mathematics

MDPI

Article

Approximating Large Scale Queueing System: An Overview and Directions for Further Research

Attahiru Sule Alfa [1,2] and Haitham Abu Ghazaleh [3,*]

[1] Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB R3T 5V6, Canada; attahiru.alfa@umanitoba.ca
[2] Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Pretoria 0002, South Africa
[3] Department of Engineering and Computer Science, Tarleton State University, Stephenville, TX 76402, USA
* Correspondence: abughazaleh@tarleton.edu

Abstract

Queueing systems are very well studied in the literature. There are a good collection of mathematical tools for studying a variety of queueing related problems. Occasionally, however, researchers doubt the applicability of some of the existing mathematical models to complex and/or large scale queueing systems they encounter in practical situations, such as in communication networks. This has led to the use of simulations and other types of approximation models for dealing with large scale systems. When approximate solutions are presented, purists often state that such solutions do not quite capture all the nuances of the real system. The question is when does a queueing system become large and require approximations, given the abundant literature on exact mathematical methods for analyzing them, and which approximation methods are appropriate to use? Is there a clear approach to readily determine whether a queuing system is large and/or complex? Thus, there is a need for an understanding of how to determine when a system of queues is deemed large and/or complex, and the need for approximation techniques in their analyses. In this paper, we briefly address this issue by providing an overview of existing tools that can be applied in analyzing any queueing problems that are deemed large and/or complex, while also focusing on approximation techniques for systems that are modeled as Markov Chains. In addition, we show that deriving a specific measure for categorizing a queueing system as large and/or complex can be an arduous task to achieve, and we propose the foundations for investigating such measures in future works. Finally, we conclude with open questions and problems with respect to modeling large scale and/or complex queueing systems and their approximations, to promote further research of effective approximation techniques for practitioners and analysts.

Keywords: queueing theory; Markov chains; queueing analysis

MSC: 90-08

Check for updates

Academic Editor: MinJae Lee

Received: 7 August 2026
Revised: 8 September 2026
Accepted: 10 September 2026
Published: 13 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

1.1. Background

Large scale and complex queueing systems are well discussed in the literature because most types of queueing systems that are pertinent to industrial and real life problems are of that nature. Such systems are encountered in communication networks, transportation systems, manufacturing systems, etc. There are several queueing models with very well

developed analytical techniques proposed in the literature, but most of them are appropriate or limited to single node applications with limited complex structures, or for limited queueing networks with common protocols/rules. Large networks and systems that consist of several queues that involve various configurations, interactions, complex protocol demands and it operations, are often quite challenging to model and analyze using existing mathematical models and tools with their limited capabilities. Thus, it is important that the next stages of research in queueing theory focus on investigating efficient and effective mathematical tools for analyzing large scale and/or complex queueing systems, which is the motivation of this work.

In this paper, we further share some insights on how one may quantitatively assess the scale and complexity of a queueing model [1,2]. Such measures are not quite well-defined yet in the literature and in a precise form. Most statements that discuss the subject often leave the definition and description of large scale complex queueing systems to be interpreted in a subjective and qualitative manner. The other challenge with deriving such metrics is that it may need to be defined in terms of the existing computing powers. Our aim is to attempt to understand what leads researchers to describe/classify a queueing system as *large scale* and/or *complex*. This would ultimately assist in developing tools for analyzing such systems for the practitioner who needs numerical values for design purposes. Various approximation techniques have been proposed for analyzing queueing models of large scale and/or complex systems, and we present some of the prominent techniques in this paper along with recommendations to further this field of research.

1.2. Scope of Work

The aim in this paper is to:

- describe the challenges with characterizing the scale and/or complexity of queueing models as well as propose an approach to develop these quantitative measures,
- review some prominent approximation techniques for analyzing large scale and/or complex queueing systems that are modeled as Markov Chains, and
- propose/encourage further research directions with developing practical techniques for analyzing large scale and/or complex queueing systems.

Before addressing both these aims in the paper, Section 2 begins with describing the common approaches used in queueing analyses. We further outline the merits and limitations of utilizing the different approaches, especially when working with large scale and/or complex queueing systems.

In Section 3, we outline the common types of queueing systems and how their analyses can potentially be non-tractable due to their size and/or complexity. We also demonstrate the difficulty of determining "how complex" a system is, and the challenges of associating a quantitative measure related to the system's complexity. We restrict our investigation to addressing the dimensionality of queueing models and defer the measures of complexity to future works. We further propose the different features of the model that can be used to characterize the size of the system and how it ultimately relates to the complexity of its analysis. In Section 4, we present known mathematically based approximation techniques that have strong potentials for analyzing large scale and/or complex queueing systems and based on their classification. We also propose a relatively simple approximation technique to demonstrate the possibilities for further research in this field. We finally conclude in Section 5 with directions and suggestions for future work in the field of queueing analyses.

In this work, we also note that the efforts of working with large scale and/or complex queueing systems can be closely related to the processor used in the computation. For example, a network of size N modeled as a queueing system may be complex to analyze using

computers with limited processing capabilities, whereas the analysis may be relatively trivial with adequate processing resources. We do not consider such cases in our work.

2. Common Approach to Queueing Analyses

2.1. Queueing Systems as Markov Chains

In principle, nearly all queueing systems can be represented/modeled as a Markov chain, including those that are multivariate with a large state space. Thus, we focus on systems that are modeled as Markov Chains and their corresponding analyses. Solving the linear equations associated with the Markov chain provides the main results that are generally used for understanding the behavior of the associated queueing system. Markov chains with large dimensions are usually difficult to analyze purely because of size. However, the analysis can be made more feasible if tools for managing their size are incorporated into the analysis, thus the need to explore the use and development of such tools. One can further conclude from this argument that the number of variables (or multi-variables) can be used as part of the measure for quantifying the effort in analyzing a queueing system, since the scale of the model representing the system is directly related to the number of states/variables that define the Markov Chain. Another factor to consider when modeling queueing systems as Markov Chains is the amount of effort required to generate the elements of the transition/generator matrix. This is especially significant when dealing with systems that involve multiple rules/protocols.

Dealing with large Markov chains is not new in the queueing literature. There is a plethora of literature on this subject that will be outlined in the subsequent Sections. Some of the well-known methods used for analyzing systems modeled as large Markov chains include *truncation* (in the case of infinite Markov chains), *decomposition* of the system, and *aggregation* or *compression* which is viewed as the *dual* of *decomposition*. A key factor in these approaches are the strength (or weakness) of the connectivity of the decomposed systems or the weakness (or strength) of the aggregation of the components that have been compressed. Our motivation in this paper is in finding ways to extend these existing approaches in the face of the modern day large size queueing systems.

In this paper, we consider both discrete-time and continuous-time Markov chains in our presentation of the topics and relative to the results reviewed from the given references.

2.2. Methods of Queueing Analyses

The four main approaches often used for analyzing queueing systems are as follows:

- Analytical methods:based on mathematical models, exact or approximate. Generally the final mathematical models are often developed as some form of a Markov chain.
- Approximation approaches: based on mathematical models as in the first item above, using some form of decomposition/aggregation.
- Simulation models: usually based on some form of Monte Carlo approach. This approach is essentially a sampling-based procedure with limited application, depending on the statistical aspect of the samples.
- Machine learning methods: based on artificial intelligence. This approach is fairly new in the queueing literature compared to the other three.

We elaborate further on these different techniques in later sections and present some of the known approaches for dealing with these types of queueing systems, identify their limitations and propose future directions for improving the analysis. An understanding on how to classify a queueing system as very large and complex can aid with selecting the appropriate analytical techniques and approximations.

2.3. Simulation of Queueing Systems

The use of simulation have been quite popular among researchers, especially when dealing with the analysis of complex systems [3]. It allows one to setup a virtual experiment of the system, mostly as a discrete event system [4], that can be repeatedly run with different variations. It is also used for verifying the analysis returned by the queueing models. Many researchers have often preferred to conduct their performance analysis using simulation without a mathematical model due to the relative simplicity in setting up a simulation. This also highlights one of the challenges in working with the development of mathematical models, which can be laborious for large and complex systems. In fact, various researchers have indicated the desire to utilize simulations in place of mathematical models for complex systems [5,6]. Hence, our motivation in this paper to promote engagement in seeking methods to advance the mathematical analysis of large scale and complex queueing systems. Simulation and mathematical modeling of the same system are two different approaches for studying a system, but one does not replace the other and they instead complement each other [7].

The increase in computing power has also driven the interest in using simulations for the performance analysis of queueing systems. Early works on computer-based simulations, such as [8–11], focused on techniques that yield efficient execution of the simulation processes due to the limited processing capabilities, including recent works such as [12] that have proposed techniques that specifically target large scale queueing systems. Nowadays, various methods and software tools have been developed and adopted for simulating complex systems such as Python programming [13] (see also Simpy), object-oriented simulation [14], stateflow-based simulation [15], and using the SimEvents library in MATLAB Simulink. These different tools have certainly helped to overcome the burdens with developing the simulations for very large and complex queueing systems, with some including a visual/graphical mechanism for setting up the simulation environment. While these tools have helped to simplify the setup of the simulations, they all generally suffer from the same issue of requiring extensive amounts of time (and computing resources) for running each instance of the simulation, and especially for the purpose of steady-state analyses.

