Submitted to *INFORMS Journal on Computing*
manuscript (Please, provide the manuscript number!)

A Robust Queueing Network Analyzer Based on Indices of Dispersion

Ward Whitt
Department of Industrial Engineering and Operations Research, Columbia University, ww2040@columbia.edu,
Wei You
Department of Industrial Engineering and Operations Research, Columbia University, wy2225@columbia.edu,

We develop a robust queueing network analyzer (RQNA) algorithm to approximate the steady-state performance of a single-class open queueing network of single-server queues with Markovian routing, allowing non-renewal external arrival processes and non-exponential service-time distributions. Each flow is partially characterized by its rate and index of dispersion for counts (IDC, i.e., scaled variance-time function). A robust queueing approximation is used to approximate the mean steady-state number of customers and workload (remaining service time) at each queue, given the rate and IDC of the arrival process and the first two moments of the service time. The RQNA algorithm includes subroutines to calculate or estimate the IDC for each external flow, subroutines to solve systems of linear equations to calculate the rates and approximate the IDC's of the internal flows and a feedback elimination procedure. Effectiveness of the RQNA algorithm is supported by heavy-traffic limits and simulations.

Key words: queueing network analyzer, index of dispersion, robust queueing, heavy traffic, queueing performance approximations, generalized Jackson network
History: August 17, 2018

1. Introduction

In this paper, we develop performance approximations for the single-class stable $(G/GI/1)^K/M$ open queueing network (OQN), with Markovian routing (the $/M$) among K single-server queues with unlimited waiting space and the first-come first-served service discipline, where (i) the external arrival processes and sequences of service times at the K queues are mutually independent, (ii) each external arrival process is a stationary and ergodic point process (the G), partially specified by its rate and index of dispersion for counts (IDC), i.e., scaled variance-time function (assumed to be finite), and (iii) at each

station the service times are independent and identically distributed (i.i.d.) with general distributions (the GI) having finite second moments.

1.1. Beyond Markov OQNs: Dependence in Arrival Processes

Many complex systems can be modeled as OQNs. Thus, one of the most important developments in queueing theory has been the theory of Markovian OQN's initiated by Jackson [29], which showed that the steady-state vector for the number at each queue in the Markovian $(M/M/1)^K/M$ special case has a product-form (mutually independent distributions at the queues) with each distribution being geometric. This initial breakthrough was followed by vigorous research leading to an elaborate and useful theory, as can be seen from [30, 40]. Even though general OQNs do not have product-form steady-state distributions, some complex models do, e.g., [7, 25, 26].

However, applications in communication, manufacturing and service systems are often complicated by significant deviations from that tractable structure. In most manufacturing systems, an external arrival process is often far less variable than a Poisson process by design, while complicated processing operations, such as those involving batching, often produce complicated variability in the arrival processes at subsequent queues; e.g., see the example in §3 of [39].

In both manufacturing and communication systems, dependence among successive interarrival times and among successive interdeparture times at a queue often occur because there are multiple classes of customers with different characteristics, e.g., [4]. Multiple classes can even cause significant dependence (i) among interarrival times, (ii) among service times and (iii) between interarrival times and service times, which all can contribute to a major impact on performance, as shown by [15] and reviewed in §9.6 of [50].

In service systems, an external customer arrival process often is well modeled by a Poisson process, because it is generated by many separate people making decisions independently, at least approximately, but dependence may be induced by over-dispersion, e.g., see [34] and references there.

Even if external arrival processes can be regarded as Poisson processes, service-time distributions are often non-exponential. Internal arrival processes are necessarily departure processes from other queues or superpositions of such processes. If some of the service-time distributions are non-exponential, then these processes cannot be renewal processes because (i) a departure process from a $M/GI/1$ queue (or any $GI/GI/1$ queue) is necessarily

non-renewal if the service-time distribution is non-exponential and (ii) the superposition of independent renewal processes cannot be renewal unless all component processes are Poisson processes (in which case the superposition process is also Poisson); e.g., see [12, 13, 14]. Indeed, for departure processes this property is consistent with our heavy-traffic limit theorem for the stationary departure process from a $GI/GI/1$ queue in [51], which we review in §4.3.1. It shows that, asymptotically, the IDC of the stationary departure process is a convex combination of the IDCs of the arrival and service processes; see (22).

1.2. A New Decomposition Approximation

Motivated by the product-form property of the Markovian OQNs, researchers investigated decomposition approximations for non-Markov OQNs, in which the steady-state queue lengths are treated as approximately independent. For example, in [45] and [39] each queue is approximated by a $GI/GI/1$ queue, where the arrival process is approximated by a renewal process partially characterized by the mean and squared coefficient of variation (scv, variance divided by the square of the mean) of an interarrival time. Another decomposition method investigated by Kim [32, 33] approximates each queue by a $MMPP(2)/GI/1$ model, where the arrival process is a Markov-modulated Poisson process with two states. (We discuss connections to this approach in Remark 5.)

While the decomposition approximations do often perform well, it was recognized that dependence in the arrival processes of the internal flows can be a significant problem. The approximation for superposition processes used in [45] already attempts to address the dependence. Nevertheless, significant problems remained, as was dramatically illustrated by comparisons of QNA in [45] to model simulations in [42], [15] and [43], as discussed in [49]. A serious effort to address this problem was made by the introduction of the IDW in Fendick and Whitt [17] and showing its connection to the normalized workload (see §2), but that did not yield systematic approximations.

We advance that approach based on the IDW further by exploiting the new functional robust queueing (RQ) method in [54], which extends the first parametric RQ approximation in Bandi, Bertsimas and Youssef et al. [3]. (We review the RQ algorithm from [54] in §3.) In that way, we develop a new decomposition approximation, where the arrival process at each queue is partially specified by its rate and index of dispersion for counts (IDC), i.e., scaled variance-time function.

By replacing a single variability parameter for each arrival process (an scv in a renewal process approximation) by an entire function, we are better able to capture the essential stochastic properties of each arrival process. Because the IDC is a scaled variance function, the decomposition approximation here based on the rate and the IDC of each arrival process is similar in spirit to QNA in [45], but the IDC captures the dependence over time in the arrival process. Indeed, a stationary renewal process is fully characterized by its rate and IDC; see [53].

1.3. Heavy-Traffic Limits

The early decomposition approximation in [45] drew heavily on the central limit theorem (CLT) and heavy-traffic (HT) limit theorems. Approximations for a single queue follow from [27, 28]. With these tools, approximations for general point processes and arrival processes were developed in [44, 46]. Heavy-traffic approximation of queues with superposition arrival processes in [47] helped capture the impact of dependence in such queues; see §4.3 of [45].

Another approach is to apply heavy-traffic (HT) limit theorems for the entire network. Such HT limits were established for feedforward OQN's in Iglehart and Whitt [27, 28] and Harrison [20, 21] and then for general OQN's by Reiman [36], but the limiting multidimensional reflected Brownian motion (RBM) is not easy to work with. A more general case with strictly bottleneck and non-bottleneck queues and general initial conditions was studied in [8]. These general heavy-traffic results for OQN's have been exploited to develop approximations for OQNs, notably by the QNET algorithm in Harrison and Nguyen [22], the Individual Bottleneck Decomposition (IBD) algorithm in Reiman [37] and the Sequential Bottleneck Decomposition (SBD) algorithm in Dai, Nguyen and Reiman [10], which combines QNET with the decomposition method. These algorithms rely on the theoretical and numerical analysis of the stationary distribution of the multi-dimensional RBM, studied in [11, 23, 24].

What we do here is closely related to the IBD algorithm in [37] and the SBD algorithm in [10], but we focus on the stationary flows instead of the steady-state queue lengths. For the stationary flows, we rely on HT limits that we established in [51, 52]. In order to establish those HT limits for the stationary flows, we exploited the HT limits for the stationary vector queue-length process in Gamarnik and Zeevi [19] and Budhiraja and Lee [6]. In order to get a general Markov process for the system state process, they assumed

that the OQN is a Generalized Jackson Network (GJN), i.e., a $(GI/GI/1)^K/M$ OQN, with renewal external arrival processes. Thus, all our HT limits for the stationary flows require this stronger assumption as well, but our approximations are intended for more general models, allowing non-renewal external arrival processes, partially characterized by their rate and IDC.

1.4. Our Main Contributions

In this paper we apply the HT limits in in [51, 52] together with the RQ algorithm for the mean workload at a single $G/GI/1$ queue in [54] to create a full algorithm; i.e., we develop the RQNA based on IDCs to approximate the steady-state performance of the $(G/GI/1)^K/M$ OQN. We also draw on [55] to calculate the IDC of each external arrival process, based on model data or statistical estimation from arrival process sample paths. In this paper we also conduct simulation experiments to evaluate the effectiveness of the new RQNA and compare it to previous algorithms in [10, 22, 45].

Our RQNA algorithm has five components:

1. the robust queueing (RQ) approximation from [54] for the mean workload (remaining service time) at a $G/GI/1$ queue partially characterized by the arrival rate and IDC of the arrival process and the mean and scv of the service time (see §3.1);

2. formulas to compute associated approximations for the expected number of customers at the $G/GI/1$ queue and the expected sojourn time in the entire $(G/GI/1)^K/M$ OQN (see §3.2);

3. algorithms to determine the IDC of each external arrival process, either by numerical calculation from a model or estimation from data or simulation (see §2.3);

4. systems of linear equations to calculate the rate and the approximate IDC of each internal arrival process at the queues within the network (see §4 and §6).

5. a feedback elimination procedure for queues with high traffic intensity and high feedback probability to refine performance (see §5).

Most of our work here is devoted to the fourth and fifth components and evaluating its effectiveness by conducting extensive simulation experiments. Our experiments indicate that RQNA performs as well or better than previous algorithms.

1.5. Organization

The rest of the paper is organized as follows. In §2 we define the indices of dispersion and briefly review methods to determine them for the external arrival processes. We also discuss

the close connection between the index of dispersion for work and the mean steady-state workload. In §3 we review the RQ algorithm for a single $G/GI/1$ queue from [54] and discuss approximations for other steady-state performance measures. In §4 we develop a framework for approximating the IDC's of the flows. In §5 we discuss feedback elimination. In §6 we present the full RQNA algorithm. We also present a more elementary version for tree-structured OQNs in §6.1. In §7 we discuss numerical experiments.

We present additional material in the appendix. First, in §A we discuss additional numerical experiments. Second, in §B we present additional theoretical support, including for our algorithm to estimate the IDC from the sample path of an arrival process. Third, in §C we provide additional heavy-traffic support.

2. The Indices of Dispersion

In §2.1 we define the two continuous-time indices of dispersion that we consider: the IDC and the IDW. We present the useful decomposition of the IDW for the $G/GI/1$ model in (3). In §2.2 we review the close connection between the IDW and mean steady-state workload from [17]. (In §A.1 we present an important illustrative example.) In §2.3 we review the IDC of a stationary renewal process, which can be the basis for numerical algorithms, including numerical transform inversion, and we present our method for estimating the IDC from data.

2.1. Definitions of the IDC and IDW

Consider a general single-server queue with arrival process $A(t)$ and service times $\{V_i : i \geq 1\}$, where V_i is the service requirement of the i-th customer. Let

$$Y(t) \equiv \sum_{i=1}^{A(t)} V_i$$

denote the cumulative work input process. We define two indices of dispersion, associated with $A(t)$ and $Y(t)$.

The *index of dispersion for counts* (IDC) associated with the arrival process A is defined as in §4.5 of [9] by

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

and the *index of dispersion for work* (IDW) associated with the cumulative input process Y is defined as in (1) of [17] by

$$I_w(t) \equiv \frac{Var(Y(t))}{E[V_1]E[Y(t)]}, \quad t \geq 0. \tag{2}$$

Clearly, these indices of dispersion are just scaled versions of the associated variance-time function, but the scaling is important for understanding, because they expose the variability over time, independent of the scale. We prefer the indices of dispersion for the same reason we prefer the scv of a nonnegative random variable to the variance, because it exposes the variability independent of the mean.

REMARK 1. (time scaling convention) In [54] we defined the IDC and IDW in terms of rate-1 processes, so that the actual rate of the process had to be inserted as part of the time argument. In contrast, here as in [51] we let the underlying processes A and Y have any given rate, so no further scaling is needed. That changes the formulas for the IDC of a superposition process, e.g., compare (36) of [54] to (35) here. To illustrate the idea, consider $A(t)$ with rate-1 and $A_\lambda(t) = A(\lambda t)$ with rate-λ. Let $I_A(t)$ denote the IDC of $A(t)$, then we have $I_{A_\lambda}(t) \equiv \mathrm{Var}(A(\lambda t))/E[A(\lambda t)] = I_A(\lambda t)$. $\square$

Since we are interested in the steady-state performance of the OQN, we assume that the processes A and Y have stationary increments. Given that these processes have constant determined rates, much of the remaining behavior is determined by the variance-time function or index of dispersion. We are interested in the variance-time *function*, because it captures the dependence through the covariances; the processes (A, Y) have independent increments for the $M/GI/1$ model, but otherwise not.

When the service times V_i are i.i.d, independent of the arrival process $A(t)$, the conditional variance formula gives a useful decomposition of the IDW

$$I_w(t) = I_a(t) + c_s^2, \quad t \geq 0, \tag{3}$$

where c_s^2 is the scv of the service-time distribution. However, even in the $(G/GI/1)^K$ OQN model we consider here, (3) need not to hold. This happens when there is customer feedback, which makes the service times necessarily correlated with the arrival process at the feedback queue. We address this issue for a large class of OQNs in §5 by eliminating near-immediate feedback at some queues.

The reference case is a Poisson arrival process, for which $I_a(t) = 1$, $t \geq 0$. However, for other $GI/GI/1$ models, including $D/GI/1$, the IDC is more complicated. Even the IDC for a determinsitic D arrival process is complicated, because the IDC is for the stationary version of the arrival process, which lets the initial point be uniformly distributed over the constant interarrival time.

2.2. The IDW and the Mean Steady-State Workload

The IDC and IDW are important because of their close connection to the mean steady-state workload $E[Z_\rho]$. The workload process $Z(t)$ is the time needed for the station to serve all customers in the system at time t. Under regularity conditions, the workload $Z(t)$ converges to the steady-state workload Z_ρ as t increases to infinity. In [17] it was shown that the IDW I_w is intimately related to a scaled mean workload $c_Z^2(\rho)$, defined by comparing to what it would be in the associated $M/D/1$ model; i.e.,

$$c_Z^2(\rho) \equiv \frac{E[Z_\rho]}{E[Z_\rho; M/D/1]} = \frac{2(1-\rho)E[Z_\rho]}{E[V_1]\rho}. \tag{4}$$

The normalization in (4) exposes the impact of variability separately from the traffic intensity. Under regularity conditions, the following finite positive limits exist and are equal:

$$\begin{aligned} \lim_{t\to\infty}\{I_w(t)\} &\equiv I_w(\infty) = c_Z^2(1) \equiv \lim_{\rho\to 1}\{c_Z^2(\rho)\}, \quad \text{and} \\ \lim_{t\to 0}\{I_w(t)\} &\equiv I_w(0) = c_Z^2(0) \equiv \lim_{\rho\to 0}\{c_Z^2(\rho)\}; \end{aligned} \tag{5}$$

see [17] and §EC.5.5 of [54].

The reference case is the classical $M/GI/1$ queue, for which we have

$$c_Z^2(\rho) = 1 + c_s^2 = I_w(t) \quad \text{for all} \quad \rho,\, t, \quad 0<\rho<1,\, t\ge 0. \tag{6}$$

In great generality, we have

$$c_Z^2(0) = 1 + c_s^2 = I_w(0) \quad \text{and} \quad c_Z^2(1) = c_A^2 + c_s^2 = I_w(\infty), \tag{7}$$

where c_A^2 is the asymptotic variability parameter, i.e., the normalization constant in the CLT for the arrival process, which coincides with the scv c_a^2 of an interarrival time for a renewal process.

Clearly, when c_A^2 is not nearly 1, $c_Z^2(\rho)$ varies significantly as a function of ρ, so that the impact of the variability in the arrival process upon the queue performance clearly depends on the traffic intensity. This important insight from [17] is the starting point for our analysis. In well-behaved models, $c_Z^2(\rho)$ as a function of ρ and $I_w(t)$ as a function of t tend to change smoothly and monotonically between those extremes, but OQNs can produce more complex behavior when both the traffic intensities at the queues and the levels of variability in the arrival and service processes at different queues vary. We illustrate in §A.1.

2.3. Estimating and Calculating the IDC

For applications, it is significant that the IDC $I_a(t)$ can readily be estimated from data from system measurements or simulation and calculated in a wide class of stochastic models. The time-dependent variance functions can be estimated from the time-dependent first and second moment functions, as discussed in §III.B of [16]. Calculation depends on the specific model structure.

