Tighter Lyapunov Truncation for Multi-Dimensional Continuous Time Markov Chains with Known Moments

Gagan Somashekar[1] Mohammad Delasay[2] Anshul Gandhi[1]
{[1]PACE lab, Department of Computer Science,[2]College of Business}, Stony Brook University
[1]{gsomashekar,anshul}@cs.stonybrook.edu [2]mohammad.delasay@stonybrook.edu

1. INTRODUCTION

Continuous Time Markov chains (CTMCs) are widely used to model and analyze networked systems. A common analysis approach is to solve the system of balance equations governing the state transitions of a CTMC to obtain its steady-state probability distribution, and use the state probabilities to derive or compute various performance measures.

In many systems, the state space of the underlying CTMC is infinite and multi-dimensional with state-dependent transitions; exact analysis of such models is challenging. For example, the exact probability distribution of the number of jobs in the Discriminatory Processor Sharing (DPS) system, first proposed by Kleinrock in 1967 [4], is still an open challenge. Likewise, obtaining the exact state probabilities of quasi-birth-and-death (QBD) processes with level-dependent transitions is known to be challenging [1]; QBDs are infinite state space multi-dimensional Markov chains in which states are organized into levels and transitions are skip-free between the levels.

A common approximation approach for such CTMCs is to truncate the state space, in one or many dimensions, and solve the resulting *truncated* CTMC with finite state-space, using analytical or numerical methods (such as matrix analytic methods). If truncation bounds are chosen carefully, the steady-state distribution of the truncated chain should approximate those of the original infinite chain accurately. However, an arbitrary truncation may result in inaccuracy; that is, the computed steady-state distribution and performance measures from the truncated CTMC may not closely approximate those of the original CTMC.

Truncation algorithms have been designed for this exact reason. For example, algorithms based on Lyapunov functions guarantee to provide truncation bounds to satisfy a desired accuracy [3]; if the maximum acceptable error is $0 < \epsilon < 1$, the probability mass of the states residing within the truncated CTMC is guaranteed to be at least $1-\epsilon$. The central idea in the state-of-the-art Lyapunov function based truncation, proposed by Dayar et al. [3], is to identify an *attractor set*, a subset of states towards which the CTMC drifts, and then truncate the infinite state space to ensure that the attractor set is part of the truncated CTMC. The drift of the Lyapunov function (the expected rate of change in its value) is finite for states in the attractor set, and is negative for states outside the attractor set, facilitating the determination of the attractor set.

An issue with such truncation methods is that they lead to loose bounds, which results in unnecessarily large truncated CTMCs and consequently, expensive time and computational effort to analyze them. Such methods are conservative due to the fact that the truncation algorithms do not leverage any properties of the original CTMC.

In this short paper, we improve the Lyapunov function based truncation to provide tighter bounds to satisfy a desired accuracy on the probability mass of multi-dimensional CTMCs for which the moments are known but the steady-state distribution is unknown. By leveraging the known moments, we scale the drift function more efficiently to obtain tighter truncation. Our truncation approach results in the same computational complexity as Dayar et al. [3] to obtain the bounds, but provides significantly tighter truncation.

We note the reliance of our approach on the moments of the state variables of the original CTMC. However, we note that our approach also applies to CTMCs for which a lower bound on the moments is known. In general, knowing the moments is not enough to obtain the steady-state distribution; the DPS system is an example of a CTMC for which the moments of the number of jobs are known [5], but its steady-state distribution is unknown and needs to be computed. In today's customer-facing online services, e.g., Amazon, performance metrics typically take the form of tail probabilities, thus requiring the steady-state distribution.

We demonstrate the effectiveness of our proposed truncation procedure in computing the steady-state distribution of the DPS system, which has been studied for decades [4], but continues to be a "class of models notoriously hard to analyze in an exact manner" [6]. Since the exact moments of its queue-length distribution are known [5], our procedure can be applied. Through our extensive numerical experiments, we show that our proposed procedure achieves on average 32%, and up to 68%, tighter truncation bounds over those obtained from the state-of-the-art Dayar et al. technique [3].