Even though the use of simulations is not within the scope of this work on the tools for approximating queueing systems formulated using mathematical models, we discuss its use in this paper due to it being a tool that is often used in conjunction with mathematical modeling. We further note the potential to utilize a hybrid approach (both simulation and mathematical modeling) for analyzing large scale and complex queueing systems that take advantage of the increasing computing capabilities.

2.4. Use of Machine Learning in Queueing Analyses

Another means of analyzing the performance of large scale and complex systems is the use of machine learning (ML) algorithms and its derivatives [16]. The popularity of such tools has surged in the recent years due to its diverse applications and with it being embedded in today as computing hardware capabilities. It had further helped accelerate the efforts in performing analyses on complex systems without the need of developing specific models for such systems [17]. Such techniques have gained the attention of many researchers, including ourselves. Recent works, such as [18–22], have proposed techniques for evaluating common performance measures (i.e., mean number in the system and mean waiting times) in queueing systems using ML-based algorithms. The results from these works have demonstrated the potential of applying these ML-based techniques for analyzing complex systems. In our recent work [23], we also proposed a novel ML-based tool that potentially can be applicable to any queueing system with finite capacity and involves tractable computations. That work needs to be developed further. Although

a thorough coverage of ML-based techniques are not within the scope of this paper, we include a brief summary of its application in analyzing queueing systems. This is for the sake of identifying alternative techniques to the reader, and to help contrast the merits between such techniques and mathematical analyses.

Despite the growing popularity of utilizing ML-based algorithms for analyzing complex systems, such methods do have their limitations. The main limitation being its reliance on real data for conducting the analysis and its sensitivity to the data. Thus, ML-based techniques are more suited for analyzing existing systems rather than investigating conceptual and yet-to-be implemented systems. One can argue that the availability of reliable and real data can also be a concern when conducting simulations or developing mathematical models of the systems under investigation. However, the study of systems using either simulation or mathematical modeling are often done with certain assumptions being made on its state transitional behaviors whereby it is modeled using a particular probability distribution, and thus does not strictly require access to real data for conducting the analysis. Another important distinction between using mathematical models and ML algorithms for analyzing complex systems is that mathematical models are more useful in developing a conceptual and theoretical understanding of the system under investigation, whereas ML-based approaches can only provide limited insights on the system's performance and specific to the data utilized in the algorithms. Nevertheless, ML-based techniques are more effective in yielding immediate results on the system's performance without imposing any, and potentially unrealistic, assumptions.

The ML-based techniques presented in the literature are also currently limited to evaluating common performance measures, namely mean number and waiting time in the system. With mathematical models, and simulations, one can ultimately evaluate the steady-state probability distribution of the system which can further be used to calculate other and specific metrics. Hence, one potential area of research is to investigate ML-techniques that can estimate the steady-state probability distribution of the system and to be used for calculating other performance metrics, similar to those done using mathematical models.

3. Queueing Models & Their Size of Variables

In this section, we demonstrate how the number of defining variables in a system affect the size of the queueing model. We later discuss the effect of other factors such as the time parameter (i.e., continuous or discrete) and the rules/protocols for operating the system. Indeed, as the number of variables increases, a system becomes more complex. We point out which variables make the system complex and can be dropped in a model with minimal impact on the analysis of the system, i.e., how to significantly reduce the problem size and minimize the impact on model representation. Associated with these approximations will be some bounds.

There are special classes of infinite Markov chains that have very well-known structures that lend themselves analytically using well established techniques. We call one class of Markov chains that fall into that structure as having analysis techniques based on *Matrix-Analytic Methods*. In general, there are three distinct structures and are as follows:

- the Quasi-birth-death (QBD) types with transition matrices of the form

$$P = \begin{bmatrix} B & C & & & & \\ E & A_1 & A_0 & & & \\ & A_2 & A_1 & A_0 & & \\ & & A_2 & A_1 & A_0 & \\ & & & \ddots & \ddots & \ddots \\ & & & & & \end{bmatrix}, \tag{1}$$

- the M/G/1 type transition matrix of the form

$$P = \begin{bmatrix} B_0 & B_1 & B_2 & B_3 & B_4 & B_5 & \cdots \\ C_0 & A_1 & A_2 & A_3 & A_4 & A_5 & \cdots \\ & A_0 & A_1 & A_2 & A_3 & A_4 & \cdots \\ & & A_0 & A_1 & A_2 & A_3 & \cdots \\ & & & \ddots & \ddots & \ddots & \ddots \\ & & & & & & \end{bmatrix}, \tag{2}$$

- the GI/M/1 type transition matrix

$$P = \begin{bmatrix} B_0 & A_0 & & & & \\ B_1 & A_1 & A_2 & & & \\ B_2 & A_2 & A_1 & A_0 & & \\ B_3 & A_3 & A_2 & A_1 & A_0 & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \tag{3}$$

where P is the transition probability matrix associated with the queue length or number in the system (depending on the setup of the model), and the block matrices of P are defined accordingly.

These three classes of Markov chains have a long-standing and well established tools for their analysis, i.e., the *Matrix-Analytic Methods* (MAM). For more details on these classes of Markov chain, see ([24,25]). In general, when a queueing system model can be set up as one of these three types of Markov chains, there is most likely a tool in the literature for analyzing it; especially if the level dependence aspect is not an issue, i.e., if it is level homogeneous implying that the rows of the matrices are repeating. The solution techniques are very dependent on obtaining one of the two matrices, R and/or G. For the QBD, the matrices are obtained as the minimal non-negative solutions to the following equations:

$$R = A_0 + RA_1 + R^2A_2 \tag{4}$$

and

$$G = A_2 + A_1G + A_0G^2. \tag{5}$$

The continuous time equivalent of these matrices have similar structures and the related equations are

$$0 = A_0 + RA_1 + R^2A_2 \tag{6}$$

and

$$0 = A_2 + A_1G + A_0G^2, \tag{7}$$

where instead of the transition probability matrix P in discrete time we have the generator matrix Q.

Ultimately, the size of the matrices R and G determine the numerical efforts required in solving for these Markov chains, hence the limitations regarding complexity.

Matrices R and G with larger dimensions would require higher computational efforts. However, there are efficient techniques that have been well studied in the literature for these computations. See the extensive literature on Matrix-analytic methods, especially those published in the MAM conferences. One limitation often pointed out with MAM tools with respect to large scale systems is the sizes of the R and G matrices, however extensive research has been carried out in those regards.

One major limitation with the classical MAM is when the Markov chain is of finite size, namely $K < \infty$, with the generator matrix given as

$$Q = \begin{bmatrix} B & C & & & & \\ E & A_1 & A_0 & & & \\ & A_2 & A_1 & A_0 & & \\ & & A_2 & A_1 & A_0 & \\ & & & \ddots & \ddots & \ddots \\ & & & & A_2 & D \end{bmatrix}. \tag{8}$$

The standard well-known tools cannot not be readily applied to such cases. One needs to employ other techniques such as the tools developed by Naumov [26,27] which involves obtain R and S as the minimal non-negative solutions to the following equations

$$0 = A_0 + RA_1 + R^2A_2 \tag{9}$$

and

$$0 = A_2 + SA_1 + S^2A_0, \tag{10}$$

and then combine those matrices into four sets of linear equations that are then used to solve for the stationary distribution of the Markov chain where $\mathbf{x}Q = \mathbf{0}$, $\mathbf{x1} = 1$, and the vector $\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_K]$, satisfy

$$\mathbf{x}Q = \mathbf{0} \tag{11}$$

if and only if

$$\mathbf{x}_k = \mathbf{a}R^{k-1} + \mathbf{b}S^{K-k-1},\ 0 < k < K, \tag{12}$$

hold and the vectors $\mathbf{x}_0, \mathbf{x}_K, \mathbf{a}, \mathbf{b}$ satisfy the linear system

$$\mathbf{x}_0B + \mathbf{a}E + \mathbf{b}S^{K-2}E = 0 \tag{13}$$

$$\mathbf{x}_0C + \mathbf{a}(A_1 + RA_2) - \mathbf{b}S^{K-1}A_0 = 0 \tag{14}$$

$$\mathbf{x}_KD + \mathbf{a}R^{K-2}A_0 + \mathbf{b}A_0 = 0 \tag{15}$$

$$\mathbf{x}_KA_2 - \mathbf{a}R^{K-1}A_2 + \mathbf{b}(A_1 + SA_0) = 0 \tag{16}$$

This result can be simplified further depending on the structures of matrix Q, especially when the behavior at the boundary is simpler, as shown in [26]. In addition, Naumov et al. in [28] have provided a more comprehensive presentation of matrix analytical methods for performance analysis of telecommunication systems. The book provides up-to-date presentation of tools that are useful for analyzing telecommunication systems. This tool should be explored further in solving large scale Markov chains with a QBD structure.

In the next sections, we present simpler versions of the finite Markov chains usually encountered in queueing systems. They occur more often in some classes of queueing system for which the method presented above may not be necessary.

3.1. The Poisson Types of Queues

The single server types of queues with Poisson arrivals and Exponential services are of the simplest types of queues to analyze. We will use that class to assist us in presenting the impact of size based on Markov chain representations; those that have finite size and can be directly analyzed using MAM.