2.3.1. The IDC's for Renewal Processes. For renewal processes, the variance $\mathrm{Var}(A(t))$ and thus the IDC $I_a(t)$ can either be calculated directly or can be characterized via their Laplace transforms and thus calculated by inverting those transforms and approximated by performing asymptotic analysis. Because we are interested in the steady-state behavior of the OQN, we are primarily interested in the equilibrium renewal process, as in §3.5 of [38]; see Remark 2.

In turns out that the variance of the equilibrium arrival renewal process $V(t) \equiv \mathrm{Var}(A(t))$ can be expressed in terms of the renewal function $m(t) \equiv E[A_0(t)]$, where A_0 is the corresponding ordinary renewal process. For a function f, let $\hat{f}$ denote the Laplace transform of f, defined by

$$\hat{f}(s) \equiv \mathcal{L}(f)(s) \equiv \int_0^\infty e^{-st} f(t) dt.$$

The following formula is taken from §2 of [51]

$$\hat{V}(s) = \frac{\lambda}{s^2} + \frac{2\lambda}{s}\hat{m}(s) - \frac{2\lambda^2}{s^3} = \frac{\lambda}{s^2} + \frac{2\lambda}{s}\frac{\hat{g}(s)}{s\,(1-\hat{g}(s))} - \frac{2\lambda^2}{s^3}, \tag{8}$$

where g is the density function of the interarrival-time dsitribution. The variance function can then be obtained numerically, which is discussed in §13 of [1]. The hyperexponential (H_2) and Erlang (E_2) special cases are described in §III.G of [17].

It is also possible to carry out similar analyses for much more complicated arrival processes. [35] applies matrix-analytic methods to give explicit representations of the variance $Var(A(t))$ for the versatile Markovian point process or Neuts process; see §5.4, especially Theorem 5.4.1. Explicit formulas for the Markov modulated Poisson process (MMPP) are given on pp. 287-289.

2.3.2. Numerical Estimation of the IDC from Data. Now we present an algorithm from [55] to numerically estimate the variance $V(t) = \mathrm{Var}(A(t))$ from a given realized sample path of the stationary point process $A(t)$. The main idea is based on Section 5.4 (iii) of [9].

Our goal is to estimate $V(t)$ for $0 < t < t_0$ using a realization of $A(t)$ for $0 < t < T$. The simplest way is to apply crude Monte Carlo method to estimate $V(t)$ for a fixed t and repeat over a finite grid of t's. This method divide the sample path of $A(t)$ into non-overlapping intervals of length t and count the number of arrivals in each interval. The variance is then estimated by the sample variance of the counts. This method is simple to implement but can be slow to converge.

To accelerate the crude Monte Carlo method, we apply three techniques: (i) we use overlapping intervals instead of non-overlapping ones, which introduces bias but reduces sample variance; (ii) we calculate $V(t)$ only over a finite grid equally spaced in the logarithm scale instead of the linear scale (further discussed in §A.1); and (iii) we re-use the tallied number of events for shorter intervals to calculate the total number of events for longer interval, which avoids repetitive counting. We discuss the three techniques in turn:

To use overlapping intervals, consider first $k = T/t$ number of non-overlapping intervals, each with length t. Now, we further divide each intervals of length t in to r intervals of the same length $\tau = t/r$. Hence we have rk number of non-overlapping intervals of length τ. Let n_i be the number of events fall in the i-th interval, consider

$$U_i \equiv A(I_i) \equiv A[i\tau, (i+r)\tau) = n_i + n_{i+1} + \cdots + n_{i+r-1}, \quad i = 0, 1, \ldots, rk - r + 1.$$

We estimate $V(t)$ with the sample variance $\bar{V}_l$ of $\{U_i\}_{i=1}^{l}$, where $l = rk - r + 1$. This estimator is in general biased but can achieve lower variance compared with the one obtained with crude Monte Carlo method. In [55] we show that this estimator of $V(t)$ is asymptotically consistent under mild conditions that $V(t)$ is differentiable with derivative $\dot{V}(t)$ having finite positive limits as $t \to \infty$. We review this theorem and proof in §B.

For the third technique, we now present a algorithm to simultaneously estimate $V(2^i\tau)$ for some $\tau > 0$ and $i = 0, 1, \ldots, l$. Let $\{I_i\}$ be the collection of non-overlapping intervals of length τ that covers $[0, T]$. Let $n_i = A(I_i)$ be the number of events on interval I_i. Then we have the following table from [9].

	time horizon t			
sample	τ	2τ	$2^2\tau$	$\cdots$
1	n_1	n_1+n_2	$n_1+n_2+n_3+n_4$	$\cdots$
2	n_2	n_2+n_3	$n_3+n_4+n_5+n_6$	$\cdots$
3	n_3	n_3+n_4	$n_5+n_6+n_7+n_8$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

We find the estimation of $V(2^i\tau)$ by calculating the sample variance of the corresponding column.

Now that we have a efficient algorithm to estimate $V(2^i\tau)$ for fixed τ, we have obtained the estimations of a grid equally spaced in logarithm scale. To obtain estimations for finer grids we shift the crude grid by picking several $\tau \le \tau_j \le 2\tau$ equally spaced in log scale and, for each j, simultaneously estimate $V(2^i\tau_j)$ for all i.

3. The Robust Queueing Algorithm

In this section, we review the RQ algorithm for single-server queues and discuss approximations for other performance measures obtained as a result.

3.1. The RQ Workload Approximation for a $G/GI/1$ Queue

In [54] RQ algorithms were developed for the mean steady-state values of both the discrete-time waiting time W and the continuous-time workload Z. We will be applying the RQ algorithm for the continuous-time workload Z. The arrival process A is assumed to be a stationary and ergodic point process, partially characterized by the arrival rate λ, and the IDC I_a defined in (1). For stationary point process, we always have $E[A(t)] = \lambda t$, see §2.7 of [41]. We further assume that the service time distribution is partially characterized by its rate μ and *squared coefficient of variation* (scv) c_s^2.

Let $Z \equiv Z(\lambda, I_a, \mu, c_s^2)$ be the steady-state workload in the $G/GI/1$ model partially characterized by the four-tuple $(\lambda, I_a, \mu, c_s^2)$, assuming that $\rho \equiv \lambda/\mu < 1$ to have model stability. The RQ algorithm provides approximation for $E[Z]$ with $(\lambda, I_a, \mu, c_s^2)$ as input data.

To obtain the RQ algorithm, we start with a reverse-time construction of the workload process as in §3 of [54]. Define the net-input process $N(t)$ as

$$N(t) \equiv Y(t) - t, \quad t \ge 0, \tag{9}$$

then the workload at time t, starting empty at time 0, is the reflection map Ψ applied to N, i.e.,

$$Z = \Psi(N)(t) \equiv N(t) - \inf_{0\le s\le t}\{N(s)\}, \quad t \ge 0. \tag{10}$$

As in §6.3 of [41], we use a reverse-time construction to represent the workload. 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

$$Z(t) \equiv \sup_{0\le s\le t}\{N(s)\}, \quad t \ge 0, \tag{11}$$

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 reverse-time process $Z(t)$ defined in (11) is nondecreasing in t and hence necessarily converges to a limit Z, For the stable stationary $G/GI/1$ model, Z corresponds to the steady-state workload and satisfies $P(Z < \infty) = 1$; see §6.3 of [41].

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.

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

In our specific settings, we have the following uncertainty set motivated from CLT

$$\begin{aligned}\mathcal{U}_\rho &\equiv \left\{\tilde{N}_\rho : \mathbb{R}^+ \to \mathbb{R} : \tilde{N}_\rho(s) \le E[N_\rho(s)] + \sqrt{2\text{Var}(N_\rho(s))},\, s \ge 0\right\}, \\ &= \left\{\tilde{N}_\rho : \mathbb{R}^+ \to \mathbb{R} : \tilde{N}_\rho(s) \le -(1-\rho)s + \sqrt{2\rho s(I_a(s) + c_s^2)/\mu},\, s \ge 0\right\},\end{aligned} \tag{12}$$

where $N_\rho(t)$ is the net input process associated with the stochastic queue with traffic intensity ρ, so

$$\begin{aligned}E[N_\rho(t)] &= E[Y_\rho(t) - t] = \rho t - t, \\ \text{Var}(N_\rho(t)) &= \text{Var}(Y_\rho(t)) = I_w(t)E[V_1]E[Y_\rho(t)] = (I_a(t) + c_s^2)\rho t/\mu.\end{aligned}$$

As in §4 of [54], and the RQ approximation based on this partial model characterization is

$$E[Z_\rho] \equiv E[Z(\lambda, I_a, \mu, c_s^2)] \approx Z_\rho^* \equiv \sup_{\tilde{N}_\rho \in \mathcal{U}_\rho} \sup_{x \ge 0} \{\tilde{N}(x)\}$$
$$= \sup_{x \ge 0} \{-(1-\rho)x + \sqrt{2\rho x (I_a(x) + c_s^2)/\mu}\}, \tag{13}$$

where the second line follows Theorem 2 of [54].

The approximation (13) is a variant of (28) in [54], assuming that we set the parameter $b_f = \sqrt{2}$, which makes the approximation asymptotically correct for the $GI/GI/1$ model in both the heavy-traffic and light-traffic limits; see Theorem 5 of [54]. We focus on the slightly more general form in (13) in terms of a general service rate because we can no longer assume unit-rate service across all stations when we move beyond single-server queues to consider a queueing network. The expression here reduces to (27) of [54] by appropriately choosing time units, i.e., by change of variable $s = \mu x$. Let $I_{a,\lambda}$ denote the IDC of the rate-λ arrival process, assuming that arrival processes with different rates are related by $A_{\lambda_1}(t/\lambda_1) = A_{\lambda_2}(t/\lambda_2)$, then

$$Z^*(\lambda, I_a, \mu, c_s^2) = \sup_{x \ge 0} \{-(1-\rho)x + \sqrt{2\rho x (I_{a,\lambda}(x) + c_s^2)/\mu}\}$$
$$= \sup_{s \ge 0} \{-(1-\rho)s/\mu + \sqrt{2\rho s (I_{a,\lambda}(s/\mu) + c_s^2)/\mu^2}\}$$
$$= \sup_{s \ge 0} \{-(1-\rho)s + \sqrt{2\rho s (I_{a,1}(\rho s) + c_s^2)}\}/\mu \tag{14}$$
$$= \sup_{s \ge 0} \{-(1-\rho)s + \sqrt{2\rho s (I_{a,\rho}(s) + c_s^2)}\}/\mu$$
$$\stackrel{\mathrm{d}}{=} Z(\rho, I_{a,\rho}, 1, c_s^2)/\mu. \tag{15}$$

The expression in (14) is exactly (27) of [54]. From (15), we see that the RQ solution in (13) is linear in the mean service time $\tau \equiv 1/\mu$ as expected.

Notice that the approximation in (13) is directly a supremum of a real-valued function, and so can be computed quite easily for any given 4-tuple $(\lambda, I_a, \mu, c_s^2)$. Indeed, the deterministic function before we take the supremum is revealing to show how the performance depends on the parameters.

Remark 2. (continuous-time stationarity) We emphasize that, in the RQ formulation, it is essential to use the continuous-time stationary version of the IDC in (1) and the IDW

in (2), instead of their discrete-time Palm stationary versions; see [41] for a comprehensive discussion. The continuous-time stationary IDC we use here yields asymptotically correct light-traffic limit, whereas the Palm stationary IDC does not. See §5.2 of [54] for more discussion. □

Theorem 5 in [54] states that the RQ algorithm is asymptotically exact in both light-traffic and heavy-traffic limits. Through extensive simulation experiments, it has been found that the mean steady-state workload $E[Z]$ can be well approximated by the IDW-based RQ algorithm, see §A.2 for a numerical example.

In the i.i.d. service time setting, the IDW reduces to the IDC plus the service scv as in (3). Thus, the main challenge for the RQNA algorithm for OQNs is developing a successful approximation for the IDC of the internal arrival process at each queue, which we discuss in §4 and §6.

3.2. Other Steady-State Performance Measures

We develop approximations for other steady-state performance measures by applying exact relations for the $G/GI/1$ queue that follow from Little's law $L = \lambda W$ and its generalization $H = \lambda G$; e.g., see [48] and Chapter X of [2] for the $GI/GI/1$ special case. Let W, Q and X be the steady-state waiting time, queue length and the number in system (including the one in service, if any, at an arbitrary time). By Little's law,

$$\begin{aligned} E[Q] &= \lambda E[W] = \rho E[W] \quad \text{and} \\ E[X] &= E[Q] + \rho = \rho(E[W] + 1). \end{aligned} \tag{16}$$

By Brumelle's formula [5] or $H = \lambda G$, (6.20) of [48],

$$E[Z] = \rho E[W] + \rho \frac{E[V^2]}{2\mu} = \rho E[W] + \rho \frac{(c_s^2 + 1)}{2\mu}, \tag{17}$$

Hence, given an approximation Z^* for $E[Z]$, we can use the approximations

$$\begin{aligned} E[W] &\approx \max\{0, Z^*/\rho - (c_s^2 + 1)/2\mu\} \quad \text{and} \\ E[Q] &\approx \lambda E[W]. \end{aligned} \tag{18}$$

Remark 3. (network performance measures) So far we only have discussed the performance measures for a single station. The total network performance measures, on the other hand, can also be derived. For example, the expected value of the total sojourn time

$T_i^{\rm tot}$, i.e. the time needed to flow through the queueing network for a customer that enters the system from station i, is easily estimated from the obtained mean waiting time at each station. Assuming Markov routing with routing matrix P, a standard argument from discrete time Markov chain theory gives the mean total number of visits $\xi_{i,j}$ to station j by a customer entering the system at station i as

$$\xi_{i,j} = \left((I-P)^{-1}\right)_{i,j},$$

where $(I-P)^{-1}$ is the so-called fundamental matrix of a absorbing Markov chain. Hence, the mean steady-state total sojourn time $E[T_i^{\rm tot}]$ is approximated by

$$E[T_i^{\rm tot}] \approx \sum_{j=1}^{K} \xi_{i,j}(W_j + 1/\mu_j). \tag{19}$$

In real world applications, customers often experiences non-Markovian routing, where routes are customer-dependent. For ways to represent those scenarios and convert them (approximately) to the current framework, see §2.3 and §6 of [45]. □

4. Approximating the IDCs of the Internal Flows

In this section we develop a framework for approximating the IDCs of the internal flows in the OQN. These flows are assumed to be continuous-time stationary point processes. As a basis for the heavy-traffic limits for the stationary flows of an $(GI/GI/1)^K/M$ OQN in [52], we showed that these stationary flows exist and are well defined in that setting.

We start in §4.1 by reviewing the OQN model and the required model data for the RQNA algorithm. We review the standard traffic rate equations in §4.2. We develop the new traffic variability equations in §4.3. As in other decomposition methods, three network operations are essential: the departure operation (flow through a queue), the splitting operation and the superposition operation.

4.1. The OQN Model Data

4.1.1. Model Assumptions. Each queue has a single server, unlimited waiting space and provides service in order of arrival. For each queue (node or station) i, $1 \le i \le K$, we have an external arrival process $A_{0,i} \equiv \{A_{0,i}(t) : t \ge 0\}$ and a sequence of i.i.d. service times $\{V_i^l; l \ge 1\}$. We assume that all these external arrival processes and service processes are mutually independent.

Each external arrival process $A_{0,i}$ is assumed to be a simple (no batches) stationary and ergodic point process (having stationary increments) with $E[A_{0,i}^2(t)] < \infty$ for all t. We assume that this external arrival process $A_{0,i}$ is partially specified by its rate $\lambda_{0,i}$ and its IDC $I_{a,i,0} \equiv \{I_{a,i,0}(t) : 0 \le t \le \infty\}$, as defined in (1). We assume that the IDC $I_{a,i,0}$ is continuous with finite limits at 0 and $+\infty$.

We assume that the service times V_i^l are distributed as V_i with cdf G_i, finite mean $1/\mu_i$ and scv $c_{s,i}^2$. Let the associated service renewal counting process be $S_i \equiv \{S_i(t) : t \ge 0\}$, where

$$S_i(t) = \max\left\{n \le 0 : \sum_{l=1}^{n} V_i^l \le t\right\}, \quad t \ge 0.$$

Let $I_{s,i} \equiv \{I_{s,i}(t); 0 \le t \le \infty\}$ be the IDC of the associated stationary renewal process. We assume that the IDC $I_{s,i}$ is continuous with limits at 0 and $+\infty$. We necessarily have $I_{s,i}(\infty) = c_{s,i}^2$.

