An Efficient Approximate Technique for Solving Fluid Models

Ana Paula Couto da Silva * Rosa M.M. Leão *
Edmundo de Souza e Silva*

Federal University of Rio de Janeiro, COPPE/PESC, CS Department
CxP 68511, Rio de Janeiro RJ 21941-972, Brazil
{anapaula, rosam, edmundo}@land.ufrj.br

1. INTRODUCTION

Stochastic fluid-flow models have been widely used as an important tool for the analysis of a variety of computer and communication models. In particular, when the event rates of the system under investigation vary in orders of magnitude, the use of fluid models results in considerable computational savings when compared to traditional models where all events are explicitly represented. This is true for instance, in the so called *performability* models [9], where events that represent structural changes in the system (e.g., failure and repair events) occur at much lower rates than those associated with some performance measure, such as the arrival and service of jobs. As another example, consider a queueing model of a communication network channel. The intervals between events associated with packet arrival and departure from a buffer may be orders of magnitude smaller than the intervals that represent changes in the arrival rate.

A fluid may be thought of a rate reward that is gained (or lost) per unit time, depending on the system state. For instance, in a queueing system, the arrival and departure of packets can be represented by a reward associated with states that model the arrival rate changes. In other words, if state i indicates that the arrival of packets is λ_i, then the reward at that state is $\lambda_i - C$ where C is the capacity of the channel.

Several methods have been proposed to analyze fluid models such as [2, 14, 7, 3, 5]. These include techniques based on spectral analysis, Laplace transforms and Uniformization, to obtain steady state or transient results. Fluid simulation is another important area of recent research (e.g. [12]).

Our interest is the efficient analytical solution of fluid models. We develop an approximation technique that transforms a continuous time Markov rate reward model into an equivalent discrete time impulse reward model. In this discrete model an impulse reward, that we call *batch fluid* is gained at state transition points. The resulting model is a QBD model with a special structure. The efficient solution of this particular QBD model is another contribution of our work.

*Rosa M.M. Leão and E. de Souza e Silva were supported in part by grants from CNPq, and FAPERJ. A.P.C. da Silva has a fellowship from CNPq.

The analysis of QBD models has been an important area of research in the last twenty years [10, 8], and substantial work has been done to obtain efficient solutions for both infinite and finite state models. For finite state models (which is the class of models we are interested in), a very clever technique is the *Folding algorithm* [15] which is a form of odd-even reduction. In [10] another two techniques are described, one that uses matrix geometric concepts and another that requires the QBD sub-diagonal matrices to be of rank 1 (for instance, the case for M/PH/1/K and PH/M/1/K models). Meo *et al* [13] developed a reduction procedure which is particularly efficient when the lower diagonal blocks (in the case of upper Hessenberg matrices) have a simple structure that is easy to invert. More recently, the work of [4] gives necessary and sufficient conditions for a QBD Markov chain to possess the so called *level-geometric* stationary distribution [10]. The resulting QBD *batch fluid* model we obtain has another special structure that yields computational gains when compared to the *Folding algorithm* applied to the *equivalent* model constructed.

A similar approach using the QBD model in order to calculate the fluid distribution can be found in [1]. In this work the authors use the QBD model as an intermediate step to calculate the exact steady state distribution. In addition, it is necessary the construction of a set of processes on a common probability space with a distribution coupling relation between them. At the end of the procedure, the Markov renewal kernel theory is used.

The model and method presented here are simpler than the approach shown in [1]. Although our procedure gives an approximate result, low-cost solution, even if approximate, are usually desirable in the preliminary phases of analysis and design, where the analyst usually needs to have only a rough idea of the system behavior.

Section 2 presents our model and develop the solution technique for steady state. (Transient measures can be obtained from the model but we do not address this issue due to space limitations.) In section 3 we present very simple examples to evaluate the accuracy of the approximation, and discuss the computational gains.