3.1.1. The M/M/1/K System

This is a system that can be modeled using only one variable; the Markov chain can be written as $\{L_t; 0 \leq L_t \leq K\}, t \geq 0$, where L_t is the number of items in the system at time t. The dimension of the generator matrix associated with this model is size $(K+1) \times (K+1)$, implying we need to solve $K+1$ linear equations, and normalizing the result. The number of non-zero elements in this generator matrix is about $3(K-1)+4$, or approximately $3K$ for simplicity, which makes the analysis simpler than that of a full generator matrix. If we consider the elements of the states $\{1, 2, \cdots, K-1\}$ of this Markov chain, there are $K-1$ of the $K+1$ rows that are repetitive. Hence, all that is needed are 3 elements and boundary conditions of this model in order to analyze this queueing system. Therefore, this is a very simple queueing model that is neither large nor complex, regardless of the size K.

If we allow the system to be level dependent (i.e., a system where the transitions are state-dependent and based on additional and complex rules/protocols) as in the general birth-death process, we will need to generate all the $3(K-1)+4$ elements of the generator matrix. Since each of the rows of the matrix representing the generator matrix of the Markov chain for this system could have lost its repetitive structure, this makes the analysis slightly more complex compared to the earlier case of level independence. Thus, the operational aspects of the queueing system, i.e., the rules and strategies, could make a queueing model more complex even if the dimension is moderate. A good example of this are retrial queues for which the Markov chains are level-dependent. In such cases the Markov chain transition matrix is no longer repeating its block matrices, and this is a very difficult problem to analyze. Some approximations have been developed for such queues. Two references which deal with such cases are [29,30]. Both references provide a long list of other earlier bibliography that include [31–33], and several others.

Let us instead consider a simple multiclass system, and suppose the arriving items in this M/M/1/K queue are of two types, with a priority system in analysis. This queue would be M(1), M(2)/M/1/K system with type 1 and type 2 classes of items. In this case, the dimension of the generator matrix has increased from $(K+1) \times (K+1)$ for the M/M/1/K to $\frac{1}{2}(K+2)(K+1) \times \frac{1}{2}(K+2)(K+1)$ for the M(1), M(2)/M/1/K system; a significant jump! One interesting observation to note in this case is that the dimension of the generator matrix has increased considerably, even though the number of items that could be in the system has not changed. The reason is due to the arrangement of the state space that requires an additional dimension, or state variable, to jointly capture the multiple classes of entities in the system. In addition, if we are interested in the type of item in service at each time, then the dimension of the generator matrix further increases to $(K+2)(K+1) \times (K+2)(K+1)$, which is another significant jump in the size of the generator matrix.

3.1.2. The Batch $M^X/M^Y/1/K$ System

The dimension of the generator matrix associated with this $M^X/M^Y/1/K$ system is still $K+1$, similar to the case of the M/M/1/K, but the number of non-zero elements is now approximately $K(X+Y+1)$. Without considering impact of the non-zero elements on the analysis, the size of this Markov chain is practically the same as that of M/M/1/K. If we consider the elements of the states $\{1, 2, \cdots, K-1\}$, they are all repetitive, and thus all that is needed are $X+Y+1$ elements and boundary conditions of this system in order to analyze this queueing system. Hence, this is a relatively simple queueing model that is neither large nor complex, but larger than that of the M/M/1/K system if we compare the number of non-zero elements.

If we instead allow the $M^X/M^Y/1/K$ system to be level dependent as in the general birth-death process, we will need to generate all the $K(X+Y+1)$ elements of the gener-

ator matrix, making this slightly more complex than the earlier case. This is due to the operational aspects of the queueing system, i.e., the rules and strategies that could make a queueing model more complex even if the dimension is moderate. Hence, the elements of each row of this generator matrix could be different for each row, i.e., level-dependent.

While all these queueing systems have QBD structures, direct use of the R and G matrices are not the most convenient approach to analyze them, partly because of the boundary behaviors with two finite boundaries. In addition, they have strictly finite size, which could be huge.

3.1.3. Rules and Protocols

The rules/protocols adopted in a system could further increase the complexity of a queueing model. It's generator matrix would have been manageable without additional rules that complicate it. The main challenge with such systems is creating the associated Markov chain generator matrix that properly incorporates the rules into the model. For example, modeling a system with multiple classes of arrivals and/or service behaviors tends to increase the size of the generator matrix and especially in the interest of keeping track of each class in the system. Such additions can complicate the system since there could be several subsystems within a Markov chain, keeping in mind that this would also affect the protocol/rules used to operate the system. Thus, the issue related to how complex a queueing model may depend on the protocols adopted in the system rather than only on the size of the transition/generator matrix of the model, and quantifying the level of complexity for such systems can be quite challenging.

The idea of using Petri-net to incorporate the rules has been explored in the literature [34]. But its role is not usually clear at times in such situations.

3.1.4. Impact of Non-Zero Elements

Even though the dimensions of the generator matrices for the two queues described previously are of the same (small) size, the non-zero elements could be significantly larger in the case of the batch system. Ultimately, the number of non-zero elements could increase to a maximum of $(K+1) \times (K+1)$ that can further impact the complexity of the analysis.

3.2. Multiserver Queues, e.g., M/M/k/K

If we consider the queueing systems above with more than one server, e.g., the M/M/k/K (k servers and K space), the dimension of the matrix associated with the Markov chain is still the same. However we now have more non-zero elements and the matrix looks more like a finite lower Hessenberg matrix. The solution technique can be straightforward, but involves more steps.

3.3. Non-Poisson Types of Queues That Can Be Analyzed by MAM

When the arrival and/or service processes are not of the Poisson/exponential types, the queueing system is slightly more complicated to analyze due to its larger state space. However, the analysis could be become more straightforward if the generator matrix exhibits the MAM structure. For instance, if we extend the simple M/M/1/K system to the $PH_n/PH_m/1/K$ system, it is straightforward to show that the non-zero elements (other than the boundaries) of the associated generator matrix is of size $3Knm$. This is a larger dimension than that of the M/M/1/K of size $3K$. Similarly, we can repeat the example of the cases when the system is state-dependent or has different rules/protocols at different states which can lead to a more complex model.

3.4. Non-Poisson Types of Queues with Phase Type Service Times and Multiple Servers

Often encountered in communication systems are queues with phase type (PH) service times, and arrivals that follow some correlated process, such as the Markovian arrival process (MAP). Examples include the MAP/PH/k systems. Even though they appear as a slight variant of the M/M/k/K, the differences are significant especially in constructing the associated Markov chain, both in discrete and continuous times. For the MAP/PH/k in discrete time, the construction of the Markov chain transition matrix is extremely complicated and involved. It was shown in He and Alfa [35] that using what is termed *counting servers for phases* (CSFP)is much more efficient than the *tracking phases for servers* (TPFS). For example, suppose the MAP is of dimension $m_a \times m_a$ and the PH is of dimension $m_s \times m_s$, it was shown that the construction of Markov chain would be of order $m_a m_s^k$ states for the interior blocks, if we used the simple and obvious TPFS approach. Whereas using the CSFP approach would lead to interior blocks not larger than order $m_a(k + m_s - 1)!)/(k(m_s - 1)!)$, which is significantly smaller than $m_a m_s^k$ and is a huge difference in state space required.

In the same consideration there are several Markov chains that occur frequently but their associated matrices are a challenge to construct efficiently. Examples of such instances include the MAP/PH/k systems, queueing systems with several identical sources that are superimposed, and several others. He and Alfa [36] presented tools for reducing the state space of the associate Markov chain and the techniques are worth pursuing further in attempts to develop efficient and effective tools for analyzing large scale queueing systems. In another paper, Alfa and He [37] showed that by studying some heterogeneous multiserver system such as PH/PH/k as a point of queue size change rather than at other points can significantly reduce the size of Markov chain.

Reduction of Markov chain construction: It is clear that a careful construction of Markov chains can considerably reduce the size of the matrix associated with it, when such efficiency is considered. Usually it involves a bit more effort at the front end of the analysis. This is one aspect that was highlighted by the works of He and Alfa [35,36] and sets up a structure from which approximations can be later developed.

3.5. Non-Poisson Types of Queues in Tandem

A system of tandem queues can especially be more complex to analyze when the arrival and/or service processes are not of the Poisson/exponential types, along with an increasing number of queues in the tandem state.

Consider a tandem queue of the form $\mathrm{PH}_n/\mathrm{PH}_{m_1}/1/\ \mathrm{K}_1 \rightarrow \cdot/\mathrm{PH}_{m_2}/1/\mathrm{K}_2$. The number of non-zero elements of the generator matrix associated with this Markov chain is approximately $9K_1K_2nm_1m_2$ for the non-boundary elements. This is of larger dimension than that of the $\mathrm{PH}_n/\mathrm{PH}_m/1/\mathrm{K}$ that is of size $3Knm$. The scale is further increased in cases when there are a larger number of queues in tandem, the system is state dependent or has different rules/protocols applied at different states. In addition, the multiple tandem structure of the system can further impact the complexity of the analysis.