We assume that departures are routed from node to node and out of the network by Markovian routing, which is independent of the arrival and service processes. We assume that each arrival eventually leaves w.p.1. Let $p_{i,j}$ denote the probability that a departure from node i is routed to node j Let $P \equiv \{p_{i,j} : 1 \le i, j \le K\}$ be the (substochastic) routing matrix. Furthermore, let $p_{i,0} \equiv 1 - \sum_j p_{i,j}$ denote the probability that a customer departs the system after completing service at from node i.

4.1.2. Model Data. For our RQNA algorithm, we assume that we are given the parameter 5-tuple $(\lambda_i, I_{a,i}, \mu_i, c_{s,i}^2, I_{s,i})$ for each queue i and the routing matrix P. For the GI service process, it suffices to specify the service-time cdf G_i; then $1/\mu_i$ is its mean and $c_{s,i}^2$ its scv, while $I_{s,i}$ can be computed from G_i as indicated in §2.3.1.

As opposed to the QNA algorithm in [45], the RQNA algorithm requires the IDC's of the external arrival processes and the service processes, in addition to the means and scv's.

If we are only given the first two moments, then we can fit a convenient cdf G_i to these parameters and use the corresponding as indicated in §3 of [44]. In particular, we would use (i) an exponential cdf when $c_{s,i}^2 = 1$, (ii) a deterministic distribution when $c_{s,i}^2 = 0$, (iii) a hyperexponential (H_2, mixture of two exponential distributions) cdf with balanced means as in (3.7) of [44] when $c_{s,i}^2 > 1$, (iv) an Erlang (E_k, sum of exponential random variables) distribution when $c_{s,i}^2 = 1/k$, and (v) a shifted-exponential distribution otherwise when $0 < c_{s,i}^2 < 1$; see (3.12) of [44].

If we are only give sample data of the processes, then we can apply the numerical algorithm in §2.3 to estimate the rate and IDC of the process.

4.2. The Traffic Rate Equations and Traffic Intensities

We use the same traffic rate equations as in Markovian $(M/M/1)^K/M$ networks to determine the internal (net) arrival rate at each queue. Let $\lambda_0 \equiv (\lambda_{0,1}, \ldots, \lambda_{0,K})$ be the external arrival rate vector; so that $\lambda_{i,j} \equiv \lambda_i p_{i,j}$ is the rate of the internal arrival stream from i to j, denoted by $A_{i,j}$. Let $\lambda \equiv (\lambda_1, \ldots, \lambda_K)$ denote the total arrival rate vector, then the (exact) traffic-rate equations are

$$\lambda_i = \lambda_{0,i} + \sum_{j=1}^{K} \lambda_{j,i} = \lambda_{0,i} + \sum_{i=1}^{K} \lambda_j p_{j,i}, \quad 1 \le i \le K, \tag{20}$$

or in matrix form

$$(I - P')\lambda = \lambda_0,$$

where I denotes the identity matrix. We assume that $I - P'$ is invertible; i.e., we assume that all customers eventually leave the system. The condition for the invertibility of $I - P'$ to hold is well known, e.g. in Theorem 3.2.1 of [31]. Hence, the vector of internal arrival rates is given by

$$\lambda = (I - P')^{-1}\lambda_0. \tag{21}$$

Then the traffic intensity at queue i is defined as usual by $\rho_i \equiv \lambda_i/\mu_i$. We assume that $\rho_i < 1$ for all i to ensure that the OQN is stable.

4.3. The Traffic Variability Equations

Paralleling §4 of [45], we develop traffic variability equations to approximate the IDC of the internal arrival process to each queue. We develop equations for each of the basic network operations: (i) departure (flow through a queue), (ii) splitting and (iii) superposition. However, we go significantly beyond [45] by having a variability function, the IDC $I_{a,i} \equiv \{I_{a,i}(t); t \ge 0\}$ instead of the single variability parameter $c^2_{a,i}$. Moreover, in treating splitting and superposition, we obtain more general formulas by relaxing customary independence assumptions (which do not hold in general OQN's allowing customer feedback).

4.3.1. The Departure Operation. For queue i, given the parameter 5-tuple $(\lambda_i, I_{a,i}, \mu_i, c^2_{s,i}, I_{s,i})$, its traffic intensity is $\rho_i = \lambda_i/\mu_i$. The departure process necessarily has the same rate λ_i. We approximate the IDC $I_{d,i}$ by a convex combination of the arrival IDC $I_{a,i}$ and the service IDC $I_{s,i}$ using a weight function that depends on both time t and the traffic intensity ρ_i.

In forming this convex combination, recall our scaling convention in Remark 1, indicating that we scale any IDC by giving it the rate of the stationary point process under consideration. Below we will assume that both processes are given the same rate λ as given for the arrival process. Given that the given stationary service process has rate μ, we convert it to rate λ by considering $I_s(\rho t)$. (This change in notation should have been made in [51], because there too the rate was taken to be λ for both I_a and I_s. Of course, this change has no impact on the heavy-traffic limits.)

In particular, we propose the approximation

$$I_{d,i}(t) \approx w_{\rho_i}(t) I_{a,i}(t) + (1 - w_{\rho_i}(t)) I_{s,i}(\rho t), \quad t \geq 0, \tag{22}$$

where the weight function w_{ρ_i} is expressed in terms of a single weight function w^* by

$$w_{\rho_i}(t) \equiv w^*((1-\rho_i)^2 \lambda_i t / h(\rho_i) c^2_{x,i}), \quad t \geq 0, \tag{23}$$

where $c^2_{x,i} \equiv c^2_{a,i} + c^2_{s,i}$ and $c^2_{a,i} = I_{a,i}(\infty)$, $h(\rho)$ is an increasing continuous *tuning function* of the traffic intensity ρ with $h(0) \equiv 0$ and $h(1) \equiv 1$, and the *canonical weight function* w^* is

$$w^*(t) \equiv 1 - \frac{1 - c^*(t)}{2t}, \quad t \geq 0. \tag{24}$$

with $c^*(t)$ being the correlation function of the stationary version of canonical one-dimensional RBM.

To elaborate on the role of RBM in the canonical weight function, let R be canonical one-dimensional RBM (having drift -1, diffusion coefficient 1) and let R_e be the stationary version, which has the exponential marginal distribution for each t with mean $1/2$. Let $c^*(t)$ be the correlation function of R_e, defined by

$$\begin{aligned} c^*(t) &\equiv \frac{E[R_e(0)R_e(t)] - E[R_e(0)]E[R_e(t)]}{\mathrm{Var}(R_e(0))} = 1 - \frac{E[R(t)^2 | R(0) = 0]}{E[R(\infty)^2]} \\ &= 2(1 - 2t - t^2)\Phi^c(\sqrt{t}) + 2\sqrt{t}\phi(\sqrt{t})(1+t), \quad t \geq 0, \end{aligned} \tag{25}$$

where Φ is the cdf of standard normal distribution and ϕ is the associated density function. The weight function w^* is a monotonically increasing function with $w^*(0) = 0$ and $w^*(\infty) = 1$. For more discussion of the correlation function and the weight function, see §3 of [51] and the references there.

The approximation in (22), for any tuning function $h(\rho)$, is supported by heavy-traffic limits for the stationary departure processes, where we push the queue of interest (denoted by h) to the heavy-traffic limit while keeping other stations strictly under-saturated. Such HT limits are established in Theorems 5.1-5.3 and Corollary 6.1 of [51] for the $GI/GI/1$ model and extended to cover the $(GI/GI/1)^K/M)$ OQN model in Corollary 4.2 of [52]. Under regularity conditions (uniform integrability, for which it suffices to have uniformly bounded finite fourth moments of the interarrival time and service time), the approximation in (22) is asymptotically correct as $\rho_h \to 1$. See [52] for technical support of (22) for general OQNs.

It remains to specify the tuning function h used in the ρ_i-dependent weight $w_{\rho_i}(t)$ in (23). It is chosen to improve the quality of approximations at queues with light-to-moderate traffic intensities. In specific, we propose

$$h(\rho) \equiv \rho^2, \quad 0 \le \rho \le 1. \tag{26}$$

This specific choice of the tuning function is motivated from Remark 5.2 of [51], where we replace γ by γ_ρ in the pre-limit weight function and recall that the usual case of $\mu_\rho = \lambda/\rho$ corresponds to $\gamma_\rho = 1/\rho$. The tuning function in (26) also corresponds to (33) of in [54]. We remark that h_ρ can be used as a tuning parameter to improve the quality of approximations. We will illustrate in our numerical examples.

Remark 4. (parallel to QNA) The convex combination in the approximation (22) is reminiscent of the convex combination for variability parameters in (38) of [45], which is a stationary-interval approximation, as discussed in [44, 45, 46]. In (38) of [45]

$$c_{d,i} \approx (1 - \rho_i^2)c_{a,i}^2 + \rho_i^2 c_{s,i}^2. \tag{27}$$

Clearly approximation (27) puts more weight on the service scv $c_{s,i}^2$ as ρ_i increases, approaching the values 0 and 1 in the extremes. This makes sense intuitively, because the queue should be busy most of the time as ρ_i increases toward 1. Thus departure times tend

to be minor variations of service times. In contrast, if ρ_i is very small, then the queue acts only as a minor perturbation of the arrival process.

Similar behavior can be seen in approximation (22). In particular, since the weight function w^* in (24) is a monotonically increasing function with $w^*(0) = 0$ and $w^*(\infty) = 1$ and since $w_{\rho_i}(t) \equiv w^*((1-\rho_i)^2\lambda_i t/h(\rho_i)c_{x,i}^2)$, we see that for each t, $w_{\rho_i}(t)$ also places less weight on $I_{a,i}(t)$ and more weight on $I_{s,i}(t)$ as ρ_i increases.

However, (22) also reveals a more subtle interaction between t and ρ. In complex stationary point processes, such as the departure processes in §A.1, the variability in a point process is not the same at all time scales. Moreover, the impact of the variability in an arrival process at a later queue depends on the traffic intensity of that later queue. The heavy-traffic limits expose the importance of the time scaling by $(1-\rho)^{-2}$. Since, $w_{\rho_i}(t) \equiv w^*((1-\rho_i)^2\lambda_i t/h(\rho_i)c_{x,i}^2)$ by (23), we see that $w_{\rho_i}((1-\rho)^{-2}t) = w^*(\lambda_i t/h(\rho_i)c_{x,i}^2)$, which tends to be nearly independent of ρ_i for larger values of ρ_i. In other words, the weight is approximately constant in scaled time. □

4.3.2. The Splitting Operation. To treat splitting, we write the split process $A_{i,j}$ as a random sum. To represent general routing, let $\theta_i^l \in \{0,1\}^K$ indicates the routing vector of the l-th departure from queue i. So at most one component of θ_i^l is 1 and the j-th component $\theta_{i,j}^l = 1$ indicates that the the l-th departure from the i-th station is routed to the j-th station. Then observe that

$$A_{i,j}(t) = \sum_{l=1}^{D_i(t)} \theta_{i,j}^l, \quad t \geq 0. \tag{28}$$

We apply the conditional-variance formula to write the variance $V_{a,i,j}(t) \equiv \mathrm{Var}(A_{i,j}(t))$ as

$$V_{a,i,j}(t) = E[Var(A_{i,j}(t)|D_i(t))] + Var(E[A_{i,j}(t)|D_i(t)]). \tag{29}$$

With the Markovian routing we have assumed, the routing decisions at each queue at each time are i.i.d. and independent of the history of the network. As a consequence, for feed-forward queueing networks, we can deduce that the collection of all routing decisions made at queue i up to time t is independent of $D_i(t)$, but not more generally. With customer feedback, there is a complicated dependence.

For the case in which independence holds, we can apply (29) to express $V_{a_{i,j}}(t)$ in terms of the variance of the departure process, $V_{d,i}(t) \equiv \mathrm{Var}(D_i(t))$; in particular,

$$V_{a,i,j}(t) = p_{i,j}^2 V_{d,i}(t) + p_{i,j}(1-p_{i,j})\lambda_i t, \tag{30}$$

or, equivalently, since $E[D_i(t)] = \lambda_i t$ and $E[A_{i,j}(t)] = p_{i,j}\lambda_i t = p_{i,j}E[D_i(t)]$,

$$I_{a,i,j}(t) = p_{i,j}I_{d,i}(t) + (1 - p_{i,j}). \tag{31}$$

The formula (31) is an initial approximation, which parallels the approximation used for splitting in (40) of [45], i.e., $c^2_{a,i,j} = p_{i,j}c^2_{d,i} + (1 - p_{i,j})$.

However, we develop a more general formula to improve the approximation in general OQNs. For that purpose, we apply the heavy-traffic FCLT for split processes in §9.5 of [50]; we give the detailed derivation in §C.2. Based on that heavy-traffic analysis, we propose the splitting IDC equation as

$$I_{a,i,j}(t) = p_{i,j}I_{d,i}(t) + (1 - p_{i,j}) + \alpha_{i,j}(t), \tag{32}$$

so that the additional correction term $\alpha_{i,j}$ is defined as

$$\alpha_{i,j}(t) \equiv I_{a,i,j}(t) - p_{i,j}I_{d,i}(t) - (1 - p_{i,j}). \tag{33}$$

In §C.2 we develop a heavy-traffic approximation for $\alpha_{i,j}(t)$ that is asymptotically correct in that heavy-traffic limit, but in general it is hard to evaluate. So we also obtain a more concrete approximation by considering a heavy-traffic limit in which only queue i enters heavy traffic in the limit. That leads to the correction term

$$\begin{aligned}\alpha_{i,j,\rho_i}(t) &\approx 2\xi_{i,j}p_{i,j}(1 - p_{i,j})w_{\rho_i}(t) \\ &= 2\xi_{i,j}p_{i,j}(1 - p_{i,j})w^*((1 - \rho_i)^{-2}\lambda_i t/(h(\rho_i)c^2_{x,i})), \quad t \geq 0,\end{aligned} \tag{34}$$

where $w_{\rho_i}(t)$ is the weight function for the departure IDC in (23), $c^2_{x,i}$, $c^2_{a,i}$ and $c^2_{s,i}$ are also as in (23), while $\xi_{i,j}$ is the $(i,j)^{\text{th}}$ entry of the matrix $(I - P')^{-1}$.

4.3.3. The Superposition Operation. In this section, we investigate the effect of the superposition operation on the IDC's. To start, consider the case in which the individual streams are mutually independent. In this case, we have

$$V_{a,i}(t) \equiv \text{Var}(A_i(t)) = \text{Var}\left(\sum_{j=0}^{K} A_{j,i}(t)\right) = \sum_{j=0}^{K} \text{Var}(A_{j,i}(t)),$$

so that

$$I_{a,i}(t) = \sum_{j=0}^{K} (\lambda_{j,i}/\lambda_i)I_{a,j,i}(t), \tag{35}$$

where $I_{a,j,i}(t) \equiv \mathrm{Var}(A_{j,i}(t))/E[A_{j,i}(t)]$. Recall that (35) differs from (36) of [54] because we are not assuming rate-1 processes in our definitons of the IDC; see Remark 1.

While (35) is exact when the streams are independent, it is not exact in general cases. Even for feed-forward networks, we may have a stream that splits and then recombines later, which introduces dependence.

For dependent streams, the variance of the superposition total arrival process at queue i can be written as

$$V_{a,i}(t) \equiv \mathrm{Var}\left(\sum_{j=0}^{K} A_{j,i}(t)\right) = \sum_{j=0}^{K} \mathrm{Var}\left(A_{j,i}(t)\right) + \beta_i(t) E[A_i(t)]$$

where $A_{0,i}$ denotes the external arrival process at station i,

$$\beta_i(t) \equiv \sum_{j \neq k} \beta_{j,i;k,i}(t), \quad \text{and} \quad \beta_{j,i;k,i}(t) \equiv \frac{\mathrm{cov}\left(A_{j,i}(t), A_{k,i}(t)\right)}{E[A_i(t)]}. \tag{36}$$

In terms of the IDC's, we have

$$I_{a_i}(t) = \sum_{j=0}^{K} (\lambda_{j,i}/\lambda_i) I_{a_{j,i}}(t) + \beta_i(t). \tag{37}$$

We do not have an exact characterization of the correction terms $\beta_i(t)$ in (36) and (37). Thus, we again apply heavy-traffic limits to generate an approximation; see Corollary 4.2 of [52]. As a consequence, we obtain the approximation

$$\beta_{j,i;k,i}(t) = \beta_{k,i;j,i}(t) \approx (\zeta_{j,i;k,i}/\lambda_i) w^*((1-\rho_j)^2 p_{j,i} \lambda_j t / h(\rho) c^2_{x,j,i}), \tag{38}$$

where w^* is the weight function in (24), $h(\rho)$ is the tuning function in (26), $c^2_{x,j,i} = p_{j,i} c^2_{a,j} + (1 - p_{j,i}) + p_{j,i} c^2_{s,j}$ and $c^2_{a,j}$ is solved from the variability equations for the asymptotic variability parameters in (44), while $\zeta_{j,i;k,i}$ are scaled covariances of Brownian limit processes. In particular,

$$\zeta_{j,i;k,i} = \nu'_j \left(\mathrm{diag}(c^2_{a,0,i}\lambda_i) + \sum_{l=1}^{K} \Sigma_l\right) \nu_k + \nu'_k \Sigma_j e_i + \nu'_j \Sigma_k e_i, \tag{39}$$

where $\nu_l \equiv p_{l,i} e'_l (I - P')^{-1}$ for $l = j, k$, e_i is the i-th unit vector, $\mathrm{diag}(c^2_{a,0,i}\lambda_i)$ is the diagonal matrix with $c^2_{a,0,i}\lambda_i$ as the i-th diagonal entry, Σ_l is the covaraince matrix of the splitting decision process at station l defined as $\Sigma_l \equiv (\sigma^l_{i,j})$ with $\sigma^l_{i,i} = p_{l,i}(1 - p_{l,i})\lambda_l$ and $\sigma^l_{i,j} = -p_{l,i} p_{l,j} \lambda_l$ for $i \neq j$. Detailed derivation of (39) appears in §C.3.2.

