 0 to 15. For each departure-count interval, the initial condition is to set the departure count random variable to 0 (and thus each moment of the departure count has initial condition equal to 0); thus, for instance, $D_{15}(15) = 0$. The first figure listed for each of the examples is a plot of the first two moments of the number-in-system across the time interval $[0, 15]$. The second figure is a plot of the number of departures in the time interval $[15, 30)$.

The plots of the departure-count moments are computed from the departure-count PMDEs as a function of τ. In the five examples plots of the first two departure-count moments when

$t = 15$ and $\tau \in (0, 15)$ are given for numerical solutions for $D_t(t + \tau)$.

Because we are interested in departure-count moments for different time intervals and for intervals of various lengths, we present summary tables for a collection of results for each example case.

4.5.1 Example 1

Consider the stationary $Ph/Ph/s/c$ case having system parameters:

$s = 5$, $c = 20$, $m_A = 3$,$m_B = 2$,

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4]$, $\qquad \boldsymbol{\beta} = [0.5, \quad 0.5]$,

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix},$$

$\boldsymbol{\lambda} = [3, \quad 3, \quad 2]$, and $\boldsymbol{\mu} = [1, \quad 0.5]$.

Consider the four points of time $p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$. Tables 8.39 and 8.40 contain the first two departure-count moments over the intervals defined by these four time points, using closure approximations and the numerically exact NEPMDEs.

Figure 4.1: Number in System Moments (Initial Conditions)

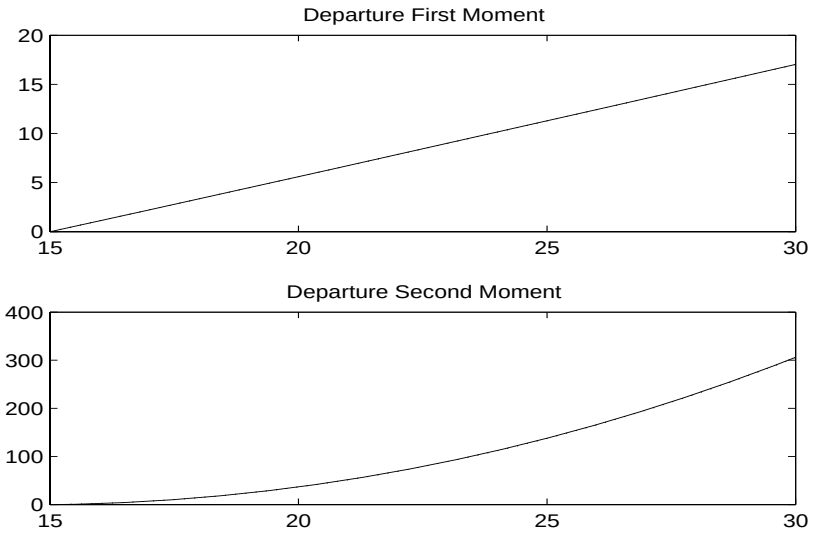


Figure 4.2: Departure Moments

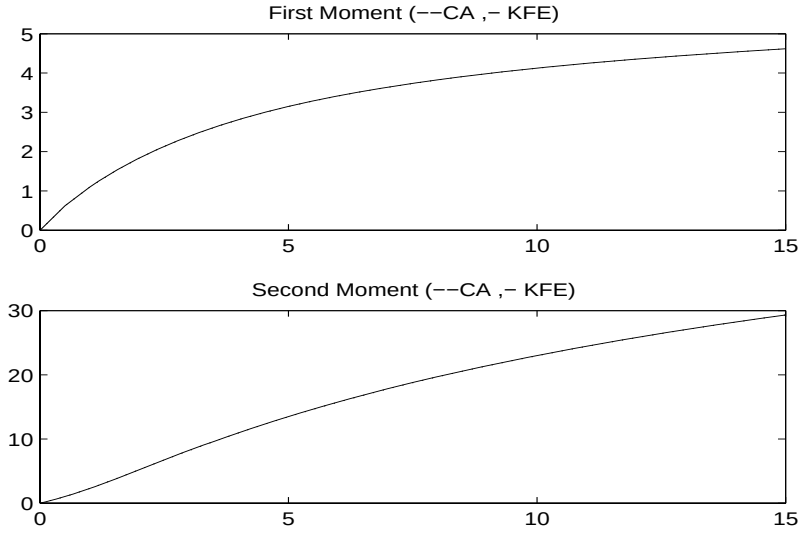


Table 4.1: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	4.4825	9.0241	13.6054	4.4869	9.0336	13.6200	-0.0970	-0.1048	-0.1074
p_2	0	4.5416	9.1229	0	4.5467	9.1331	0	-0.1125	-0.1125
p_3	0	0	4.5812	0	0	4.5864	0	0	-0.1126

Table 4.2: Second Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	24.4076	89.8370	197.6544	24.4827	90.2037	198.3203	-0.3067	-0.4066	-0.3358
p_2	0	25.0418	91.8734	0	25.1182	92.2137	0	-0.3045	-0.3690
p_3	0	0	25.4746	0	0	25.5463	0	0	-0.2807

4.5.2 Example 2

Consider the time-dependent $Ph_t/Ph_t/s/c$ case having system parameters:

$s = 3$, $c = 20$, $m_A = 3$,$m_B = 3$,

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4]$, $\qquad \boldsymbol{\beta} = [0.3, \quad 0.3, \quad 0.4]$,

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.3 & 0.2 & 0.4 \\ 0.3 & 0.1 & 0.2 & 0.4 \\ 0.2 & 0.2 & 0.3 & 0.3 \end{pmatrix},$$

$\boldsymbol{\lambda} = [3 + 3\sin(t/0.03\pi), \quad 3 + 3\sin(t/0.3\pi), \quad 2 + 2\sin(t/0.05\pi)]$, and $\boldsymbol{\mu} = [5/2, \quad 5/2, \quad 5]$.

Figure 4.3: Number in System Moments (Initial Conditions)

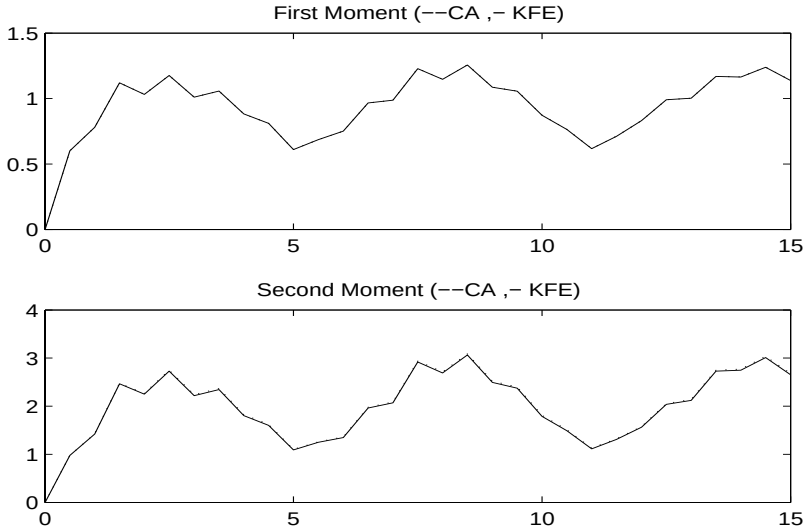
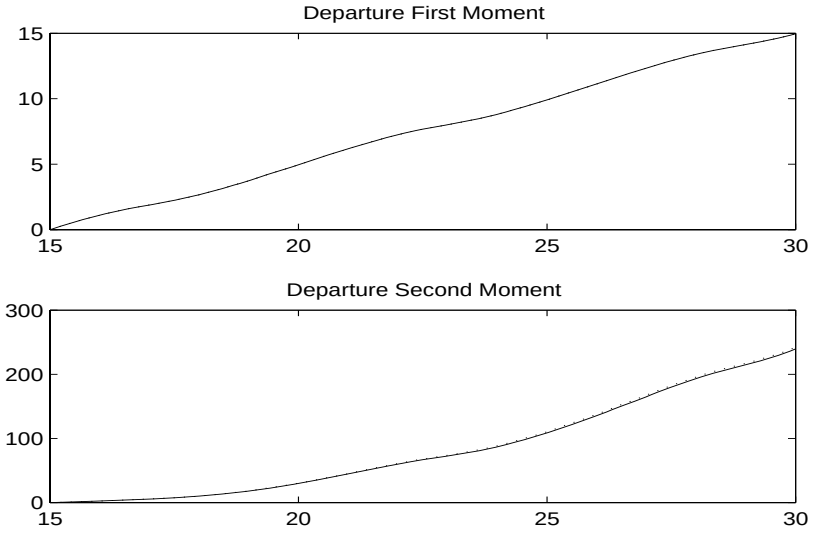


Figure 4.4: Departure Moments



4.5.3 Example 3

Consider the time-dependent $Ph_t/Ph_t/s/c$ case having parameters:

$s = 3$, $c = 20$, $m_A = 3$,$m_B = 3$,

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4]$, $\quad \boldsymbol{\beta} = [0.3, \quad 0.3, \quad 0.4]$,

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.3 & 0.2 & 0.4 \\ 0.3 & 0.1 & 0.2 & 0.4 \\ 0.2 & 0.2 & 0.3 & 0.3 \end{pmatrix},$$

$\boldsymbol{\lambda} = [3 + 3\sin(t/0.9\pi), \quad 3 + 3\sin(t/0.8\pi), \quad 2 + 2\sin(t/0.5\pi)]$, and $\boldsymbol{\mu} = [2, \quad 2, \quad 4]$.

Figure 4.5: Number in System Moments (Initial Conditions)

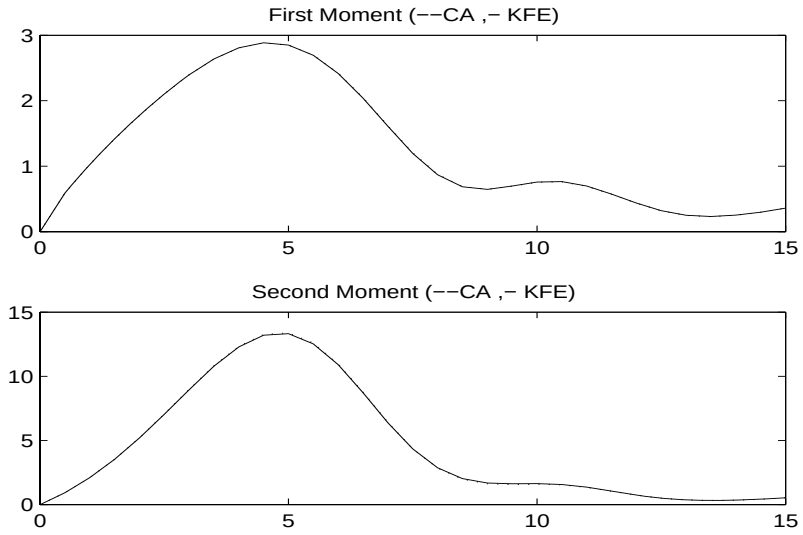
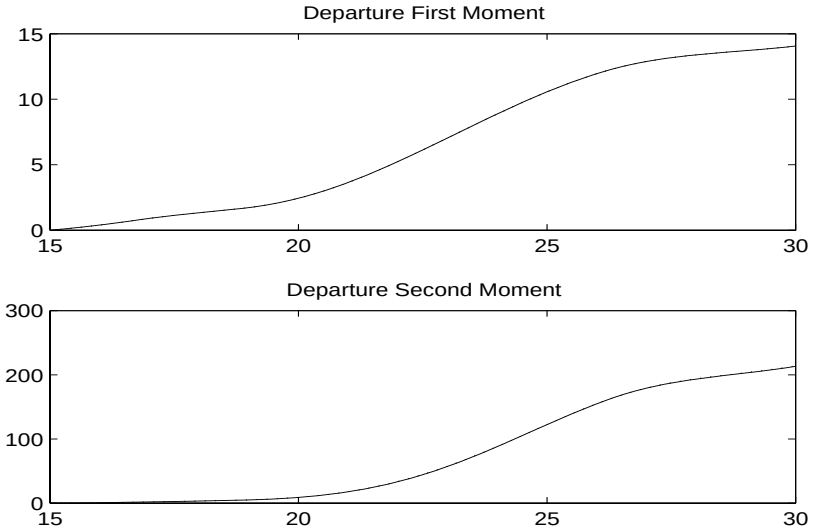


Figure 4.6: Departure Moments



4.5.4 Example 4

Consider the stationary $M/M/s/c$ case having system parameters:

$s = 3$, $c = 20$, $m_A = 1$,$m_B = 1$,

$\alpha = [1]$, $\beta = [1]$,

$$\mathbf{a} = \begin{pmatrix} 0 & 1 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 0 & 1 \end{pmatrix},$$

$\lambda = [3]$, and $\mu = [3/2]$.

Figure 4.7: Number in System Moments (Initial Conditions)

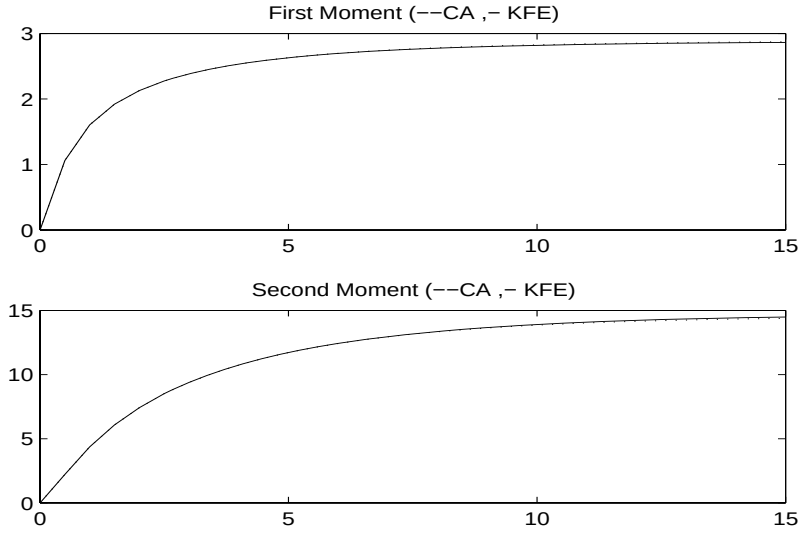
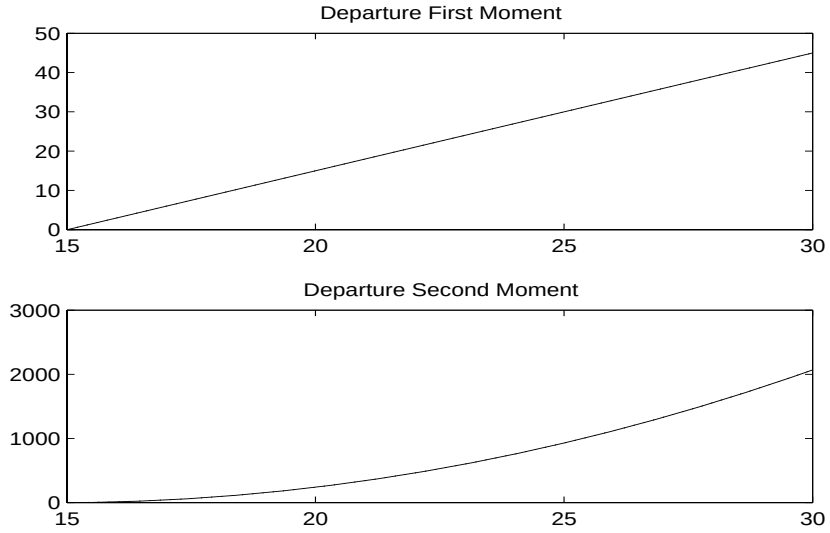


Figure 4.8: Departure Moments



4.5.5 Example 5

Consider the stationary $Ph_t/Ph_t/s/c$ case having parameters:

$s = 3$, $c = 20$, $m_A = 3$,$m_B = 2$

$\boldsymbol{\alpha} = [0.2, \quad 0.2, \quad 0.6]$ $\qquad$ $\boldsymbol{\beta} = [1/3, \quad 2/3]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [10, \quad 10, \quad 0.411]$ $\qquad$ $\boldsymbol{\mu} = [1, \quad 2/5]$

Figure 4.9: Number in System Moments (Initial Conditions)

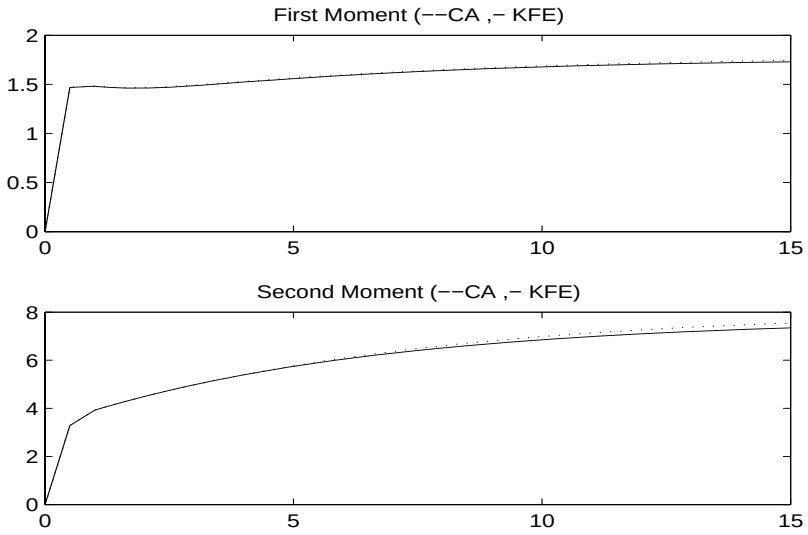
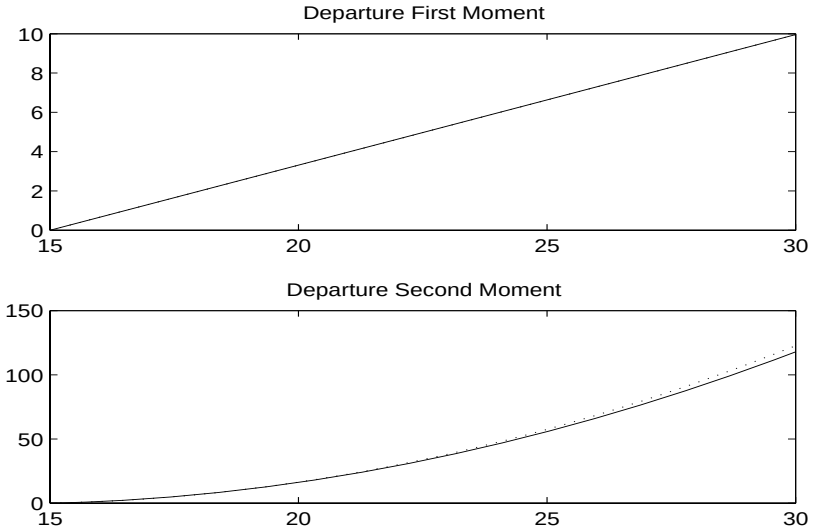


Figure 4.10: Departure Moments



4.5.6 $Ph_t/Ph_t/s/c$ Examples Conclusion

In this section we examined the algorithm presented in Chapter 3 that fits departure-count moments from a queueing node to a $\widetilde{Ph_t}$ distribution and then uses that distribution as the input process. We examined departure-count first-moment approximations over small time intervals and departure-count second-moment approximations over larger time intervals. The plots in Examples 1 to 5 show that the closure algorithm provides good departure-count moment approximations for intervals $[t, t+\tau)$ as τ increases from 0. We conclude that the closure approximations provide good departure-count moment approximations for intervals of all lengths.