Consider instead a type of tandem queue in which $\mathrm{PH}_n/\mathrm{PH}_{m_1}/1/\ \mathrm{K}_1$ is feeding into two parallel queues $\cdot/\mathrm{PH}_{m_2}/1/\mathrm{K}_2$ and $\cdot/\mathrm{PH}_{m_3}/1/\mathrm{K}_3$. The number of non-zero elements of the generator matrix associated with this Markov chain is approximately $27K_1K_2K_3nm_1m_2m_3$ for the non-boundary elements. This is of much larger dimension than that of the tandem queue discussed previously and its complexity can further increase in the cases when the system is state dependent or has different rules/protocols at different states. The same observations can made for the discrete time equivalents of these queueing systems and similar dimensions can be evaluated for their transition matrices.

The previous examples demonstrate how the complexity of a system of tandem queues can significantly increase when additional queues, structures, and rules/protocols

are included in the system. Even though the analysis of such systems is still based on Markov chains, one may need to solve a very large set of linear equations, unless there is a special structure that can be exploited in the analysis. Thus, various factors contribute to the measure of *how large a queueing system is*. Hence, the Markov chain dimensions and the rules of operations determine how large and complex a queueing system is.

While it is relatively straightforward to measure the size of a Markov chain, it is generally not simple to quantify the complexity of rules/protocols adopted in the system. For example, the complexity is seen as standard (or low) if the service rule adopted in the system First Come First Served (FCFS), Last Come First Served or Service in Random Order. However, if instead consider a FCFS system with an additional requirement that a completed service returns for an additional service by placing it back in the queue based on some rules and with a particular probability, then we have two protocols involved in the system and the complexity is now higher, even though the dimension is still the same.

3.6. Discrete Time Queues

Discrete time queues have some additional characteristics that make them more complicated to analyze than their continuous time counterparts. For the simple Geo/Geo/1/K system, there are practically at least five main versions of its protocols discussed in the literature (see El-Taha [38]): (1) early arrival system (EAS), (2) the late arrival system with immediate access (LAS-IA), (3) the late arrival system with delayed access (LAS-DA), (4) the late arrivals with arrivals first system (LA-AF), and (5) the late arrivals with departures first rule (LA-DF), with all having different transition matrices, specifically at the boundary. However, the main issue is not the case of the Geo/Geo/1/K system but when we have more than two processes, e.g., arrival, service and vacation processes. In such cases, the model becomes very complex, even in the case of the Geo/Geo/1/K. If we introduce batch systems, level dependence, and/or non-Geometric arrival/service processes, then the system becomes more complicated. Hence, the complexity of the analysis can become quite significant even with subtle differences in the rules adopted in the system.

In this paper, we focus more on the continuous case of queueing systems. What we present in that case can be later extended to the discrete time case.

3.7. Summary

In summary, a large scale queueing system is one in which there are several nodes connected and traffic movements between all the nodes can be captured by a form of Markov chain represented by large matrices for which the states are connected in a complex manner. Even though they can all be analyzed as Markov chains – a well studied tool in the literature of applied stochastic modeling – the issue we face is the size of the resulting Markov chain blocks; often too large. While the size of the Markov chain is clearly a factor, another major aspect that also leads to complexity in a queueing system is the rules/protocols guiding the transitions; the more complex these are the higher the complexity of the system. Thus, a large scale complex queueing model is one that embodies these two main features. Determining the complexity of a queueing model is related to a measure that captures these two components with definable quantities.

4. Dealing with Non-MAM Based Queueing Systems, with Infinite and Finite Buffers

When we come across some queueing systems that do not have any structures related to MAM, then we have to find methods for approximating them, especially if they have a large structure. Such types of queueing systems are encountered in computer systems, reliability, software systems, and in many other areas.

The key issue here is to accept, realistically, the necessity to focus more on developing approximation techniques for analyzing large scale queueing systems. Such systems are a class of problems applied researchers have to contend with, and how to develop effective and efficient tools for analyzing them is critical. Since most mathematical models for queueing systems are based on Markov chain analysis, we focus on the approximations based on this. Specifically we present the following approximations:

- Truncation methods
- Decomposition and aggregation methods
- Censored Markov chain approach
- Lumpability
- QBD Compression Method

The first four techniques are well-known to researchers and practitioners. The QBD Compression Method is a heuristic technique that we propose as a simple and alternative method. While the main purpose of this paper is provide an overview of existing techniques, the newly proposed method is used to highlight the prospects of researching alternative and simpler techniques and to encourage its efforts.

Other approximation techniques have also been proposed in the literature, such as the technique by Naumov et al. in [28] for dealing with large-scale queueing systems with random resource requirements. While such techniques have been shown to have significant merits, they are limited to specific types of systems and our overview focuses on techniques that can be applied for all types of systems.

4.1. Truncation Methods

In a truncation method [39], we attempt to reduce the size of a large Markov chain with transition matrix P of dimension N, finite or infinite (usually infinite), to a smaller one P_m of size $m << N$. Let the state space of the Markov chain be arranged in the order of importance, hence the first m states are deemed very important to study. Then the remaining $N - m$ states are grouped into some form of augmentation. There are several methods for creating the augmentation, ranging from the Northwest corner truncation, fixed state augmentation, last state augmentation and linear augmentation. The approach is based on the following steps. Suppose the generator matrix of the original Markov chain is of the form

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N-1} & p_{1,N} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,N-1} & p_{2,N} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{N-1,1} & p_{N-1,2} & \cdots & p_{N-1,N-1} & p_{N-1,N} \\ p_{N,1} & p_{N,2} & \cdots & p_{N,N-1} & p_{N,N} \end{bmatrix}. \tag{17}$$

The truncated version will be of the form

$$P(m) = \begin{bmatrix} p_{1,1}(m) & p_{1,2}(m) & \cdots & p_{1,m-1}(m) & p_{1,m}(m) \\ p_{2,1}(m) & p_{2,2}(m) & \cdots & p_{2,m-1}(m) & p_{2,m}(m) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{m-1,1}(m) & p_{m-1,2}(m) & \cdots & p_{m-1,m-1}(m) & p_{m-1,m}(m) \\ p_{m,1}(m) & p_{m,2}(m) & \cdots & p_{m,m-1}(m) & p_{m,m}(m) \end{bmatrix}. \tag{18}$$

A simple example of the approximation is the last state augmentation in which $p_{i,j}(m) = p_{i,j}, \forall j < m$, and $\forall i < m$, however, $p_{i,m}(m)$ and $p_{m,j}(m), \forall (i,j)$ are determined by some form of approximations.

There are several methods for estimating the final columns and final rows of this new matrix $P(m)$, as mentioned earlier. A major direction for research on this topic is to study

how different augmentations result in the error of approximations, i.e., the best methods for determining the last row and column of this truncated matrix with regards to practical problems. The research should be queueing system biased, rather than generic.

4.2. Aggregation and Decomposition

Aggregation, sometimes used in conjunction with decomposition, is a powerful tool for studying large scale queueing systems that do not have simple structures such as MAM.

Decomposition and Aggregation Techniques for Markov Chains

The two techniques have been used extensively in manufacturing systems and, to some extent, in communication networks. The decomposition approach basically reduces the huge Markov chain to a set of smaller, but conceptually connected, Markov chains. The interconnection between the set of smaller manageable Markov chains are defined, or approximated, and attempts to reconnect the original Markov chain from the smaller ones are carried out. The aggregation approach is essentially the reverse in which each of the smaller subsystems are analyzed individually and then recombined based on some rules. In this case the huge Markov chain is compressed into a small one in which the combined behaviors of a set of the nodes are grouped into one pseudo (or imaginary) Markov chain, which is much smaller and then analyzed accordingly. Later the compressed Markov chain is then expanded back to re-capture the smaller Markov chains back. Some of the well-known tools in this category can be found in [40–47], and several references quoted in those papers.

Generally, decomposition has been used more frequently in the case of queueing networks because they often have some natural or physical groupings that can be exploited. A physical decomposition of the network is considered when one can see weak connections between some groups of nodes. Thus, the subgroups of networks are analyzed individually and then all of them are combined using some knowledge of how they all interact.

- **Decomposition:** Decompositional methods for solving large discrete time Markov chain was well discussed by Haviv [48]. The idea is of breaking the problem into subproblems that can be solved independently, with the global solution obtained by "pasting" together the subproblem solutions. The idea is based on having a system that is nearly-completely-decomposable (NCD), nearly uncoupled, or nearly separable systems. NCD is based on reasonably connected subgroups. Steady state probability vectors are based on a two-level hierarchy (i) within clusters, and (ii) across clusters. For more details, see the book by Courtois [49]. The idea of NCD was used by Gambian et. al. [50] in analyzing some class of Markov chains presented as efficient combinatorial aggregation algorithms for computing two key parameters needed to solve the linear systems associated with such models.
  In studying decomposition, we consider a Markov chain $\mathbf{M}$ of size $N < \infty$, and represent the complexity of analyzing this system as $O(N^2)$. Suppose we can decompose this Markov chain to K Markov chains, labeled as $\mathbf{M_k}, k = 1, 2, \cdots, K$, with sizes $N_1, N_2, \cdots, N_k$, each with complexity $O(N_k^2)$, $k = 1, 2, \cdots, K$, with $O(N_k^2) << O(N^2)$, such that when the results of each of the Markov chains $\mathbf{M_k}, k = 1, 2, \cdots, K$ are combined through their loose connections, we get a good approximation of $\mathbf{M}$. While this approach has been well studied in the literature, the main challenge is usually how to develop the connectivity of the K smaller Markov chains to get a good approximation of $\mathbf{M}$.
  In the detailed paper [48], Haviv presented six aggregation/disaggregation methods for Markov chains. Note, to the reader, is that in [48] the stochastic matrix is represented in a column stochastic form. Some of the methods had previously appeared in

other papers [51,52]. In a later paper, Haviv [48] discussed the error bounds for each of the six methods and it is worthwhile to explore further these methods.