4.4. The IDC Equation System

We now assemble the building blocks into a system of linear equations (for each t) that describes the IDC's in the OQN. Combining (22), (32) and (37), we obtain *the IDC equations*. These are equations that should be satisfied by the unknown IDCs. For $1 \le i \le K$, the equations are

$$\begin{aligned} I_{a,i}(t) &= \sum_{j=1}^{K} (\lambda_{j,i}/\lambda_i) I_{a,j,i}(t) + (\lambda_{0,i}/\lambda_i) I_{a,0,i}(t) + \beta_i(t), \\ I_{a,i,j}(t) &= p_{i,j} I_{d,i}(t) + (1 - p_{i,j}) + \alpha_{i,j}(t), \\ I_{d,i}(t) &= w_i(t) I_{a,i}(t) + (1 - w_i(t)) I_{s,i}(\rho t). \end{aligned} \tag{40}$$

The coefficient parameters $p_{i,j}$, $\lambda_{i,j}$ and λ_i are determined by the system parameters introduced in §4.1 and the traffic rate equations in §4.2. The external arrival IDC $I_{a_{0,i}}(t)$ and the service IDC $I_{s_i}(t)$ are assumed to be calculated via exact or numerical inversion of Laplace Transforms, or estimated from data, see §2.3.

The weight functions

$$w_i(t) \equiv w^*((1-\rho_i)^2 \lambda_i t/(h(\rho_i) c_{x,i}^2)) \tag{41}$$

involves a pre-determined function $w^*(t)$ defined in (24), the tuning function in (26), the effective arrival rate λ_i from (20), the traffic intensity ρ_i and a limiting variability parameter $c_{x,i}^2 \equiv I_{a,i}(\infty) + c_{s,i}^2$, as discussed in §4.3.1.

To solve for the limiting variability parameters $I_{a,i}(\infty)$, we let $t \to \infty$ in (40) and denote $c_{a,i}^2 \equiv I_{a,i}(\infty), c_{a,i,j}^2 \equiv I_{a,i,j}(\infty)$ and $c_{d,i}^2 \equiv I_{d,i}(\infty)$. Furthermore, we define

$$\begin{aligned} c_{\alpha_{i,j}}^2 &\equiv \alpha_{i,j}(\infty) = 2\xi_{i,j} p_{i,j}(1 - p_{i,j}), \\ c_{\beta_i}^2 &\equiv \beta_i(\infty) = \frac{2}{\lambda_i} \sum_{j<k} \zeta_{j,i;k,i}, \end{aligned}$$

where we used $w^*(\infty) = 1$ in (34) and (38). Hence, we have the *limiting variability equations*:

$$\begin{aligned} c_{a,i}^2 &= \sum_{j=1}^{K} (\lambda_{j,i}/\lambda_i) c_{a,j,i}^2 + (\lambda_{0,i}/\lambda_i) c_{a,0,i}^2 + c_{\beta_i}^2, \\ c_{a,i,j}^2 &= p_{i,j} c_{d,i}^2 + (1 - p_{i,j}) + c_{\alpha_{i,j}}^2, \\ c_{d,i}^2 &= c_{a,i}^2, \quad 1 \le i \le K. \end{aligned} \tag{42}$$

where we used the fact that $w_i(t) \to 1$ as $t \to \infty$.

For a concise matrix notation, let

$$\begin{aligned}
\mathbf{I}(t) &\equiv (I_{a,1}(t), \ldots, I_{a,K}(t), I_{a,1,1}(t), \ldots, I_{a,K,K}(t), I_{d,1}(t), \ldots, I_{d,K}(t)), \\
\mathbf{b}(t) &\equiv (b_{a,1}(t), \ldots, b_{a,K}(t), b_{a,1,1}(t), \ldots, b_{a,K,K}(t), b_{d,1}(t), \ldots, b_{d,K}(t)), \\
\mathbf{M}(t) &\equiv (M_{m,n}(t)) \in \mathbb{R}^{(2K+K^2)^2}, \quad m,n \in \{a_1, \ldots, a_K, a_{1,1}, \ldots, a_{K,K}, d_1, \ldots, d_K\}, \\
\mathbf{c}^2 &\equiv (c^2_{a,1}, \ldots, c^2_{a,K}, c^2_{a,1,1}, \ldots, c^2_{a,K,K}, c^2_{d,i}, \ldots, c^2_{d,K}),
\end{aligned}$$

where

$$\begin{aligned}
&b_{a,i}(t) \equiv \frac{\lambda_{0,i}}{\lambda_i} I_{a,0,i}(t) + \beta_i(t), \quad b_{a,i,j} \equiv (1 - p_{i,j}) + \alpha_{i,j}(t), \\
&b_{d,i}(t) \equiv (1 - w_i(t)) I_{s,i}(t); \quad M_{a_i, a_{j,i}(t)} = \frac{\lambda_{j,i}}{\lambda_i}, \\
&M_{a_{i,j}, d_i}(t) = p_{i,j}, M_{d_i, a_i}(t) = w_i(t), \quad \text{and} \quad M_{m,n}(t) = 0 \text{ otherwise.}
\end{aligned}$$

Then the IDC equations can be expressed concisely as

$$(\mathbf{E} - \mathbf{M}(t))\mathbf{I}(t) = \mathbf{b}(t), \tag{43}$$

while the limiting variability equations can be expressed as

$$(\mathbf{E} - \mathbf{M}(\infty))\mathbf{c}^2 = \mathbf{b}(\infty), \tag{44}$$

where $\mathbf{E} \in \mathbb{R}^{(2K+K^2)^2}$ is the identity matrix.

The following theorem states that these equations have unique solutions.

THEOREM 1. *Assume that $I - P'$ is invertible. Then $\mathbf{E} - \mathbf{M}(t)$ is invertible for each fixed $t \in \mathbb{R}^+ \cup \{\infty\}$. Hence, for any given t and $\mathbf{b}$, the IDC equations in* (40) *or* eqn: IDC equations *have the unique solution*

$$\mathbf{I}(t) = (\mathbf{E} - \mathbf{M}(t))^{-1}\mathbf{b}(t)$$

and the limiting variability equations in (44) *have the unique solution*

$$\mathbf{c} = (\mathbf{E} - \mathbf{M}(\infty))^{-1}\mathbf{b}(\infty).$$

Proof. Let $\delta_{i,j}$ be the Kronecker delta functioon. Then substituting the equations for $I_{a,j,i}(t)$ and $I_{d,i}(t)$ into the equation for $I_{a,i}(t)$, we obtain an equation set for $I_{a,i}(t)$ with coefficient matrix $(\delta_{i,j} - (\lambda_{j,i}/\lambda_i)p_{j,i}w_j(t)) \in \mathbb{R}^{K^2}$. Note that $(\lambda_{j,i}/\lambda_i)w_j(t) \leq 1$ for $t \in \mathbb{R}^+ \cup \{\infty\}$, the invertibility of $I - P'$ implies that the equations for $I_{a,i}(t)$ have an unique solution. Substituting in the solution for $I_{a,i}(t)$, we obtain solutions for $I_{a,i,j}(t)$ and $I_{d,i}(t)$. $\square$

The correction terms, defined in (33) and (36), provides a way to treat general queueing network settings such as customer feedback and dependence among flows. In deployment, one needs to specify or approximate $\alpha_{i,j}(t)$ and $\beta_i(t)$. Clearly, well defined correction term are of crucial important in obtaining an accurate RQNA algorithm. In §6, we discuss specific $\alpha_{i,j}(t)$ and $\beta_i(t)$ supported by the heavy-traffic limit theorems.

Remark 5. (the Kim [32, 33] $MMPP(2)$ decomposition) As indicated in §1, a decomposition approximation of queueing networks based on $MMPP(2)/GI/1$ queues is investigated in Kim [32, 33]. The waiting time of such system is known from [18]. The four rate parameters in the MMPP(2) are determined from the approximations of the mean, IDC and the third moment process of the arrival process at a pre-selected time t_0 and the limiting variability parameter of the arrival process. The IDC and third moment processes are approximated by the network equations with correction terms motivated from the Markovian routing settings.

At first glance, the IDC equations proposed here are quite similar to the network equations used in [32], see (20), (22) and (31) there. However, the three methods are different in three significant aspects. First, our approach does not fit the flows to special processes (MMPP in [32]), instead we partially characterize the flows by the IDC and apply the RQ algorithm reviewed in §3. Secondly, the entire IDC function is utilized in the RQ algorithm, whereas [32] used IDC evaluated at a pre-selected time t_0 to fit the parameters of the MMPP. Thirdly, we rely on more detailed heavy-traffic limit to propose asymptotically exact correction terms, see §6. $\square$

5. Feedback Elimination

In this section, we discuss the case in which customers can return (feedback) to a station after receiving service there. The possibility of feedback introduces dependence between the arrival process and the service times, even when the service times themselves are mutually independent. As a result, the decomposition $I_w(t) = I_a(t) + c_s^2$ in (3) is no longer valid.

Indeed, assuming that it is, as we do so far, can introduce serious errors, as we show in our simulation examples. We address this problem by introducing a feedback elimination procedure. We start with the so-called immediate feedback in §5.1 and generalize it into near-immediate feedback in §5.2.

5.1. Immediate Feedback Elimination

In Section III of [45] it is observed that it is often helpful to pre-process the model data by eliminating immediate feedback for queues with feedback. We now review how that can be done.

We consider a single queue with i.i.d. feedback. In this case, all feedback is *immediate feedback*, meaning that the customer feeds back to the station immediately after completing service, without first going through another station. For a $G/GI/1$ model allowing feedback, all feedback is necessarily immediate because there is only one station. Normally, the immediate feedback returns the customer back to the end of the line at the same station. However, in the immediate feedback elimination procedure, the approximation step is to put the customer back at the head of the line. Thus, the customer receives a geometrically random number of service times all at once. Clearly this does not alter the queue length process or the workload process, because the approximation step is work-conserving.

The modified system does not have a feedback flow. Let N_p denote a geometric random variable with success probability $1-p$ and support $\mathbb{N}^+$, the positive natural numbers, then the new service time can be expressed as

$$S_p = \sum_{i=1}^{N_p} S_i, \tag{45}$$

where S_i's are i.i.d. copies of the original service times. This modification in service times results in a change in the service scv. By the conditional variance formula, the scv of the total service time is $\tilde{c}_s^2 = p + (1-p)c_s^2$. The new service IDC in the modified system is the IDC of the stationary renewal process associated with the new service times. To obtain the new service IDC, we need only find the Laplace Transform of the new service distribution, then apply the algorithm in §2.3.1. Let g_p denote the density function of the new service time, we have

$$\hat{g}_p(s) \equiv E\left[\exp\left(-s\sum_{i=1}^{N_p} S_i\right)\right] = E\left[E\left[\exp\left(-s\sum_{i=1}^{N_p} S_i\right)\middle| N_p\right]\right]$$

$$= E\left[\prod_{i=1}^{N_p} E\left[\exp\left(-sS_i\right)\right]\right] = E\left[\hat{g}^{N_p}(s)\right] = M_p(\hat{g}(s)),$$

where $\hat{g}(s)$ is the Laplace transform of the original service distribution and M_p is the probability generating function of the geometric random variable described above.

For the mean waiting time, we need to adjust for per-visit waiting time by multiplying the waiting time in the modified system by $(1-p)$. Note that $(1-p)^{-1}$ is the mean number of visit by a customer in the original system.

In §4.1 of [52] it is shown that the modified system after the immediate feedback elimination procedure shares the same HT limits of the queue length process, the external departure process, the workload process and the waiting time process. Hence, the immediate feedback elimination procedure as an approximation is asymptotically exact in the heavy-traffic limit.

5.2. Near-Immediate Feedback

Now, we consider general OQNs, where the feedback does not necessarily happen immediately, meaning that a customer may visit other stations before coming back to the feedback station. To treat general OQNs, we extend the immediate feedback concept to *near-immediate feedback*, which depends on the traffic intensities of the queues on the feedback path. Near-immediate feedback is then defined as feedback that does not go through any station with higher traffic intensity. The RQNA algorithm eliminates all near-immediate feedback.

An alternative algorithm eliminates only all near-immediate feedback from the bottleneck queues, where a *bottleneck queue* is a station with a traffic intensity that equals the highest traffic intensity in the network. For each bottleneck queue in the network, by the definition of near-immediate feedback, we eliminate all feedback at this queue when we analyze the mean workload at that queue, even if the feedback flow passes through other bottleneck queues.

To help understand near-immediate feedback, consider a modified OQN with one bottleneck queue, denoted by h, while all non-bottleneck queues have service times set to 0 so that they serve as instantaneous switches. In the reduced network, we define an external arrival $\hat{A}_0$ to the bottleneck queue to be any external arrival that arrive at the bottleneck queue for the first time. Hence, an external arrival may have visited one or multiple non-bottleneck queues before its first visit to the bottleneck queue. In particular, the external

arrival process can be expressed as the superposition of (i) the original external arrival process $A_{0,h}$ at station h; and (ii) the Markov splitting of the external arrival process $A_{0,i}$ at station i with probability $\hat{p}_{i,h}$, for $i \neq h$, where $\hat{p}_{i,h}$ denote the probability of a customer that enters the original system at station i ends up visiting the bottleneck station h. For the explicit formula of $\hat{p}_{i,h}$, see Remark 3.2 of [52].

In §4.2 of [52], we showed that this reduced network is asymptotically equivalent in the HT limit to the single-server queue with i.i.d. feedback that we considered in §5.1. In particular, the arrival process of the equivalent single-station system is $\hat{A}_0$ as described above, the service times remain unchanged and the feedback probability is $\hat{p}$, which is exactly the probability of a near-immediate feedback in the original system; see (3.9) of [52] for the expression of $\hat{p}$. Hence we showed that eliminating all feedback at the bottleneck queue as described above prior to analysis is asymptotically correct in HT for OQNs with a single bottleneck queue in terms of the queue length process, the external departure process, the workload process and the waiting time process. Moreover, the different variants of the algorithm - eliminating all near immediate feedback or only the near-immediate feedback at the bottleneck queues - are asymptotically exact in the HT limit for an OQN with a single-bottleneck queue, because only the bottleneck queues have nondegenerate HT limit. In contrast, if there are multiple bottleneck queues, the HT limit requires multidimensional RBM, which is not used in our RQNA.

6. The RQNA Algorithm

As basic input parameters, the RQNA algorithm requires the model data specified in §4.1:

1. Network topology specified by the routing matrix P;

2. External arrival processes specified by (i) the interarrival distribution, if renewal; or (ii) rate λ and IDC; or (iii) a realized sample path of the stationary external arrival process;

3. Service renewal process specified by (i) the service distribution; or (ii) the rate and IDC; or (iii) a realized sample path of the service renewal process.

Combining the traffic-rate equation, the limiting variability equation, the IDC equation and the feedback elimination procedure, we have obtained a general framework for the RQNA algorithm, which we summarize in Algorithm 1.

The general framework here allows different choices of (1) the tuning function in (23), (2) the correction terms $\alpha_{i,j}$ in §4.3.2 and β_i in §4.3.3 and (3) the feedback elimination procedure.

Algorithm 1: A general framework of the RQNA algorithm for the approximation of the system performance measures.

Require: Specification of the correction terms $\alpha_{i,j}(t)$ in §4.3.2 and $\beta_i(t)$ in §4.3.3, a set of stations to perform feedback elimination as specified in §5 and the flows to eliminate for each of the selected station.

Output : Approximation of the system performance measures.