2. THE MODEL AND THE COMPUTATIONAL ALGORITHM

We consider a homogeneous continuous time Markov chain (CTMC) $\mathcal{X} = \{X(t), t \geq 0\}$ with finite state space $\mathcal{S} = \{1, \ldots, M\}$ and infinitesimal generator $\mathbf{Q}$, where $q_i = \sum_{j \neq i} q_{ij}$ is the exponential rate out of state i. We assume that a rate reward (or fluid) r_i is associated with state i, that is a reward of r_i is accumulated per unit time while in state i. Let $CR(t)$ be the cumulative rate reward during $(0, t)$, i.e., $CR(t) = \int_0^t r_{\mathcal{X}(\tau)} d\tau$. We assume that $0 \leq CR(t) \leq B$. We are interested in calculating $E[CR] = \lim_{t \to \infty} E[CR(t)]$.

Let γ be a given parameter value that satisfies: $\gamma \leq \gamma_{\max} = \frac{r_l}{q_l}$ such that: $l = \text{argmin}_i \left\{ \frac{|r_i|}{q_i} \right\}$. For each state i choose $\Lambda_i = |r_i/\gamma|$. Clearly, Λ_i satisfies: $\Lambda_i \geq q_i$. In what follows we transform the fluid model defined above (or equivalent the Markov reward rate model) into a discrete time Markov chain which has an special structure.

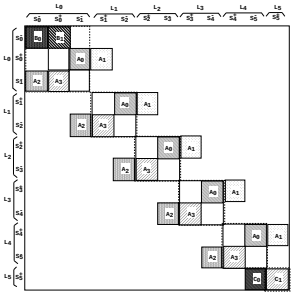
Let $\mathcal{Y}$ be a CTMC with the same state space as $\mathcal{X}$ constructed as follows. Associate to each state i an event ψ_i that triggers with rate Λ_i at the end of exponentially distributed intervals. Let p_{ij} be the probability of moving from i to j when ψ_i triggers. We set $p_{ij} = q_{ij}/\Lambda_i$. Note that $\mathcal{X}$ and $\mathcal{Y}$ are different processes, but we can easily obtain the steady state probabilities for each state of $\mathcal{X}$ by solving process $\mathcal{Y}$.

Let δ_i be the random variable which is equal to the reward gained in a visit to state i of $\mathcal{Y}$. If $CR(t)$ is unbounded, i.e. there is no limit for the total amount of reward that can be accumulated by the system, then $E[\delta_i] = \gamma$, independently of i.

The approximation we propose is to assume that the amount of reward gained when $\mathcal{Y}$ transitions from i to j is equal to γ. In other words, due to the transformation of process $\mathcal{X}$ into $\mathcal{Y}$, the *expected* amount of fluid gained per visit to state i is $|\gamma|$ independently of i. (Note that γ can be positive or negative, depending on the state visited, but the absolute value is state independent due to the choice of γ and Λ_i.) The key to the approximation is to assume that the process gains an impulse reward γ per visit to a state. We call parameter γ the *batch fluid* received at each visit to any of the states in $\mathcal{Y}$.

We proceed by constructing a discrete time Markov chain (DTMC) $\mathcal{Y}^*$ from process $\mathcal{Y}$. The state space $\mathcal{S}^*$ of $\mathcal{Y}^*$ is such that $\mathcal{S}^* = \{(i, l) : 1 \leq i \leq M, 0 \leq l \leq \lfloor B/\gamma \rfloor\}$ and, for each state in $\mathcal{Y}$ there are $\lfloor B/\gamma \rfloor + 1$ states in $\mathcal{Y}^*$. The transition probability matrix $\mathbf{P}^* = [p^*_{ij,kl}]$ is obtained from $\mathcal{Y}$ as follows: if $r_i > 0$ then $p^*_{ij,kl} = p_{ij}$ for $l = \min\{k+1, B\}$; if $r_i < 0$, $p^*_{ij,kl} = p_{ij}$ for $l = \max\{k-1, 0\}$; and 0 otherwise. From the values of $p^*_{ij,kl}$ it is easy to see that, each time $\mathcal{Y}$ makes a transition, process $\mathcal{Y}^*$ increments the amount of *batch fluid* received in excess of the fluid transmitted, or decrements the amount drained from the reservoir while in the previous state.