- **Aggregation:** Aggregation approach, on the other hand, starts by assuming that the Markov chain **M** can be studied as a smaller imaginary (or virtual) Markov chain of size $M << N$ where the behavior of all the activities of the larger Markov chain can be captured approximately. Selecting the aggregated states is driven by which states are more important for the study rather than by lumpability criteria. This aggregated Markov chain is then analyzed appropriately.

4.3. Censored Markov Chain

Censored Markov chain [53] is another technique that can be used for studying large scale Markov chains, especially if they have a structure close to being manageable by MAM, but is not necessary. Consider the transition matrix P of dimension $N \times N$. Suppose the main interest to us are the states $1, 2, \cdots, m$, hence states $m+1, m+2, \cdots, N$ can be censored without losing too much important information. Let us now write the matrix P as

$$P = \begin{bmatrix} P_{AA} & P_{AB} \\ P_{BA} & P_{BB} \end{bmatrix}, \tag{19}$$

where P_{AA} is of dimension $m \times m$.

If the stationary distribution of the matrix P is $[\pi_A, \pi_B]$, then the stationary distribution of the states of interest will be given by solving

$$\pi_A = \pi_A[P_{AA} + P_{AB}(I - P_{BB})^{-1}P_{BA}], \tag{20}$$

where $\pi_A \mathbf{1} = 1$.

The size of the Markov chain is reduced using this technique and the extent of the reduction depends on how many states are really of interest. There are still two matrices that need to be computed, P_{AA} and $(I - P_{BB})^{-1}$. The larger of the two determines how much the original transition matrix of the Markov chain has been reduced. The key in this approach is to determine which states are more important and include them in the states captured by P_{AA}. Even though the augmentation block $P_{AB}(I - P_{BB})^{-1}P_{BA}$ consists of a block $(I - P_{BB})^{-1}$ which may be large, it can probably be approximated using appropriate approaches.

When we consider censored Markov chains under some mild assumptions, for nearly uncoupled chains, the best augmentation in the sense of minimizing the resulting error-bound was presented by Haviv [54]. Specifically, it is shown in [54] that for the case of nearly uncoupled chain, and under some mild conditions, minimizing the error-bound of the best augmentation is based mainly on P_{AA}, the sub-stochastic matrix generated from P.

It is immediately clear that solving for the stationary distribution of the states of interest has been reduced to finding the stationary distribution of m states compared to n, where $n >> m$. Censored Markov chains have potential for being creatively used to approximate large size Markov chains, and hence large scale queueing systems. In a major work, Grassmann et al. [55] showed how to use the idea to create what they call state reduction. From their work, it is clear that this approach has potential of being used for large scale Markov chains and needs to be revisited thoroughly.

4.4. Lumpability of Markov Chains

The idea of combining some states of Markov chains in order to reduce its size is known as aggregation. However, a type of aggregation that still lead to a Markov chain are known as lumpability as discussed in [56,57]. This idea has potential of getting developed

further to incorporate its weak lumpability into applications in large scale queueing systems. A major theorem in [56] states that: a Markov chain is lumpable with respect to a partitioned state space $\{A_1, A_2, \cdots, A_r\}$, for every starting vector π, and associated condition for this to be possible are stated in its Theorem 6.3.2 for the case of transition probability matrices, i.e., for the discrete time Markov chains. We discuss that case as presented in [56], keeping in mind that the idea is also applicable to continuous time Markov chains. While most Markov chains do not necessarily meet the criteria required, they also presented some conditions for what they call weak *lumpability*, an idea we feel needs to be explored while considering practical problems for which the criteria for lumpability may not be met directly. To further explore lumpability and its properties, we consider a discrete time Markov chain with transition probability matrix P, which has n states, and suppose it is partitioned into $r < n$ states as mentioned above, with matrices U and V defined as

$$VUPV = PV, \tag{21}$$

where U is a matrix of dimension $r \times n$ and V a matrix of dimension $n \times r$. The ith row of U is the probability vector of having equal components for states A_i and 0 for the remaining states. V is an $n \times r$ matrix with the jth column a vector with $1's$ in the components corresponding to states A_j and $0's$ otherwise, Thus, the lumped transition matrix is given as follows (see [56]).

$$\hat{P} = UPV. \tag{22}$$

Indeed, most real life practical problems, and especially ones that are of large scale complex types, will hardly meet these criteria and thus the idea of weak lumpability was also advanced. While the conditions for the weak lumpability of this type of Markov chain is a slight relaxation of the strict lumpability, they might also be difficult to meet in practical situations. The idea of proportional lumpability was advanced as a reasonable approximation to most Markov chains, but mainly focused on continuous time Markov chains [58]. However, the result can still be applied to discrete time systems by employing the known detail that a continuous time Markov chain with generator Q can be easily converted to a discrete time Markov chain with transition matrix P based on the relationship

$$P = Q\Delta t + I, \tag{23}$$

where Δt is selected sufficiently small so that the probability of two or more transitions occurring in Δt is small and of order $o(t)$.

To help explain better the technique, let us return to the models in Section 3 which were developed using the continuous-time approach for a Markov chain with generator matrix Q. In order to obtain a proportional lumpable system from it, we refer to the following definition given by Piazza in [58].

Definition 1 ([58]). *For a continuous time Markov chain (CTMC) $X(t)$ with a state space S and infinitesimal generator Q, let κ denote a function from S to R^+. We say that a CTMC $X'(t)$ with infinitesimal generator Q' is a perturbation of $X(t)$ with respect to κ if $X'(t)$ is obtained from $X(t)$ by perturbing its rates such that for all $s' \in S$ with $s \neq s'$*

$$q'(s, s') = \frac{q(s, s')}{\kappa(s)} \tag{24}$$

Theorem 1 in [58], based on this definition, provides equilibrium distribution for the proportional lumpability as follows.

Theorem 1 ([58]). *For any perturbation $X'(t)$ of the original chain $X(t)$ with respect to κ, where $X'(t)$ has an infinitesimal generator Q' and equilibrium distribution π', the equilibrium distribution π of $X(t)$ adheres to the following property: Let $K = \sum_{s \in S} \frac{\pi'(s)}{\kappa(s)}$, then $\forall s \in S$, we have*

$$\pi(s) = \frac{\pi'(s)}{K\kappa(s)}. \tag{25}$$

Examples are provided in [58] and the ideas presented in that paper should be explored further. In later works, they introduce alternative characterizations of proportional lumpability.

In other works, Buchholz [59] has developed approximations for lumpability that have potential for applications to real live problems. This subject also needs to be explored further. Franceshinis and Muntz [60] later developed bounds for quasi-lumpable Markov chains by approximating the chain to make it lumpable, implying you get estimations, not exact answers, for the original process. The idea in that work should be considered in developing lumpability for practical problems as bounds are key to assessing the approximations. Also Georgiou et al. [61] have shown how lumpability can be used for real life problems. It appears that this approach has great potentials that have not been fully explored yet. Proportional lumpability allows for exact stationary performance analysis of a simplified lumped Markov chain, deriving precise results for the original complex model using a scaling factor called $\kappa(s)$.

We strongly encourage researchers to explore further the lumpability technique and associated approximations as tools for dealing with large scale queueing system, via Markov chain formulation.

4.5. QBD Compression Method

In this sub-section, we propose a heuristic method of approximating the QBD (Quasi-Birth-Death) model for a large-scale system into a smaller QBD that is more computationally tractable. We consider the case of discrete time Markov chains and similar analyses can be extended for the case of continuous time. We present a technique that may be used to approximate a block matrix in a QBD, or a sub-block matrix within a block, into a single probability. The proposed approximation is based on the premise that a Phase-distribution (ψ, T) can be approximated as a Geometric distribution Geom(p) with parameter p.

Consider a random variable $\mathcal{T}$ that is modeled as a discrete Phase-type (PH) distribution with parameters (ψ, T) of order n, such that

$$Pr\{\mathcal{T} = k\} = \boldsymbol{\psi} T^{k-1}\mathbf{t}, \quad k = 1, 2, \cdots \tag{26}$$

where $\mathbf{t} = \mathbf{1} - T\mathbf{1}$. For a given PH distribution, the same random variable can be approximated and modeled using a Geometric distribution with parameter p of the following form.

$$Pr\{\mathcal{T} = k\} = p^{k-1}(1-p), \quad k = 1, 2, \cdots \tag{27}$$

Consider the following example of a PH distribution which we will use to find p that best approximates it to Geometric distribution.

$$\boldsymbol{\psi} = [1, 0, 0] \qquad T = \begin{bmatrix} 0.4 & 0.05 & 0.25 \\ 0.25 & 0.2 & 0.35 \\ 0.25 & 0.2 & 0.3 \end{bmatrix}, \tag{28}$$

For evaluating the probability p that best approximates the PH distribution, a simple search algorithm was used that generates the probability distribution Geom(p) for

$0.001 \leq p \leq 0.999$ and determines the probability p that yields a probability distribution that is most similar to the PH distribution. We utilize the *Chi-Square Goodness-of-Fit Test* χ^2 for a given pair of probability distributions $\mathbf{x}$ and $\mathbf{y}$, with x_i and y_i being the probabilities in state i. A single and unique p is assumed to yield the minimum metric.

$$\text{Chi-Square Goodness-of-Fit Test:} \qquad \chi^2 = \sum_{i=1}^{k} \frac{(y_i - x_i)^2}{x_i}$$

For this example, $p = 0.723$ was determined as the best solution (with $\chi^2 = 0.0042$), and thus Geom(0.723) would be the best approximation to the PH distribution. This example demonstrates the potential of utilizing the Geom(p) as an approximation for a transient probability matrix and further illustrate its application to a queueing model.