1 Solve the traffic rate equations by $\lambda = (I - P')^{-1}\lambda_0$ as in §4.2 and let $\rho_i = \lambda_i/\mu_i$;

2 Solve the limiting variability equations by $\mathbf{c} = (\mathbf{E} - \mathbf{M}(\infty))^{-1}\mathbf{b}(\infty)$ specified in §4.4;

3 Solve the IDC equations by $\mathbf{I}(t) = (\mathbf{E} - \mathbf{M}(t))^{-1}\mathbf{b}(t)$ for the total arrival IDCs, where we use $\mathbf{c}$ from Step 2 in (41);

4 Select a set of stations to perform feedback elimination, as in §5. For each selected station, identify the flows to eliminate, then identify the corresponding feedback probability, the modified service IDC as in §5.1 as well as the reduced network. Repeat Step 1 to Step 3 on the reduced network to obtain the modified IDW (as the sum of the modified total arrival IDC and the modified service scv) at the selected station.

5 Apply the RQ algorithm in (13) to obtain the approximations for the mean steady-state workload at each station.

6 Apply the formulas in §3.2 to obtain approximations for the expected values of the steady-state queue length and waiting time at each queue and the total sojourn time for the system.

As default, we use the tuning function in (26), the correction terms in (34) and (38). For the feedback elimination procedure, we apply near-immediate feedback elimination to all stations.

6.1. RQNA for Tree-Structured Queueing Networks

We also develop a more elementary algorithm for tree-structured OQNs. A *tree-structured queueing network* is an OQN whose topology forms a directed tree. Recall that a directed tree is a connected directed graph whose underlying undirected graph is a tree. The queueing network in this setting contains either re-combining after splitting nor customer feedback. The tree-structured network is a special case of feed-forward network in which the superposed flows ate each node have no common origin.

This special structure greatly simplifies the IDC-based RQNA algorithm. First, feedback elimination is unnecessary because there is no customer feedback. Second, for any internal flow $A_{i,j}$ that is non-zero, we must have $\alpha_{i,j} = 0$ for the correction term in (33), because the tree structure implies that the two processes D_i^* and $\Theta_{i,j}^*$ are mutually independent. In particular, by definition,

$$\alpha_{i,j}^*(t) \equiv 2\text{cov}(p_{i,j}D_i^*(t), \Theta_{i,j}^*(\lambda_i t))/E[A_{i,j}^*(t)] = 0.$$

Finally, the tree structure implies that $\beta_i = 0$ for the correction term for superposition because all superposed processes are independent.

With these simplifications of the correction terms, the equations in (40), yield, for $1 \le i, j \le K$,

$$\begin{aligned} I_{a_i}(t) &= \sum_{j=1}^{K} \frac{\lambda_{j,i}}{\lambda_i} I_{a_{j,i}}(t) + (\lambda_{0,i}/\lambda_i) I_{a_{0,i}}(t), \\ I_{a_{i,j}}(t) &= p_{i,j} I_{d_i}(t) + (1 - p_{i,j}), \\ I_{d_i}(t) &= w_i(t) I_{a_i}(t) + (1 - w_i(t)) I_{s_i}(t). \end{aligned}$$

The IDC equations in this setting inherit a special structure that allows a recursive algorithm. Note that the stations in the tree-structured network can be partitioned into disjoint layers $\{\mathcal{L}_1, \ldots, \mathcal{L}_l\}$ such that for station $i \in \mathcal{L}_k$, it takes only the input flows from $j \in \bigcup_{j=1}^{k-1} \mathcal{L}_j$ for $1 \le k \le l$. To simplify the notation, we sort the node in the order of their layers and assign arbitrary order to nodes within the same layer. If $i \in \mathcal{L}_k$, then $\bigcup_{j=1}^{k-1} \mathcal{L}_j \subset \{1, 2, \ldots, i-1\}$, so that $\lambda_{j,i} = 0$ for all $j \ge i$. Hence, by substituting in the equations for I_{d_i} and $I_{a_{i,j}}$ into that of I_{a_i}, we have

$$\begin{aligned} I_{a_i}(t) &= \sum_{j=1}^{K} \frac{\lambda_{j,i}}{\lambda_i} \left(p_{j,i} \left(w_j(t) I_{a_j}(t) + (1 - w_j(t)) I_{s_j}(t) \right) + (1 - p_{j,i}) \right) + \frac{\lambda_{0,i}}{\lambda_i} I_{a_{0,i}}(t), \\ &= \sum_{j<i} \frac{\lambda_{j,i}}{\lambda_i} \left(p_{j,i} \left(w_j(t) I_{a_j}(t) + (1 - w_j(t)) I_{s_j}(t) \right) + (1 - p_{j,i}) \right) + \frac{\lambda_{0,i}}{\lambda_i} I_{a_{0,i}}(t). \end{aligned} \tag{46}$$

Note that (46) exhibits a lower-triangular shape so that we can explicitly write down the solution in the order of the stations. We summarize the procedure in Algorithm 2. With the total arrival IDCs, we simply continue to Step 5 and 6 in Algorithm 1 to obtain approximations to the system performance measures.

Algorithm 2: The RQNA algorithm for approximating the IDC's in a tree-structured queueing networks.

Require: The queueing network has tree structure.

Output : Solution to the IDC equations (43).

for $i = 1$ *to* n **do**
- $\lambda_i \leftarrow \lambda_{0,i} + \sum_{j<i} \lambda_j p_{j,i}$;
- $\rho_i \leftarrow \lambda_i/\mu_i$;
- $c^2_{a,i} \leftarrow \sum_{j<i} \frac{\lambda_{j,i}}{\lambda_i} c^2_{a,j,i} + \frac{\lambda_{0,i}}{\lambda_i} c^2_{a,0,i}$;
- $c^2_{x,i} \leftarrow c^2_{a,i} + c^2_{s,i}$;
- $w_i(t) \leftarrow w^*((1-\rho_i)^2 \lambda_i t/(\rho_i c^2_{x,i}))$;
- $I_{a_i}(t) \leftarrow \sum_{j<i} \frac{\lambda_{j,i}}{\lambda_i} \left(p_{j,i}\left(w_j(t) I_{a,j}(t) + (1-w_j(t)) I_{s,j}(t)\right) + (1-p_{j,i})\right) + \frac{\lambda_{0,i}}{\lambda_i} I_{a,0,i}(t)$;
- $I_{d_i}(t) \leftarrow w_i(t) I_{a,i}(t) + (1-w_i(t)) I_{s,i}(t)$;
- **for** $j < i$ **do**
  - $I_{a,i,j}(t) \leftarrow p_{i,j} I_{d,i}(t) + (1-p_{i,j})$;
- **end**

end

return $\mathbf{I}(t)$.

7. Numerical Studies

We have conducted a wide range of simulation experiments to confirm the effectiveness of RQNA. Some of them have been done in our previous papers and so are here put in §A of the appendix. We briefly discuss these first. Afterwards, we discuss examples of networks with significant near-immediate feedback from [10]. We show that the near-immediate feedback in these examples makes a big difference in the performance descriptions. Hence our predictions with and without feedback elmination are very different. We find that our RQNA with near-immediate feedback elimination performs as well or better than the other algorithms.

7.1. Examples from Previous Papers

An important innovation in this work is in the way we look at the performance of an OQN. On the one hand, we take a limited view, looking only at the mean steady-state workload at each queue in the OQN. However, we look at this mean steady-state workload as a function of the traffic intensity of that queue; i.e., we look at the normalized mean

workload $c_Z^2(\rho)$ as a function of ρ. That perspective is achieved by scaling the mean service time at the queue so that the traffic intensity ρ varies across its full range $0 < \rho < 1$.

Hence, our performance measures are not single mean values, but instead are sets of mean values in the normalized form $c_Z^2(\rho)$ in (4). Thus, we are evaluating not one OQN, but instead a continuum of OQNs. In our experiments we achieve these continua approximately by looking at large finite subsets.

7.1.1. An Example Having an IDW with Six Modes. To illustrate how the normalized mean workload $c_Z^2(\rho)$ can vary as a function of ρ and how that behavior can be captured by the IDW $I_w(t)$ as a function of t, in §A.1 we consider the $EHEHE \to M$ example from §EC.8.2 of [54]. This example has 5 single-server queues in series, where the variability increases and then decreases 5 times, with the traffic intensities at successive queues decreasing. That makes the external arrival process and the earlier queues relevant only as the traffic intensity increases. Specifically, the example can be denoted by

$$E_{10}/H_2/1 \to \cdot/E_{10}/1 \to \cdot/H_2/1 \to \cdot/E_{10}/1 \to \cdot/M/1. \tag{47}$$

In particular, the external arrival process is a rate-1 renewal process with Erlang E_{10} interarrival times, thus $c_a^2 = 0.1$. The 1^{st} queue has hyperexponential H_2 service times with mean 0.99 and $c_s^2 = 10$ thus the traffic intensity at this queue is 0.99. (Throughout this paper, we fix the third H_2 parameter by stipulating that it also has balanced means, as on p. 137 of [44].) The 2^{nd} queue has E_{10} service time with mean and thus traffic intensity 0.98. The 3^{rd} queue has H_2 service times with mean 0.70 and $c_s^2 = 10$. The 4^{th} queue has E_{10} service times with mean and thus traffic intensity 0.5. The last (5^{th}) queue has an exponential service-time distribution. with mean and traffic intensity ρ.

Figure 3 (left) shows the IDW at that last queue over the interval $[10^{-2}, 10^5]$ in log scale, while Figure 3 (right) shows the impact of ρ on the performance of that last queue. Figure 3 shows that the IDW and the normalized mean workload are not nearly constant, showing that a single variability parameter cannot possibly perform well at all traffic intensities. The limits of these functions at the endpoints are consistent with (5), but these functions have substantial variation in between. These functions have six modes if we consider the left and right end points.

7.1.2. Evaluation of RQ and RQNA for 10 Queues in Series. Next, the example in §A.2 taken from §5.2 of [54] evaluates both our approximation for the IDC and the normalized workload. For this example, we consider 10 single-server queues in series. The external arrival process is a rate-1 renewal process with H_2 interarrival times, having $c_a^2 = 5$. The first 9 queues all have Erlang service times with $c_a^2 = 0.5$ denoted by E_2, i.e., the sum of 2 i.i.d. exponential random variables. The first 8 queues have mean service time and thus traffic intensity 0.6, while the 9^{th} queue has mean service time and thus traffic intensity 0.95. The difference in variability level of the arrival and service process introduces complex variability structure underneath the first 9 queues in series. The 10^{th} queue serves as a test queue and has an exponential service-time distribution with mean and traffic intensity ρ, which is allowed to vary from 0 to 1 in order to expose the complex impact of the variability on the performance measure of the test queue.

The RQNA algorithm in this case is a simple special case of Algorithm 2. The IDC's of the external flows (I_{a_1} for external arrival at station 1 and I_{s_i} service flows) can be derived by explicitly inverse (8), see §III.G of [17]. For internal flows, we apply the departure approximation in (22) recursively, so that for $2 \le i \le 9$,

$$\begin{aligned} I_{d_1}(t) &= w_1 I_{a_1}(t) + (1 - w_1) I_{s_1}(t), \quad \text{and} \\ I_{d_i}(t) &= w_i I_{d_{i-1}}(t) + (1 - w_i) I_{s_i}(t), \end{aligned} \tag{48}$$

where we used (23) with $h(\rho) = \rho^2$ as in (26) with $\rho_i = 0.6$ for $1 \le i \le 8$, $\rho_9 = 0.95$, $\lambda_i = 1$. For the variability parameters, we note that $c_{x_i}^2 \equiv c_{a_i}^2 + c_{s_i}^2 = c_{a_i}^2 + 0.5$ and that, for $2 \le i \le 9$,

$$c_{a_i}^2 \equiv I_{a_i}(\infty) = I_{d_{i-1}}(\infty) = I_{a_{i-1}}(\infty) = \cdots = I_{a_1}(\infty) = c_{a_1}^2 = 5.$$

With $I_{a_{10}}(t) = I_{d_9}(t)$, we can now apply the RQ algorithm in (13) to obtain approximation of the steady-state mean workload.

Figure 4 (left) shows that the IDC approximation in the RQNA algorithm performs very well, while Figure 4 (right) shows that both RQ (with directly estimated IDC) and RQNA are accurate, just as for the more complex example in §A.1.

7.1.3. Comparisons with Previous Algorithms for Queues in Series. In §A.4 we compare the performance of our RQNA algorithm to the performance of QNA from [45], QNET

from [22], SBD from [10] and RQ from [54], for the example with 9 queues in series considered by [43]. This example was introduced by [43] to illustrate the heavy-traffic bottleneck phenomenon. This example is also discussed in §5 of [53].

In particular, we consider an OQN with 9 stations in tandem, each with i.i.d. exponential service times. Station 1 has the only external arrival process, which is a rate-1 general renewal process. The traffic intensities at the first 8 queues are set to $\rho_i = 0.6$ for $1 \le i \le 8$, while the last queue has the significantly higher traffic intensity $\rho_9 = 0.9$. As in [43], two specific external renewal arrival processes are considered: (i) deterministic interarrival times with $c_{a_0}^2 = 0$; and (ii) highly variable H_2 interarrival times with $c_{a_0}^2 = 8$ (and again balanced means).

The heavy-traffic bottleneck phenomenon illustrates that the variability of the external arrival process can have only very limited impact on the performance of the following queues, especially after passing through several queues, and yet dramatically affect the performance of a later queue with a much higher traffic intensity. This phenomenon is a result of complicated long-range dependence embedded in the arrival processes, introduced by flowing through a queue (the departure processes), as discussed in §A.1 and revealed by the departure approximation in (22). This example was introduced to show the limitation of traditional decomposition methods, e.g. the QNA algorithm, and is often used as a benchmark for different approximation methods, see §3.3 of [10].

Table 6 (for low variability) and Table 7 (for high variability) compare the various approximations of the mean steady-state waiting time at each station, as well as the total waiting time in the system, to simulation estimates.

We make the following observations from this experiment:

1. The new RQNA algorithm does better than the QNA and QNET methods on total time spent waiting in queue, and is comparable with the SBD method, even though RQNA does not require solving an RBM.

2. The RQNA algorithm does exceptionally well at the final bottleneck queue and is competitive with all other methods for approximating the mean waiting time. The new RQNA method is based on heavy-traffic limits just as the previous methods methods, but focuses on the flows, and exploits RQ instead of analyzing an RBM.

3. The RQNA algorithm can benefit from further improvement for light-to-medium traffic intensities. As demonstrated in Table 7, the mean waiting times at queues 3-8 are

pushed too much towards the $M/M/1$ values in the departure IDC approximation for light to medium traffic intensity. That remains to be a direction for future research.

In §5 of [53] we also consider a variant of this experiment in which the third parameter of the H_2 interarrival-time distributon in the renewal external arrival process is changed, so that the distribution no longer has balanced means. Since this change leaves the first two moments unchanged, all previous approximations remain unchanged. However, since RQ and RQNA depend on the entire distribution of the interarrival-time distribution, it changes. Table 2 of [53] shows that RQ and RQNA perform better than the other methods in these modified models. For third parameter $r = 0.9$, 0.5 and 0.1, the simulation estimates of the total mean waiting time are 28.8, 45.3 and 47.5, the corresponding RQNA approximations are 33.8, 40.1 and 43.7 and the SBD approximation is always 49.8. The maximum relative errors of RQNA and SBD for these three cases are 20% and 73%.

7.2. Examples from [10] with Significant Feedback

As discussed in §5, the RQNA algorithm can benefit from the feedback elimination procedure when customer feedback is present. In this section, we show various numerical examples to support feedback elimination. For that purpose, we discuss two examples from [10] with significant feedback. We remark that the SBD algorithm performed remarkably well in these examples.

7.2.1. Feedback Elimination: A Three-Station Example. In this section, we look at the suite of three-station examples §3.1 of [10] depicted in Figure 1. This example is designed to have three stations that are tightly coupled with each other, so that the dependence among the queues and the flows is fairly complicated.

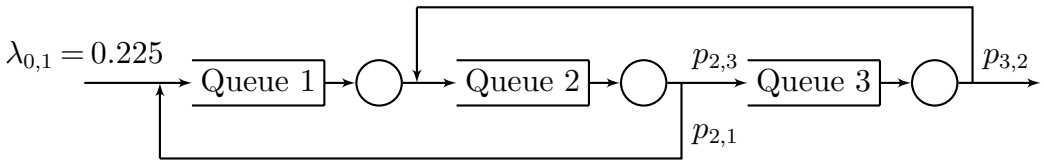

Figure 1 **A three-station example.**

In this example, we have three stations in tandem but also allow customer feedback from station 2 to station 1 and from station 3 to station 2, with probability $p_{2,1} = p_{2,3} = p_{3,2} = 0.5$. The only external arrival process is a Poisson process which arrives at station 1 with rate $\lambda_{0,1} = 0.225$, hence by (20) the effective arrival rate is $\lambda_1 = 0.675, \lambda_2 = 0.9$ and $\lambda_3 = 0.45$.