We next test our (nodal) departure-count closure approximations by examining the accuracy of the downstream node performance measures where the downstream node has its arrival process approximated by the departure-process approximation from the upstream node. In the next section we test the quality of our departure-count approximations for tandem queues.

4.6 Two $Ph_t/Ph_t/s/c$ Nodes in Tandem

In this section we use the results obtained by numerically solving the NEPMDEs and closing the departure-count PMDEs to analyze two $Ph_t/Ph_t/s/c$ nodes in tandem. We obtain the first two departure-count moments over different time intervals using the NEPMDES and

CAs.

Next we consider the number-in-system KFEs of two queues in tandem.

4.6.1 KFEs of the Number-in-System of Two $Ph_t/Ph_t/s/c$ Nodes in Tandem

The number of system-states for the number-in-system of a single $Ph_t/Ph_t/s/c$ node is

$$m_B^{-1} m_A (s + m_B(c - s + 1)) \binom{m_B+s-1}{s} \tag{4.16}$$

The state space of a $Ph_t/Ph_t/s/c$ node is represented by the ordered $(m_B + 2)$-tuple, $(n_1, \ldots, n_{m_B}; \ell, q)$ where n_i is the number of entities in service phase i, ℓ is the phase of the arrival process, and q is the number of entities waiting in the queue. Recall that the state space of a $Ph_t/Ph_t/s/c$ node can be partitioned into two subspaces, $\Omega_1 \equiv \{0, 1, \ldots, s-1\}$ and $\Omega_2 \equiv \{s, \ldots, c\}$. The KFE's for states in Ω_1 and for states in Ω_2 have two distinct forms, as shown in Appendix 1.

The state-space of the tandem $Ph_t/Ph_t/s/c$ queue is represented by

$$\left((n_1, \ldots, n_{m_B^{(1)}}; \ell, q^{(1)})^{(1)}, (n_1, \ldots, n_{m_B^{(2)}}, q^{(2)})^{(2)} \right),$$

where $m_B^{(i)}$ and $q^{(i)}$ are the number of service phases and number in queue at the i^{th} node, respectively. The number of system states for this tandem queue network is:

$$\begin{aligned} & m_A \left(s^{(1)} + m_B^{(1)} \left(c^{(1)} - s^{(1)} + 1 \right) \right) \left(s^{(2)} + m_B^{(2)} \left(c^{(2)} - s^{(2)} + 1 \right) \right) \times \\ & \binom{m_B^{(1)}+s^{(1)}-1}{s^{(1)}} \binom{m_B^{(2)}+s^{(2)}-1}{s^{(2)}} \left(m_B^{(1)} m_B^{(2)} \right)^{-1}, \end{aligned} \tag{4.17}$$

where the superscript $^{(i)}$ indicates node i. For a relatively small system having parameters $c^{(i)} = 30$, $s^{(i)} = 3$ and $m_A = m_B^{(i)} = 3$ for $i = 1, 2$ the number of states is $236,883$; thus the number of KFE differential-difference equations is too large for practical implementation. Evaluating our approximation for such cases by numerically integrating the full set of KFE's for comparison is simply impractical. For such cases we use Monte Carlo simulation experiments instead of numerical integration.

4.6.2 Tandem Queues with Ph_t Service and Ph_t Arrival

We use the same approach used for the $Ph_t/M_t/s/c$ system (Section 4.6.2) to test the quality of our departure-count moment approximations for the $Ph_t/Ph_t/s/c$ system . We use the departure-count moment approximations to approximate the behavior of two queues in tandem with time-dependent Ph_t service distributions and finite capacity.

The arrival process to the second queue, which is the departure process from the first node, is approximated by a $\widetilde{Ph_t}$ distribution. We use the Fitting Algorithm (FA) in Section 3.6.1 to obtain the approximating $\widetilde{Ph_t}$ distribution for the $Ph_t/Ph_t/s/c$ system. The FA as presented in Section 3.6.1 is intended for the $Ph_t/M_t/s/c$ system. To apply the FA on a $Ph_t/Ph_t/s/c$ system, we present two adjustments to the $Ph_t/M_t/s/c$ FA:

1. Use the $Ph_t/Ph_t/s/c$ NEDMA and CM algorithm at step 1 of the $Ph_t/M_t/s/c$ FA.

2. Patch the $Ph_t/Ph_t/s/c$ number-in-system PMDEs at the second node. We present the $Ph_t/Ph_t/s/c$ FA patching procedure in the next section.

Patching and the Second Node Arrival Process

Patching of the number-in-system PMDEs at the second node might be required when transitioning across consecutive SMIs. Patching can also be described as setting the initial values of the number-in-system PMDEs for an SMI to the terminating values of the previous SMI. The patching process depends on the $\widetilde{Ph_t}$ distribution being used in the consecutive SMI. See Section 4.6.2 for a more detailed description on patching.

Let θ_i be the parameter used if the distribution in the i^{th} SMI is Erlang for $i = 1, \ldots, n_1$.

Let α_i be the parameter used if the distribution in the i^{th} SMI is hyper-exponential for $i = 1, \ldots, n_1$.

Let λ_{ij} be the parameter used in the j^{th} FMI of the i^{th} SMI for $i = 1, \ldots, n_1$ and $j = 1, \ldots, n_2$.

Assume the transition takes place between the i^{th} interval and the $(i+1)^{th}$ interval as shown in figure 3.15. The number-in-system probability equations (3.48 and 3.49) change as follows:

- Transitioning from an Erlang-4 to an Erlang -4 in the i^{th} to $(i+1)^{th}$:

$$\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = 1) = \theta_{i+1}\mathrm{P}(N(iT^-) \in \Omega_r) \quad \text{for} \quad r = 1, 2 \quad \text{and} \quad i = 1, \ldots, n_1.$$

$$\mathrm{P}(N(iT^+) \in \Omega_r; A(iT^+) = k) = (1-\theta_{i+1})\mathrm{P}(N(iT^-) \in \Omega_r)/3 \quad \text{for} \quad r = 1, 2, \quad k = 2, 3, 4 \quad \text{and} \quad i = 1, \ldots, n_1.$$

The conditional number-in-system moments change, but the partial number-in-system moments remain the same as we cross intervals.

$\mathrm{E}[N^j(iT^+), \Omega_r; \ell] = \mathrm{E}[N^j(iT^-), \Omega_r; \ell]$, but $\mathrm{E}[N^j(iT^+)|\Omega_r; \ell] \neq \mathrm{E}[N^j(iT^-)|\Omega_r; \ell]$ for $r = 1, 2$, $k = 1, 2, 3, 4$ and $j = 1, 2$.

- Transition occurs between two distributions with a different number of phases :

  The transition between two distributions with a different number of phases results in a different number of number-in-system PMDEs. For example, a transition between a hyper-exponential (two phases) and an Erlang-4 (four phases) increases the number of number-in-system PMDEs since the number of phases in the arrival process increases. Let $m_A^{(i)}$ be the number of phases of the $\widetilde{Ph_t}$ process in the i^{th} SMI. So if $m_A^{(i)} \neq m_A^{(i+1)}$ at time iT^+, the initial conditions of Equations 6.1 to 6.8 need to be approximated. At time iT^+, the following patching algorithm is performed to obtain the initial conditions:

  Let $F[\bullet(t); \ell]$ represent any partial moment or probability at time t in Equations 6.1 to 6.8.

  1. Compute $h = \sum_{\ell=1}^{m_A^{(i)}} F[\bullet(iT^-); \ell]$.
  2. If the $\widetilde{Ph_t}$ distribution of the $i+1$ SMI is Erlang-4 $\left(m_A^{(i+1)} = 4\right)$:

     $F[\bullet(iT^+); \ell = 1] = \theta_{i+1} h$

     $F[\bullet(iT^+); \ell = j] = (1 - \theta_{i+1})h$ for $j = 2, 3, 4$
  3. If the $\widetilde{Ph_t}$ distribution of the $i+1$ SMI is hyper-exponential $\left(m_A^{(i+1)} = 2\right)$:

     $F[\bullet(iT^+); \ell = j] = h/2$ for $j = 1, 2$

4. If the $\widetilde{Ph_t}$ distribution of the $i+1$ SMI is exponential $\left(m_A^{(i+1)} = 1\right)$:

$$F[\bullet(iT^+); \ell = 1] = h$$

Notice that the transition between a balanced mean hyper-exponential to another balanced means hyper exponential does not require adjusting the initial conditions (patching) since the probability of being in an arrival phase remains 1/2 throughout and obviously the number of phases does not change. In Section 3.6.1 we present a low variability distribution (figure 3.16) which does not require patching and can be used in place of the Erlang-4 as an approximating distribution. The disadvantage of this distribution is that it requires an extra phase to cover the same second moment range covered by the Erlang-4 distribution.

4.6.3 Solving the Number-in-System KFEs of the Second Node: The KFE FA

In the previous section, patching is performed for the number-in-system PMDEs (Equations 6.1 to 6.8) of node 2. Due to the large number of differential equation, it is not efficient to perform patching on the number-in-system KFEs of node 2. As an alternative to patching we construct a different approximate $\widehat{Ph_t}$ which enables solving the second node number-in-system KFEs without patching.

We use an approximate $\widehat{Ph_t}$ distribution with a constant number of phases (four phases), where the parameters are chosen depending on the variability and rate of the count process being approximated over an SMI.

Consider adjusting the FA at node 2 as follows (the KFE FA):

Solve the node 2 number-in-system KFEs (Equations 5.1 and 5.2) for the i^{th} SMI using the following approximating arrival Ph_t process $\left(\widehat{Ph_t}\right)$,

- If the arrival-count distribution has low variability

$$\mathrm{E}[D^2_{(i-1)T}(iT)] < 0.99\left(\mathrm{E}[D_{(i-1)T}(iT)]^2 + \mathrm{E}[D_{(i-1)T}(iT)]\right),$$

  use the following parameters to represent the approximate arrival $\widehat{Ph_t}$ process (Erlang-4):

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\boldsymbol{\alpha} = [1, \quad 0, \quad 0, \quad 0]$$

$$\boldsymbol{\lambda} = \left[\frac{\lambda}{\theta}, \quad \frac{3\lambda}{1-\theta}, \quad \frac{3\lambda}{1-\theta}, \quad \frac{3\lambda}{1-\theta}\right]$$

- If the arrival-count distribution has high variability

$$\mathrm{E}[D^2_{(i-1)T}(iT)] > 1.01\left(\mathrm{E}[D_{(i-1)T}(iT)]^2 + \mathrm{E}[D_{(i-1)T}(iT)]\right),$$

  use the following parameters to represent the approximate arrival $\widehat{Ph_t}$ process (bal-

anced means hyper-exponential):

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\boldsymbol{\alpha} = \left[\frac{p}{2}, \quad \frac{p}{2}, \quad \frac{1-p}{2}, \quad \frac{1-p}{2}\right]$$

$$\boldsymbol{\lambda} = [\lambda, \quad \lambda, \quad \lambda_2, \quad \lambda_2] \quad \text{where} \quad \lambda_2 = \frac{p\lambda}{1-p}$$

- Otherwise use the following parameters to represent the approximate arrival $\widehat{Ph_t}$ process (exponential):

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\boldsymbol{\alpha} = \left[\frac{1}{4}, \quad \frac{1}{4}, \quad \frac{1}{4}, \quad \frac{1}{4}\right]$$

$$\boldsymbol{\lambda} = [\lambda, \quad \lambda, \quad \lambda, \quad \lambda]$$

The FA requires patching when transitioning across two consecutive SMIs when the transition is from/to an Erlang-4 or the transition involves two different distribution families (Erlang-4

to hyper-exponential, hyper-exponential to exponential, ...). As a result, the $\widetilde{Ph_t}$ phase probabilities are constant within an SMI (see Section 3.6.1), where a phase probability i ($\pi_i(t) = \mathrm{P}(A(t) = i)$) is the probability the arrival $\widehat{Ph_t}$ process is in phase i at time t . The disadvantage of not patching when solving the number-in-system KFEs is that the $\widehat{Ph_t}$ phase probabilities are not necessarily constant within an SMI when the transition is from/to an Erlang-4 or the transition involves two different distribution families. Although the $\widehat{Ph_t}$ phase state probabilities are not always constant within an SMI, they converge to their steady state values as the SMI length increases.

There is more than one four-phase representation to model the exponential and the balanced mean hyper-exponential distributions. For example it is possible to represent the exponential distribution as follows:

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{\alpha} = [1, \quad 0, \quad 0, \quad 0]$$

$$\boldsymbol{\lambda} = [\lambda, \quad \infty, \quad \infty, \quad \infty]$$

Such a representation guarantees constant $\widehat{Ph_t}$ phase state probabilities within an SMI, but it is not practical for numerical integration techniques (such as the Runge-Kutta numerical

integration technique) due to the infinite phase departure rates.

4.6.4 Examples on Tandem Queues with Ph_t Service and Ph_t Arrival

We use three approaches to approximate the number-in-system at node 2:

1. The Closure Approximations Approach :

   • At node 1, use the CM algorithm to obtain approximations on the first departure-count moments over all FMIs and second departure-count moments over all SMIs.

   • Use the FA algorithm to obtain the approximate arrival process ($\widetilde{Ph_t}$) and approximate the number-in-system at node 2 by closing the corresponding number-in-system PMDEs (Equations 6.1 to 6.8) .

2. The KFE FA Approach:

   • At node 1, use the NEDMA to obtain numerically exact results on the first departure-count moments over all FMIs and second departure-count moments over all SMIs.

   • Use the KFE FA algorithm to obtain the approximate arrival process ($\widehat{Ph_t}$) and approximate the number-in-system at node 2 by numerically solving the corresponding number-in-system KFEs (Equations 5.1 to 5.2) .

3. The Monte-Carlo Simulation Approach:

- Use Monte-Carlo simulation to simulate the tandem queue network and obtain 1000 simulated values of the number-in-system at node 2 for each of the following points in time t =5, 10, 15, 20, 30, 45, 65, where for each point in time we re-run the simulation and use a different seed for the random number stream.

- Construct a 95% confidence interval on number-in-system at node 2 for each of the chosen points in time. Notice that since we use a different seed for the random number stream for each point in time, the means and half widths of the confidence intervals created at each point in time are independent.

We present two examples and plot the number-in-system at node 2 using the three approaches. The numerical results obtained from the closure approximation approach are plotted using dotted lines and the KFE FA numerical results are plotted on the same plot using solid lines. The limits of the confidence intervals of the Monte-Carlo simulation approach are also plotted on the same plot and the limits of the 95% confidence interval are represented by $*$s as shown in Figures 4.11 and 4.12. We also plot the number-in-system at node 1 by closing the number-in-system PMDEs (dotted plot), solving the number-in-system KFEs (solid plot) and constructing a 95% confidence interval using Monte-Carlo simulation.

Example 1

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 5, c = 20, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.5, \quad 0.5]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\lambda} = [3, \quad 3, \quad 2]$, $\boldsymbol{\mu} = [1, \quad 0.5]$.