Consider the example of a pair of tandem queues where the entity departing the first queue enters the second queue, as shown in Figure 1. We model this system as a discrete time Markov chain where the arrival into the first queue is a Bernoulli process with probability α, and the service times in the first and second queues are Geometrically distributed with probabilities β_1 and β_2, respectively. Let K_1 and K_2 be the capacity of the first and second queue, where the capacity includes the entities in service and that are waiting for service.

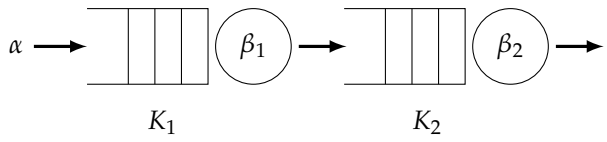


Figure 1. Tandem queueing system example.

In this example, we assume that the entity departing the first queue after a service completion will remain in that queue if the second queue is full, and the next entity will not start its service. In such a state, we say that the departing entity is blocked from transitioning to the second queue and service of remaining entities is paused until the block entity is admitted to the second queue when space becomes available. The state space that models the behavior of the model is given as follows,

$$\Delta = \left\{ (\ell_1 \,,\, \ell_2) \,; \ell_1 = 1, 2, \cdots, K_1 \,,\, \ell_2 = 1, 2, \cdots, K_2, K_2^{'} \right\},$$

where ℓ_q is the number of packets in the queue q (including the one in service), and the state $\ell_2 = K_2^{'}$ refers to the case where the buffer in the second queue is full and is blocking the departure from the first queue.

The system can be modeled as a QBD with the following transition probability matrix,

$$\mathbf{P} = \begin{bmatrix} B & C & & & & & \\ E & A_1 & A_0 & & & & \\ & A_2 & A_2 & A_0 & & & \\ & & A_2 & A_1 & A_0 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & & A_2 & D \end{bmatrix}, \tag{29}$$

where $D = A_1 + A_0$. The derivation of the block matrices can be readily derived for this model, and the invariant probability vector $\boldsymbol{x}$ of the system described by the transition probability matrix $\mathbf{P}$ can be evaluated by solving for $\boldsymbol{x} = \boldsymbol{x}\mathbf{P}$.

For this QBD, each block and sub-block matrices in it can be approximated to a single probability. We perform the approximation by assigning each block matrix as the matrix of transient probabilities T in a PH distribution, and assume that $\boldsymbol{\psi} = [1, 0, 0, \cdots, 0]$.

To demonstrate the performance of the approximation, we assume the example of the system with $\alpha = 0.4$, $\beta_1 = 0.5$, $\beta_2 = 0.6$, $K_1 = 5$, and $K_2 = 4$. After running the approximation for each block matrix in $\mathbf{P}$, we end up with the following approximation $\tilde{\mathbf{P}}$ for the transition probability matrix of the same system. Note that for running the approximations for the non-square block matrices (namely C and E), we could either expand their dimensions with rows or columns of 0's to transform them in to a square matrix, or determine their probability by simply subtracting from 1 the other approximated probabilities on the same row. We opted for the latter in this example.

$$\tilde{\mathbf{P}} = \begin{bmatrix} 0.6 & 0.4 & & & & & \\ 0.3 & 0.5 & 0.2 & & & & \\ & 0.3 & 0.5 & 0.2 & & & \\ & & 0.3 & 0.5 & 0.2 & & \\ & & & 0.3 & 0.5 & 0.2 & \\ & & & & 0.3 & 0.7 & \end{bmatrix} \tag{30}$$

The invariant probabilities were calculated for the model using the transition probability matrix $\mathbf{P}$ and its approximated form $\tilde{\mathbf{P}}$. Figure 2 shows the comparison between the two probability distributions, and how the approximated $\tilde{\mathbf{P}}$ yielded a very similar probability distribution. For this example, the similarity metric of the two distributions is $\chi^2 = 6.4 \times 10^{-4}$. We further evaluated the mean number $E[\ell_1] = 1.78$ using the matrix $\mathbf{P}$, whereas $E[\ell_1] = 1.74$ for $\tilde{\mathbf{P}}$. Note that the approximation limited the analysis to examining the system state ℓ_1, and the transition probability matrix $\mathbf{P}$ could instead be reformulated with state ℓ_2 as the first level state for analyzing instead the system in terms of state ℓ_2.

Figure 3 shows another example that compares the probability distributions using the actual and approximated transition probability matrices of the same tandem queue system where the arrival probability is increased to $\alpha = 0.55$, with $\beta_1 = 0.5$, $\beta_2 = 0.6$, $K_1 = 8$, and $K_2 = 5$. For this example, $\chi^2 = 3.3 \times 10^{-4}$, the mean number $E[\ell_1] = 5.60$ for the matrix $\mathbf{P}$, whereas $E[\ell_1] = 5.79$ for $\tilde{\mathbf{P}}$.

The results from this example illustrates the merits of applying the proposed heuristic method to approximate the transition probability matrices. For an n-state system, the proposed approximation can be implemented to reduce the analysis of the QBD to an m-state system, where $1 \leq m < n$. This reduction in state can scale down the complexity of the system's analysis, but limits the performance metrics evaluations to the reduce m states. Overall, these results demonstrate the potential to explore further such approximations for simplifying the analyses of large scale QBDs, and to examine methods for improving the accuracy of the approximations.

Further work is needed to rigorously assess the performance of this proposed technique in terms of its approximation accuracy, convergence behavior, and error bounds. These assessments will be carried out in our future work, and the main intent of presenting this technique is to demonstrate the prospects of researching alternative techniques toward the efforts of approximating large scale and complex queueing models.

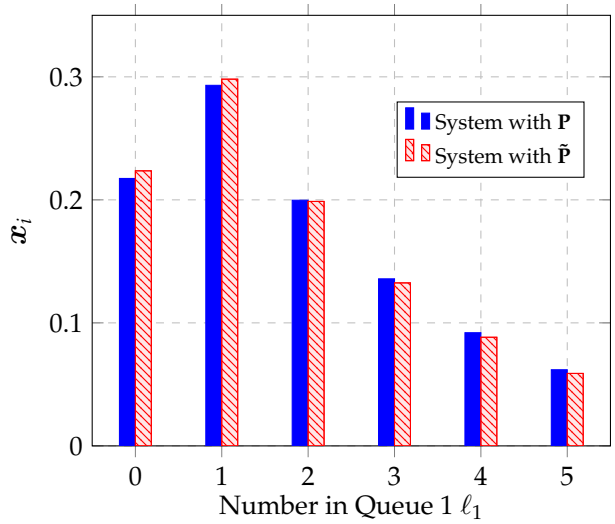


Figure 2. A Comparison between the invariant probabilities for the System modeled by **P** and $\tilde{\mathbf{P}}$, with $\alpha = 0.4$, $\beta_1 = 0.5$, $\beta_2 = 0.6$, $K_1 = 5$, and $K_2 = 4$.

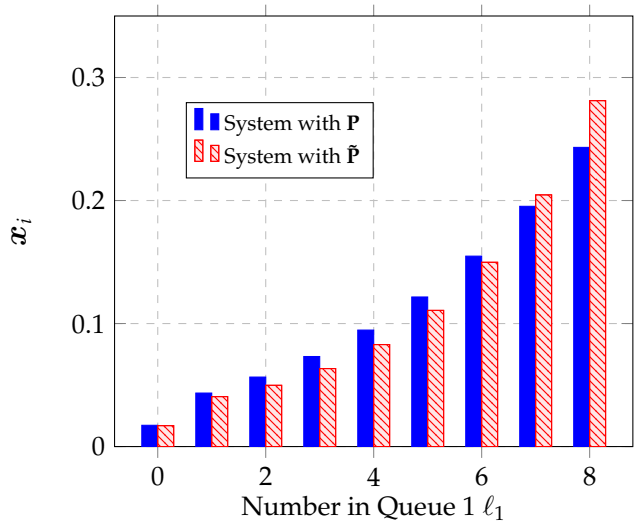


Figure 3. A Comparison between the invariant probabilities for the System modeled by **P** and $\tilde{\mathbf{P}}$, with $\alpha = 0.55$, $\beta_1 = 0.5$, $\beta_2 = 0.6$, $K_1 = 8$, and $K_2 = 5$.