Matrix $\mathbf{P}^*$ is clear a QBD process and it has a special structure that allows us to solve it more efficiently than previous approaches. We partition the state space of $\mathcal{Y}^*$ into subsets $\mathcal{S}_l^+$ and $\mathcal{S}_l^-$ such that state $(i, l) \in \mathcal{S}_l^+$ iff $r_{(i,l)} > 0$ and $(i, l) \in \mathcal{S}_l^-$ iff $r_{(i,l)} < 0$. (To simplify the explanation we assume that all reward rates are different than zero. Many models of interest satisfy this assumption though.) We define another partition of the state space as follows: $\mathcal{L}_0 = \mathcal{S}_0^- \bigcup \mathcal{S}_0^+ \bigcup \mathcal{S}_1^-$, $\mathcal{L}_l = \mathcal{S}_l^+ \bigcup \mathcal{S}_{l+1}^-$ for $l > 0$, $\mathcal{L}_L = \mathcal{S}_L^+$ where $L = \lfloor B/\gamma \rfloor$. Figure 1 illustrates the structure of the state transition probability matrix $\mathbf{P}^*$ for the DTMC $\mathcal{Y}^*$, the partitions defined above and the sub-matrices obtained.



1: $\mathbf{P}^*$ and the partitions

Clearly, the obvious partition should be to choose $\mathcal{L}_l = \mathcal{S}_l^+ \bigcup \mathcal{S}_l^-$, for $l \geq 0$ and the model could be solved with the Folding algorithm. However, organizing the matrix as in Figure 1 will produce further computational gains.

The algorithm we propose works with subsets $\mathcal{S}_l^+$ and $\mathcal{S}_l^-$ as follows. We eliminate the "even sub-blocks" (those with even values for l) corresponding to $\mathcal{S}_l^+$ for $0 < l < \lfloor B/\gamma \rfloor$ (i.e., all even blocks except those in the boundary.) Due to the structure of the blocks, no matrix inversion is needed in this step. We then eliminate all even sub-blocks corresponding to $\mathcal{S}_l^-$. The resulting matrix after these two block eliminations has the same structure as the original matrix. Therefore, we can repeat the steps above until only the boundary blocks are present. Combining the two sub-blocks eliminations, we obtain, for each step:

$$
\begin{aligned}
\mathbf{A}_0^{(i)} &= \mathbf{A}_0^{(i-1)} + \mathbf{A}_1^{(i-1)} \mathbf{A}_0^{(i-1)} (\mathbf{I} - \mathbf{A}_3^{(i-1)} \mathbf{A}_0^{(i-1)})^{-1} \mathbf{A}_2^{(i-1)} \\
\mathbf{A}_1^{(i)} &= (\mathbf{A}_1^{(i-1)})^2 + \mathbf{A}_1^{(i-1)} \mathbf{A}_0^{(i-1)} (\mathbf{I} - \mathbf{A}_3^{(i-1)} \mathbf{A}_0^{(i-1)})^{-1} \mathbf{A}_3^{(i-1)} \mathbf{A}_1^{(i-1)} \\
\mathbf{A}_2^{(i)} &= \mathbf{A}_2^{(i-1)} (\mathbf{I} - \mathbf{A}_3^{(i-1)} \mathbf{A}_0^{(i-1)})^{-1} \mathbf{A}_2^{(i-1)} \\
\mathbf{A}_3^{(i)} &= \mathbf{A}_3^{(i-1)} + \mathbf{A}_2^{(i-1)} (\mathbf{I} - \mathbf{A}_3^{(i-1)} \mathbf{A}_0^{(i-1)})^{-1} \mathbf{A}_3^{(i-1)} \mathbf{A}_1^{(i-1)}
\end{aligned}
$$

The stationary probability vectors for the blocks can be easily calculated from the balance equations and working backwards with the sub-blocks. Furthermore, from the solution of $\mathcal{Y}^*$ we can trivially obtain that for $\mathcal{X}$.

