Beyond Jackson: New Decomposition Approximations for Queueing Networks Based on Indices of Dispersion

Ward Whitt* Wei You†

ww2040@columbia.edu weiyou@ust.hk

January 1, 2022

Abstract

The product-form structure of Markov open queueing networks stemming from Jackson (1957) motivated decomposition approximations for open networks with non-Poisson arrival processes and non-exponential service times. Whitt and You (2021) developed a new decomposition approximation, where each queue is treated as a $G/GI/1$ model and each flow is partially characterized by its rate and a scaled version of the variance-time curve, called the index of dispersion for counts (IDC). A robust queueing technique is used to generate approximations of the mean steady-state performance at each queue from the IDC of the total arrival flow and the service specification at that queue. There remain many opportunities to improve the algorithm and extend it to more general models.

Keywords: *queueing networks, non-Markov queueing networks, robust queueing, index of dispersion, decomposition approximations, heavy traffic*

*Emeritus Professor, Department of Industrial Engineering and Operations Research, Columbia University

†Department of Industrial Engineering and Decision Analytics, Hong Kong University of Science and Technology

1 Introduction

One of the great successes of queueing, for both theory and applications, was the development of the theory of product-form Markovian queueing networks, e.g., as in Kelly [9], stemming from the seminal paper of Jackson [8]. The product-form theory motivated considering decomposition approximations for more general open queueing networks, e.g., with non-exponential service-time distributions and non-Poisson arrival processes, as developed by [10, 14]. In these early decomposition approximations, the arrival processes were partially characterized by their rate and a single variability parameter, corresponding to the variance of an interarrival time in a renewal-process approximation.

In [23] we developed a new decomposition algorithm to approximate the steady-state performance of a single-class open queueing network of single-server queues with unlimited waiting space, the first-come first-served discipline and Markovian routing. The algorithm allows non-renewal external arrival processes, general service-time distributions and customer feedback. Each flow is partially characterized by its rate and a scaled version of the variance-time curve, called the *Index of Dispersion for Counts* (IDC). To elaborate, let A be an arrival counting process at a queue, i.e., $A(t)$ counts the total number of arrivals in the interval $[0, t]$. We assume that A is a stationary point process as in [5, 12]. We partially characterize A by its rate and its IDC, a function of non-negative real numbers $I_A : \mathbb{R}^+ \to \mathbb{R}^+$ defined as in §4.5 of Cox and Lewis [3] by

$$I_A(t) \equiv \frac{\mathrm{Var}(A(t))}{E[A(t)]}, \quad t \geq 0. \tag{1}$$

A reference case is the Poisson process, where $I_A(t) = 1$ for all $t \geq 0$. As regularity conditions, we assume that $E[A(t)]$ and $\mathrm{Var}(A(t))$ are finite for all $t \geq 0$. For renewal processes, it suffices to assume that the time between renewals has a finite second moment. The required IDC functions for the external arrival processes can be calculated from the model primitives or estimated from data. Approximations for the IDC functions of the internal flows are calculated by solving a set of linear equations. The theoretical basis is provided by heavy-traffic limits for the flows established in our previous papers [18, 19, 22].

Building on the ideas of Bandi et al. [1], in [19] we developed a new *Robust Queueing* (RQ) technique to generate approximations of the mean steady-state performance at each queue from the IDC of the total arrival flow and the mean μ^{-1} and squared coeficient of variation (scv) c_s^2 of the service-time distribution at that queue. To obtain the RQ algorithm, we start with a reverse-time construction of the workload process as in §3 of [19]. Given the net-input process $N(t) \equiv Y(t) - t$, $t \geq 0$, the workload at time t, starting empty at time 0,

is obtained from the reflection map Ψ applied to N, i.e.,

$$Z = \Psi(N)(t) \equiv N(t) - \inf_{0 \le s \le t} \{N(s)\}, \quad t \ge 0. \tag{2}$$