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

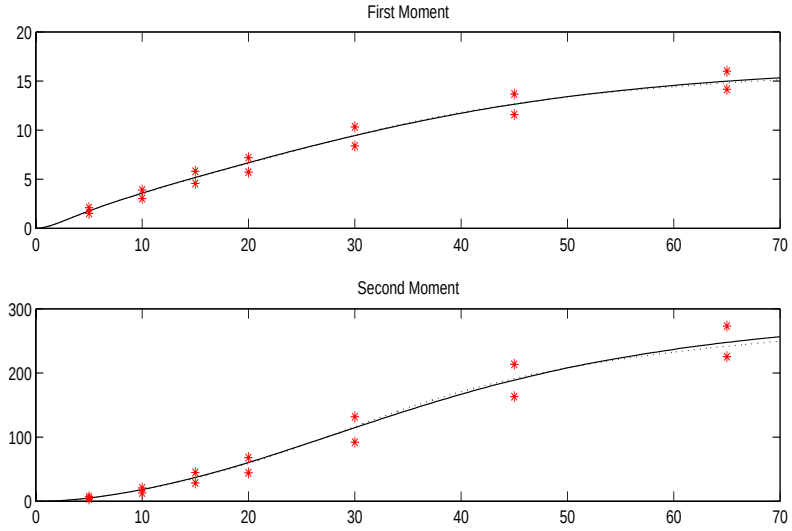
$s = 3, c = 20, m_B = 2$

$$\mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.5]$, $\boldsymbol{\mu} = [1, \quad 0.5]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 4.11: Node 2 Number-in-System Moments



Example 2

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 20, m_A = 2, m_B = 3$

$\boldsymbol{\alpha} = [1, \quad 0] \quad \boldsymbol{\beta} = [1, \quad 0, \quad 0]$

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1, \quad 1]$, $\boldsymbol{\mu} = [3, \quad 3, \quad 3]$.

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

$s = 3, c = 25, m_B = 3$

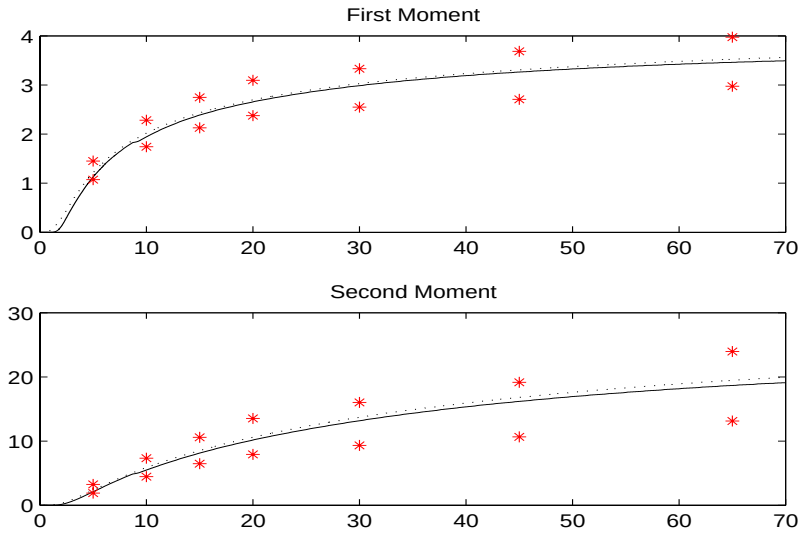
$$\mathbf{b} = \begin{pmatrix} 0.2 & 0.2 & 0.5 & 0.1 \\ 0.3 & 0.2 & 0.1 & 0.4 \\ 0.2 & 0.2 & 0.2 & 0.4 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.3, \quad 0.2]$, $\boldsymbol{\mu} = [1, \quad 1, \quad 1/2]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

The closure approximation and KFE FA approaches gave similar results and are within the confidence intervals constructed by the Monte-Carlo Simulation approach as can be seen in

Figure 4.12: Node 2 Number-in-System Moments



Figures 4.11 and 4.12. We test the quality our algorithms for different $Ph_t/Ph_t/s/c$ tandem queues and conclude that our algorithms provide good number-in-system approximations. We present more $Ph_t/Ph_t/s/c$ systems in Appendix 5.

4.7 Conclusion

We divide the algorithms presented in this chapter into 3 Categories:

1. Algorithms to obtain first/second departure-count moments over the FMIs/SMIs:

   • The NEDMA numerically solves the NEPDEs and obtains the first two moments of the $Ph_t/Ph_t/s/c$ departure-count process over any chosen set of time intervals.

   • The CM algorithm is a closure approximation algorithm which provides approx-

imations on the first two moments of the $Ph_t/Ph_t/s/c$ departure-count process over any chosen set of time intervals.

Using the NEDMA or the CM algorithm, we obtain numerically exact results or approximations on the departure-count first moment over all the FMIs and the departure-count second moments over all the SMIs.

2. Algorithms to fit the first and second departure-count moments obtained from node 2 to a Ph_t process.

   • At node 1, the FA uses the CM algorithm to obtain approximation on the first two departure-count moments over the FMIs and the SMI and approximates a $\widetilde{Ph_t}$ arrival process to node 2.

   • At node 1, the KFE FA uses the NEDMA to obtain numerically exact results on the first two departure-count moments over the FMIs and the SMIs and approximates a $\widehat{Ph_t}$ arrival process to node 2.

3. Algorithms that use an approximated Ph_t process to approximate the number-in-system at node 2:

   • At node 2, the FA uses the $\widetilde{Ph_t}$ as the arrival process to node 2 and then patches the resulting number-in-system PMDEs over the SMIs to enable the numerical integration of the PMDE.

   $Ph_t/Ph_t/s/c$ FA

- At node 2, the KFE FA uses the $\widehat{Ph_t}$ as the arrival process to node 2 and then solves the number-in-system KFEs.

The number of differential equations needed to solve the KFE FA is a function of the capacity of both nodes and is

$$3m_B^{(1)-1}m_A\left(s^{(1)}+m_B^{(1)}\left(c^{(1)}-s^{(1)}+1\right)\right)\binom{m_B^{(1)}+s^{(1)}-1}{s^{(1)}}$$

$$+4m_B^{(2)-1}\left(s^{(2)}+m_B^{(2)}\left(c^{(2)}-s^{(2)}+1\right)\right)\binom{m_B^{(2)}+s^{(2)}-1}{s^{(2)}}$$

The number of differential equations needed to solve the FA is not a function of the system capacity and is

$$m_A\left(m_B^{(1)2}+2m_B^{(1)}+4\right)+4\left(m_B^{(2)2}+2m_B^{(2)}+4\right)$$

Note that the algorithms presented do not provide numerically exact results on the first two moments of the number-in-system at node 2. Numerically integrating the tandem network KFEs provides numerically exact results but as discussed in Section 4.6.1, the number of KFEs is huge as given in Equation 4.17. So we use Monte-Carlo simulation to test the accuracy of our approximations.

Chapter 5

Appendix 1

We use MATLAB to numerically solve the KFEs. The function we use is ode45, which uses the Runge Kutta numerical integration technique. The function returns all the state probabilities over time. This is represented in an (axb), L_{axb} matrix, where a is the maximum number required to represent the system-state probabilities, and b is the number of integration intervals used within the specified time interval. Obviously L can get very large and due to computer space limitations we need $a = n_s$. So we present an algorithm that maps each state into a number in the interval $[1, n_s]$. At every iteration in the Runge Kutta numerical integration, we call the mapping to represent the states. This mapping might not be straight forward, but alternate simpler representations can result in huge matrices which imposes computer space limitations on the test cases we can run.

The KFEs are presented in [14] and they are

$$
\begin{aligned}
P_{n_1\cdots n_{m_B};\ell,0}\left(t\right)' &= -\lambda_\ell\left(t\right)P_{n_1\cdots n_{m_B};\ell,0}\left(t\right)-\sum_{i=1}^{m_B}n_i\mu_i\left(t\right)\left[1-b_{ii}\left(t\right)\right]P_{n_1\cdots n_{m_B};\ell,0}\left(t\right)\\
&\quad+\sum_{i=1}^{m_A}a_{i\ell}\left(t\right)\lambda_i\left(t\right)P_{n_1\cdots n_{m_B};i,0}\left(t\right)\\
&\quad+\sum_{i=1}^{m_B}\delta_{[n_i>0]}\sum_{\substack{j=1\\ j\neq i}}^{m_B}b_{ji}\left(t\right)\left[n_j+1\right]\mu_j\left(t\right)P_{n_1\cdots n_i-1\cdots n_j+1\cdots n_{m_B};\ell,0}\left(t\right)\\
&\quad+\sum_{i=1}^{m_A}a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)\left\{\sum_{j=1}^{m_B}\delta_{[n_j>0]}\beta_j\left(t\right)P_{n_1\cdots n_j-1\cdots n_{m_B};i,0}\left(t\right)\right\}\\
&\quad+\sum_{i=1}^{m_B}b_{i,m_B+1}\left(t\right)\left[n_i+1\right]\mu_i\left(t\right)P_{n_1\cdots n_i+1\cdots n_{m_B};\ell,0}\left(t\right)
\end{aligned}
\tag{5.1}
$$

for $0\leq n_i\leq s-1$; $i=1,\cdots,m_B$; $0\leq\sum_{j=1}^{m_B}n_j\leq s-1$; $\ell=1,\cdots,m_A$ and $t\geq 0$.

$$
\begin{aligned}
P_{n_1\cdots n_{m_B};\ell,q}(t)' &= -\lambda_\ell(t)\,P_{n_1\cdots n_{m_B};\ell,q}(t) \\
&\quad -\sum_{i=1}^{m_B} n_i\mu_i(t)\,[1-b_{ii}(t)]P_{n_1\cdots n_{m_B};\ell,q}(t) \\
&\quad +\sum_{i=1}^{m_A} a_{i\ell}(t)\,\lambda_i(t)\,P_{n_1\cdots n_{m_B};i,q}(t) \\
&\quad +\sum_{i=1}^{m_B}\delta_{[n_i>0]}\sum_{\substack{j=1\\ j\neq i}}^{m_B} b_{ji}(t)\,\delta_{[n_j<s]}\,[n_j+1]\,\mu_j(t)\,P_{n_1\cdots n_i-1\cdots n_j+1\cdots n_{m_B};\ell,q}(t) \\
&\quad +\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\left\{\left(1-\delta_{[q>0]}\right)\sum_{j=1}^{m_B}\delta_{[n_j>0]}\beta_j(t)\right. \\
&\quad \left. P_{n_1\cdots n_j-1\cdots n_{m_B};i,q}(t)+\delta_{[q>0]}P_{n_1\cdots n_{m_B};i,q-1}(t)\right\} \\
&\quad +\left(1-\delta_{[q<c-s]}\right)\sum_{i=1}^{m_A} a_{i,m_A+1}(t)\,\alpha_\ell(t)\,\lambda_i(t)\,P_{n_1\ldots n_{m_B};i,q}(t) \\
&\quad +\delta_{[q<c-s]}\sum_{i=1}^{m_B}\sum_{\substack{j=1\\ j\neq i}}^{m_B}\delta_{[n_j>0]}b_{i,m_B+1}(t)\,\beta_j(t)\,\delta_{[n_i<s]}\,[n_i+1]\,\mu_i(t) \\
&\quad P_{n_1\cdots n_i+1\cdots n_j-1\cdots n_{m_B};\ell,q+1}(t) \\
&\quad +\delta_{[q<c-s]}\sum_{i=1}^{m_B} b_{i,m_B+1}(t)\,\beta_i(t)\,n_i\mu_i(t)\,P_{n_1\cdots n_{m_B};\ell,q+1}(t)
\end{aligned}
\tag{5.2}
$$

for $0\leq n_i\leq s$; $i=1,\cdots,m_B$; $0\leq\sum_{j=1}^{m_B} n_j=s$; $\ell=1,\cdots,m_A$; $q=0,\cdots,c-s$ and $t\geq 0$.

The KFEs are solved using Runge-Kutta algorithm. In order to generate the KFEs we need an algorithm to generate the equations. So the first step in numerically solving the KFEs for the $Ph_t/Ph_t/s/c$ system is to map every state to a non-negative natural number. A state can be easily represented by two integers and a vector of length m_B. The first integer q represents the number in queue. The second integer $a_r\in[1,m_A]$ represents the current arrival phase. The vector s_r represents the number being served at each service phase. So

$0 \leq s_r(i) \leq s$ for $i \in [1, m_B]$ represents the number being served in service phase i.

The number of states is represented by the integer input parameters:

m_A: number of phases in the arrival distribution.

m_B: number of phases in the service distribution.

s: number of servers.

c: capacity of the system.

Where the number of states is $s_n = m_B^{-1} m_A(s + m_B(c - s + 1))\mathrm{C}(m_B + s - 1, s)$

The mapping we are looking for is from $(q, a_r, s_r) \rightarrow i \in [1, s_n]$.
It is obvious that: $(\sum_{i=1}^{m_B} s_r(i) < s \Rightarrow q = 0)$ and $(q > 0 \Rightarrow \sum_{i=1}^{m_B} s_r(i) = s)$

Divide the states into two groups. The first group has $q = 0$ and the second has $q > 0$. The number of states that s_r can take in group 1 is $\mathrm{C}(m_B + s, s)$ which is the number of ways to place s identical balls into $(m_B + 1)$ differentiable urns. The number of states that s_r can take in group 2 is $\mathrm{C}(m_B + s - 1, s)$ which is the number of ways to place s identical balls into m_B differentiable urns.

If we are able to map the states of s_r from each group to a number in $[1, s_n]$ then the rest of the algorithm consists of solving the KFEs.

Group 1:

For the case where $m_B = 3$ and $s = 2$, the mapping that needs to be constructed should return:

$000 \to 1$ $100 \to 2$ $200 \to 3$ $010 \to 4$ $110 \to 5$ $020 \to 6$ $101 \to 7$

$011 \to 8$ $002 \to 9$.

For any m_B and s consider a state $(i_1 i_2 i_3 \cdots i_{m_B}) \in$ group 1. Each component of $i_1, i_2, \cdots, i_n, \cdots, i_{m_B}$ contributes a number to the final mapping. Let $f(i_n, d_n)$ be this number returned by the n^{th} component where $d_n = \sum_{j=n+1}^{m_B} i_j$.

Let $A(u, b)$= number of ways to place b indistinguishable balls into u distinguishable urns.

$A(u, b) = \mathrm{C}(u + b - 1, b)$.

$f(i_1, d_1) = i_1 + 1$ (+1 is for the $00 \ldots 0$ state).

If $n > 1$ and $i_n = 0$ then $f(i_n, d_n) = 0$.

If $n > 1$ and $i_n \neq 0$ then $f(i_n, d_n) = \sum_{k=0}^{i_n - 1} A(n, s - d_n - k)$.

Let F be the number returned. So $F = \sum_{n=1}^{m_B} f(i_n, d_n)$

Group 2:

When $q > 0$, the previous example with $m_B = 3$ and $s = 2$ has the following states with their corresponding mappings.

$200 \to 1$ $110 \to 2$ $020 \to 3$ $011 \to 4$ $002 \to 5$

The number of states for group 1 was $A(3, 2)$. The number of states for group 2 is $A(2, 2)$. So the mapping used for group 2 is similar to that of group 1. The only difference is that the numbers of urns is one less in group 2.

In the example above, eliminate the first element of each state to get:(eliminate one urn

since in group 2 all b balls have to be used when forming combinations)

200→00→1 110→10→2 020→20→3 011→11→4 002→02→5.

The final step is to complete the mapping by returning a non-negative integer

Let s_1 be the number of states that can be formed by s_r in group1 and let s_2 be the number of states that can be formed by s_r in group 2.

$s_1 = \mathrm{C}(m_B + s, s)$ and $s_2 = \mathrm{C}(m_B + s - 1, s)$

For group1 the final mapping is $(F + (a_r - 1)s_1)$. For group 2 the final mapping is $(F + s_1 m_A + ((q-1)m_A + (a_r - 1))s_2)$.

Chapter 6

Appendix 2

The NIS PMDEs as derived in [14] are presented in this Appendix.