5. Conclusions & Directions for Future Research

One may still question the need for queueing analysis approximation techniques, given the continuing increase of computer processing power and capabilities. Many of the queueing models for large scale and/or complex systems can still be processed with adequate computing resources. However, they can be computationally expensive and not suitable to real-time applications, especially when they are incorporated in Digital Twins applications. An example of applications where processing powers and delays can be performance-limiting is the scheduling and decision-making processes in cognitive wireless sensor nodes [62]. Various results have proposed queueing models for scheduling decisions, such as spectrum sensing, and often do not factor the processing requirements and impact on the node's resources. Another example include the works on utilizing queueing models for scheduling task offloading in energy-constrained mobile devices [63]. Applying approximation tools on the queueing models would help to save the node's energy and execution time of the results. There is also a growing interest in the use of queueing models in Digital Twins [64] and they can only be effective in such applications if the results from the models can also be obtained efficiently and in real time.

The use of machine learning and its derivatives have been quite popular in the discussed applications, as well as others, and in lieu of mathematical analysis. The reason for its preference is likely due to the larger effort of setting up the queueing model for such complex systems, along with effort of running the analysis. However, deriving models using machine learning do have their limitations, as described earlier, and those solutions do not exploit the merits of queueing models. These shortcomings could be addressed by the development of better approximation techniques for analyzing large scale and/or complex queueing systems.

In the previous sections, we have highlighted some of the different practices that could be further explored to aid in the analysis of complex queues. Aside from investigating alternative and more tractable approximation techniques, we propose the following additional endeavors to support the efforts of improving on the existing techniques.

- **Analysis of Error Bounds:** Approximation of large scale queues is inevitable when it comes to solving real life major problems. The application of approximation techniques tend to reduce the accuracy of the results from the analysis. These erroneous results can be significant. Hence, the need to further study the error bounds associated with different approximations. This would allow practitioners and analysts to better assess the severity of the anticipated errors from employing certain approximation techniques, as well as decide on the suitability of the approximations associated with their applications. Furthermore, these bounds can serve as measures for researching improvements to these approximation techniques.
- **Measure of Complexity:** Determining such measures can aid with better understanding the scale and complexity of analyzing queueing systems, as well as determining the appropriate techniques to employ in the analysis. The metric can further allow for a more meaningful comparison of the different queueing systems in terms of their analytical scale and complexity. The purpose of this measure can be similar to the Big **O** notation used in describing the scale and complexity of computing algorithms. Thus, we recommend researching an adaptable method for deriving the measure of complexity of queueing systems. In Section 3, we provided an example of an approach that can be taken to derive this metric. We also want to note that this metric is separate from the Kendall Notation (i.e., A/S/c/K/N/D) used for classifying and describing queueing systems, however, the notation could be used as a factor in deriving the complexity metric and we recommend its exploration in future works.
- **Improved State Aggregation Technique:** Some of the existing approximation techniques described in Section 4 involve aggregating or censoring certain states in the Markov chain for the purpose of reducing the complexity of the analysis. However, the task of state aggregation and/or censoring requires some initial computational efforts and is also done at the cost of reducing the granularity of the system analysis. Such limitations can be deemed acceptable if it can significantly improve efforts in the queueing analysis. However, deciding on which states to aggregate and censor can be a challenge and is often dependent on the system being analyzed as well as the desired performance metrics in the analysis. In addition to investigating better state aggregation and censoring techniques with reduced efforts, we also propose researching smarter methods for applying these techniques along with an approach to decide on which states can be "sufficiently" aggregated or censored for the analysis.
- **State Reconstruction of Markov Chain:** Solutions such as those proposed by He and Alfa in [35,36] offer a method for reconstructing and reorganizing the states in the Markov chain with the aim of reducing the dimensions of the model significantly. These techniques can aid with reducing the computational efforts and resources for analyzing large scale and complex queueing systems. Note that these state reconstruc-

tive techniques are not approximations of a Markov chain with large dimensions, and the analysis of the model with the reconstructed states returns the same results as the original model. However, the application of such methods do require additional efforts in modifying the setup of the Markov chain, but should only need to be done once during the development of the model. Even though these state reconstruction techniques are non-approximation methods, unlike the other techniques discussed throughout this article, we bring attention to it in this work as it tends to achieve a similar goal of the approximation techniques. Thus, we propose further researching improved state reconstruction techniques that allow for a more tractable analysis of large scale and complex queueing systems, along with the possibility of combining it with other approximation techniques.

Analysis of queueing systems continue to be essential in various applications and resource-constrained systems. With the increasing scale and complexity of these system, there is a growing need for better approximation and analytical techniques, and especially in systems that require results in real time and under certain constraints. The alternative methods (e.g., simulation, and machine learning techniques) may be sufficient and appropriate in certain applications, but the analytical and Markov chain based solutions also have their merits and are well suited for certain types of analyses. Our aim in this paper is to provide an overview of the existing approximation techniques for analyzing queueing systems, and to further promote future research in developing better techniques for analyzing large scale and complex queueing systems.

Author Contributions: Conceptualization, A.S.A.; Methodology, A.S.A. and H.A.G.; Formal analysis, A.S.A. and H.A.G.; Investigation, A.S.A. and H.A.G.; Writing—original draft, A.S.A.; Writing—review and editing, A.S.A. and H.A.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Shah, D. Three Metrics for Stochastic Networks: Capacity, Queue-Size and Complexity. In Proceedings of the IEEE Third International Conference on Communication Systems and Networks (COMSNETS 2011), Bangalore, India, 4–8 January 2011.
2. Paiva, J.R.B.; Gomes, V.M.; Oliveira, S.G.; Domingues, E.G.; Alves, A.J.; da Cruz, G.; Calixto, W.P. Calculation of system complexity based on the connections: Methodology and applications. In Proceedings of the IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC), Florence, Italy, 7–10 June 2016; pp. 1–6.
3. Vangheluwe, H. Foundations of Modelling and Simulation of Complex Systems. In *Proceedings of the Seventh International Workshop on Graph Transformation and Visual Modeling Techniques*; Electronic Communications of the EASST: Berlin, Germany, 2008; Volume 10.
4. Cassandras, C.G.; Lafortune, S. *Introduction to Discrete Event Systems*; Springer: Cham, Switzerland, 2021.
5. Maria, A. Introduction to Modelling and Simulation. In Proceedings of the 1997 Winter Simulation Conference, Atlanta, GA, USA, 7–10 December 1997; pp. 7–13.
6. Loper, M.L.; Register, A. Chapter 1: Introduction to Modeling and Simulation. In *Modeling and Simulation in the Systems Engineering Life Cycle Simulation Foundations, Methods and Applications*; Springer: Berlin/Heidelberg, Germany, 2015.
7. Velten, K.; Schmidt, D.M.; Kahlen, K. *Mathematical Modeling and Simulation: Introduction for Scientists and Engineers*; Wiley-VCH: Hoboken, NJ, USA, 2024.
8. White, J.A.; Schmidt, J.W.; Bennett, G.K. *Analysis of Queueing Systems*; Academic Press: Cambridge, MA, USA, 1975.
9. Glynn, P.W.; Iglehart, D.L. Simulation methods for queues: An overview. *Queueing Syst.* **1988**, *3*, 221–255. [CrossRef]
10. Pawlikowski, K. Steady-state simulation of queueing processes survey of problems and solutions. *Acm Comput. Surv. (CSUR)* **1990**, *22*, 123–170. [CrossRef]