With a slight abuse of notation, let $Z(t)$ be the workload at time 0 of a system that started empty at time $-t$. Then $Z(t)$ can be represented as

$$Z(t) \equiv \sup_{0 \le s \le t} \{N(s)\}, \quad t \ge 0, \tag{3}$$

where N is defined in terms of Y as before, but Y is interpreted as the total work in service time to enter over the interval $[-s, 0]$. That is achieved by letting V_k be the k^{th} service time indexed going backwards from time 0 and $A(s)$ counting the number of arrivals in the interval $[-s, 0]$. The workload process $Z(t)$ defined in (3) is nondecreasing in t and hence necessarily converges to a limit Z. In the ordinary stochastic queueing model, $N(s)$ is a stochastic process and hence $Z(t)$ is a random variable. However, in Robust Queueing practice, $N(s)$ is viewed as a deterministic instance drawn from a pre-determined uncertainty set $\mathcal{U}$ of input functions, while the workload Z^* for a Robust Queue is regarded as the worst case workload over the uncertainty set, i.e.

$$Z^* \equiv \sup_{\tilde{N} \in \mathcal{U}} \sup_{x \ge 0} \{\tilde{N}(x)\}.$$

Motivated by the central limit theorem, we use the uncertainty set

$$\mathcal{U} \equiv \left\{ \tilde{N} : \mathbb{R}^+ \to \mathbb{R} : \tilde{N}(s) \le E[N(s)] + b\sqrt{\text{Var}(N(s))},\ s \ge 0 \right\}, \tag{4}$$

where $N(t)$ is the net input process associated with the stochastic queue, so

$$\begin{aligned} E[N(t)] &= E[Y(t) - t] = \rho t - t, \\ \text{Var}(N(t)) &= \text{Var}(Y(t)) = (I_a(t) + c_s^2)\rho t/\mu. \end{aligned}$$

The RQ approximation based on this partial model characterization is

$$E[Z_\rho] \approx Z_\rho^* \equiv \sup_{\tilde{N}_\rho \in \mathcal{U}_\rho} \sup_{x \ge 0} \{\tilde{N}(x)\} = \sup_{x \ge 0} \{-(1-\rho)x + b\sqrt{\rho x (I_A(x) + c_s^2)/\mu}\}, \tag{5}$$

where $b = \sqrt{2}$; see Theorem 2 of [19]. For additional discussion about the motivation for our approach, including why robust optimization can yield good approximations as well as bounds, see [19] and §EC.3 of the e-companion to [19].

2 Problem Statement

The problem we pose here is to improve the RQNA in [23] and extend it to more general models.

2.1. Improvements. 2.1.1. Allowing multiple bottleneck queues. The RQNA developed in [23] exploits the special case of the FCLT for the flows in Theorem 3.1 of [22] in which only a single queue in the network is a bottleneck. That leads to tractable approximations involving one-dimensional *Reflected Brownian Motion* (RBM) supporting the approximations in [23, 24]. The proposed improvement is to allow more bottleneck queues and thus exploit multidimensional RBM. That evidently requires an algorithm for multidimensional RBM as well as new approximation formulas, but there is great promise of better approximations for non-tree network models.

2.1.2. Sophisticated statistical fitting. There is also great potential to exploit large system data sets together with advanced statistical techniques, e.g., machine learning, within this framework to fit the covariance functions of the internal flows, e.g., $Cov(A_{i,j}(t), A_{k,l}(t))$, that play a critical role in non-tree networks; e.g., see (28) of [23] and §§5.2-5.3 of [24]. The RQNA provides an especially promising framework to conduct such investigations.

2.2 Extensions. There are also many opportunities to extend the basic model. Such extensions are no doubt best motivated from the needs of concrete applications, as illustrated by the extensions of QNA in [14] discussed in [11]. Here are some:

2.2.1. Allow multiple servers at each queue.

2.2.2. Allow multiple classes and/or more general routing.

2.2.3. Allow time-varying arrival processes.

2.2.4. Treat closed queueing network models.

3 Discussion

3.1. Performance Comparisons with Alternative Algorithms. Section 6 of [23] is devoted to comparisons of RQNA predictions to simulation and other algorithms for difficult network examples with extensive near-immediate feedback from [4], which are the most challenging for RQNA. These examples are difficult for RQNA because the feedback induces strong dependence among the flows and the service times, as illlustrated by the case of immediate feedback; see §III of [14] and §4 of [23]. Our study shows that RQNA without our special techniques to eliminate near-immediate feedback performs quite poorly, but when we incorporate these special techniques from §4 of [23], RQNA (elim) performs as well as the

Sequential Bottleneck Decomposition (SBD) algorithm from [4], which in turn outperforms QNA from [14] and QNET from [7].