For the service distributions, we consider the same sets of parameters as in [10], summarized in Table 1 and 2. Note that Case 2 is relatively more challenging because there are two bottlneck stations; in contrast, all the other cases have only one.

Table 1 Traffic intensity of the four cases in the three-station example.

Case	ρ_1	ρ_2	ρ_3
1	0.675	0.900	0.450
2	0.900	0.675	0.900
3	0.900	0.675	0.450
4	0.900	0.675	0.675

Table 2 Variability of the service distributions of the four cases in the three-station example.

Case	$c^2_{s,1}$	$c^2_{s,2}$	$c^2_{s,3}$
A	0.00	0.00	0.00
B	2.25	0.00	0.25
C	0.25	0.25	2.25
D	0.00	2.25	2.25
E	8.00	8.00	0.25

We now compare the RQNA approximations and four previous algorithms as in §A.4, with the simulated mean sojourn times at each station, as well as total sojourn time of the network. The sojourn time for each station is defined as the waiting time plus the service time at that station, whereas the total sojourn time of the network is defined as in (19). We consider two cases of the RQNA algorithm: (1) the plain RQNA algorithm without feedback elimination, as in Algorithm 1 and (2) the RQNA algorithm with feedback elimination, as discussed in §5.

For RQNA with feedback elimination, we apply feedback elimination to each station that has at least one feedback flow that only passes through stations with equal or lower traffic intensities. We eliminate all such flows in the feedback elimination procedure. Take Case 1 for example, we do not apply feedback elimination for Station 1 because all feedback customers go through Station 2, which has higher traffic intensity; we will, however, eliminate the flow from 2 to 1 as well as the flow from 3 to 2 for Station 2, since both Station 1 and 3 have lower traffic intensities. As another example, for both Station 2 and 3 in case 4, we eliminate the flow from 3 to 2, but we do not eliminate the flow from 2 to 1, since Station 2 and 3 share the same traffic intensity while Station 1 has higher traffic intensity.

Tables 3 and 4 expand Tables II and III in [10] by adding values for (1) the mean total sojourn time and (2) the RQ and RQNA approximations, with and without feedback elimination. For each table, we indicate by an asterisk in the last column the stations where elimination is applied.

Table 3 A comparison of six approximation methods to simulation for the total sojourn time in the three-station example in Figure 1 with parameters specified in Table 1 and 2.

Case		Simulation	QNA	QNET	SBD	RQ	RQNA	RQNA (elim)
A	1	40.39 (3.75%)	20.5 (-49%)	diverging	43.0 (6.4%)	73.9 (83%)	83.5 (107%)	44.8 (11.0%)
	2	59.58 (3.29%)	36.0 (-40%)	56.7 (-4.9%)	58.2 (-2.4%)	78.0 (31%)	94.3 (58%)	69.3 (16.4%)
	3	40.72 (4.78%)	24.0 (-41%)	38.7 (-5.0%)	40.2 (-1.3%)	57.2 (41%)	74.7 (83%)	43.3 (6.3%)
	4	42.12 (3.36%)	26.2 (-38%)	41.8 (-0.7%)	42.7 (1.3%)	59.3 (41%)	75.1 (78%)	41.2 (-2.2%)
B	1	52.40 (2.64%)	42.0 (-20%)	52.6 (0.4%)	50.2 (-4.2%)	72.4 (38%)	93.7 (79%)	53.1 (1.4%)
	2	91.52 (3.77%)	94.1 (2.8%)	83.7 (-8.5%)	95.3 (4.1%)	109 (20%)	169 (85%)	94.5 (3.2%)
	3	61.68 (3.44%)	72.2 (17%)	61.9 (0.4%)	60.9 (-1.3%)	79.4 (29%)	133 (115%)	60.5 (-1.9%)
	4	63.34 (2.83%)	75.8 (20%)	64.1 (1.3%)	64.7 (2.1%)	83.0 (31%)	135 (113%)	62.4 (-1.4%)
C	1	44.24 (1.96%)	31.3 (-29%)	37.0 (-16%)	47.1 (6.4%)	75.7 (71%)	91.4 (106%)	42.1 (-4.8%)
	2	92.42 (4.23%)	87.4 (-5.4%)	91.2 (-1.4%)	91.6 (-0.83%)	106 (15%)	156 (68%)	96.0 (3.8%)
	3	44.26 (4.69%)	33.2 (-25%)	44.0 (-0.7%)	45.0 (1.7%)	61.3 (38%)	84.2 (90%)	44.0 (-0.6%)
	4	50.20 (1.04%)	41.4 (-18%)	51.1 (1.7%)	52.2 (4.0%)	67.4 (34%)	91.2 (82%)	45.9 (-8.6%)
E	1	134.4 (4.77%)	265 (97%)	155 (15%)	116 (-14%)	158 (17%)	305 (127%)	120 (-11%)
	2	213.1 (3.47%)	308 (45%)	228 (7.1%)	206 (-3.3%)	234 (10%)	367 (72%)	173 (-19%)
	3	138.7 (3.97%)	244 (76%)	161 (16%)	135 (-2.5%)	163 (17%)	300 (116%)	136 (-2.0%)
	4	155.1 (4.37%)	252 (63%)	168 (8.2%)	147 (-5.0%)	178 (15%)	312 (101%)	148 (-4.8%)

Table 4 A comparison of six approximation methods to simulation for the sojourn time at each station of the three-station example in Figure 1 for Case D as specified in Table 1 and 2.

Case	Station	Simulation	QNA	QNET	SBD	RQ	RQNA	RQNA (elim)
D1	1	2.476 (0.61%)	2.24 (-9.4%)	2.48 (0.3%)	2.47 (-0.1%)	2.47 (-0.28%)	2.68 (7.8%)	2.68 (7.8%)
	2	10.85 (3.21%)	14.9 (37%)	11.6 (6.5%)	11.4 (5.2%)	19.8 (83%)	28.4 (162%)	11.1* (2.7%)
	3	2.544 (0.63%)	2.53 (-0.8%)	2.54 (-0.0%)	2.59 (1.6%)	2.57 (1.2%)	2.53 (-0.7%)	2.53 (-0.7%)
	Total	55.81 (2.58%)	71.4 (28%)	58.8 (5.3%)	58.2 (4.3%)	91.8 (64%)	127 (127%)	57.6 (3.3%)
D2	1	11.35 (3.29%)	8.01 (-29%)	10.8 (-4.5%)	11.1 (-1.9%)	13.7 (20%)	16.6 (46%)	11.3* (0.1%)
	2	2.643 (1.25%)	2.96 (12%)	2.75 (4.0%)	2.82 (6.7%)	2.85 (7.8%)	3.06 (16%)	3.06 (16%)
	3	26.87 (2.04%)	32.9 (22%)	26.8 (-0.4%)	24.9 (-7.5%)	27.5 (2.2%)	36.4 (35%)	31.1* (16%)
	Total	98.36 (1.82%)	102 (3.4%)	97.2 (-1.2%)	94.4 (-4.0%)	104 (6.0%)	132 (34%)	105 (7.1%)
D3	1	11.39 (3.04%)	7.95 (-30%)	11.0 (-3.5%)	11.3 (-0.5%)	15.8 (39%)	16.5 (45%)	11.3* (-0.5%)
	2	2.290 (1.27%)	2.90 (27%)	2.53 (10%)	2.26 (-1.4%)	2.57 (12%)	3.04 (33%)	2.10* (-8.2%)
	3	2.220 (0.59%)	2.40 (7.9%)	2.38 (7.0%)	2.59 (16%)	2.39 (7.6%)	2.43 (9.6%)	2.43 (9.6%)
	Total	47.72 (2.51%)	40.2 (-16%)	47.8 (0.2%)	48.2 (1.0%)	62.6 (31%)	66.6 (39%)	47.5 (0.51%)
D4	1	11.30 (6.39%)	7.97 (-29%)	10.9 (-3.2%)	11.3 (0.3%)	14.2 (26%)	16.43 (45%)	11.3* (0.3%)
	2	2.414 (1.12%)	2.93 (21%)	2.64 (9.5%)	2.60 (7.7%)	2.65 (10%)	3.05 (26%)	2.10* (-13%)
	3	5.886 (1.05%)	6.83 (16%)	6.31 (7.3%)	6.17 (4.8%)	6.47 (10%)	6.85 (16%)	5.95* (1.1%)
	Total	55.24 (4.37%)	49.3 (-11%)	56.0 (1.4%)	56.7 (2.7%)	69.3 (25%)	75.5 (37%)	54.3 (-1.7%)
Average absolute relative error			20.24%	4.72%	4.52%	21.61%	42.60%	5.51%

We observed that the plain RQNA algorithm works well for stations with moderate to low traffic intensities, but not so satisfactory for congested stations. On the other hand, the accuracy of the RQNA algorithm with feedback elimination is on par with, if not better than the best previous algorithm.

7.2.2. A 10-Station Example with Feedback. We conclude with the 10-station OQN example with feedback considered in §3.5 of [10]. It is depicted here in Figure 2.

The only exogenous arrival process is Poission with rate 1. For each station, if there are two routing destinations, the departing customer follows Markovian routing with equal probability, each being 0.5. The vector of mean service times is $(0.45, 0.30, 0.90, 0.30, 0.38571, 0.20, 0.1333, 0.20, 0.15, 0.20)$, so that the traffic intensity vector is $(0.6, 0.4, 0.6, 0.9, 0.9, 0.6, 0.4, 0.6, 0.6, 0.4)$. The scv's at these stations are $(0.5, 2, 2, 0.25, 0.25, 2, 1, 2, 0.5, 0.5)$, where we assume a Erlang distribution if $c_s^2 < 1$, an exponential distribution if $c_s^2 = 1$ and a hyperexponential distribution if $c_s^2 > 1$.

In particular, note that stations 4 and 5 are bottleneck queues, having equal traffic intensity, far greater than the traffic intensities at the other queues. Moreover, these two stations are quite closely coupled. Thus, at first glance, we expect that SBD with two-dimensional RBM should perform very well, which proves to be correct. Moreover, this example should be challenging for RQNA because it is based on heavy-traffic limits for OQNs with only a single bottleneck, thus involving only one-dimensional RBM.

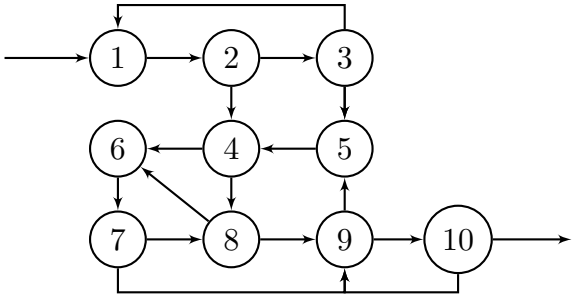

Figure 2 A ten-station with customer feedback example.

In Table 5, we report the simulation estimates and approximattions for the steady-state mean sojourn time (waiting time plus service time) at each station, as well as the total sojourn time of the system, calculated as in (19). For the approximations, we compare QNA from [45], QNET from [22], SBD from [10], RQ from [54] (with estimated IDC), as well as the RQNA algorithms here. The simulation, QNA, QNET and SBD columns are taken from Table XIV of [10].

Again, we consider two versions of RQNA algorithm, the first one does not eliminate feedback, while the second one (marked by 'elim') applies the feedback elimination procedure. As before, in eliminating customer feedback, for each station, we identify the near-immediate feedback flows as the flows that come back to the station after completing service, without passing through any station with a higher traffic intensity. We then

eliminate all near-immediate feedback flows, apply plain RQNA algorithm on the reduced network and use the new RQNA approximation as the approximation for that station.

Table 5 A comparison of six approximation methods to simulation for the mean steady-state sojourn times at each station of the open queueing network in Figure 2.

Station	Simulation	QNA	QNET	SBD	RQ	RQNA	RQNA (elim)
1	0.99 (0.86%)	0.97 (-2.8%)	1.00 (0.2%)	1.00 (0.4%)	0.97 (-2.0%)	1.09 (9.2%)	1.00* (0.4%)
2	0.55 (0.69%)	0.58 (6.0%)	0.56 (2.6%)	0.55 (0.2%)	0.55 (-0.1%)	0.56 (1.3%)	0.56 (1.4%)
3	2.82 (1.93%)	2.93 (4.2%)	2.90 (3.2%)	2.76 (-2.0%)	2.96 (5.0%)	3.40 (21%)	2.75* (-2.5%)
4	1.79 (3.71%)	1.34 (-25%)	1.41 (-21%)	1.76 (-1.6%)	2.34 (31%)	3.51 (97%)	2.11* (18%)
5	2.92 (4.77%)	2.49 (-15%)	2.44 (-17%)	2.81 (-3.6%)	3.77 (29%)	9.07 (211%)	3.35* (15%)
6	0.58 (0.78%)	0.64 (10%)	0.62 (7.4%)	0.59 (2.2%)	0.60 (3.8%)	0.70 (20%)	0.49* (-16%)
7	0.24 (0.28%)	0.24 (-1.7%)	0.26 (7.1%)	0.27 (11%)	0.23 (-3.0%)	0.24 (-1.3%)	0.24 (-1.3%)
8	0.58 (0.67%)	0.64 (9.6%)	0.61 (4.6%)	0.60 (1.7%)	0.61 (3.9%)	0.70 (20%)	0.59* (0.6%)
9	0.34 (0.63%)	0.32 (-6.1%)	0.35 (2.0%)	0.43 (26%)	0.33 (-4.2%)	0.73 (111%)	0.42* (21%)
10	0.29 (0.19%)	0.30 (2.4%)	0.29 (1.4%)	0.28 (-1.7%)	0.28 (-1.5%)	0.26 (-8.7%)	0.26 (-8.7%)
Total	22.0 (2.45%)	20.3 (-7.9%)	20.4 (-7.3%)	22.4 (1.7%)	26.1 (18%)	44.5 (102%)	24.2* (9.9%)

We make the following observations from this numerical example:

1. Particular attention should be given to the two bottleneck stations: 4 and 5. Note that QNA and QNET produce $15-25\%$ error, which is satisfactory, but SBD does far better with only $1-4\%$ error.

2. The RQNA algorithm without feedback elimination can perform very poorly with high traffic intensity and high feedback probability, presumably due to the break down of (3).

3. With feedback elimination, the RQNA algorithm performs significantly better and is competitive with previous algorithms in this complex setting, producing $15-18\%$ error at stations 4 and 5. The performance of RQNA at the tightly coupled bottleneck queues evidently suffers because the current RQNA depends heavily on one-dimensional RBM.

8. Conclusions

In this paper we developed a new robust queueing network queueing analyzer (RQNA) based on indices of dispersion. The indices of dispersion are scaled variance time curves. They enable the approximations to exploit the dependence in the arrival processes over time to describe the mean workload as a function of the traffic intensity at each station.

After reviewing the indices of dispersion in §2 and the robust queueing approximation for a single queue in §3, we developed the important variability linear equations for the IDCs of the internal arrival processes in §4. We then introduced the extra step of feedback

elimination in §5. These approximations draw heavily on heavy-traffic limits in [51, 52, 54] involving one-dimensional RBM. We put all this together into a full algorithm in §6, developing a simplified version for networks with tree structure in §6.1.

We then evaluated the performance of the new RQNA-IDC by making comparisons with simulations for various examples in §7 and §A. These experiments confirm that RQNA-IDC is remarkably effective. They also point to directions for future research, including developing refined approximations for the flows that exploit multi-dimensional RBM instead of just one-dimensional RBM.

Acknowledgments

This research was supported by NSF grant CMMI 1634133.

References

[1] Abate J, Whitt W (1992) The Fourier-series method for inverting transforms of probability distributions. *Queueing Systems* 10:5–88.

[2] Asmussen S (2003) *Applied Probability and Queues* (New York: Springer), second edition.

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

[4] Bitran GR, Tirupati D (1988) Multiproduct queueing networks with deterministic routing: decompositioon approach and the notion of interference. *Management Science* 34:75–100.

[5] Brumelle S (1971) On the relation between customer averages and time averages in queues. *J. Appl. Prob.* 8(3):508–520.

[6] Budhiraja A, Lee C (2009) Stationary distribution convergence for generalized Jackson networks in heavy traffic. *Mathematics of Operations Research* 34(1):45–56.

[7] Chao X (1995) A queueing network model with catastrophes and product form solution. *Operations Research Letters* 18(2):75–79.

[8] Chen H, Mandelbaum A (1991) Stochastic discrete flow networks: diffusion approximations and bottlenecks. *The Annals of Probability* 19(4):1463–1519.

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

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

[11] Dai JG, Harrison JM (1992) Reflected Brownian motion in an orthant: numerical methods for steady-state analysis. *The Annals of Applied Probability* 65–86.

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

[13] Daley DJ (1976) Queueing output processes. *Adv. Appl. Prob.* 8(2):395–415.