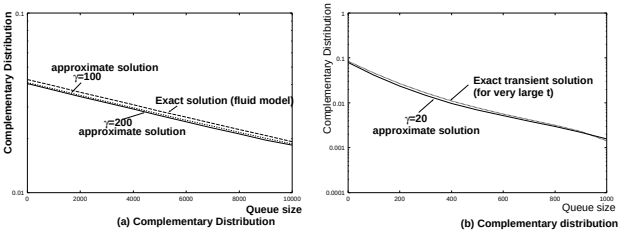
The algorithm above works with even blocks and only one matrix must be inverted at each step. The algorithm is similar to the odd-even permutation of the folding algorithm but, unlike that algorithm, it does not require an extra inversion when the reduced matrix has an odd number of blocks [10]. Furthermore, since it works with subsets $\mathcal{S}_l^+$ and $\mathcal{S}_l^-$, it requires the inversion of a smaller matrix than the one that must be inverted at each step of the Folding algorithm.

From these observations it is not difficult to see that the computational complexity of the algorithm above is reduced by one order of magnitude when compared to the Folding al-

gorithm applied to $\mathcal{Y}^*$. Furthermore, due to the fluid approximation we propose, much larger gains can be obtained, as illustrated in the simple examples in section 3.

3. EXAMPLE

Figure 2 shows the results of two very simple examples that illustrate the potential of the approach. The first is a histogram model of a video source that feeds a single server queue, and our objective is to evaluate the accuracy of the approximation. This fluid model has 8 states, each associated with a different reward rate. Figure 2(a) plots the results obtained with the proposed technique and those calculated using the approach of [2], for buffer size of 10,000. In this example we used two values for parameter γ (100 and 200) to illustrate the sensitivity of the technique with γ. As can be observed, the maximum error was smaller than 10^{-2}.



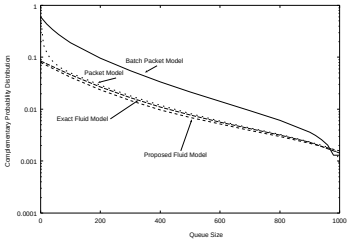
2: Examples.

For the second example we construct a 44 state fluid model with 30 different flow values (or reward rate values) obtained from traces collected at the output link of our department. Figure 2(b) plots the buffer size complementary distribution for maximum buffer size equal to 1,000. This model is too large to be solved with the method of [2] and so we used the approach of [11]. (This approach obtains transient measures, and so we used a very large value of t as an approximation for the steady state result.) The value of parameter γ is 20.

In both examples the approximate technique applied to the simple fluid models is almost two orders of magnitude cheaper to solve than the corresponding MC packet level queueing models. Further savings can be obtained depending on the structure of $\mathbf{A}_0$ and $\mathbf{A}_3$ in Figure 1.

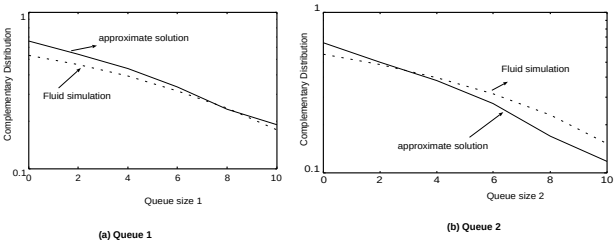
Figure 3 illustrates the different results obtained when a fluid model is used in comparison with a packet model. The second model in this section is used for the comparison. The example shows, as expected, the good agreement between the packet and fluid models, except for the boundary values of the buffer size. The figure also plots the results for a model in which batch arrivals and departures are used as an approximation for a packet level model, in order to lower the number of levels in the buffer. (In the packet batch example, all rates are re-scaled so that the buffer contains 50 levels, each corresponding to batches of 200 packets.) As can be observed, the batch model provides a very poor approximation of the fluid model, and the error can be as high as 400% in this case, for buffer level of 200 packets.