As noted in Sections 1.2 and 7 of [23], RQNA has shown to be especially effective for tree networks. Indeed, RQNA is provably superior for a single queue. First, Theorem 5 of [19] shows that it is asymptotically exact in both light and heavy traffic for the $G/GI/1$. Second, Corollary 2 of [20] shows that a $GI/GI/1$ queue is fully characterized by the four tuple consisting of the rate and IDC of the arrival and service processes. Dramatic examples are provided by Tables 2 and 3 from [20], which show comparisons for queues in series exhibiting the heavy-traffic bottleneck phenomenon from [13]. These tables show the mean waiting times at each of nine exponential queues in series fed by a rate-1 renewal arrival process. The first eight queues have mean service time, and thus traffic intensity, 0.6, while the last queue has mean service time, and thus traffic intensity, 0.9, making it a bottleneck. The interarrival time has a hyperexponential (H_2) distribution with a squared coefficient of variation $c_a^2 = 8.0$ but three possible values for the remaining third parameter: $r = 0.1$, $r = 0.5$ (the common case of balanced means) and $r = 0.9$. (The case $r = 0.1$ makes the process like a batch Poisson process, while the case $r = 0.9$ makes it like a Poisson process; see §V of [15].)

These examples are discussed in the third paragraph of §1.2.2 of [23]. To highlight the potential advantages of RQNA over the other methods, we now focus on the performance at individual queues. For that purpose, Table 1 below compares the approximate mean waiting times determined by the four methods RQNA, QNA, QNET and SBD to simulation for the first and last queues for each value of r.

Since the first queue has a renewal arrival process, the approximations for QNA, QNET and SBD all coincide with the heavy-traffic approximation used in (44) of [14], which is independent of r. However, these three approximations differ at the last queue, which does not have a renewal arrival process. The main point is that only RQNA captures the impact of r (necessarily indirectly, because r is not used directly). From this perspective, RQNA performs far better than the other methods. In this example, also SBD outperforms QNA and QNET. For further perspective on the limitations of approximations for one $GI/GI/1$ queue based on the first two meoments, see [2].

3.2 Extensions. 3.2.1. Allowing multiple servers at each queue. As can be seen from §5.2 of [14], multiple servers at each node was allowed for QNA. An approach to robust queueing with multiple servers was offered by [1], but does not seem very compelling. New ideas seem to be needed to extend [18, 19] to multiple servers.

Table 1: A comparison of four approximations of the expected mean waiting time at one queue to simulation for the first and last queue of nine queues in series from Tables 2 and 3 of [20]. The interarrival-time distribution is hyperexponential (H_2) with squared coefficient of variation $c_a^2 = 8.0$ and third parameter r. All the service-time distributions are exponential (M).

first queue $\rho = 0.6$	Sim	RQNA	QNA	QNET	SBD
$r = 0.9$	1.16	1.13	4.05	4.05	4.05
$r = 0.5$	3.36	3.95	4.05	4.05	4.05
$r = 0.1$	5.69	5.84	4.05	4.05	4.05
last queue $\rho = 0.9$	Sim	RQNA	QNA	QNET	SBD
$r = 0.9$	19.6	27.2	8.9	6.0	36.4
$r = 0.5$	29.2	29.1	8.9	6.0	36.4
$r = 0.1$	29.6	29.3	8.9	6.0	36.4

3.2.2. Allowing multiple classes and/or more general routing. A provision for multiple classes, where each class had its own routing was provided in §2.3 of [14]. The algorithm aggregated the input data to convert it into an associated approximate Markovian routing.