For Ω_1 the zero NIS PMDE

$$\begin{aligned} P'\left(I\left(t\right)=1,A\left(t\right)=\ell\right) &= -\lambda_\ell\left(t\right)P\left(I\left(t\right)=1,A\left(t\right)=\ell\right) \\ &\quad +\sum_{i=1}^{m_A}a_{i\ell}\left(t\right)\lambda_i\left(t\right)P\left(I\left(t\right)=1,A\left(t\right)=i\right) \\ &\quad +\sum_{i=1}^{m_A}a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)P\left(I\left(t\right)=1,A\left(t\right)=i\right) \\ &\quad -\sum_{i=1}^{m_A}a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)P\left(N\left(t\right)=s-1,A\left(t\right)=i\right) \\ &\quad +\sum_{i=1}^{m_B}b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)E\left[N_i\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right] \end{aligned} \tag{6.1}$$

For Ω_2 the zero NIS PMDE

$$\begin{aligned} P'\left(I\left(t\right)=2,A\left(t\right)=\ell\right) &= -\lambda_{\ell}\left(t\right)P\left(I\left(t\right)=2,A\left(t\right)=\ell\right) \\ &\quad +\sum_{i=1}^{m_A}a_{i\ell}\left(t\right)\lambda_i\left(t\right)P\left(I\left(t\right)=2,A\left(t\right)=i\right) \\ &\quad +\sum_{i=1}^{m_A}a_{i,m_A+1}\left(t\right)\alpha_{\ell}\left(t\right)\lambda_i\left(t\right)P\left(N\left(t\right)=s-1,A\left(t\right)=i\right) \\ &\quad +\sum_{i=1}^{m_A}a_{i,m_A+1}\left(t\right)\alpha_{\ell}\left(t\right)\lambda_i\left(t\right)P\left(I\left(t\right)=2,A\left(t\right)=i\right) \\ &\quad -\sum_{i=1}^{m_B}b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)E\left[N_i\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right] \end{aligned} \tag{6.2}$$

For Ω_1 the first NIS PMDE

$$\begin{aligned} E'\left[N_i\left(t\right),\ell,1\right] &= -\lambda_{\ell}\left(t\right)E\left[N_i\left(t\right),\ell,1\right] \\ &\quad +\sum_{j=1}^{m_A}a_{j\ell}\left(t\right)\lambda_j\left(t\right)E\left[N_i\left(t\right),j,1\right] \\ &\quad -\mu_i\left(t\right)E\left[N_i\left(t\right),\ell,1\right] \\ &\quad +\sum_{j=1}^{m_B}b_{ji}\left(t\right)\mu_j\left(t\right)E\left[N_j\left(t\right),\ell,1\right] \\ &\quad +\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_{\ell}\left(t\right)\lambda_j\left(t\right)E\left[N_i\left(t\right),j,1\right] \\ &\quad +\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_{\ell}\left(t\right)\lambda_j\left(t\right)\beta_i\left(t\right)P\left(I\left(t\right)=1,A\left(t\right)=j\right) \\ &\quad -\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_{\ell}\left(t\right)\lambda_j\left(t\right)E\left[N_i\left(t\right),N\left(t\right)=s-1,A\left(t\right)=j\right] \\ &\quad -\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_{\ell}\left(t\right)\lambda_j\left(t\right)\beta_i\left(t\right)P\left(N\left(t\right)=s-1,A\left(t\right)=j\right) \\ &\quad -b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)E\left[N_i\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right] \\ &\quad +\sum_{j=1}^{m_B}b_{j,m_B+1}\left(t\right)\mu_j\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right] \end{aligned} \tag{6.3}$$

For Ω_2 the first NIS PMDE

$$\begin{aligned} E'\left[N\left(t\right),\ell,2\right] &= -\lambda_\ell\left(t\right)E\left[N\left(t\right),\ell,2\right] \\ &\quad + \sum_{i=1}^{m_A} a_{i\ell}\left(t\right)\lambda_i\left(t\right)E\left[N\left(t\right),i,2\right] \\ &\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)sP\left(N\left(t\right)=s-1,A\left(t\right)=i\right) \\ &\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)E\left[N\left(t\right),i,2\right] \\ &\quad + \sum_{i=1}^{m_A} a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)P\left(I\left(t\right)=2,A\left(t\right)=i\right) \\ &\quad - \sum_{i=1}^{m_A} a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)P\left(N\left(t\right)=c,A\left(t\right)=i\right) \\ &\quad - \sum_{i=1}^{m_B} b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)E\left[N_i\left(t\right),I\left(t\right)=2,A\left(t\right)=\ell\right] \\ &\quad - \sum_{i=1}^{m_B} b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)\left(s-1\right)E\left[N_i\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right] \end{aligned} \tag{6.4}$$

For Ω_1 the p^{th} NIS PMDE

$$
\begin{aligned}
E'\left[N_i^p(t),\ell,1\right] &= -\lambda_\ell(t)E\left[N_i^p(t),\ell,1\right] \\
&\quad +\sum_{j=1}^{m_A} a_{j\ell}(t)\lambda_j(t)E\left[N_i^p(t),j,1\right] \\
&\quad +\sum_{\substack{j=1\\ j\neq i}}^{m_B} b_{ji}(t)\mu_j(t)\sum_{h=1}^{p}\binom{p}{h}E\left[N_i^{p-h}(t)N_j(t),\ell,1\right] \\
&\quad +(1-b_{ii}(t))\mu_i(t)\sum_{h=0}^{p-1}\binom{p}{h+1}(-1)^{h+1}E\left[N_i^{p-h}(t),\ell,1\right] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\alpha_\ell(t)\lambda_j(t)E\left[N_i^p(t),j,1\right] \\
&\quad +\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\alpha_\ell(t)\lambda_j(t)\beta_i(t)\sum_{h=1}^{p}\binom{p}{h}E\left[N_i^{p-h}(t),j,1\right] \\
&\quad -\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\alpha_\ell(t)\lambda_j(t)\beta_i(t)\sum_{h=1}^{p}\binom{p}{h}E\left[N_i^{p-h}(t),N(t)=s-1,A(t)=j\right] \\
&\quad -\sum_{j=1}^{m_A} a_{j,m_A+1}(t)\alpha_\ell(t)\lambda_j(t)E\left[N_i^p(t),N(t)=s-1,A(t)=j\right] \\
&\quad +b_{i,m_B+1}(t)\mu_i(t)\sum_{h=0}^{p-1}\binom{p}{h+1}(-1)^{h+1}E\left[N_i^{p-h}(t),N(t)=s,A(t)=\ell\right] \\
&\quad +\sum_{j=1}^{m_B} b_{j,m_B+1}(t)\mu_j(t)E\left[N_i^p(t)N_j(t),N(t)=s,A(t)=\ell\right]
\end{aligned}
\tag{6.5}
$$

For Ω_2 the p^{th} NIS PMDE

$$\begin{aligned}
E'\left[N^p\left(t\right),\ell,2\right] &= -\lambda_\ell\left(t\right)E\left[N^p\left(t\right),\ell,2\right] \\
&\quad +\sum_{i=1}^{m_A} a_{i\ell}\left(t\right)\lambda_i\left(t\right)E\left[N^p\left(t\right),i,2\right] \\
&\quad +\sum_{i=1}^{m_A} a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)s^p P\left(N\left(t\right)=s-1,A\left(t\right)=i\right) \\
&\quad +\sum_{i=1}^{m_A} a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)\sum_{h=0}^{p}\tbinom{p}{h}E\left[N^{p-h}\left(t\right),i,2\right] \\
&\quad -\sum_{i=1}^{m_A} a_{i,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_i\left(t\right)\sum_{h=1}^{p}\tbinom{p}{h}c^{p-h}P\left(N\left(t\right)=c,A\left(t\right)=i\right) \\
&\quad +\sum_{i=1}^{m_B} b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)\sum_{h=1}^{p}\tbinom{p}{h}\left(-1\right)^h E\left[N_i\left(t\right)N^{p-h}\left(t\right),I\left(t\right)=2,A\left(t\right)=\ell\right] \\
&\quad -\sum_{i=1}^{m_B} b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)\left(s-1\right)^p E\left[N_i\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right]
\end{aligned} \tag{6.6}$$

For Ω_1 the required Cross-Product NIS PMDE terms

$$
\begin{aligned}
&E'\left[N_i\left(t\right)N_j\left(t\right),\ell,1\right]=-\lambda_\ell\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),\ell,1\right]\\
&-\mu_i\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),\ell,1\right]\\
&-\mu_j\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),\ell,1\right]\\
&+\sum_{k=1}^{m_A}a_{k\ell}\left(t\right)\lambda_k\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),k,1\right]\\
&-b_{ji}\left(t\right)\mu_j\left(t\right)E\left[N_j\left(t\right),\ell,1\right]\\
&-b_{ij}\left(t\right)\mu_i\left(t\right)E\left[N_i\left(t\right),\ell,1\right]\\
&+\sum_{k=1}^{m_B}b_{ki}\left(t\right)\mu_k\left(t\right)E\left[N_j\left(t\right)N_k\left(t\right),\ell,1\right]\\
&+\sum_{k=1}^{m_B}b_{kj}\left(t\right)\mu_k\left(t\right)E\left[N_i\left(t\right)N_k\left(t\right),\ell,1\right]\\
&+\sum_{k=1}^{m_A}a_{k,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_k\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),k,1\right]\\
&+\sum_{k=1}^{m_A}a_{k,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_k\left(t\right)\left(\beta_i\left(t\right)E\left[N_j\left(t\right),k,1\right]+\beta_j\left(t\right)E\left[N_i\left(t\right),k,1\right]\right)\\
&-\sum_{k=1}^{m_A}a_{k,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_k\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),N\left(t\right)=s-1,A\left(t\right)=k\right]\\
&-\sum_{k=1}^{m_A}a_{k,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_k\left(t\right)\beta_i\left(t\right)E\left[N_j\left(t\right),N\left(t\right)=s-1,A\left(t\right)=k\right]\\
&-\sum_{k=1}^{m_A}a_{k,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_k\left(t\right)\beta_j\left(t\right)E\left[N_i\left(t\right),N\left(t\right)=s-1,A\left(t\right)=k\right]\\
&-\left(b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)+b_{j,m_B+1}\left(t\right)\mu_j\left(t\right)\right)E\left[N_i\left(t\right)N_j\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right]\\
&+\sum_{k=1}^{m_B}b_{k,m_B+1}\left(t\right)\mu_k\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right)N_k\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right]
\end{aligned}
\tag{6.7}
$$

For Ω_2 the required Cross-Product NIS PMDE terms

$$
\begin{aligned}
&E'\left[N_i\left(t\right)N\left(t\right),\ell,2\right]=-\lambda_\ell\left(t\right)E\left[N_i\left(t\right)N\left(t\right),\ell,2\right]\\
&-\mu_i\left(t\right)E\left[N_i\left(t\right)N\left(t\right),\ell,2\right]\\
&+\sum_{j=1}^{m_A}a_{j\ell}\left(t\right)\lambda_j\left(t\right)E\left[N_i\left(t\right)N\left(t\right),j,2\right]\\
&+\sum_{j=1}^{m_B}b_{ji}\left(t\right)\mu_j\left(t\right)E\left[N_j\left(t\right)N\left(t\right),\ell,2\right]\\
&+\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_j\left(t\right)sE\left[N_i\left(t\right),N\left(t\right)=s-1,A\left(t\right)=j\right]\\
&+\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_j\left(t\right)\beta_i\left(t\right)sP\left(N\left(t\right)=s-1,A\left(t\right)=j\right)\\
&+\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_j\left(t\right)E\left[N_i\left(t\right)N\left(t\right),j,2\right]\\
&+\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_j\left(t\right)E\left[N_i\left(t\right),I\left(t\right)=2,A\left(t\right)=j\right]\\
&-\sum_{j=1}^{m_A}a_{j,m_A+1}\left(t\right)\alpha_\ell\left(t\right)\lambda_j\left(t\right)E\left[N_i\left(t\right),N\left(t\right)=c,A\left(t\right)=j\right]\\
&+b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)E\left[N_i\left(t\right),I\left(t\right)=2,A\left(t\right)=\ell\right]\\
&+\sum_{j=1}^{m_B}b_{j,m_B+1}\left(t\right)\mu_j\left(t\right)\beta_i\left(t\right)E\left[N_j\left(t\right)N\left(t\right),\ell,2\right]\\
&-\sum_{j=1}^{m_B}b_{j,m_B+1}\left(t\right)\mu_j\left(t\right)E\left[N_i\left(t\right)N_j\left(t\right),I\left(t\right)=2,A\left(t\right)=\ell\right]\\
&-\sum_{j=1}^{m_B}b_{j,m_B+1}\left(t\right)\mu_j\left(t\right)\beta_i\left(t\right)E\left[N_j\left(t\right),I\left(t\right)=2,A\left(t\right)=\ell\right]\\
&-\sum_{j=1}^{m_B}b_{j,m_B+1}\left(t\right)\mu_j\left(t\right)\beta_i\left(t\right)\left(s-1\right)E\left[N_j\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right]\\
&-\sum_{j=1}^{m_B}b_{j,m_B+1}\left(t\right)\mu_j\left(t\right)\left(s-1\right)E\left[N_i\left(t\right)N_j\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right]\\
&+b_{i,m_B+1}\left(t\right)\mu_i\left(t\right)\left(s-1\right)E\left[N_i\left(t\right),N\left(t\right)=s,A\left(t\right)=\ell\right]
\end{aligned}
\tag{6.8}
$$

Chapter 7

Appendix 3

In order to compare the performance of the methods over time we choose the following ten points (10, 20, 30,..., 100) and obtain the total absolute error for each method. We take the averages of the absolute error and present the results in Table 2.2 which also contains the average percent error of the first and second number in system moments over the ten points in time as well as the run time in minutes required to solve the CAs and the KFEs.

Example 1

$s = 3, c = 30, m_A = 4, m_B = 2$

$\boldsymbol{\alpha} = [1, \quad 0, \quad 0, \quad 0], \quad \boldsymbol{\beta} = [0.8, \quad 0.2]$

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [12, \quad 9, \quad 6, \quad 8]$, $\boldsymbol{\mu} = m[2, \quad 1]$.

Figure 7.1: Moments E1 (m=1)

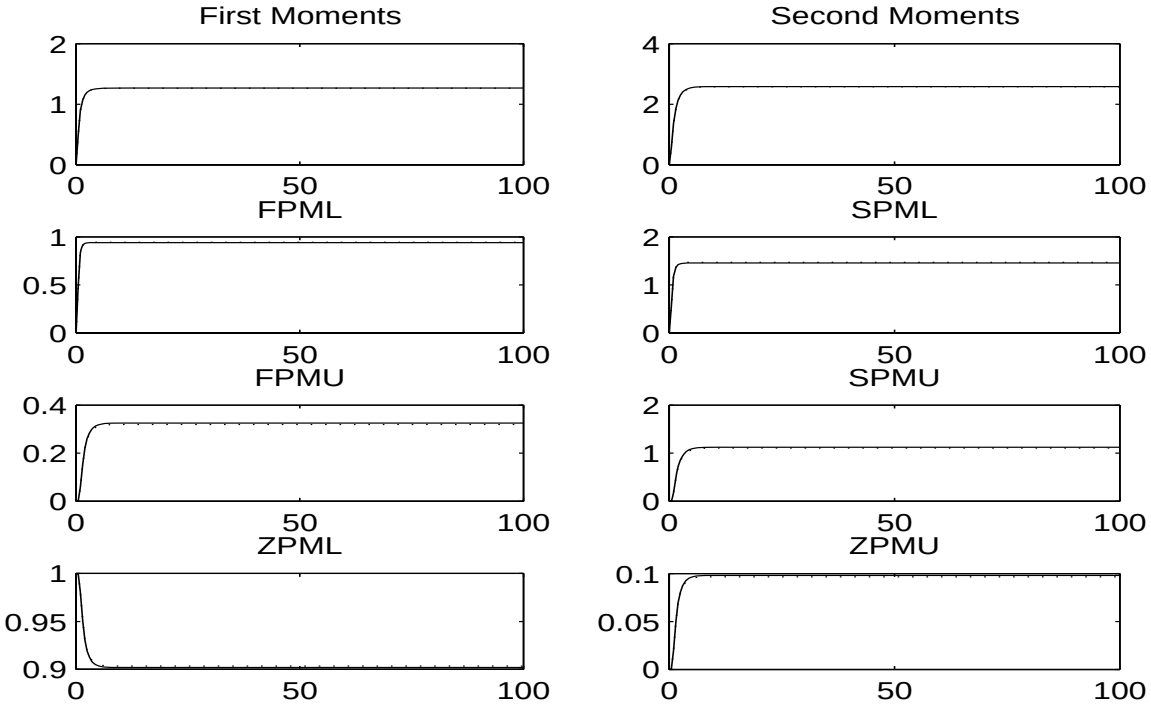


Table 7.1: Example 1

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.0139	0.1516	0.0137	1.1225	0.1066	0.1710	2.8753	0.0784
$m = 0.1$	0.0565	0.0785	0.0570	1.5038	0.2541	0.3820	1.8768	0.0539
$m = 0.5$	0.0456	0.2454	0.0456	1.3150	0.2108	0.8546	2.3596	0.0544

Figure 7.2: Moments E1 (m=0.1)