2. PROVIDING A TIGHTER TRUNCATION

Let $\{N(t), t \geq 0\}$ be an ergodic k-dimensional CTMC with state space S and generic state $n = (n_1, n_2, \ldots, n_k)$. Let N_i denote the random variable corresponding to n_i, with $N(t) = (N_1(t), N_2(t), \ldots, N_k(t))$. Let $\pi(n) = \pi(n_1, n_2, ..., n_k)$ denote the steady-state probability of being in state n, and let Q denote the infinitesimal generator matrix. Without loss of generality, assume that the CTMC is infinite in m-dimensions and finite in the remaining $(k-m)$ dimensions.

The stability of a Markov chain can be established if a ***Lyapunov*** function that maps the state space to positive real numbers is found such that its ***drift*** (the expected rate of change in its value) is negative outside a finite subset of the state space, known as the ***attractor set*** (C), and is bounded in this finite subset. Formally, if $N(t)$ is ergodic, there exists a Lyapunov function $g : S \to R_{\geq 0}$ and set $C \subset S$, with $\overline{C} = S \setminus C$, such that for some $\gamma > 0$ [3],

1. $(\mathrm{d}/\mathrm{d}t)\ E[g(N(t))|N(t) = n] \leq -\gamma,\ \forall n \in \overline{C}$,
2. $(\mathrm{d}/\mathrm{d}t)\ E[g(N(t))|N(t) = n] < \infty,\ \forall n \in C$, and

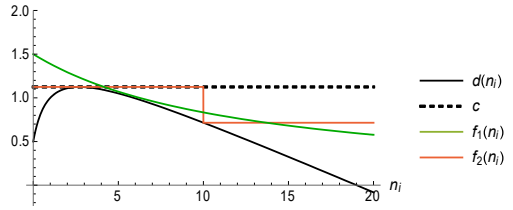


Figure 1: Comparison of our state-dependent drift bounds ($f_1(n_i)$ and $f_2(n_i)$) with the supremum bound (c).

3. $\{n \in S | g(n) \leq r\}$ is finite, $\forall r < \infty$,

where $d(n) = (\mathrm{d}/\mathrm{d}t)\, E[g(N(t))|N(t) = n]$ is the value of the drift function in state n.

Dayar et al. use the above conditions to derive an upper bound on the probability mass in $\overline{C}$. The authors define a function $g^*(n) = g(n)/(c+\gamma)$, where $c = \sup_{n\in S} d(n)$ (note that c is finite from condition 2) and γ is as defined in condition 1. Consequently, using condition 1, we have:

$$d^*(n) = \frac{d(n)}{c+\gamma} \leq \frac{c}{c+\gamma} - I_{\overline{C}}, \tag{1}$$

where $I_{\overline{C}} = 1$ if $n \in \overline{C}$ and 0 otherwise. If d is the vector of the drift function values, g is the vector of the Lyapunov function values, and π is the vector of steady-state probabilities for all the states, then, by the definition of drift [3]:

$$\begin{aligned} d^T = Qg^T &\implies \pi d^T = \pi Q g^T = 0 \\ &\implies \pi d^{*T} = \pi Q g^{*T} = 0. \end{aligned} \tag{2}$$

Using Eqs. (1) and (2), a bound on the probability mass in $\overline{C}$ is obtained as follows:

$$\begin{aligned} 0 = \sum_{n\in S} d^*(n)\cdot\pi(n) &\leq \sum_{n\in S}\pi(n)\cdot\frac{c}{c+\gamma} - \sum_{n\in\overline{C}}\pi(n) \\ \implies \sum_{n\in\overline{C}}\pi(n) &\leq \sum_{n\in S}\pi(n)\cdot\frac{c}{c+\gamma} = \frac{c}{c+\gamma}. \end{aligned} \tag{3}$$

This guarantees that the probability mass in C is at least $1 - c/(c+\gamma)$. Hence, the value of γ obtained by solving $c/(c+\gamma) = \epsilon$, where $0 < \epsilon < 1$, guarantees that a truncated CTMC containing C satisfies the accuracy $1-\epsilon$ on the loss of probability mass outside the truncated CTMC. Once γ is found, the set C can be found as follows:

$$C = \{n \in S \mid d(n) > -\gamma\}. \tag{4}$$

Eq. (3) provides an upper bound on the probability mass in $\overline{C}$; the actual mass could be much smaller than $c/(c+\gamma)$. The authors in [3] acknowledge this issue. Indeed, our experiments in Section 3.1 show that the truncation bounds obtained using the above technique are quite loose.