3.2.3. Allowing time-varying arrival processes. A significant start was provided for a single queue with time-varying arrivals in [21], but much more is required to treat networks. For additional background on queues with time-varying arrivals, see [6, 17] and references there.

3.2.4. Treating non-Markov closed networks of queues. Various approaches were explored in the 1980's. Variants of the fixed-population-mean method were effective for large models; e.g., see [16], especially §X.

References

[1] Bandi C, Bertsimas D, Youssef N (2015) Robust queueing theory. *Operations Research* 63(3):676–700.

[2] Chen Y, Whitt W (2020) Algorithms for the Upper Bound Mean Waiting Time in the $GI/GI/1$ Queue. *Queueing Systems* 94:327–356.

[3] Cox DR, Lewis PAW (1966) *The Statistical Analysis of Series of Events* (London: Methuen).

[4] Dai J, Nguyen V, Reiman MI (1994) Sequential bottleneck decomposition: an approximation method for generalized Jackson networks. *Operations Research* 42(1):119–136.

[5] Daley DJ, Vere-Jones D (2008) *An Introduction to the Theory of Point Processes* (Oxford, U. K.: Springer), second edition.

[6] Green LV, Kolesar PJ, Whitt W (2007) Coping with time-varying demand when setting staffing requirements for a service system. *Production Oper. Management* 16:13–29.

[7] Harrison JM, Nguyen V (1990) The QNET method for two-moment analysis of open queueing networks. *Queueing Systems* 6(1):1–32.

[8] Jackson JR (1957) Networks of waiting lines. *Operations Research* 5(4):518–521.

[9] Kelly FP (2011) *Reversibility and Stochastic Networks* (Cambridge University Press), revised edition.

[10] Kuehn PJ (1979) Approximate analysis of general queuing networks by decomposition. *IEEE Transactions on Communications* 27(1):113–126.

[11] Segal M, Whitt W (1989) A Queueing Network Analyzer for Manufacturing. Bonatti M, ed., *Teletraffic Science for New Cost-Effective Systems, Networks and Services. ITC 12, Proceedings of the 12th International Teletraffic Congress*, 1146–1152 (Elsevier, North-Holland).

[12] Sigman K (1995) *Stationary Marked Point Processes: An Intuitive Approach* (New York: Chapman and Hall/CRC).

[13] Suresh S, Whitt W (1990) The heavy-traffic bottleneck phenomenon in open queueing networks. *Operations Research Letters* 9(6):355–362.

[14] Whitt W (1983) The queueing network analyzer. *Bell Laboratories Technical Journal* 62(9):2779–2815.

[15] Whitt W (1984) On approximations for queues, III: Mixtures of exponential distribtuoons. *AT&T Bell Laboratories Technical Journal* 63(1):163–175.

[16] Whitt W (1984) Open and closed models for networks of queues. *AT&T Bell labs Technical Journal* 63(9):1911–1979.

[17] Whitt W (2018) Time-varying queues. *Queueing Models and Service Management* 1(2):79–164.

[18] Whitt W, You W (2018) Heavy-traffic limit of the GI/GI/1 stationary departure process and its variance function. *Stochastic Systems* 8(2):143–165.

[19] Whitt W, You W (2018) Using robust queueing to expose the impact of dependence in single-server queues. *Operations Research* 66(1):184–199.

[20] Whitt W, You W (2019) The advantage of indices of dispersion in queueing approximations. *Operations Research Letters* 47(2):99–104.

[21] Whitt W, You W (2019) Time-varying robust queueing. *Operations Research* 67(6):1766–1782.

[22] Whitt W, You W (2020) Heavy-traffic limits for stationary network flows. *Queueing Systems* 95:53–68.

[23] Whitt W, You W (2022) A robust queueing network analyzer based on indices of dispersion. *Naval Research Logistics* 69(1):36–56.

[24] Whitt W, You W (2022) Supplement to "a robust queueing network analyzer based on indices of dispersion", http://www.columbia.edu/∼ww2040/allpapers.html.