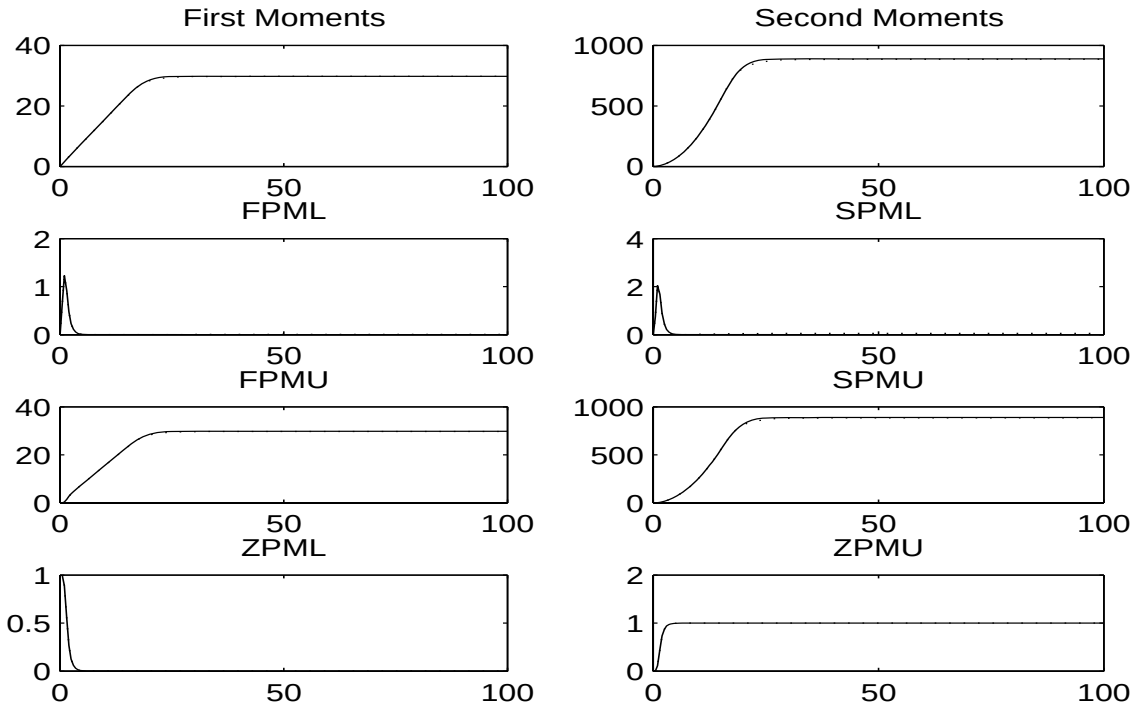
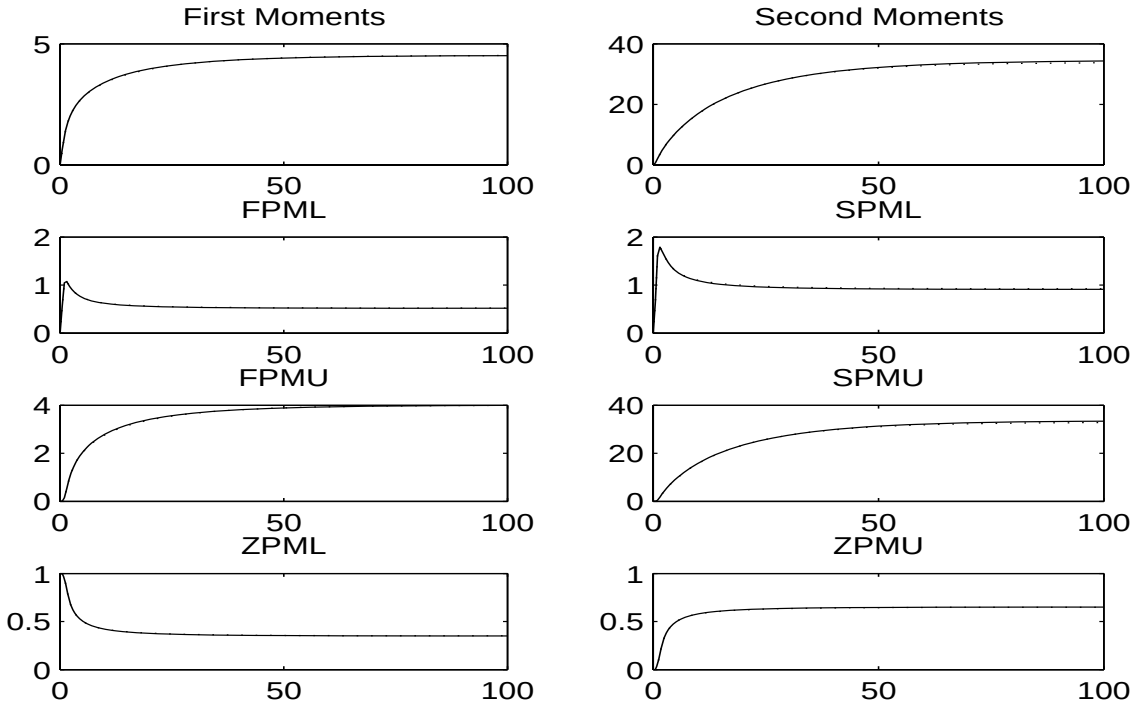


Figure 7.3: Moments E1 (m=0.5)



Example 2

$s = 3, c = 20, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.5, \quad 0.5]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\lambda} = [2.7 + 0.5\sin(t/3\pi), \quad 2.7 + 0.5\sin(t/3\pi), \quad 1.7 + 0.5\sin(t/3\pi)]$, $\boldsymbol{\mu} = m[2, \quad 0.5]$.

Figure 7.4: Moments E2 (m=1)

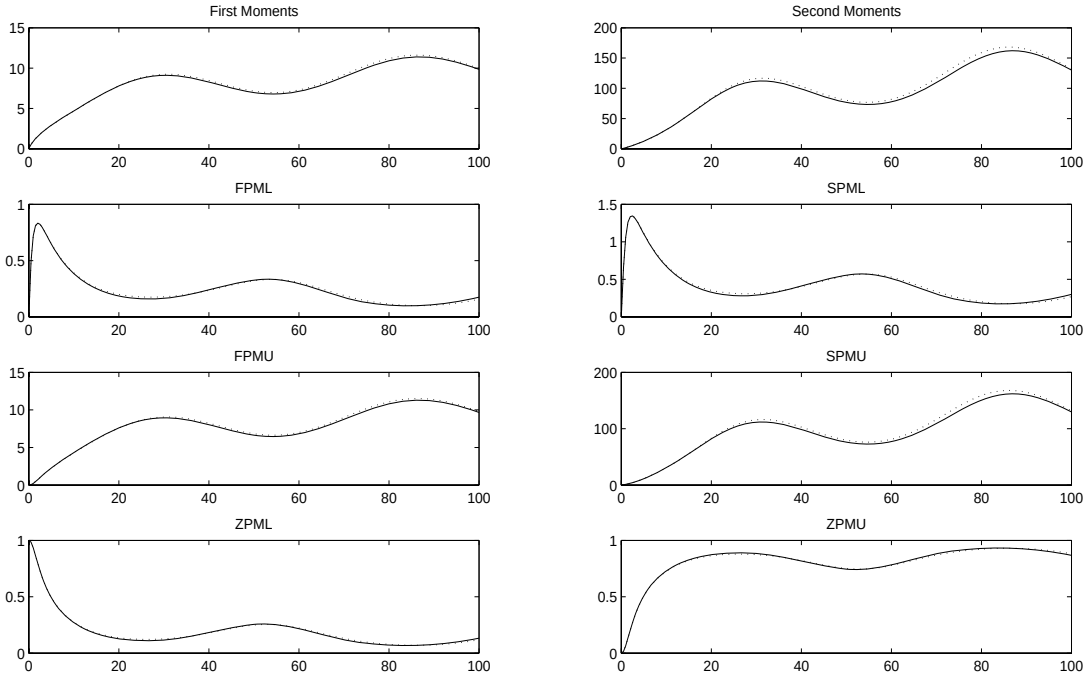


Figure 7.5: Moments E2 (m=3)

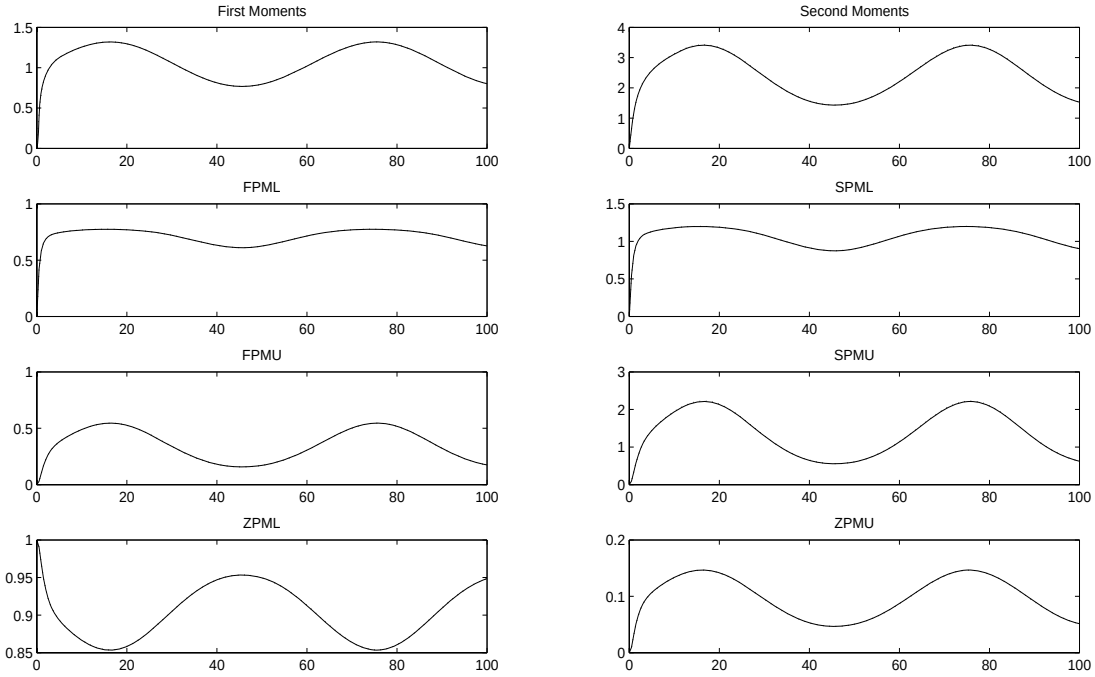


Table 7.2: Example 2

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.0717	0.1359	0.0717	1.1481	1.7713	3.5925	0.4846	0.0190
$m = 3$	0.0031	0.0594	0.0031	0.7891	0.0358	0.2891	1.1143	0.0380

Example 3

$s = 3, c = 30, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.5, \quad 0.3, \quad 0.2], \quad \boldsymbol{\beta} = [0.6, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [4, \quad 6, \quad 4]$, $\boldsymbol{\mu} = m[1, \quad 1.5]$.

Figure 7.6: Moments E3 (m=1)

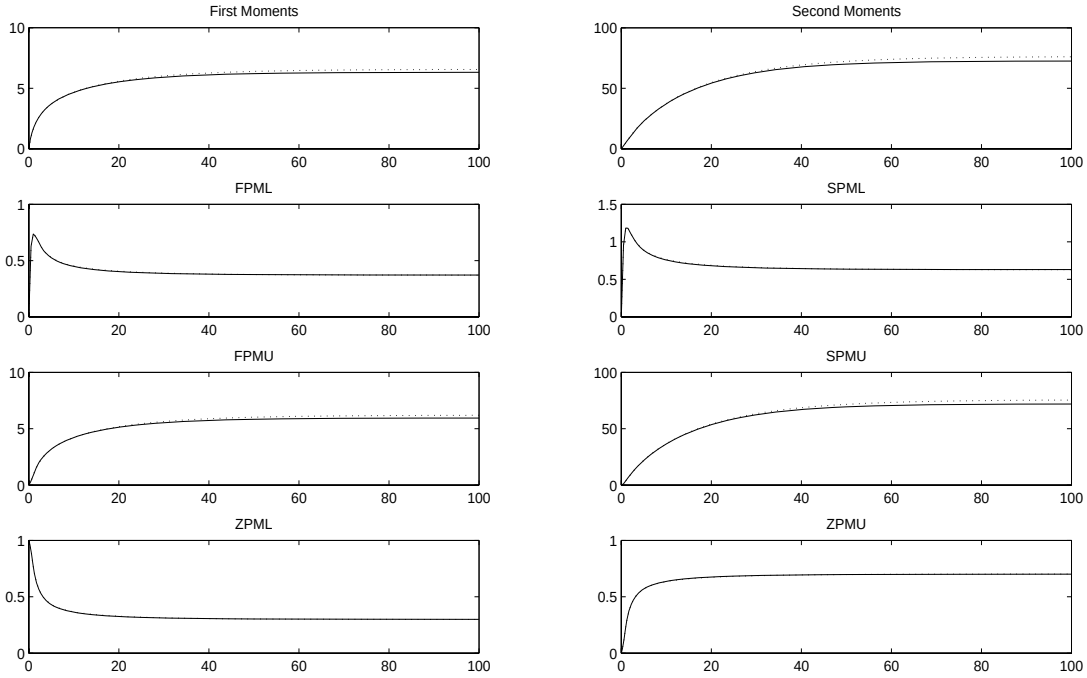


Figure 7.7: Moments E3 (m=3)

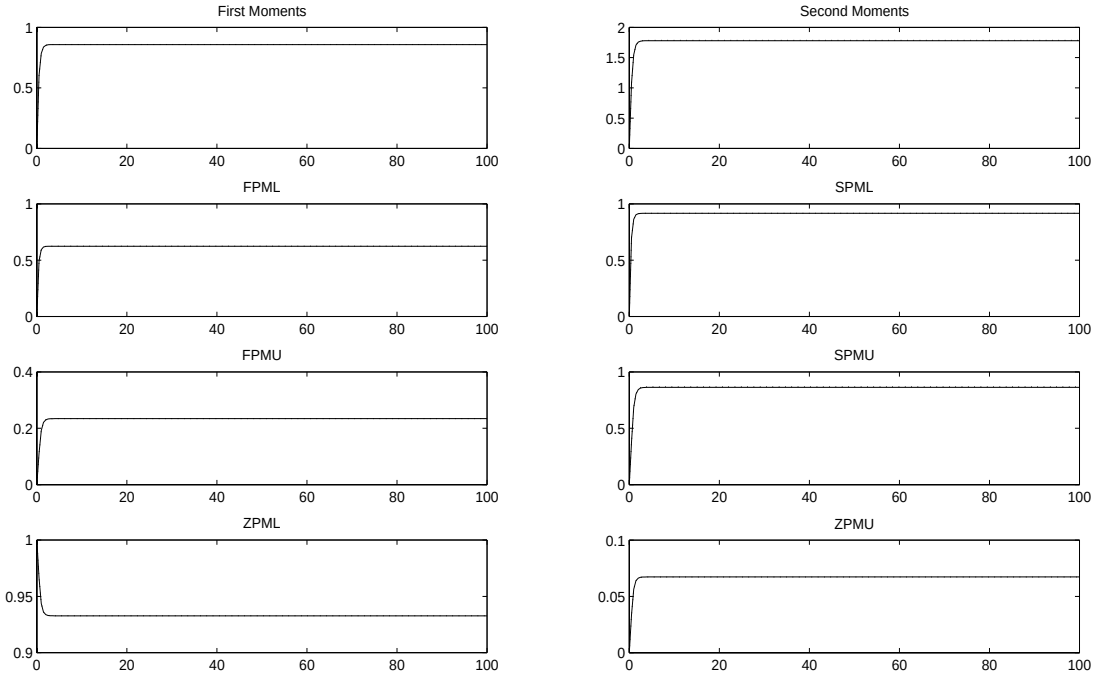


Table 7.3: Example 3

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.0623	0.2042	0.0623	0.7786	2.6082	3.0279	1.2688	0.0260
$m = 3$	0.0013	0.0267	0.0013	0.5341	0.0547	0.3061	2.1615	0.0492

Example 4

$s = 3, c = 30, m_A = 3, m_B = 3$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.4, \quad 0.1, \quad 0.5]$

$$\mathbf{a} = \begin{pmatrix} 0.3 & 0.1 & 0.3 & 0.3 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.2 & 0.2 & 0.2 & 0.4 \\ 0.2 & 0.1 & 0.2 & 0.5 \\ 0.2 & 0.3 & 0.2 & 0.3 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1.4 + 0.5\sin(t/6\pi), \quad 2.2 + 0.8\sin(t/6\pi), \quad 1.6 + 0.6\sin(t/6\pi)]$, $\boldsymbol{\mu} = m[0.5, \quad 1, \quad 0.6]$.