This approximate model can also be extended to the case



3: Fluid versus packet model results.

where there are two fluid queues that share the same channel with capacity C. The amount C is divided between the two queues using a parameter w_i. If one of the queues is empty, all capacity is given to the other one. Figure 4 shows the results of the distribuiton of the two queues. A fluid simulation result with the TANGRAM-II tool [6] is used in order to compare the results.



4: Two Fluid Queues.

4. REFERENCES

[1] S. Ahn and V. Ramaswami. Fluid flow models and queues - a connection by stochastic coup ling. *Stochastic Models*, 19(3):325–348, 2003.

[2] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic Theory of a Data-Handling System with Multiple Sources. *The Bell System Technical Journal*, 61(8):1871–1894, 1982.

[3] M. Bladt, B. Meini, M. F. Neuts, and B. Sericola. Distributions of Reward Functions on Continuous-Time Markov Chains. In *Matrix-analytic methods*, pages 39–62. 2002.

[4] T. Dayar and F. Quessette. Quasi-birth-and-death processes with level-geometric distribution. *SIAM Journal on Matrix Analysis and Applications*, 24(1):pp. 281–291, 2002.

[5] E. de Souza e Silva and H. Gail. An algorithm to calculate transient distributions of cumulative rate and impulse based reward. *Stochastic Models*, 14(3):509–536, 1998.

[6] E. de Souza e Silva and R. M. M. Leão. The TANGRAM-II Environment. In *Computer Peformance Evaluation - Modelling Techniques and Tools - $11^{\underline{th}}$ International Conference (TOOLS2000)*, volume 1786, pages 366–369. Springer, Março 2000.

[7] A. I. Elwalid and D. Mitra. Fluid Models for the Analysis and Design of Statistical Multiplexing with

Loss Priorities on Multiple Classes of Bursty Traffic. In *Infocom*, 1992.

[8] W. Grassmann and D. Stanford. Matrix Analytic Methods. In W. Grassmann, editor, *Computational Probability*, pages 153–203. Kluwer, 2000.

[9] B. R. Haverkort, R. Marie, G. Rubino, and K. S. Trivedi. *Performability Modelling: Techniques and Tools*. Wiley, 2001.

[10] G. Latouche and V. Ramaswami. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. SIAM, 1999.

[11] R. M. Leão, E. de Souza e Silva, and M. C. Diniz. Traffic Engineering using Reward Models. In Elsevier, editor, *Proc. of the International Teletraffic Congress - ITC17*, volume 4, pages 1101–1112, December 2001.

[12] B. Liu, D. Figueiredo, Y. Guo, J. Kurose, and D. Towsley. A study of networks simulation efficiency: Fluid simulation vs. packet level simulation. In *Infocom 2001*, 2001.

[13] M. Meo, E. de Souza e Silva, and M. Marsan. Efficient solution for a class of Markov chain models of telecommunication systems. *Performance Evaluation*, 27&28:603–625, Outubro 1996.

[14] D. Mitra. Stochastic Theory of a Fluid Model of Producers and Consumers Coupled by a Buffer. *Adv. Appl. Prob*, 20:646–676, 1988.

[15] J. Ye and S. Q. Li. Folding Algorithm: A Computational Method for Finite QBD Processes with Level-Dependent Transitions. *IEEE Trans. Commun.*, 42(2):pp. 625–639, 1994.