[14] Disney RL, Konig D (1985) Queueing networks: a survey of their random processes. *SIAM Review* 27(3):335–403.

[15] Fendick KW, Saksena V, Whitt W (1989) Dependence in packet queues. *IEEE Trans Commun.* 37:1173–1183.

[16] Fendick KW, Saksena V, Whitt W (1991) Investigating dependence in packet queues with the index of dispersion for work. *IEEE Trans Commun.* 39(8):1231–1244.

[17] Fendick KW, Whitt W (1989) Measurements and approximations to describe the offered traffic and predict the average workload in a single-server queue. *Proceedings of the IEEE* 71(1):171–194.

[18] Fischer W, Meier-Hellstern K (1993) The Markov-modulated Poisson process (MMPP) cookbook. *Performance evaluation* 18(2):149–171.

[19] Gamarnik D, Zeevi A (2006) Validity of heavy traffic steady-state approximations in generalized Jackson networks. *Advances in Applied Probability* 16(1):56–90.

[20] Harrison JM (1973) The heavy traffic approximation for single server queues in series. *Journal of Applied Probability* 10(3):613–629.

[21] Harrison JM (1978) The diffusion approximation for tandem queues in heavy traffic. *Advances in Applied Probability* 10(4):886–905.

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

[23] Harrison JM, Reiman MI (1981) Reflected Brownian motion on an orthant. *The Annals of Probability* 302–308.

[24] Harrison JM, Williams RJ (1987) Multidimensional reflected Brownian motions having exponential stationary distributions. *The Annals of Probability* 115–137.

[25] Harrison JM, Williams RJ (1990) On the quasireversibility of a multiclass Brownian service station. *Annals of Probability* 18(3):1249–1268.

[26] Henderson W, Taylor PG (1990) Product form in networks of queues with batch arrivals and batch services. *Queueing systems* 6(1):71–87.

[27] Iglehart DL, Whitt W (1970) Multiple channel queues in heavy traffic, I. *Advances in Applied Probability* 2(1):150–177.

[28] Iglehart DL, Whitt W (1970) Multiple channel queues in heavy traffic, II: Sequences, networks and batches. *Advances in Applied Probability* 2(2):355–369.

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

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

[31] Kemeny JG, Snell JL (1976) *Finite Markov Chains* (New York: Springer).

[32] Kim S (2011) Modeling cross correlation in three-moment four-parameter decomposition approximation of queueing networks. *Operations Research* 59(2):480–497.

[33] Kim S (2011) The two-moment three-parameter decomposition approximation of queueing networks with exponential residual renewal processes. *Queueing Systems* 68:193–216.

[34] Kim S, Whitt W (2014) Are call center and hospital arrivals well modeled by nonhomogeneous Poisson processes? *Manufacturing and Service Oper. Management* 16(3):464–480.

[35] Neuts MF (1989) *Structured Stochastic Matrices of $M/G/1$ Type and their Application* (New York: Marcel Dekker).

[36] Reiman MI (1984) Open queueing networks in heavy traffic. *Math. Oper. Res.* 9(3):441–458.

[37] Reiman MI (1990) Asymptotically exact decomposition approximations for open queueing networks. *Operations research letters* 9(6):363–370.

[38] Ross SM (1996) *Stochastic Processes* (New York: Wiley), second edition.

[39] Segal M, Whitt W (1989) A queueing network analyzer for manufacturing. Bonatti M, ed., *Teletraffic Science for New Cost-Effective Systems, Networks and ServicesProceedings: ITC 12, Proceedings of the 12th International Teletraffic Congress*, 1146–1152 (Elsevier, North-Holland).

[40] Serfozo R (2012) *Introduction to Stochastic Networks*, volume 44 (Springer Science & Business Media).

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

[42] Sriram K, Whitt W (1986) Characterizing superposition arrival processes in packet multiplexers for voice and data. *IEEE Journal on Selected Areas in Communications* SAC-4(6):833–846.

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

[44] Whitt W (1982) Approximating a point process by a renewal process: two basic methods. *Oper. Res.* 30:125–147.

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

[46] Whitt W (1984) Approximations for departure processes and queues in series. *Naval Research Logistics (NRL)* 31(4):499–521.

[47] Whitt W (1985) Queues with superposition arrival processes in heavy traffic. *Stochastic Processes and Their Applications* 21:81–91.

[48] Whitt W (1991) A review of $L = \lambda W$. *Queueing Systems* 9:235–268.

[49] Whitt W (1995) Variability functions for parametric-decomposition approximations of queueing networks. *Management Science* 41(10):1704–1715.

[50] Whitt W (2002) *Stochastic-Process Limits* (New York: Springer).

[51] Whitt W, You W (2018) Heavy-traffic limit of the $GI/GI/1$ stationary departure process and its variance function, stochastic Systems, published online May 11, 2018, doi:10.1287/stsy.2018.0011.

[52] Whitt W, You W (2018) Heavy-traffic limits for the stationary flows in generalized Jackson networks, in preparation, Columbia University, http://www.columbia.edu/∼ww2040/allpapers.html.

[53] Whitt W, You W (2018) On approximations for the $GI/GI/1$ queue and generalized Jackson open queueing networks using indices of dispersion", Columbia University, http://www.columbia.edu/∼ww2040/allpapers.html.

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

[55] Whitt W, You W (2019) Algorithms to compute the index of dispersion of a stationary point process, in preparation, Columbia University, http://www.columbia.edu/∼ww2040/allpapers.html.

Appendix

In this appendix we provide addtional supporting material. First, in §A we discuss additional numerical experiments. Second, in §B we present supporting technical details. Third, in §C we provide additional heavy-traffic support.

A. Additional Numerical Experiments

A.1. An Important Illustrative Example

An important innovation in this work is in the way we look at the performance of an OQN. On the one hand, we take a limited view, looking only at the mean steady-state workload at each queue in the OQN. However, we look at this mean steady-state workload as a function of the traffic intensity of that queue; i.e., we look at the normalized mean workload $c_Z^2(\rho)$ as a function of ρ. That perspective is achieved by adjusting the mean service time at the queue so that the traffic intensity ρ varies across its full range $0 < \rho < 1$.

Hence, our performance measures are not single mean values but instead a continuum of mean values in the normalized form $c_Z^2(\rho)$ in (4). Thus, we are evaluating not one OQN, but instead a continuum of OQNs. In our experiments we achieve these continua approximately by looking at large finite subsets.

To illustrate how the normalized mean workload $c_Z^2(\rho)$ can vary as a function of ρ and how that behavior can be captured by the IDW $I_w(t)$, we consider the $EHEHE \to M$ example from §EC.8.2 of [54]. This example has 5 single-server queues in series, where the variability increases and then decreases 5 times, with the traffic intensities at successive queues decreasing. That makes the external arrival process and the earlier queues relevant only as the traffic intensity increases. Specifically, the example can be denoted by

$$E_{10}/H_2/1 \to \cdot/E_{10}/1 \to \cdot/H_2/1 \to \cdot/E_{10}/1 \to \cdot/M/1. \tag{49}$$

In particular, the external arrival process is a rate-1 renewal process with Erlang E_{10} interarrival times, thus $c_a^2 = 0.1$. The 1^{st} queue has hyperexponential H_2 service times with mean 0.99 and $c_s^2 = 10$ thus the traffic

intensity at this queue is 0.99. (Throughout this paper, we fix the third H_2 parameter by stipulating that it also has balanced means, as on p. 137 of [44].) The 2^{nd} queue has E_{10} service time with mean and thus traffic intensity 0.98. The 3^{rd} queue has H_2 service times with mean 0.70 and $c_s^2 = 10$. The 4^{th} queue has E_{10} service times with mean and thus traffic intensity 0.5. The last (5^{th}) queue has an exponential service-time distribution. with mean and traffic intensity ρ.

Figure 3 (left) shows the IDW at that last queue over the interval $[10^{-2}, 10^5]$ in log scale, while Figure 3 (right) shows the impact of ρ on the performance of that last queue.

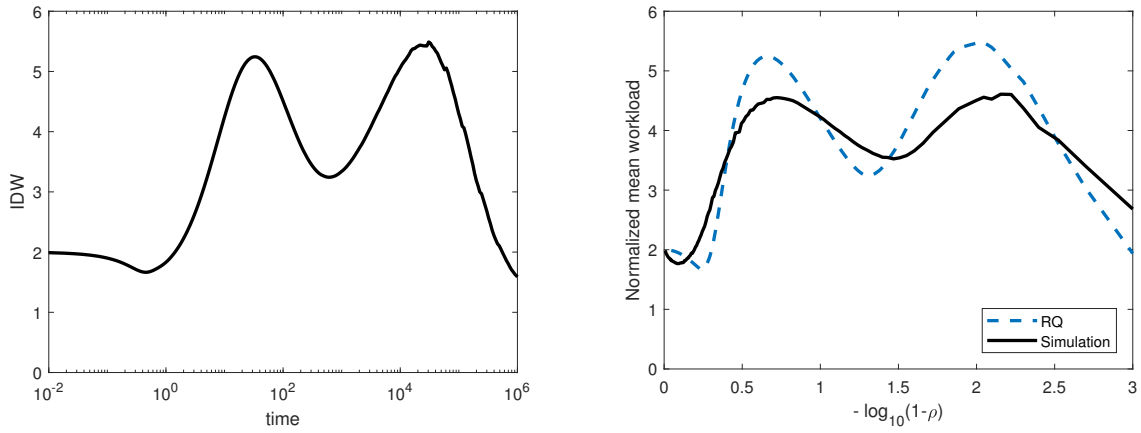

Figure 3 **The IDW at the last queue over the interval $[10^{-2}, 10^5]$ in log scale (left) and a comparison between simulation estimates of the normalized workload $c_Z^2(\rho)$ at the last queue as a function of traffic intensity ρ with the RQ approximation (right).**

Looking backwards starting from the 4^{th} queue, i.e., the queue just before the last queue, the Erlang service act to smooth the arrival process at the last queue. Thus, for sufficiently low traffic intensities ρ at the last queue, the last queue should behave essentially the same as a $E_{10}/M/1$ queue, which has $c_a^2 = 0.1$, but as ρ increases, the arrival process at the last queue should inherit the variability of the previous service times and the external arrival process, and altering between $H_2/M/1$ and $E_{10}/M/1$ as the traffic intensity at the last queue increases. This implies that the normalized workload $c_Z^2(\rho)$ as a function of ρ should have four internal modes, as we see. (If we also count the left and right ends, there are six modes.) Clearly, the IDW has the same qualitative property as the normalized workload as well as the RQ approximation.

This example was carefully designed to expose the complicated dependence structure that can be introduced by a series of queues with different level of variability in service. To a large extent, the scaling can be explained by the heavy-traffic limits which involve a time scaling by $(1-\rho)^{-2}$. Consistent with that HT scaling, our departure process IDC approximation is a time-varying convex combination of the service IDC and arrival IDC, where the weight function for the i-th station includes a time scaling of $(1-\rho_i)^{-2}$ with respect to the traffic intensity ρ_i; e.g., see (22). We obtain good separation for the chosen traffic intensities, because $(1-\rho_i)^{-2} = 10^4$ for $\rho_i = 0.99$, 2.5×10^{-3} for $\rho_i = 0.98$, 11.1 for $\rho_i = 0.7$ and 4.0 for $\rho_i = 0.5$. That is why we use the log scaling for the IDW and IDC throughout this paper.

In closing, we observe that in this example, as in other feed-forward OQNs, we can regard the mean workload at the last queue as a direct function of the IDW at that queue as well as the arrival rate and mean service time. However, there is a more complex relation in OQN's with customer feedback. Then the performance at individual queues should be regarded as a function of the IDC's, arrival rates and mean service times at all the queues. In §6 we develop different RQNA algorithms for the two cases.

A.2. Comparison with RQ: Ten Queues in Series

This example is taken from §5.2 of [54], where we consider 10 single-server queues in series. The external arrival process is a rate-1 renewal process with H_2 interarrival times, having $c_a^2 = 5$. The first 9 queues all have Erlang service times with $c_a^2 = 0.5$ denoted by E_2, i.e., the sum of 2 i.i.d. exponential random variables. The first 8 queues have mean service time and thus traffic intensity 0.6, while the 9^{th} queue has mean service time and thus traffic intensity 0.95. The difference in variability level of the arrival and service process introduces complex variability structure underneath the first 9 queues in series. The 10^{th} queue serves as a test queue and has an exponential service-time distribution with mean and traffic intensity ρ, which is allowed to vary from 0 to 1 in order to expose the complex impact of the variability on the performance measure of the test queue.

The RQNA algorithm in this case is a simple special case of Algorithm 2. The IDC's of the external flows (I_{a_1} for external arrival at station 1 and I_{s_i} service flows) can be derived by explicitly inverse (8), see §III.G of [17]. For internal flows, we apply the departure approximation in (22) recursively, so that for $2 \le i \le 9$,

$$\begin{aligned} I_{d_1}(t) &= w_1 I_{a_1}(t) + (1-w_1) I_{s_1}(t), \quad \text{and} \\ I_{d_i}(t) &= w_i I_{d_{i-1}}(t) + (1-w_i) I_{s_i}(t), \end{aligned} \tag{50}$$

where we used (23) with $h(\rho) = \rho^2$ as in (26) with $\rho_i = 0.6$ for $1 \le i \le 8$, $\rho_9 = 0.95$, $\lambda_i = 1$. For the variability parameters, we note that $c_{x_i}^2 \equiv c_{a_i}^2 + c_{s_i}^2 = c_{a_i}^2 + 0.5$ and that, for $2 \le i \le 9$,

$$c_{a_i}^2 \equiv I_{a_i}(\infty) = I_{d_{i-1}}(\infty) = I_{a_{i-1}}(\infty) = \cdots = I_{a_1}(\infty) = c_{a_1}^2 = 5.$$

With $I_{a_{10}}(t) = I_{d_9}(t)$, we can now apply the RQ algorithm in (13) to obtain approximation of the steady-state mean workload.

Figure 4 reports on two aspects the performance of the RQNA algorithm at the (10^{th}) test queue: (i) the approximation of the IDW, and (ii) the RQNA approximation of the steady-state mean workload. Figure 4 (left) shows that the IDC approximation in the RQNA algorithm performs very well, while Figure 4 (right) shows that both RQ (with directly estimated IDC) and RQNA are accurate, just as for the more complex example in §A.1.

A.3. A Single-Server Queue with i.i.d. Feedback

We start the minimal example with customer feedback, i.e., single-server queue with i.i.d. customer feedback.

In specific, we look at two settings: (1) H_2 external arrival and service distribution, both with balanced mean but $c_a^2 = 6$ and $c_s^2 = 2$, the external arrival rate is set to 1 and the feedback probability is $p = 0.5$; and (2) E_2 external arrival distribution so that $c_a^2 = 1/2$ and H_2 service distribution with balanced mean and $c_s^2 = 6$, the external arrival rate is again 1 but the feedback probability is $p = 0.75$.

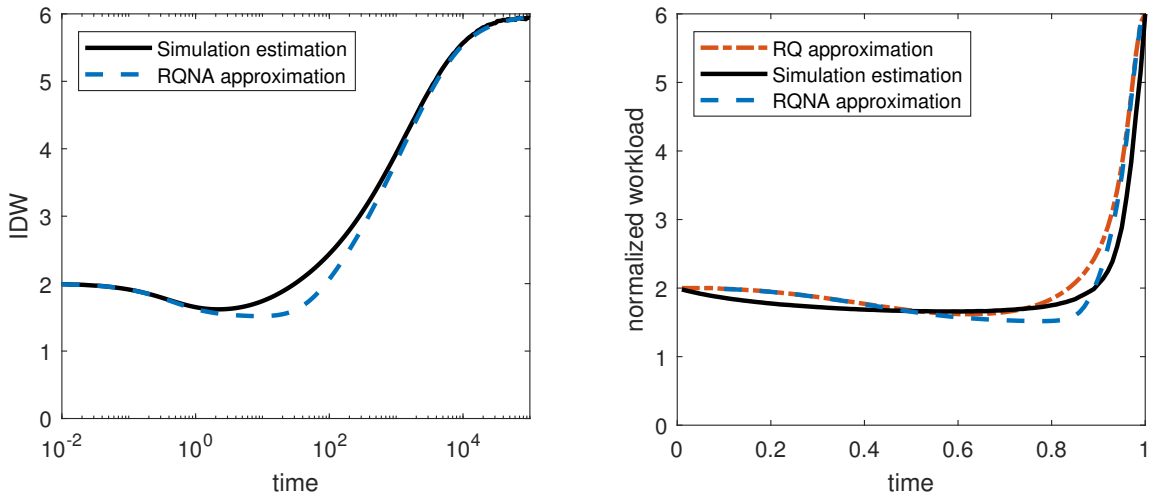

Figure 4 **Contrasting the RQNA approximation of the IDW at the 10-th queue and simulation estimated IDW (left) in the ten queues in series example. Simulation estimation of the steady-state mean workload, the RQ approximation in (13) and the RQNA approximation from Algorithm 2 shown in the right plot.**

To exposed the impact of the traffic intensity on the mean steady-state workload, we allow traffic in tensity to vary in the full range of $(0, 1)$.