Figure 7.8: Moments E4 (m=1)

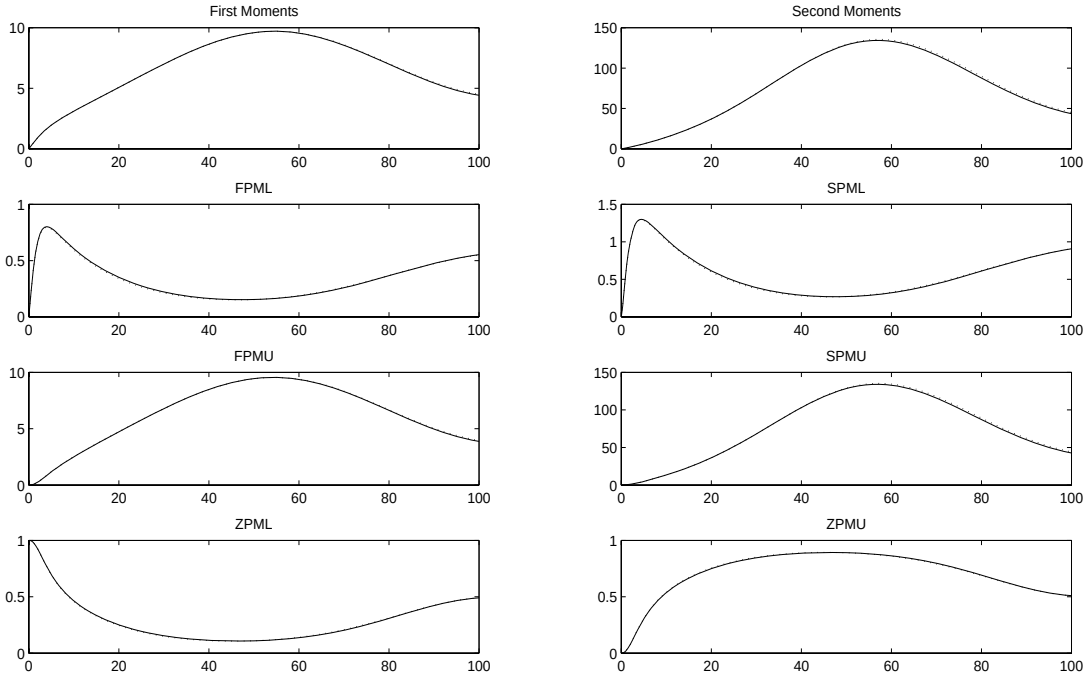


Figure 7.9: Moments E4 (m=3)

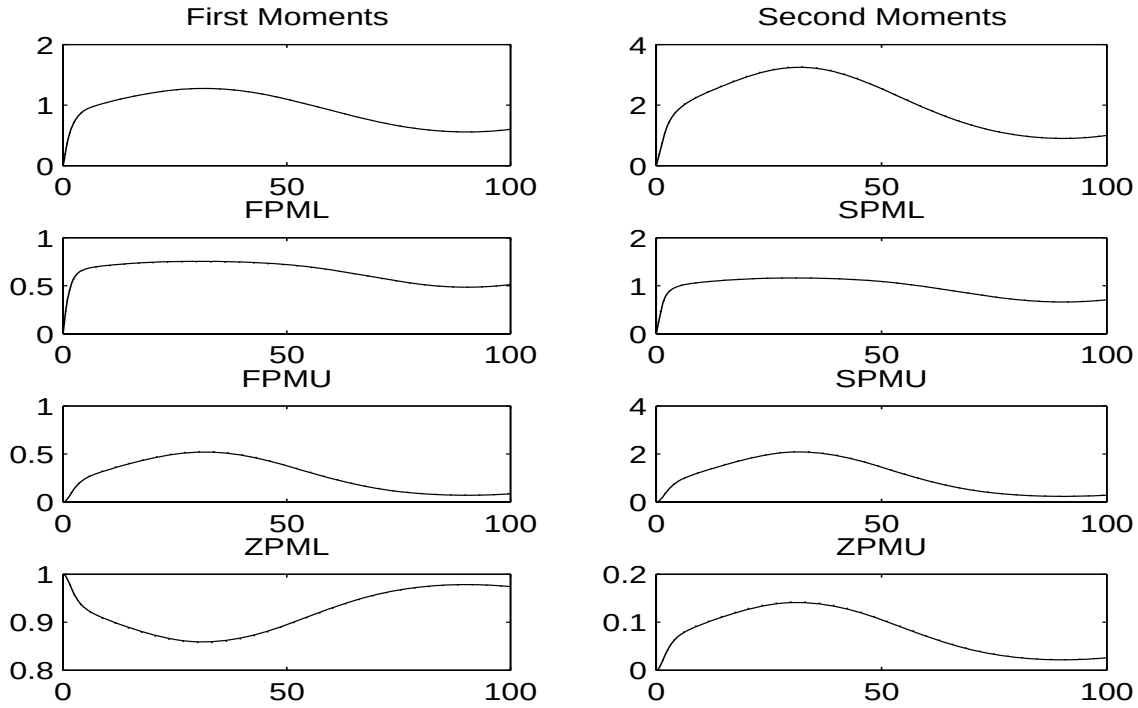


Table 7.4: Example 4

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.0436	0.1235	0.0436	1.0814	0.4672	1.2436	0.9063	0.0148
$m = 3$	0.0186	0.0411	0.0186	0.7052	0.0498	0.7890	1.2924	0.0193

Example 5

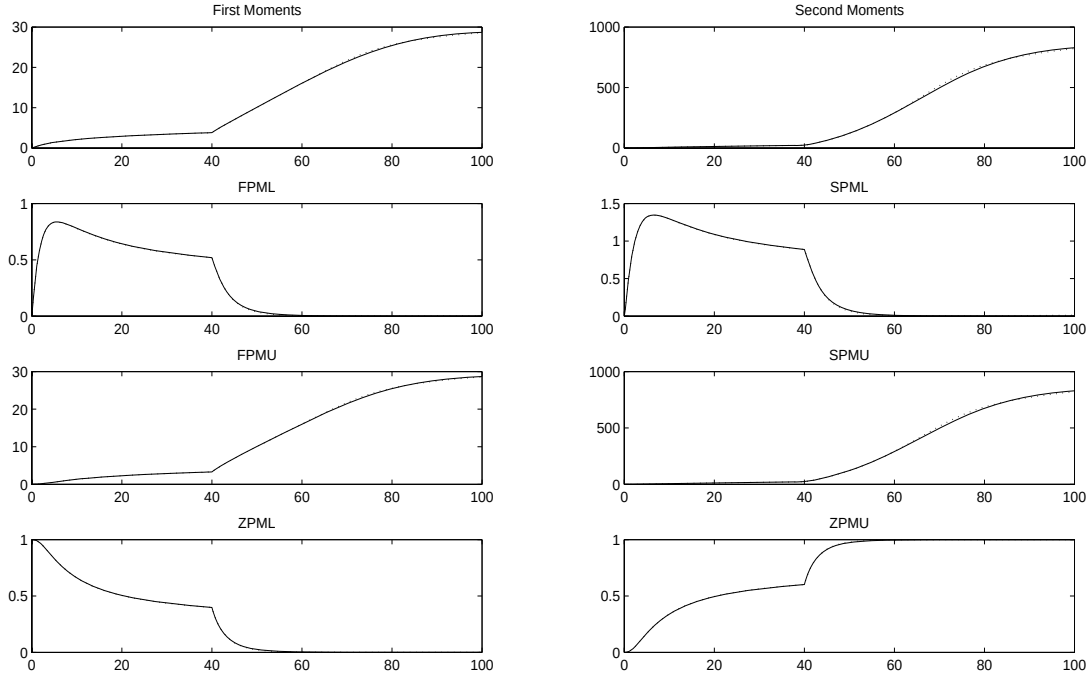
$s = 3, c = 30, m_A = 3, m_B = 3$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.4, \quad 0.1, \quad 0.5]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.2 & 0.2 & 0.2 & 0.4 \\ 0.2 & 0.1 & 0.2 & 0.5 \\ 0.2 & 0.3 & 0.2 & 0.3 \end{pmatrix}$$

$$\boldsymbol{\lambda} = \begin{Bmatrix} [1, \quad 0.5, \quad 1] & t < 40 \\ [2, \quad 3, \quad 2] & t \geq 40 \end{Bmatrix} \qquad , \boldsymbol{\mu} = m[0.4, \quad 0.3, \quad 0.4].$$

Figure 7.10: Moments E5 (m=1)



Example 6

Table 7.5: Example 6

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.0862	0.1103	0.0862	1.2112	0.3435	0.8445	0.4742	0.0104

$s = 4, c = 20, m_A = 4, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.1, \quad 0.3]$ $\qquad$ $\boldsymbol{\beta} = [0.6, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.2 & 0.3; \\ 0.3 & 0.1 & 0.1 & 0.3 & 0.2; \\ 0.3 & 0.2 & 0.1 & 0.2 & 0.2; \\ 0.3 & 0.1 & 0.2 & 0.3 & 0.1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.4 & 0.4 & 0.2 \end{pmatrix}$$

$\boldsymbol{\lambda} = [3 + 0.5\sin(t/(3\pi)), \quad 3 + 0.5\sin(t/(3\pi)), \quad 2 + 0.5\sin(t/(3\pi)), \quad 2 + 0.5\sin(t/(3\pi))]$

$\boldsymbol{\mu} = m[0.5, \quad 0.5].$

Table 7.6: Example 6

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.0556	0.1746	0.0556	1.3271	0.9900	2.1577	0.2943	0.0122
$m = 0.2$	0.0884	0.0919	0.0884	1.4645	0.3312	0.8350	0.2076	0.0115
$m = 10$	0.0101	0.0042	0.0104	0.4521	0.0411	0.6501	1.1857	0.0417

Example 7

$s = 3, c = 20, m_A = 3, m_B = 3$

$\boldsymbol{\alpha} = [0.6\,|\sin(t/0.03\pi)|\,, \quad 0.6(1 - |\sin(t/0.03\pi)|), \quad 0.4]$

$\boldsymbol{\beta} = [0.4, \quad 0.6\,|\cos(t/0.03\pi)|\,, \quad 0.6(1 - |\cos(t/0.03\pi)|)]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.7\,|\sin(t/0.3\pi)| & 0.7(1 - |\sin(t/0.3\pi)|) & 0.3\,|\sin(t/0.01\pi)| & 0.3(1 - |\sin(t/0.01\pi)|) \\ 0.8\,|\sin(t/0.3\pi)| & 0.8(1 - |\sin(t/0.3\pi)) & 0.2\,|\cos(t/0.01\pi)| & 0.2(1 - |\cos(t/0.01\pi)|) \end{pmatrix}$$

$$\mathbf{b} = \begin{pmatrix} 0.75\,|\sin(t/0.05\pi)| & 0.75(1 - |\sin(t/0.05\pi)|) & 0.25\,|\sin(t/0.02\pi)| & 0.25(1 - |\sin(t/0.02\pi)|) \\ 0.3 & 0.1 & 0.2 & 0.4 \\ 0.7\,|\sin(t/0.3\pi)| & 0.7(1 - |\sin(t/0.3\pi)|) & 0.3\,|\cos(t/0.01\pi)| & 0.3(1 - |\cos(t/0.01\pi)|) \end{pmatrix}$$

$\boldsymbol{\lambda} = [3 + 3\,|\sin(t/0.03\pi)|\,, \quad 3 + 2\,|sin(t/0.3\pi)|\,, \quad 2 + |\sin(t/0.05\pi)|]$

$\boldsymbol{\mu} = [2m(0.5 + 0.5\,|\sin(t/0.03\pi)|), \quad 2m(0.5 + 0.5\,|\sin(t/0.3\pi)|), \quad 2m(1 + |\sin(t/0.05\pi)|)]$

Table 7.7: Example 7

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.0856	0.0791	0.0856	1.2413	0.5988	1.2053	7.5622	0.2156
$m = 4$	0.1386	0.1520	0.1386	0.9173	0.6499	3.9989	7.6281	0.4151

Example 8

$s = 3, c = 20, m_A = 2, m_B = 2$

$\boldsymbol{\alpha} = [1/2, \quad 1/2] \qquad\qquad \boldsymbol{\beta} = [2/3, \quad 1/3]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1, \quad 100], \qquad \boldsymbol{\mu} = [1, \quad 100]$

Table 7.8: Example 8

M1	M2	M3	M4	% FM	% SM	KFE time	CA time
0.0846	0.3024	0.0846	0.3128	4.3535	13.3123	21.8172	0.8094

Example 9

$s = 3, c = 20, m_A = 4, m_B = 4$

$\boldsymbol{\alpha} = [1, \quad 0, \quad 0, \quad 0]$ $\qquad$ $\boldsymbol{\beta} = [1, \quad 0, \quad 0, \quad 0]$

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [4, \quad 4, \quad 4, \quad 4]$ $\qquad$ $\boldsymbol{\mu} = 1.5[1, \quad 1, \quad 1, \quad 1]$

Table 7.9: Example 9

M1	M2	M3	M4	% FM	% SM	KFE time	CA time
0.3025	0.2159	0.3013	1.4206	4.3551	4.5988	3.7617	0.0789

Example 10

$s = 3, c = 30, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4]$ $\qquad$ $\boldsymbol{\beta} = [0.6, \quad 0.3, \quad 0.1]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0.5 & 0.5 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0.8 & 0 & 0.2 \\ 0 & 0 & 0.8 & 0.2 \\ 0.3 & 0 & 0 & 0.7 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1 + \cos(2\pi t), \quad 2 + \sin(\pi t), \quad 3 + \cos(0.5\pi t)]$

$\boldsymbol{\mu} = [1.5 + \cos(2\pi t), \quad 3 + \sin(\pi t), \quad 1 + 0.5\cos(0.5\pi t)]$

Figure 7.11: Moments E10

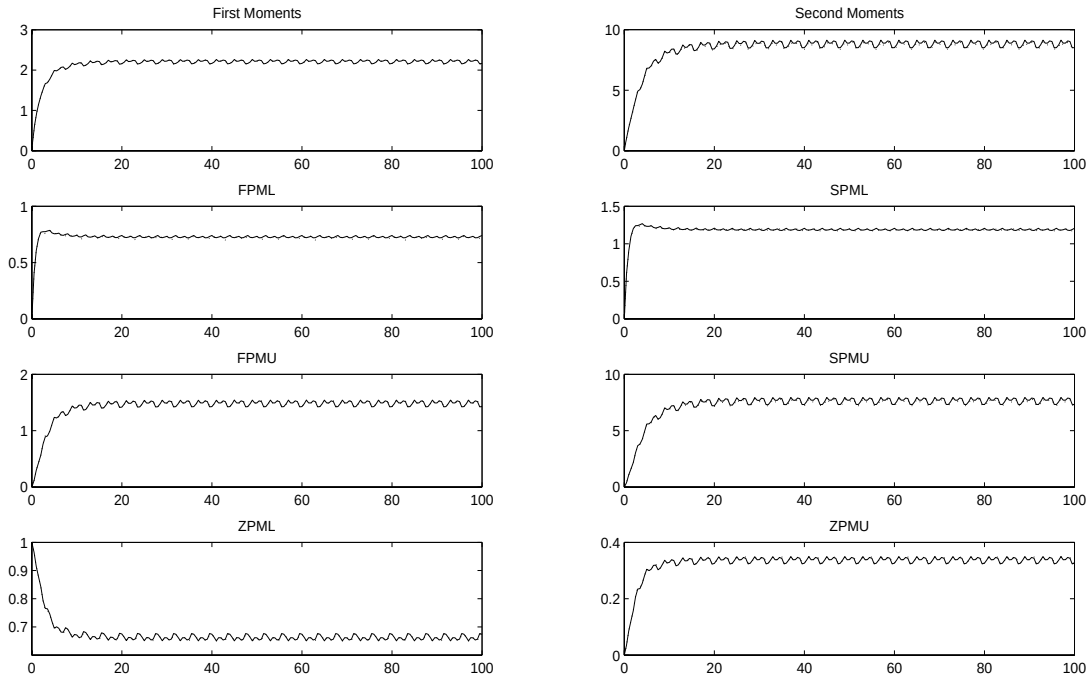


Table 7.10: Example 10

M1	M2	M3	M4	% FM	% SM	KFE time	CA time
0.0282	0.1275	0.0282	1.0291	0.0870	1.4419	3.5732	0.1378

Example 11

$s = 3, c = 20, m_A = 3, m_B = 3$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.3, \quad 0.3, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.3 & 0.2 & 0.4 \\ 0.3 & 0.1 & 0.2 & 0.4 \\ 0.2 & 0.2 & 0.3 & 0.3 \end{pmatrix}$$

$\boldsymbol{\lambda} = [3 + 3\sin(t/0.03\pi), \quad 3 + 3\sin(t/0.3\pi), \quad 2 + 2\sin(t/0.05\pi)],$

$\boldsymbol{\mu} = m[0.5, \quad 0.5, \quad 1]$

Table 7.11: Example 11

	M1	M2	M3	M4	% FM	% SM	KFE time	CA time
$m = 1$	0.1039	0.1002	0.1042	1.1934	0.7788	1.7328	3.8870	0.1237
$m = 3$	0.0271	0.1426	0.0271	0.9078	0.1568	0.3551	1.6966	0.0505
$m = 5$	0.0202	0.0511	0.0202	0.7020	0.0876	0.9940	1.7688	0.0583

Chapter 8

Appendix 4

Consider the four points of time p_1, p_2, p_3, and p_4. The tabes in this appendix contain the first two departure-count moments over the intervals defined by these four time points for the $Ph_t/Ph_t/s/c$ systems presented in Appendix 3, using the CM closure approximations and the numerically exact NEPMDEs.