11. Wang, J.J.; Abrams, M. Approximate Time-Parallel Simulation of Queueing Systems with Losses. In Proceedings of the 1992 Winter Simulation Conference, Arlington, VA, USA, 13–16 December 1992.
12. Vasilyev, S.A.; Tsareva, G. Simulation of Large-Scale Queueing System. In Proceedings of the 32nd European Conference on Modelling and Simulation, Wilhelmshaven, Germany, 22–25 May 2018; pp. 485–490.
13. Dolgopolovas, V.; Dagienė, V.; Minkevičius, S.; Sakalauskas, L. Python for Scientific Computing Education: Modeling of Queueing Systems. *Sci. Program.* **2014**, *22*, 164306. [CrossRef]
14. Wei, S.; Yong-Ke, C.; Ai-Jun, Y.; Wei, H.; Chao, Z. Research on Simulation of Queuing System Based on OOP. In Proceedings of the 2018 IEEE International Conference on Computer and Communication Engineering Technology (CCET), Beijing, China, 18–20 August 2018; pp. 34–38.
15. Lv, X.-Z.; Fan, B.-X.; Wang, X.-W.; Zhao, X.-H. A modeling and simulation method of queue systems based on stateflow. In Proceedings of the 2016 3rd International Conference on Systems and Informatics (ICSAI), Shanghai, China, 19–21 November 2016; pp. 541–545.
16. Ajewole, O.R.; Osagiede, A.A.; Okoro, J.O. A Review on the Application of Machine Learning Techniques for Predictions of Performance Measures of Queue Systems. In Proceedings of the 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG), Omu-Aran, Nigeria, 2–4 April 2024; pp. 1–7.
17. Sherzer, E.; Senderovich, A.; Baron, O.; Krass, D. Can machines solve general queueing systems? In Proceedings of the IEEE 2022 Winter Simulation Conference (WSC), Singapore, 11–14 December 2022.
18. Sherzer, E. Computing the steady-state probabilities of the number of customers in the system of a tandem queueing system, a Machine Learning approach. *Eur. J. Oper. Res.* **2025**, *326*, 141–156. [CrossRef]
19. Garyfallos, S. Deep Learning Methods for Transient Performance Analysis and Optimization of Non-Markovian Nonstationary Queueing Systems. Ph.D. Thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, February 2026.
20. Abdali, N.; Heidari, S.; Alipour-Vaezi, M.; Jolai, F.; Aghsami, A. A priority queueing-inventory approach for inventory management in multi-channel service retailing using machine learning algorithms. *Kybernetes* **2025**, *54*, 2563–2591. [CrossRef]
21. Chocron, E.; Cohen, I.; Feigin, P. Delay Prediction for Managing Multiclass Service Systems: An Investigation of Queueing Theory and Machine Learning Approaches. *IEEE Trans. Eng. Manag.* **2024**, *71*, 4469–4479. [CrossRef]
22. Efrosinin, D.; Vishnevsky, V.; Stepanova, N.; Sztrik, J. Use Cases of Machine Learning in Queueing Theory Based on a GI/G/K System. *Mathematics* **2025**, *13*, 776. [CrossRef]
23. Alfa, A.S.; Abu Ghazaleh, H. Machine Learning Tool for Analyzing Finite Buffer Queueing Systems. *Mathematics* **2025**, *13*, 346. [CrossRef]
24. Neuts, M. *A Matrix-Geometric Solutions in Stochastic Models: An Algorithm Approach*; Johns Hopkins University Press: Baltimore, MD, USA, 1981.
25. Neuts, M. *Structured Stochastic Matrices of M/G/1 Type and Their Applications*; Marcel Dekker, Inc.: New York, NY, USA, 1989.
26. Naumov, V. Matrix-multiplcative approach to quasi-birth-and-death processes analysis. In *Matrix-Analytic Methods in Stochastic Models*; Chakravarthy, A., Ed.; Marcel Dekker, Inc.: New York, NY, USA, 1997; pp. 87–106.
27. Naumov, V.A.; Gaidamaka, Y.V.; Samouylov, K.E. On Truncation of the Matrix-Geometric Stationary Distributions. *Mathematics* **2019**, *7*, 798. [CrossRef]
28. Naumov, V.A.; Gaidamaka, Y.V.; Yarkina, N.; Samouylov, K.E. *Matrix and Analytical Methods for Performance Analysis of Telecommunication Systems*; Springer: Cham, Switzerland, 2021. [CrossRef]
29. Klimenok, V.I.; Dudin, N. Multi-dimensional asymptotically quasi-toeplitz Markov chains and their application in queueing theory. *Queueing Syst.* **2006**, *54*, 245–259. [CrossRef]
30. Alfa, A.S. Discrete-time analysis of the GI/G/1 system with Bernoulli retrials: An algorithmic approach. *Ann. Oper. Res.* **2006**, *141*, 51–66. [CrossRef]
31. Neuts, M.F.; Rao, B.M. Numerical investigation of a multi-server retrial model. *Queueing Syst.-Theory Appl.* **1990**, *7*, 169–190. [CrossRef]
32. Falin, G.I. A survey on retrial queues. *Queueing Syst.-Theory Appl.* **1990**, *7*, 127–168. [CrossRef]
33. Diamond, J.E.; Alfa, A.S. The MAP/PH/1 retrial queue. *Commun. Stat. Stoch. Model.* **1998**, *14*, 1151–1177. [CrossRef]
34. Trivedi, K.S. Probability, Statistics with Reliability, Queueing, analysis and Computer Science Applications. In *Probability Statistics with Reliability, Queueing, Analysis and Computer Science Applications*; Wiley: Hoboken, NJ, USA, 2016.
35. He, Q.M.; Alfa, A.S. Construction of Markov chains for discrete time MAP/PH/K. *Perform. Eval.* **2015**, *93*, 17. [CrossRef]
36. He, Q.M.; Alfa, A.S. Space reduction for a class of multidimensional Markov chains: A summary and some applications. *Inf. J. Comput.* **2018**, *30*, 1–215. [CrossRef]
37. Alfa, A.S.; He, Q.M. Point of queue size change analysis of the PH/PH/k system with heterogeneous servers. *Oper. Res. Lett.* **2017**, *45*, 581–584. [CrossRef]
38. El-Taha, M. A review of Birth-Death and Other Markovian Discrete-Time Queues. *Adv. Oper. Res.* **2023**, *2023*, 6620393. [CrossRef]
39. van Dijk, N.M. Truncation of Markov Chains with Applications to Queueing. *Oper. Res.* **1991**, *39*, 1018–1026. [CrossRef]

40. MacGregor Smith, J. Queue decomposition and finite closed queueing network models. *Comput. Oper. Res.* **2015**, *53*, 176–193. [CrossRef]
41. Dallery, Y.; Frein, Y. On Decomposition Methods for Tandem Queueing Networks with Blocking. *Oper. Res.* **1993**, *41*, 386–399. [CrossRef]
42. Sadre, R. Decomposition-Based Analysis of Queueing Networks. Ph.D. Thesis, University of Twente, Enschede, The Netherlands, 1995.
43. Whitt, W.; You, W. New Decomposition Approximations for Queueing Networks. *Queueing Syst.* **2022**, *100*, 365–367. [CrossRef]
44. Huisman, T.; Boucherie, R.J. Decomposition and Aggregation in Queueing Networks. In *Queueing Networks*; International Series in Operations Research & Management Science; Springer: Boston, MA, USA, 2011; Volume 154.
45. Mohammadi, M.; Dauzère-pérès, S.; Yugma, C.; Karimi-Mamaghan, M. A queue-based aggregation approach for performance evaluation of a production system with an AMHS. *Comput. Oper. Res.* **2020**, *115*, 104838. [CrossRef]
46. Conway, A.E.; Georganas, N.D. Decomposition and aggregation by class in closed queueing networks. *IEEE Trans. Softw. Eng.* **1986**, *SE-12*, 1025–1040. [CrossRef]
47. Islam, F. Simplifying Layered Queueing Network Models. Ph.D. Thesis, Electrical & Computer Engineering, Carleton University, Ottawa, ON, Canada, 2018.
48. Haviv, M. Aggregation/disaggregation methods for computing the stationary distribution of a Markov chain. *SIAM J. Numer. Anal.* **1987**, *24*, 952–966. [CrossRef]
49. Courtois, P. *Decomposiability: Queueing and Computer System Applications*; Academic Press: Orlando, FL, USA, 1977.
50. Gambin, A.; Krzyzanowski, P.; Pokaroswki, P. Aggregation algorithms for perturbed Markov chains with applications to networks modelling. *SIAM J. Comput.* **2005**, *31*, 46–73.
51. Chatelin, F.; Miranker, W.L. Acceleration by aggregation successive approximation methods. *Linear Algebra Appl.* **1982**, *43*, 17–47. [CrossRef]
52. Vantilborgh, H. Aggregation with an error of $O(e^2)$. *J. Assoc. Comput. Mach.* **1985**, *32*, 161–190. [CrossRef]
53. Zhao, Y.Q.; Liu, D. The Censored Markov Chain and the Best Augmentation. *J. Appl. Probab.* **1996**, *33*, 623–629. [CrossRef]
54. Haviv, M. On censored Markov chains, best augmentations and aggregation/disaggregation procedures. *Comput. Oper. Res.* **1999**, *26*, 1125–1132. [CrossRef]
55. Grassmann, W.; Taksar, M.I.; Heyman, D.P. Regenerative analysis and steady state distributions for Markov chains. *Oper. Res.* **1985**, *33*, 1107–1116. [CrossRef]
56. Kemeny, J.G.; Snell, J.L. *Finite Markov Chains*; Springer: New York, NY, USA, 1960.
57. Rubino, G.; Sericola, B. *Markov Chains and Dependability Theory*; Cambridge University Press: Cambridge, UK, 2014.
58. Piazza, C.; Rossi, S.; Smuseva, D. Efficient algorithm for proportional lumpability and its application to selfish mining in public blockchains. *Algorithms* **2024**, *17*, 159. [CrossRef]
59. Buchholz, P. Exact and ordinary lumpability in finite Markov chains. *J. Appl. Probab.* **1994**, *31*, 59–75. [CrossRef]
60. Franceschinis, G.; Muntz, R.R. Bounds for quasi-lumpable Markov chains. *Perform. Eval.* **1993**, *20*, 223–243. [CrossRef]
61. Georgiou, K.; Domazakis, G.; Pappas, D.; Yannacopoulos, A.N. Markov chain lumpability and applications to credit risk modelling in compliance with the International Financial Reporting Standard 9 Framework. *Eur. J. Oper. Res.* **2020**, *292*, 1146–1164. [CrossRef]
62. PalunčIć, F.; Alfa, A.S.; Maharaj, B.T.; Tsimba, H.M. Queueing Models for Cognitive Radio Networks: A Survey. *IEEE Access* **2018**, *6*, 50801–50823. [CrossRef]
63. Chen, H.; Todd, T.D.; Zhao, D.; Karakostas, G. Wireless and Service Allocation for Mobile Computation Offloading With Task Deadlines. *IEEE Trans. Mob. Comput.* **2024**, *23*, 5054–5068. [CrossRef]
64. Okegbile, S.D.; Cai, J.; Alfa, A. Queue-Aware Age of Information Optimization Framework for Large-Scale Digital Twin Networks. In Proceedings of the ICC 2026—IEEE International Conference on Communications, Glasgow, UK, 24–28 May 2026.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.