2.1 Our approach: bounding the drift

Our goal is to find tighter Lyapunov truncation bounds than those obtained via Dayar et al. [3] without sacrificing accuracy. A tighter truncation reduces the state space size, and results in more efficient and less time-expensive computation to obtain the steady-state probabilities of the states that encompass most of the distribution mass.

Dayar et al. bound the drift function in Eq. (1) by the trivial upper bound of $c = \sup_{n\in S} d(n)$. The advantage of using the supremum is that the bound on $\sum_{n\in\overline{C}}\pi(n)$ in Eq. (3) can be easily obtained as $\sum_{n\in S}\pi(n)\cdot c/(c+\gamma) = c/(c+\gamma)$. To illustrate this, the solid black plot in Fig. 1 is the drift as a function of the state variable n_i, $d(n_i)$, for

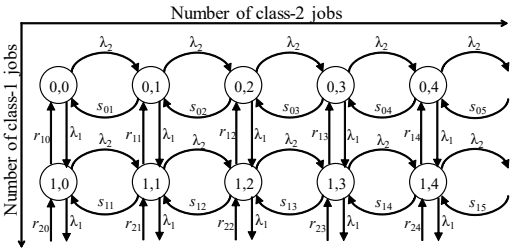


Figure 2: M/M/1-DPS with two customer classes; for state (i,j), i and j are the number of class-1 and 2 jobs.

the DPS chain that we analyze in Section 3.1; the dashed black line is the supremum, c. The drift is a state-dependent function, however, the supremum (the bounding function of the drift in [3]) is a fixed function that does *not* adapt to changes in the state variate. Hence, the supremum is a loose upper bound on the drift function; as Fig. 1 shows, c deviates substantially from the drift for higher values of n_i.

The key idea in our approach is to employ a *state-dependent* function that mimics, to some extent, the changes in the drift function in response to the state variate to provide tighter upper bounds for the drift function; examples of such functions include a decaying function (e.g., $f_1(n_i)$ in Fig. 1) or a step function (e.g., $f_2(n_i)$ in Fig. 1). However, when using a state-dependent generic bounding function, $f(n)$, in place of c in Eq. (3), the weighted sum $\sum_n \pi(n)\cdot f(n)$ may not be easily obtained in closed-form, making it difficult to solve for the set C. We formalize this challenge below.

Consider a generic bounding function, $f(n)$, that bounds the drift, $d(n)$, i.e., $f(n) \geq \max(d(n), 0)$, $\forall n \in S$. Defining $h(n) = d(n)/(f(n)+\gamma)$, we have, similar to Eq. (1):

$$h(n) = \frac{d(n)}{f(n)+\gamma} \leq \frac{f(n)}{f(n)+\gamma} - I_{\overline{C}}. \tag{5}$$

Using Eqs. (2) and (5), we have (similar to Eq. (3)):

$$\sum_{n\in\overline{C}}\pi(n) \leq \sum_{n\in S}\pi(n)\cdot\frac{f(n)}{f(n)+\gamma} = E\left[\frac{f(n)}{f(n)+\gamma}\right]. \tag{6}$$

The set C can be obtained by setting the right-hand-side of Eq. (6) to ϵ, solving for γ, and then using Eq. (4). However, this requires knowing the expectation of $f(n)/(f(n)+\gamma)$.

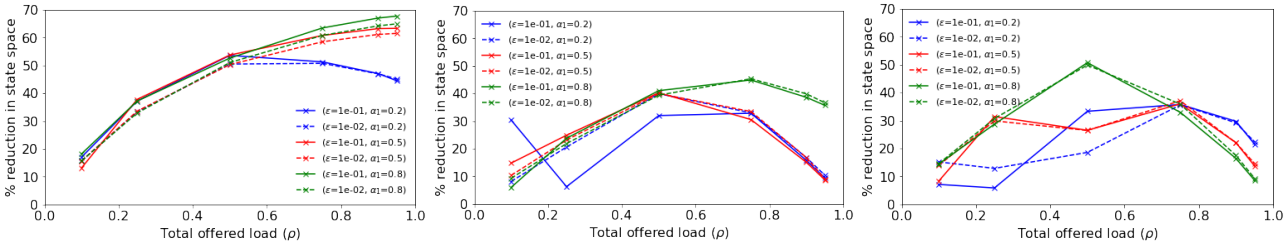
2.2 Bounding the drift using a step function

We demonstrate the applicability of our approach using a step function, illustrated in Fig. 1 as $f_2(n)$, that initially is equal to the supremum, and drops to a lower value, c_1, along one dimension of the state space (for simplicity), to provide a tighter upper bound on the drift for larger n.