Example 1

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.1: First Moments $m = 0.1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	2.0652	4.1135	6.1489	2.0214	4.0346	6.0428	2.1666	1.9566	1.7561
p_2	0	2.0484	4.0837	0	2.0132	4.0214	0	1.7457	1.5497
p_3	0	0	2.0354	0	0	2.0082	0	0	1.3532

Table 8.2: Second Moments $m = 0.1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	6.3390	21.0644	44.0123	6.1866	20.5940	43.1239	2.4629	2.2843	2.0602
p_2	0	6.2549	20.7956	0	6.1503	20.4884	0	1.7020	1.4994
p_3	0	0	6.1900	0	0	6.1278	0	0	1.0146

Table 8.3: First Moments $m = 0.5$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	8.0604	16.1646	24.2988	8.0655	16.1752	24.3145	-0.0625	-0.0655	-0.0644
p_2	0	8.1042	16.2384	0	8.1097	16.2490	0	-0.0684	-0.0654
p_3	0	0	8.1342	0	0	8.1393	0	0	-0.0624

Table 8.4: Second Moments $m = 0.5$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	67.8084	265.6247	600.2122	71.0778	272.0438	605.2188	-4.5997	-2.3596	-0.8272
p_2	0	68.7309	268.6040	0	71.9503	274.7723	0	-4.4745	-2.2449
p_3	0	0	69.3670	0	0	72.5372	0	0	-4.3704

Example 2

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.5: First Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	3.8906	7.8186	11.7504	3.8966	7.8411	11.7967	-0.1531	-0.2862	-0.3923
p_2	0	3.9280	7.8598	0	3.9445	7.9001	0	-0.4177	-0.5103
p_3	0	0	3.9318	0	0	3.9556	0	0	-0.6027

Table 8.6: Second Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	18.7787	68.0579	148.0946	19.1769	69.3198	150.6899	-2.0767	-1.8205	-1.7223
p_2	0	19.1537	68.8966	0	19.6548	70.4264	0	-2.5494	-2.1722
p_3	0	0	19.2373	0	0	19.7925	0	0	-2.8055

Table 8.7: First Moments $m = 3$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	5.0696	9.9779	14.5857	5.0696	9.9779	14.5856	-0.0009	0.0001	0.0009
p_2	0	4.9083	9.5161	0	4.9083	9.5160	0	0.0008	0.0016
p_3	0	0	4.6078	0	0	4.6077	0	0	0.0025

Table 8.8: Second Moments $m = 3$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	30.9107	110.1209	228.1993	30.8985	109.8401	227.8434	0.0395	0.2556	0.1562
p_2	0	29.1968	100.8594	0	129.1555	100.4403	0	0.1419	0.4172
p_3	0	0	26.0791	0	0	26.0148	0	0	0.2469

Example 3

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.9: First Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	11.1484	22.3663	33.6329	11.1659	22.3980	33.6754	-0.1564	-0.1419	-0.1261
p_2	0	11.2178	22.4845	0	11.2321	22.5095	0	-0.1274	-0.1110
p_3	0	0	11.2666	0	0	11.2773	0	0	-0.0947

Table 8.10: Second Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	137.4	526.4	1170.1	137.3	527.6	1173.6	0.0657	-0.2420	-0.2933
p_2	0	139.0	531.9	0	1139.0	533.0	0	0.0498	-0.2033
p_3	0	0	140.2	0	0	140.1	0	0	0.1139

Table 8.11: First Moments $m = 1.5$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	11.4281	22.8566	34.2851	11.4279	22.8563	34.2848	0.0019	0.0012	0.0009
p_2	0	11.4285	22.8570	0	11.4284	22.8569	0	0.0006	0.0002
p_3	0	0	11.4285	0	0	11.4285	0	0	0.0002

Table 8.12: Second Moments $m = 1.5$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	147.5	554.5	1222.4	145.3	552.8	1221.6	1.4925	0.3109	0.0676
p_2	0	147.4	554.3	0	145.3	552.8	0	1.4135	0.2726
p_3	0	0	147.5	0	0	145.3	0	0	1.4911

Example 4

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.13: First Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	2.4325	4.9565	7.5463	2.4303	4.9514	7.5380	0.0928	0.1041	0.1106
p_2	0	2.5240	5.1138	0	2.5211	5.1077	0	0.1150	0.1191
p_3	0	0	2.5898	0	0	2.5866	0	0	0.1231

Table 8.14: Second Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	8.2514	29.1666	63.7645	8.1893	29.0167	63.5223	0.7583	0.5166	0.3814
p_2	0	8.8052	30.9399	0	8.7330	30.7615	0	0.8258	0.5798
p_3	0	0	9.2169	0	0	9.1385	0	0	0.8570

Table 8.15: First Moments $m = 3$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	3.1552	6.4167	9.7498	3.1553	6.4168	9.7500	-0.0029	-0.0025	-0.0021
p_2	0	3.2615	6.5946	0	3.2615	6.5947	0	-0.0022	-0.0017
p_3	0	0	3.3332	0	0	3.3332	0	0	-0.0013

Table 8.16: Second Moments $m = 3$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	13.4198	48.6204	106.9861	13.2930	48.0755	105.6231	0.9538	1.1333	1.2904
p_2	0	14.2256	51.1837	0	14.0891	50.5907	0	0.9690	1.1721
p_3	0	0	14.7862	0	0	14.6413	0	0	0.9897

Example 5

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.17: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	1.1867	2.4166	3.6773	1.1860	2.4149	3.6746	0.0614	0.0701	0.0741
p_2	0	1.2299	2.4905	0	1.2289	2.4885	0	0.0786	0.0801
p_3	0	0	1.2607	0	0	1.2596	0	0	0.0816

Table 8.18: Second Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	2.5556	8.1092	16.8854	2.5509	8.1046	16.8973	0.1844	0.0561	-0.0701
p_2	0	2.7065	8.5585	0	2.7002	8.5487	0	0.2307	0.1147
p_3	0	0	2.8177	0	0	2.8106	0	0	0.2536

$p_1 = 34$, $p_2 = 38$, $p_3 = 42$, and $p_4 = 46$

Table 8.19: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	1.3095	2.6920	4.2964	1.3088	2.6906	4.2953	0.0540	0.0533	0.0273
p_2	0	1.3803	2.9815	0	1.3833	2.9884	0	-0.2174	-0.2310
p_3	0	0	1.6003	0	0	1.6054	0	0	-0.3211

Table 8.20: Second Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	3.0002	9.8505	22.5721	2.9930	9.8322	22.5424	0.2400	0.1865	0.1319
p_2	0	3.2602	11.7617	0	3.2660	11.7937	0	-0.1764	-0.2713
p_3	0	0	4.1125	0	0	4.1253	0	0	-0.3094

Example 6

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.21: First Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	1.6710	3.3976	5.1481	1.6702	3.3965	5.1477	0.0524	0.0327	0.0063
p_2	0	1.7266	3.4770	0	1.7264	3.4776	0	0.0137	-0.0158
p_3	0	0	1.7504	0	0	1.7512	0	0	-0.0448

Table 8.22: Second Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	4.3856	14.6541	31.0410	4.3909	14.6933	31.1670	-0.1217	-0.2669	-0.4045
p_2	0	4.6355	15.2983	0	4.6448	15.3520	0	-0.2013	-0.3497
p_3	0	0	4.7515	0	0	4.7658	0	0	-0.3008

Table 8.23: First Moments $m = 10$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	2.2906	4.5045	6.5856	2.2906	4.5045	6.5856	0	0	0
p_2	0	2.2139	4.2951	0	2.2140	4.2951	0	0	0
p_3	0	0	2.0811	0	0	2.0811	0	0	0

Table 8.24: Second Moments $m = 10$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	7.6105	24.9516	50.1839	7.6147	24.9713	50.2318	-0.0550	-0.0789	-0.0955
p_2	0	7.1891	22.9020	0	7.1933	22.9199	0	-0.0584	-0.0781
p_3	0	0	6.4872	0	0	6.4904	0	0	-0.0488

Table 8.25: First Moments $m = 0.2$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	0.4056	0.8148	1.2249	0.4043	0.8121	1.2208	0.3156	0.3341	0.3388
p_2	0	0.4092	0.8194	0	0.4078	0.8165	0	0.3525	0.3504
p_3	0	0	0.4101	0	0	0.4087	0	0	0.3483

Table 8.26: Second Moments $m = 0.2$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	0.5684	1.4748	2.7195	0.5664	1.4689	2.7085	0.3664	0.3983	0.4060
p_2	0	0.5759	1.4886	0	0.5737	1.4830	0	0.3756	0.3782
p_3	0	0	0.5779	0	0	0.5759	0	0	0.3414

Example 7

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.27: First Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	2.0173	3.9149	5.9474	2.0185	3.9313	5.9762	-0.0603	-0.4181	-0.4817
p_2	0	1.9107	3.9454	0	1.9166	3.9608	0	-0.3099	-0.3876
p_3	0	0	2.0338	0	0	2.0440	0	0	-0.4971

Table 8.28: Second Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	6.0517	19.1348	41.1180	6.1387	19.4800	41.8323	-1.4176	-1.7716	-1.7076
p_2	0	5.5372	19.4286	0	5.6409	19.7664	0	-1.8387	-1.7088
p_3	0	0	6.1504	0	0	6.2800	0	0	-2.0637

Table 8.29: First Moments $m = 4$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	3.8334	7.5951	11.4522	3.8342	7.5928	11.4383	-0.0199	0.0302	0.1222
p_2	0	3.7550	7.6005	0	3.7587	7.6003	0	-0.0977	0.0024
p_3	0	0	3.8507	0	0	3.8396	0	0	0.2898

Table 8.30: Second Moments $m = 4$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	19.3393	68.3463	149.3419	18.6870	65.6266	142.9498	3.4904	4.1443	4.4715
p_2	0	18.6831	68.7903	0	18.0583	65.8218	0	3.4600	4.5099
p_3	0	0	19.6493	0	0	18.8053	0	0	4.4881

Example 8

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.31: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	7.9169	15.8358	23.7551	7.9087	15.8190	23.7297	0.1043	0.1061	0.1069
p_2	0	7.9188	15.8382	0	7.9103	15.8210	0	0.1081	0.1086
p_3	0	0	7.9194	0	0	7.9107	0	0	0.1095

Table 8.32: Second Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	91.7371	321.2385	654.6721	80.8566	290.5898	626.0131	13.4565	10.5471	4.5780
p_2	0	91.7806	321.3410	0	80.8907	290.6662	0	13.4625	10.5533
p_3	0	0	91.7933	0	0	80.8997	0	0	13.4656

Example 9

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.33: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	3.8892	7.8098	11.7446	3.8834	7.7938	11.7225	0.1486	0.2049	0.1883
p_2	0	3.9206	7.8554	0	3.9104	7.8391	0	0.2609	0.2080
p_3	0	0	3.9348	0	0	3.9287	0	0	0.1554

Table 8.34: Second Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	18.3624	66.8450	146.4530	16.6848	63.3336	140.9333	10.0548	5.5444	3.9165
p_2	0	18.6561	67.6505	0	16.9039	64.0721	0	10.3657	5.5850
p_3	0	0	18.7982	0	0	17.0538	0	0	10.2291

Example 10

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.35: First Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	5.6092	11.2248	16.8430	5.6090	11.2241	16.8418	0.0048	0.0065	0.0068
p_2	0	5.6156	11.2337	0	5.6151	11.2328	0	0.0081	0.0077
p_3	0	0	5.6181	0	0	5.6177	0	0	0.0072

Table 8.36: Second Moments

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	37.6182	139.1095	303.3368	37.3419	138.1745	302.2564	0.7399	0.6767	0.3575
p_2	0	37.7125	139.3683	0	37.4280	138.4066	0	0.7601	0.6948
p_3	0	0	37.7505	0	0	37.4647	0	0	0.7628

Example 11

$p_1 = 16, p_2 = 20, p_3 = 24$,and $p_4 = 28$

Table 8.37: First Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	2.5855	5.1995	7.8438	2.5869	5.2091	7.8528	-0.0548	-0.1840	-0.1145
p_2	0	2.6140	5.2583	0	2.6222	5.2659	0	-0.3119	-0.1441
p_3	0	0	2.6443	0	0	2.6437	0	0	0.0224

Table 8.38: Second Moments $m = 1$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	9.2006	31.9604	68.8603	9.2846	32.3072	69.4458	-0.9044	-1.0735	-0.8432
p_2	0	9.3650	32.6774	0	9.5107	33.0174	0	-1.5327	-1.0300
p_3	0	0	9.5829	0	0	9.6706	0	0	-0.9076

Table 8.39: First Moments $m = 5$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	3.8604	7.7121	12.2905	3.8600	7.7111	12.2904	0.0093	0.0129	0.0004
p_2	0	3.8518	8.4301	0	3.8512	8.4304	0	0.0159	-0.0040
p_3	0	0	4.5784	0	0	4.5793	0	0	-0.0195

Table 8.40: Second Moments $m = 5$

	CA			NE			% Error		
	p_2	p_3	p_4	p_2	p_3	p_4	p_2	p_3	p_4
p_1	19.1778	68.7298	165.9516	18.9178	67.6156	163.9091	1.3746	1.6479	1.2461
p_2	0	19.2911	81.0758	0	18.9359	79.8778	0	1.8754	1.4998
p_3	0	0	25.7294	0	0	25.6416	0	0	0.3424

Chapter 9

Appendix 5

Example 1

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 25, m_A = 2, m_B = 3$

$\boldsymbol{\alpha} = [0.6, \quad 0.4], \quad \boldsymbol{\beta} = [0.4, \quad 0.3, \quad 0.3]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.4 & 0.5 \\ 0.6 & 0.1 & 0.3 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0.3 & 0.2 & 0.5 \\ 0.2 & 0 & 0.3 & 0.5 \\ 0.2 & 0.2 & 0 & 0.6 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1, \quad 2]$, $\boldsymbol{\mu} = [2, \quad 1, \quad 3]$.

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

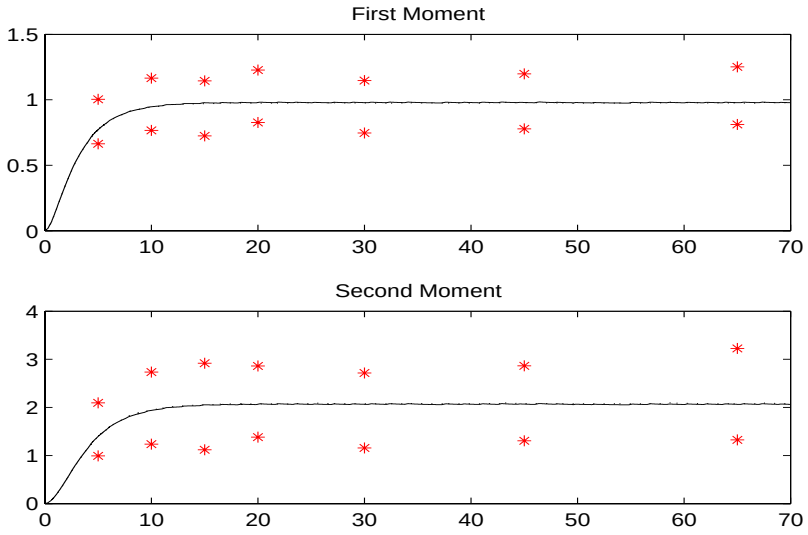
$s = 3, c = 25, m_B = 3$

$$\mathbf{b} = \begin{pmatrix} 0 & 0.7 & 0 & 0.3 \\ 0.2 & 0 & 0.2 & 0.6 \\ 0 & 0.3 & 0 & 0.7 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.25, \quad 0.5, \quad 0.25]$, $\boldsymbol{\mu} = [1, \quad 2, \quad 0.5]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.1: Node 2 Number-in-System Moments



Example 2

The $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 25, m_A = 2, m_B = 3$

$\boldsymbol{\alpha} = [0.6, \quad 0.4], \quad \boldsymbol{\beta} = [0.4, \quad 0.3, \quad 0.3]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0.4 & 0.6 \\ 0.5 & 0 & 0.5 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0.6 & 0.2 & 0.2 \\ 0.3 & 0 & 0.3 & 0.4 \\ 0.2 & 0.2 & 0 & 0.6 \end{pmatrix}$$

$\boldsymbol{\lambda}(t) = [2 + \cos(\pi t/4), \quad 3 + \cos(\pi t/7)]$, $\boldsymbol{\mu}(t) = [2, \quad 1.5 + \cos(\pi t/4), \quad 1.2 + \cos(\pi t/4)]$.

The $Ph_t/Ph_t/s/c$ parameters of node 2 are:

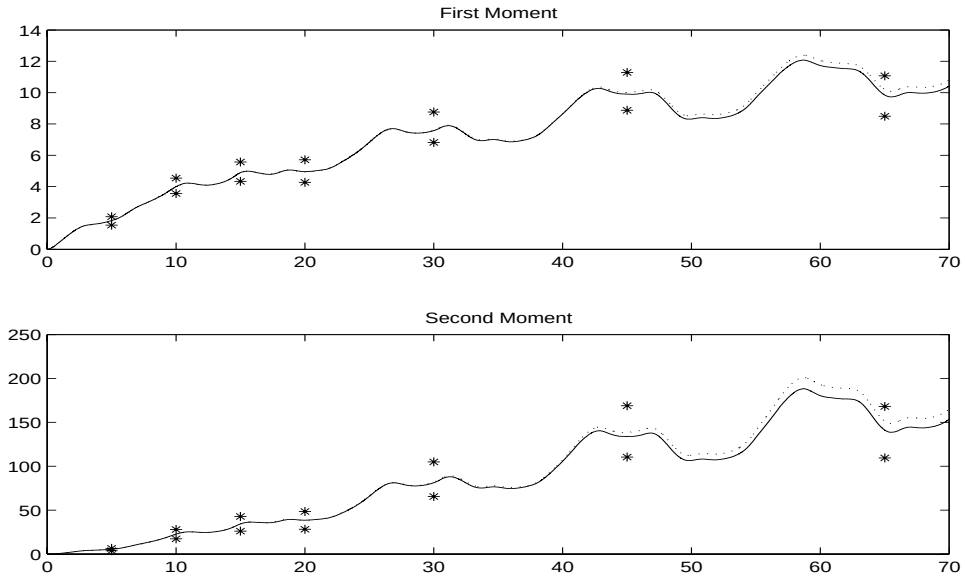
$s = 3, c = 25, m_B = 3$

$$\mathbf{b} = \begin{pmatrix} 0 & 0.6 & 0.2 & 0.2 \\ 0.3 & 0 & 0.3 & 0.4 \\ 0.2 & 0.2 & 0 & 0.6 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.3, \quad 0.5, \quad 0.2]$, $\boldsymbol{\mu}(t) = [1.3 + \cos(\pi t/2), \quad 1.1 + \cos(\pi t/4), \quad 1.4 + \cos(\pi t/8)]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.2: Node 2 Number-in-System Moments



Example 3

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 20, m_A = 2, m_B = 3$

$\boldsymbol{\alpha} = [1, \quad 0], \quad \boldsymbol{\beta} = [1, \quad 0, \quad 0]$

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [3, \quad 3]$, $\boldsymbol{\mu} = [9, \quad 9, \quad 9]$.