We improve the upper bound of the drift along an arbitrary infinite dimension, say dimension $i \in \{1, 2, \cdots, m\}$ corresponding to the i^{th} state variable, N_i. We formally define the step function, which drops to $c_1 \leq c$ for $n_i > a$, as Eq. (7), and substitute it in Eq. (6) to get Eq. (8).

$$f_2(n) = \begin{cases} c &= \sup_{\forall n\in S} d(n), & \forall n_i \leq a, \\ c_1 &= \sup_{\forall n\in\{S|n_i>a\}} d(n), & \forall n_i > a. \end{cases} \tag{7}$$

$$\begin{aligned} \sum_{n\in\overline{C}}\pi(n) &\leq \frac{c}{c+\gamma}\sum_{n_i\leq a}\pi(n) + \frac{c_1}{c_1+\gamma}\sum_{n_i>a}\pi(n) \\ &= \frac{c}{c+\gamma} - \left(\frac{c}{c+\gamma} - \frac{c_1}{c_1+\gamma}\right)\sum_{n_i>a}\pi(n). \end{aligned} \tag{8}$$



(a) Case 1: $\rho_1 < \rho_2$ (b) Case 2: $\rho_1 = \rho_2$ (c) Case 3: $\rho_1 > \rho_2$

Figure 3: Reduction in state space over Dayar et al. for different total (ρ) and per-queue (ρ_1 and ρ_2) offered loads.

The Paley-Zygmund inequality says that for a positive random variable X and $0 < \theta < 1$, $\Pr(X > \theta E[X]) \geq (1-\theta)^2 E[X]^2/E[X^2]$. Applying this inequality for N_i and setting $a = \theta E[N_i]$, we can write Eq. (8) as follows:

$$\sum_{n \in \overline{C}} \pi(n) \leq \frac{c}{c+\gamma} - \left(\frac{c}{c+\gamma} - \frac{c_1}{c_1+\gamma}\right)(1-\theta)^2 \frac{E[N_i]^2}{E[N_i^2]}. \quad (9)$$

The above would result in tighter truncation bounds than Dayar et al. when $c_1 \neq c$. However, such bounds depend on the availability of the first and second moments of the marginal distribution of N_i.

3. APPLICATION TO THE DPS SYSTEM

We demonstrate the applicability of our technique for the DPS system, which is an M/M/1 queue with multiple customer classes that operates under a disproportionate processor sharing policy [4]. In DPS, the server capacity is processor shared based on a given weight vector $\alpha = (\alpha_1, \alpha_2, ..., \alpha_k)$, where α_i is the weight associated with class-i. If there are N_i jobs of class-i, each class-j job gets a fraction $\alpha_j / \sum_{i=1}^{k} \alpha_i N_i$ of the server's capacity.

For evaluation, we consider $k = 2$ customer classes. Fig. 2 shows the corresponding CTMC with two infinite dimensions where λ_i and μ_i are the arrival and service rates of customer class $i \in \{1, 2\}$. The transition rates from state (i, j) to $(i-1, j)$ and $(i, j-1)$ are $r_{i,j} = i\alpha_1\mu_1/(i\alpha_1 + j\alpha_2)$ and $s_{i,j} = j\alpha_2\mu_2/(i\alpha_1 + j\alpha_2)$, respectively.

While the DPS model was introduced in the late 1960s, the exact probability distribution of the underlying CTMC continues to remain elusive due to the complex and non-repeating structure of its multi-dimensional infinite CTMC. However, the exact moments of the DPS system are known [5], enabling the application of our truncation bounds from Eq. (9).

3.1 Evaluation

As reported in the stability literature [2], function $g(n_1, n_2) = (\alpha_1 n_1)/2\lambda_1 + (\alpha_2 n_2)/2\lambda_2$ is a feasible Lyapunov function for the DPS model, and its drift in state (n_1, n_2) is:

$$d(n_1,n_2) = \lambda_1 \left(g(n_1+1,n_2) - g(n_1,n_2)\right) + \lambda_2 \left(g(n_1,n_2+1) - g(n_1,n_2)\right) + s_1 \left(g(n_1-1,n_2) - g(n_1,n_2)\right) + s_2 \left(g(n_1,n_2-1) - g(n_1,n_2)\right). \quad (10)$$

The first step in finding the truncation bounds is to define the function in Eq. (7) by setting an appropriate value for a, which in turn is determined via θ since $a = \theta E[N_i]$; i is the dimension along which the bound is being improved. Noting that a smaller θ provides a tighter bound in Eq. (9), we set $\theta = 0.01$. We then derive a by obtaining $E[N_i]$ via the known moments of DPS [5]. We then compute c and c_1 via Eq. (7). Finally, using the known second moment [5], we compute the right-hand-side of Eq. (9); by setting this to ϵ, the tolerance for probability mass loss, we solve for γ, which in turn gives us the attractor set C via Eq. (4). The chain is then truncated such that it includes all the states in C. In our experiments, we truncate the DPS CTMC along the two dimensions at $m_1 = \max_{(n_1,n_2)\in C} n_1$ and $m_2 = \max_{(n_1,n_2)\in C} n_2$.

To evaluate the improvement in truncation over Dayar et al., we numerically experiment with 72 different parameter sets for λ_1, λ_2, μ_1, and μ_2, spanning the offered load in the range $[0.1, 0.95]$; offered load is expressed as $\rho = \rho_1 + \rho_2$, where $\rho_i = \lambda_i/\mu_i$. For each parameter set, we vary $\alpha_1 \in \{0.2, 0.5, 0.8\}$ ($\alpha_2 = 1 - \alpha_1$) and the truncation error $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$. We obtain truncation bounds by employing the step function over either dimension ($i = \{1, 2\}$ in Eq. (7)), and use the tighter of the two.

We find that the ρ, ρ_1, and ρ_2 values impact the truncation improvements over Dayar et al. significantly; we thus present the results along these parameters in Fig. 3, which plots the improvements for $\epsilon = 10^{-1}$ and $\epsilon = 10^{-2}$; results are similar for other ϵ values. We see that our approach provides as much as 68% reduction in the state space over Dayar et al.; in other words, the truncated chain can be up to 68% smaller while maintaining the same accuracy level. In general, the improvement is highest for moderate offered loads ($\rho \approx 0.5$). Across all experiments, the average improvement is around 32%. For $0.5 \leq \rho \leq 0.95$, the average improvement is 38%; since the truncated CTMC contains more states for higher loads, the absolute reduction in state space (number of states) is much higher for this range.

We also compare the obtained moments from the truncated chains with the exact ones provided in [5]. For $\epsilon = 0.1$, the average errors for the first three moments are around 10^{-4}%, 10^{-3}%, and 2×10^{-3}%, respectively. We further validate our approach by comparing the steady-state distribution of the truncated CTMC for $\alpha_1 = \alpha_2 = 0.5$ with that of the classical processor sharing system, which is a DPS with $\alpha_1 = \alpha_2$. The maximum observed difference in per-state probability, for $\epsilon = 0.1$, is only 10^{-6}%.

Acknowledgment: This work was supported by NSF grants CNS-1617046 and CNS-1750109.

4. REFERENCES

[1] Hendrik Baumann and Werner Sandmann. Numerical solution of level dependent quasi-birth-and-death processes. *Procedia Computer Science*, 1(1):1561–1569, 2010.

[2] G. De Veciana and T. Konstantopoulos. Stability and performance analysis of networks supporting elastic services. *IEEE/ACM Transactions on Networking*, 9(1):2–14, 2001.

[3] T. Dayar et al. Infinite level-dependent QBD processes and matrix-analytic solutions for stochastic chemical kinetics. *Advances in Applied Probability*, 43(4):1005–1026, 2011.

[4] Leonard Kleinrock. Time-shared systems: A theoretical treatment. *J. ACM*, 14(2):242–261, 1967.

[5] Kiran M. Rege and Bhaskar Sengupta. Queue-length distribution for the discriminatory processor-sharing queue. *Operations Research*, 44(4):653–657, 1996.

[6] Petra Vis. *Performance analysis of multi-class queueing models*. PhD thesis, September 2017.