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

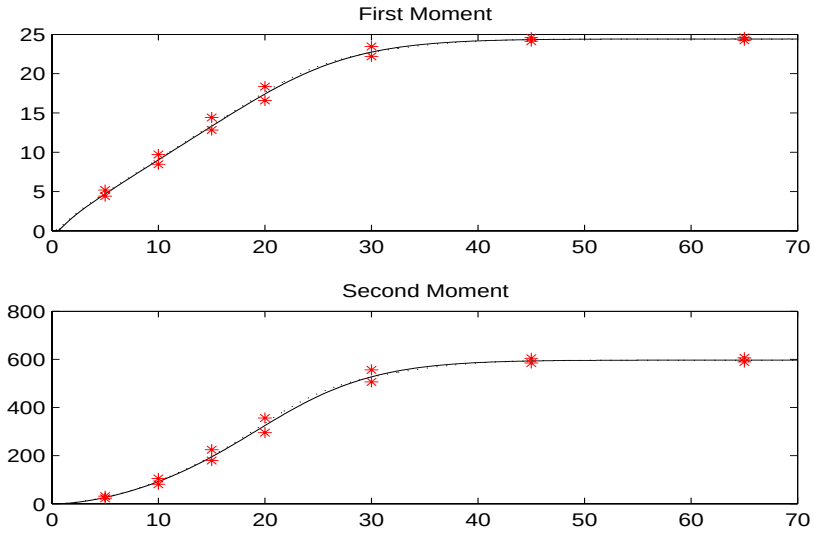
$s = 3, c = 25, m_B = 3$

$$\mathbf{b} = \begin{pmatrix} 0.2 & 0.2 & 0.5 & 0.1 \\ 0.3 & 0.2 & 0.1 & 0.4 \\ 0.2 & 0.2 & 0.2 & 0.4 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.3, \quad 0.2]$, $\boldsymbol{\mu} = [1, \quad 1, \quad 0.5]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.3: Node 2 Number-in-System Moments



Example 4

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 20, m_A = 2, m_B = 3$

$\boldsymbol{\alpha} = [1, \quad 0], \quad \boldsymbol{\beta} = [1, \quad 0, \quad 0]$

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [0.4, \quad 0.4]$, $\boldsymbol{\mu} = [1.2, \quad 1.2, \quad 1.2]$.

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

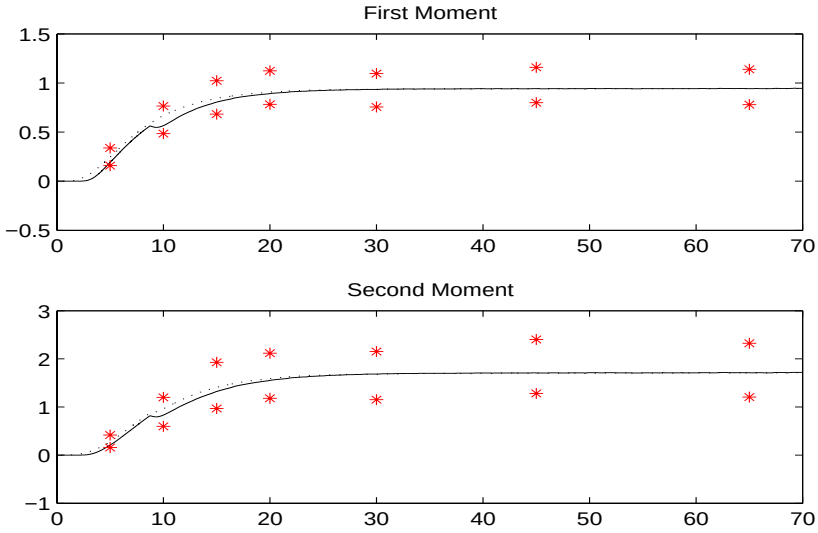
$s = 3, c = 25, m_B = 3$

$$\mathbf{b} = \begin{pmatrix} 0.2 & 0.2 & 0.5 & 0.1 \\ 0.3 & 0.2 & 0.1 & 0.4 \\ 0.2 & 0.2 & 0.2 & 0.4 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.3, \quad 0.2]$, $\boldsymbol{\mu} = [1, \quad 1, \quad 0.5]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.4: Node 2 Number-in-System Moments



Example 5

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 25, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.1, \quad 0.6], \quad \boldsymbol{\beta} = [0.6, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0.1 & 0.2 & 0.7 \\ 0.2 & 0 & 0.1 & 0.7 \\ 0.2 & 0.2 & 0 & 0.6 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1, \quad 3, \quad 6], \boldsymbol{\mu} = [1, \quad 0.5].$

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

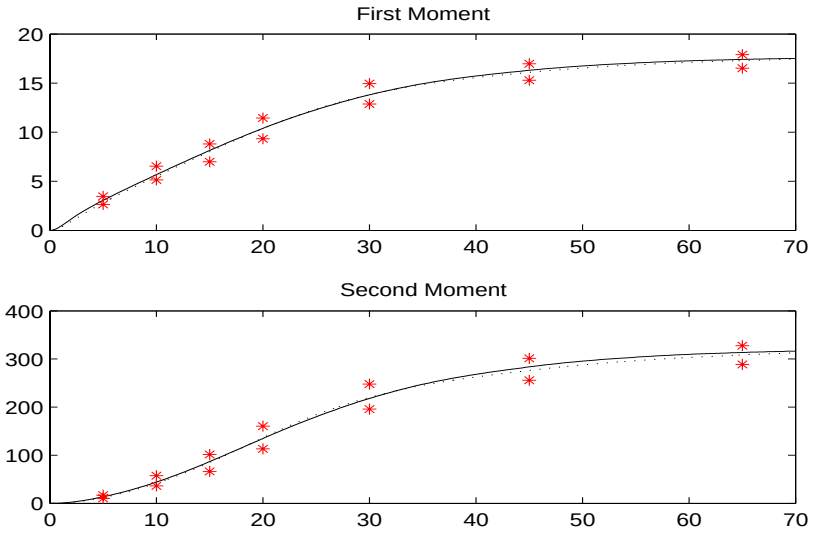
$s = 3, c = 20, m_B = 2$

$$\mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.5]$, $\boldsymbol{\mu} = [1, \quad 0.5]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.5: Node 2 Number-in-System Moments



Example 6

The $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 25, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.1, \quad 0.6], \quad \boldsymbol{\beta} = [0.6, \quad 0.4]$

$$\mathbf{a} = \begin{pmatrix} 0 & 0.1 & 0.2 & 0.7 \\ 0.2 & 0 & 0.1 & 0.7 \\ 0.2 & 0.2 & 0 & 0.6 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$\boldsymbol{\lambda}(t) = [2 + \sin(\pi t/9), \quad 3 + 2\sin(\pi t/10), \quad 6 + 3\sin(\pi t/12)]$, $\boldsymbol{\mu}(t) = [2 + \sin(\pi t/8), \quad 1 + 0.5\cos(\pi t/8)]$.

The $Ph_t/Ph_t/s/c$ parameters of node 2 are:

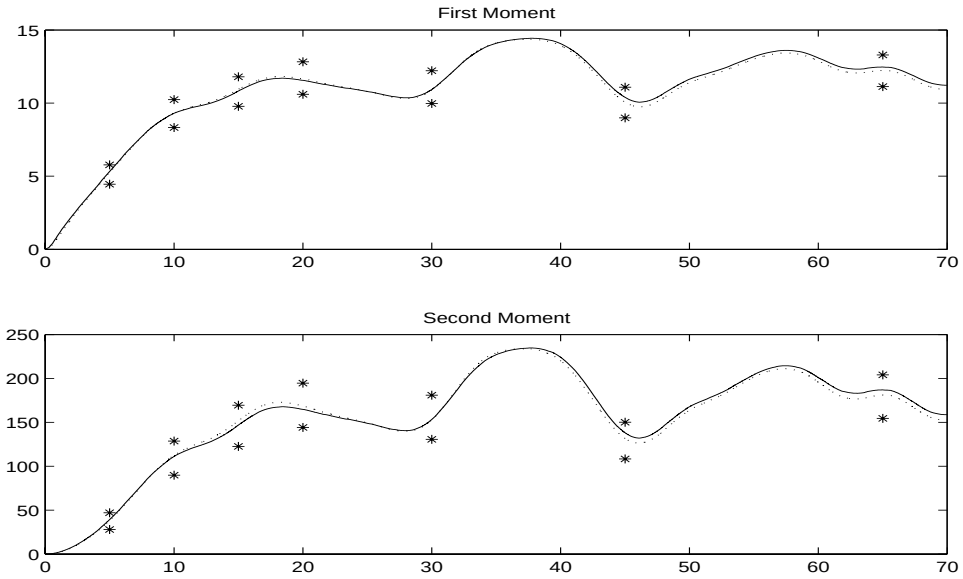
$s = 3, c = 20, m_B = 2$

$$\mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.5]$, $\boldsymbol{\mu}(t) = [2 + \cos(\pi t/10), \quad 1 + 0.5\sin(\pi t/10)]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.6: Node 2 Number-in-System Moments



Example 7

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 20, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.5, \quad 0.5]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\lambda} = [9, \quad 9, \quad 6]$, $\boldsymbol{\mu} = [3, \quad 1.5]$.

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

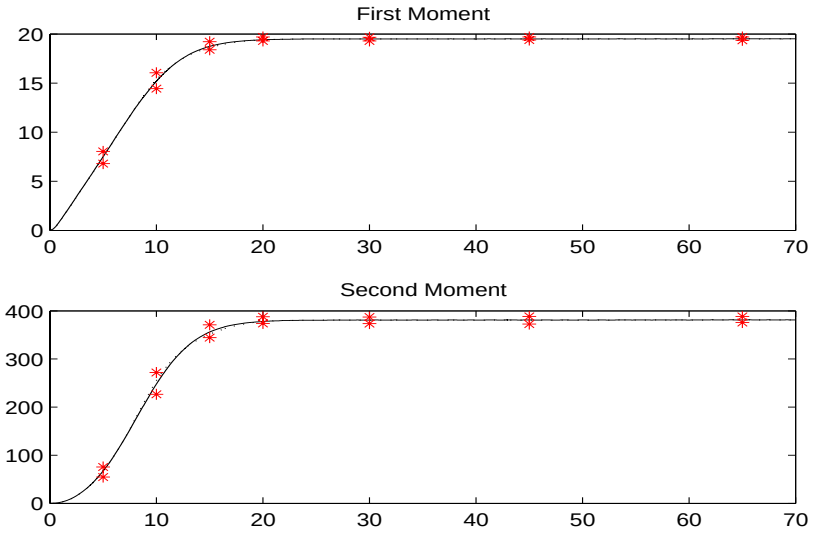
$s = 3, c = 20, m_B = 2$

$$\mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.5]$, $\boldsymbol{\mu} = [1, \quad 0.5]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.7: Node 2 Number-in-System Moments



Example 8

The stationary $Ph_t/Ph_t/s/c$ parameters of node 1 are:

$s = 3, c = 20, m_A = 3, m_B = 2$

$\boldsymbol{\alpha} = [0.3, \quad 0.3, \quad 0.4], \quad \boldsymbol{\beta} = [0.5, \quad 0.5]$

$$\mathbf{a} = \begin{pmatrix} 0.1 & 0.1 & 0.3 & 0.5 \\ 0.3 & 0.1 & 0.1 & 0.5 \\ 0.3 & 0.2 & 0.1 & 0.4 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\lambda} = [1.2, \quad 1.2, \quad 0.8]$, $\boldsymbol{\mu} = [0.4, \quad 0.2]$.

The stationary $Ph_t/Ph_t/s/c$ parameters of node 2 are:

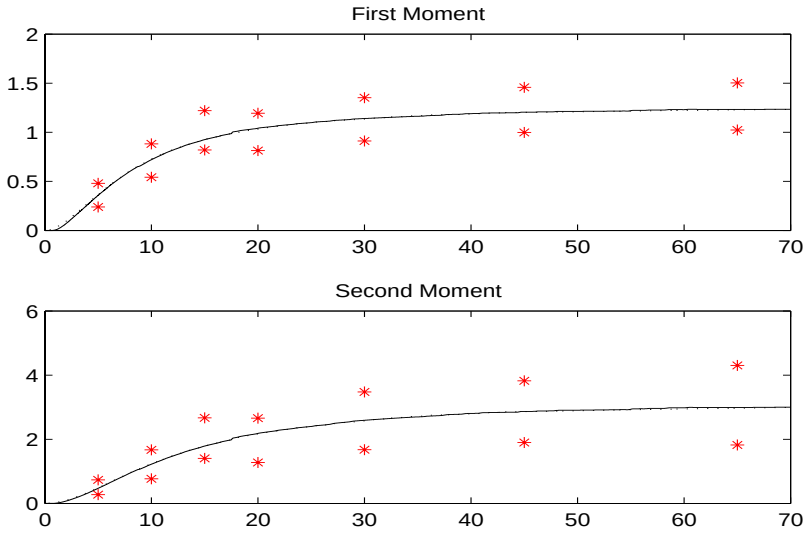
$s = 3, c = 20, m_B = 2$

$$\mathbf{b} = \begin{pmatrix} 0.1 & 0.6 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

$\boldsymbol{\beta} = [0.5, \quad 0.5]$, $\boldsymbol{\mu} = [1, \quad 0.5]$.

The SMI and FMI are determined by $\Theta = 70, n_1 = 8, n_2 = 30$ where Θ, n_1 and n_2 are defined in Section 4.6.2.

Figure 9.8: Node 2 Number-in-System Moments



Chapter 10

Bibliography

Bibliography

[1] S. L. Albin. Approximating a Point Process by a Renewal Process II:Superposition Arrival Processes to Queues *Operations Research Vol.32,No. 5,* September-October (1984).

[2] S. Asmussen, Applied Probability And Queues. *New York : Springer,* c2003.

[3] G. M. Clark. Use of Polya distributions in approximate solutions to nonstationary $M/M/s$ queues *Comm ACM 24 (4)* ,206-218 (1981) .

[4] D. J. Daley. Queueing Output Processes *Adv, Appl. Prob. 8,* 395-415 (1976)

[5] M. J. Johnson and M. R. Taaffe . The Denseness of Phase Distributions. *School of Industrial Engineering Purdue University Research Memorandum*, (1988) 88-20.

[6] M. J. Johnson and M. R. Taaffe. Matching Moments to Phase Distributions:Mixture of Erlang Distributions of Common Order *Commun.Statis,-Stochastic Models, 5(4)*,711-743 (1989).

[7] M. J. Johnson and M. R. Taaffe. Matching Moments to Phase Distributions: Nonlinear Programming Approaches *Stochastic Models, 6, 2*,259-281 (1990).

[8] S. Karlin and H. M. Taylor. A First Course in Stochastic Processes. *New York : Academic Press,* (1975).

[9] APPENDIX W. Nasr and M. R. Taaffe. Chapter 2 (Approximating System-State Probability Mass Functions for the Time-Dependent $Ph_t/Ph_t/s/c$ Queueing Model)

[10] B. L. Nelson and M. R. Taaffe. The $Ph_t/Ph_t/\infty$ Queueing System: Part I - The Single Node. *INFORMS JOC 16 (3)*, (2004) 266-274.

[11] M. F. Neuts, Probability. *Allyn and Bacon, INC,* 1973.

[12] K. L. Ong and M. R. Taaffe. Approximating nonstationary $Ph_t/M_t/s/c$ queueing systems. *Ann. Oper. Res. 9*,(1987) 103-116.

[13] K. L. Ong and M. R. Taaffe. Approximating $Ph_t/Ph_t/1/c$ nonstationary queueing systems. *Mathematics and Computers in Simulation 30*, (1988) 441-452.

[14] J. E. Rueda and M. R. Taaffe. The $Ph_t/Ph_t/s/c$ Queueing Model and Approximation. *Technical Report, The Grado Department of Industrial and Systems Engineering*, (2004).

[15] W. Whitt. The Queueing Network Analyzer *Bell System Technical Journal 62,* 2779-2815 (1983).

[16] W. Whitt. Approximating a Point Process by a Renewal Process I:Two Basic Methods *Operations Research Vol.30,No. 1,* January-February (1982).

[17] W. Whitt. Approximations for the GI/G/m Queue *Production and Operations Management Vol. 2,No. 2,* Spring (1993)

[18] W. Whitt. Approximations for Departure Processes and Queues in Series *Naval Research Logistics Quarterly, Vol. 31,*pp. 499-521 (1984).

[19] W. Whitt. Heavy Traffic Limit Theorems for Queues: A Survey. *Mathematical Methods in Queueing Theory* (Lecture Notes in Econ. and Math. Systems No. 98) pp. 307-350 (1974).