Figure 5 reports various robust queueing approximation of the two examples. We observer that feedback elimination produces exact values in the HT limit, however, it does not capture the correct LT limit. On the other hand, the RQ-IDW algorithm, as well as the RQNA-IDC algorithms with suitable tuning function gives exact LT limit, but incorrect HT limit.

A.4. Comparisons with Previous Algorithms for Queues in Series

In this section, we compare the performance of our RQNA algorithm to the performance of QNA from [45], QNET from [22], SBD from [10] and RQ from [54], for the example with 9 queues in series considered by [43]. This example was introduced by [43] to illustrate the heavy-traffic bottleneck phenomenon.

In particular, we consider an OQN with 9 stations in tandem, each with i.i.d. exponential service times. Station 1 has the only external arrival process, which is a rate-1 general renewal process. The traffic intensities at the first 8 queues are set to $\rho_i = 0.6$ for $1 \le i \le 8$, while the last queue has the significantly higher traffic intensity $\rho_9 = 0.9$. As in [43], two specific external renewal arrival processes are considered: (i) deterministic interarrival times with $c_{a_0}^2 = 0$; and (ii) highly variable H_2 interarrival times with $c_{a_0}^2 = 8$ (and again balanced means).

The heavy-traffic bottleneck phenomenon illustrates that the variability of the external arrival process can have only very limited impact on the performance of the following queues, especially after passing through several queues, and yet dramatically affect the performance of a later queue with a much higher traffic intensity. This phenomenon is a result of complicated long-range dependence embedded in the arrival processes, introduced by flowing through a queue (the departure processes), as discussed in §A.1 and revealed by the

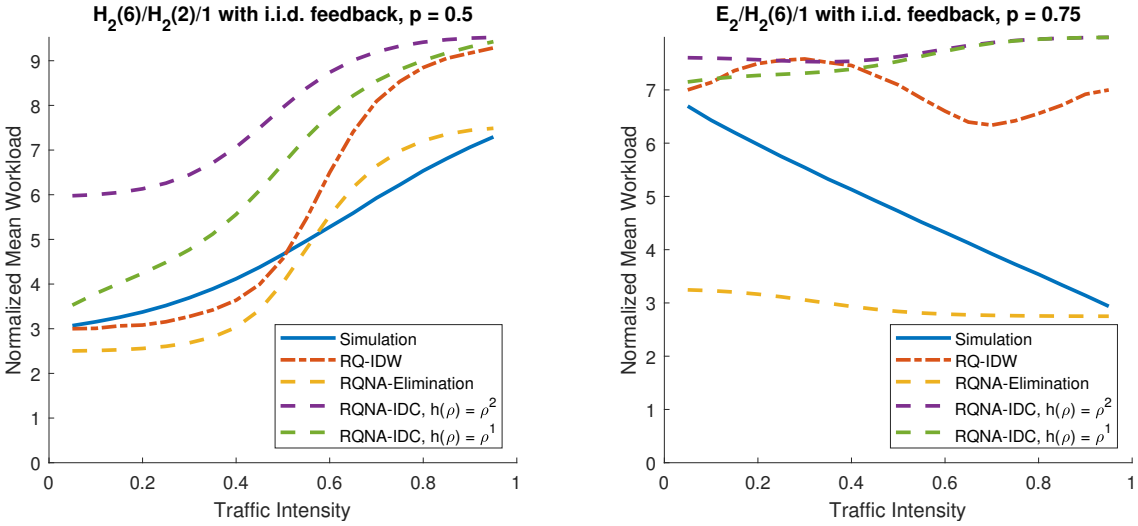

Figure 5 **Contrasting the RQ algorithm with simulated IDW in §3.1, the RQNA algorithm with feedback elimination in §5.1 and the RQNA-IDC algorithm described in §6 with the simulation estimation of the mean steady-state workload, as functions of the traffic intensity ρ. For the RQNA-IDC algorithm, we display results for two different tuning functions $h(\rho)$ as specified in the legend.**

departure approximation in (22). This example was introduced to show the limitation of traditional decomposition methods, e.g. the QNA algorithm, and is often used as a benchmark for different approximation methods, see §3.3 of [10].

Table 6 (for low variability) and Table 7 (for high variability) compare the various approximations of the mean steady-state waiting time at each station, as well as the total waiting time in the system, to simulation estimates.

In the parentheses, we include (i) the relative half-width of the 95% confidence interval for simulation estimates (column Sim); and (ii) the relative error of the approximations compared to the simulation estimates. The first 5 columns in Table 6 and Table 7 are taken directly from Tables VIII and IX of [10], but the simulation and QNA approximations come from [43]. The last three columns are the approximations obtained from the RQNA algorithm discussed in this paper with various choice of the tuning function $h(\rho)$. The RQNA approximations of the workload are transformed into the approximations of the waiting time by (18).

To put these performance measures in perspective, note that in an $M/M/1$ queue with arrival rate 1 we would have $EW = \rho^2/(1-\rho)$, which would be 0.90 at the first 8 queues, but 8.1 at the last queue. For the D arrival process in Table 6, we expect that EW will be smaller; for the the H_2 arrival process in Table 7, we expect EW to be higher, but we see a big impact at the last queue, more than might be expected.

We make the following observations from this experiment:

1. The new RQNA algorithm does better than the QNA and QNET methods on total time spent waiting in queue, and is comparable with the SBD method, even though RQNA does not require solving an RBM.

2. The RQNA algorithm does exceptionally well at the final bottleneck queue and is competitive with all other methods for approximating the mean waiting time. The new RQNA method is based on heavy-traffic

Table 6: A comparison of four approximation methods to simulation for 9 exponential (M) queues in series fed by a deterministic arrival process with $c_a^2 = 0$.

Queue	Sim	QNA	QNET	SBD	RQ	RQNA $h(\rho) = \rho$	RQNA $h(\rho) = \rho^2$	RQNA $h(\rho) = \rho^3$
1	0.290 (2.41%)	0.45 (55%)	0.45 (55%)	0.45 (55%)	0.30 (2.3%)	0.30 (2.3%)	0.30(2.3%)	0.30 (2.3%)
2	0.491 (1.43%)	0.61 (24%)	0.66 (35%)	0.66 (35%)	0.55 (13%)	0.58 (19%)	0.53 (8.1%)	0.48 (-2.8%)
3	0.607 (1.32%)	0.72 (19%)	0.74 (22%)	0.74 (22%)	0.70 (15%)	0.72 (19%)	0.66 (9.4%)	0.60 (-1.1%)
4	0.666 (1.20%)	0.78 (17%)	0.79 (18%)	0.79 (19%)	0.77 (16%)	0.79 (19%)	0.74 (11%)	0.68 (2.1%)
5	0.706 (1.42%)	0.83 (18%)	0.82 (16%)	0.82 (16%)	0.80 (14%)	0.83 (18%)	0.79 (12%)	0.73 (3.9%)
6	0.731 (1.78%)	0.85 (16%)	0.84 (14%)	0.84 (15%)	0.83 (13%)	0.86 (18%)	0.82 (13%)	0.77 (5.7%)
7	0.748 (1.34%)	0.87 (16%)	0.85 (14%)	0.85 (14%)	0.84 (12%)	0.88 (17%)	0.85 (13%)	0.80 (7.2%)
8	0.775 (1.68%)	0.88 (14%)	0.86 (11%)	0.86 (11%)	0.85 (9.2%)	0.89 (15%)	0.86 (11%)	0.82 (6.2%)
9	5.031 (4.31%)	7.99 (59%)	6.97 (39%)	4.05 (-20%)	4.95 (-2.0%)	4.97 (-1.3%)	4.50 (-11%)	4.11 (-18%)
Total	10.05	14.0 (39%)	18.6 (-59%)	10.1 (0.09%)	10.6 (5.3%)	10.8 (7.6%)	10.1 (0.13%)	9.00 (-10%)

Table 7: A comparison of four approximation methods to simulation for 9 exponential (M) queues in series fed by a highly-variable H_2 renewal arrival process with $c_a^2 = 8$.

Queue	Sim	QNA	QNET	SBD	RQ	RQNA $h(\rho) = \rho$	RQNA $h(\rho) = \rho^2$	RQNA $h(\rho) = \rho^3$
1	3.284 (3.50%)	4.05 (23%)	4.05 (23%)	4.05 (23%)	3.95 (20%)	3.95 (20%)	3.95 (20%)	3.95 (20%)
2	2.321 (4.18%)	2.92 (26%)	1.81 (22%)	1.82 (-22%)	2.61 (12%)	1.58 (-32%)	1.95 (-15%)	2.39 (3.0%)
3	1.914 (3.40%)	2.19 (14%)	1.47 (-23%)	1.49 (-22%)	2.04 (6.7%)	0.98 (-49%)	1.07 (-44%)	1.33 (-31%)
4	1.719 (4.07%)	1.73 (0.64%)	1.16 (-33%)	1.19 (-31%)	1.72 (0.31%)	0.92 (-47%)	0.94 (-41%)	0.98 (-43%)
5	1.598 (3.69%)	1.43 (-11%)	1.07 (-33%)	1.10 (-31%)	1.53 (-4.1%)	0.90 (-44%)	0.91 (-43%)	0.93 (-43%)
6	1.478 (4.13%)	1.24 (-16%)	1.03 (-31%)	1.06 (-28%)	1.41 (-4.6%)	0.90 (-39%)	0.90 (-39%)	0.91 (-39%)
7	1.423 (3.23%)	1.12 (-21%)	1.00 (-30%)	1.03 (-28%)	1.33 (-6.8%)	0.90 (-37%)	0.90 (-37%)	0.90 (-37%)
8	1.413 (4.67%)	1.04 (-26%)	0.98 (-30%)	1.01 (-29%)	1.27 (-10%)	0.90 (-36%)	0.90 (-36%)	0.90 (-36%)
9	30.12 (16.8%)	8.90 (-71%)	6.04 (-80%)	36.5 (21%)	36.9 (23%)	29.1 (-3.5%)	32.8 (9.0%)	35.3 (17%)
Total	45.27	24.6 (-46%)	18.6 (-59%)	49.8 (10%)	52.8 (17%)	40.1 (-11%)	44.4 (-2.0%)	47.6 (5.1%)

limits just as the previous methods methods, but focuses on the flows, and exploits RQ instead of analyzing an RBM.

3. The RQNA algorithm can benefit from further improvement for light-to-medium traffic intensities. As demonstrated in Table 7, the mean waiting times at queues 3-8 are pushed too much towards the $M/M/1$ values in the departure IDC approximation for light to medium traffic intensity. That remains to be a direction for future research.

B. Supporting Technical Details

In this section we provide theoretical support for our algorithm to estimate the IDC from data in §2.3.

We now review Theorem 2 from [55], which states that the estimator of the of the variance function $V(t)$ is asymptotically consistent under mild regularity conditions that $V(t)$ is differentiable with derivative $\dot{V}(t)$ having finite positive limits as $t \to \infty$, i.e.,

$$\dot{V}(t) \to \sigma^2 \text{ as } t \to \infty,$$

for an appropriate constant σ^2. This condition is also used in §3.3 of [54].

THEOREM 2 **(Consistency of the estimator)**. *Let A be a time-stationary and ergodic point process with variance function $V(t)$ that is differentiable with derivative $\dot{V}(t)$ having finite positive limit as $t \to \infty$, i.e.,*

$$\dot{V}(t) \to \sigma^2 \text{ as } t \to \infty.$$

Then we have

$$\lim_{l\to\infty} bias(\bar{V}_l) = 0$$

for $l = rk - r + 1$, $r = t/\tau, k = T/t$ and $\bar{V}_k$ is the sample variance of $\{U_i\}_{i=1}^k$. Furthermore,

$$\lim_{l\to\infty} \bar{V}_l = V(t), \text{ w.p.1.}$$

Proof. Let $K = rk - r + 1$ be the sample size, and assume that $V(t) = I(t)t < Ct$ for some constant C. Then

$$\begin{aligned}
E\left[\bar{V}\right] &= \frac{1}{K-1}\sum_{i=1}^{K} E\left[U_i^2\right] - \frac{1}{K(K-1)} E\left[\left(\sum_{i=1}^{K} U_i\right)^2\right] \\
&= \frac{1}{K-1}\left(\sum_{i=1}^{K} E\left[U_i^2\right] - \frac{1}{K} E\left[\sum_{i=1}^{K} U_i^2 + 2\sum_{i>j} U_i U_j\right]\right) \\
&= E\left[U_1^2\right] - E\left[U_1\right]^2 - \frac{2}{K(K-1)}\sum_{i<j} \mathrm{cov}(U_i, U_j) \\
&= V(t) - \frac{2}{K(K-1)}\left(\sum_{j<i<j+r} \mathrm{cov}(U_i, U_j) + \sum_{i>j+r+1} \mathrm{cov}(U_i, U_j)\right) \\
&= V(t) - \frac{2}{K(K-1)}\left(\sum_{i=1}^{r-1}(K-i)\mathrm{cov}(U_1, U_{i+1}) + \sum_{i=r}^{K-1}(K-i)\mathrm{cov}(U_1, U_{i+1})\right) \\
&\equiv V(t) - (A+B)
\end{aligned}$$

The covariance terms can be expressed as

$$\mathrm{cov}(U_1, U_{1+i}) = \begin{cases} V(t-i\tau)+V(t+i\tau)-V(t)-V(i\tau), & i=1,2,\ldots,r-1 \\ V(t+i\tau)-2V(i\tau)+V(i\tau-t), & i=r,r+1,\ldots,K-1 \end{cases} \tag{51}$$

Using the bound on $I(t)$, we have

$$\begin{aligned} A &= \frac{2}{K(K-1)} \sum_{i=1}^{r-1} (K-i)\mathrm{cov}(U_1, U_{i+1}) \\ &\le \frac{2}{K} \sum_{i=1}^{r-1} (V(t-i\tau)+V(t+i\tau)) \\ &\le \frac{4Ct(r-1)}{K} \le \frac{4Ct}{k-1}, \end{aligned}$$

and

$$\begin{aligned} B &= \frac{2}{K(K-1)} \sum_{i=r}^{K-1} (K-i)\mathrm{cov}(U_1, U_{i+1}) \\ &\le \frac{2}{K} \sum_{i=r}^{K-1} ((V(t+i\tau)-V(i\tau))-(V(i\tau)-V(i\tau-t))) \\ &\le \frac{2t}{K} \sum_{i=r}^{K-1} \left(\frac{V(t+i\tau)-V(i\tau)}{t} - \frac{V(i\tau)-V(i\tau-t)}{t} \right) \\ &\to 0, \text{ as } k \to \infty, \end{aligned}$$

where we used the regularity condition that $\dot{V}(t) \to \sigma^2$ as $t \to \infty$, and the fact that the average converges to 0 if the summands converge to 0.

Note that

$$\bar{V}_k \equiv \frac{1}{k-1} \sum_{i=1}^{k} U_i^2 - \frac{1}{k(k-1)} \left(\sum_{i=1}^{k} U_i \right)^2$$

By Continuous Mapping Theorem, we need only prove that both $\{U_i\}$ and $\{U_i^2\}$ follows Strong Law of Large Number (SLLN). This in turns is implied by the Strong Ergodic Theorem for stationary and ergodic sequence. The stationarity of both sequences are implied by the time-stationarity of the point process $N(t)$. The ergodicity of both sequence follows from the ergodicity of the underlying process $N(t)$. □

C. Additional Heavy Traffic Results

In this section, we provide detailed HT limits.

C.1. Heavy-Traffic Limits for Departure Processes

We first provide theoretical support for the approximation (22) of the departure IDC. That approximation is ultimately supported by the heavy-traffic limit theorem obtained in Corollary 4.2 of [52]. To use that result, we start by presenting a slight variant of it. We refer to §3.2 of [52] for the notations used here.

LEMMA 1. *Under the assumption of Corollary 4.2 of* [52]*, the HT limit of the departure process of the bottleneck station h can be written as*

$$D_h^* = \tilde{Q}_h^*(0) + \tilde{A}_h^* - \tilde{Q}_h^*, \tag{52}$$

where

$$\tilde{A}_h^* = e_h'(I - P')^{-1}\left(A_0^* + (\Theta^*)'\mathbf{1}\right) \tag{53}$$

and

$$\tilde{Q}_h^* = \frac{1}{1-\hat{P}_h} Q_h^* = \psi\left(\tilde{Q}_h^*(0) + \tilde{A}_h^* - S_h^* - \lambda_h e\right). \tag{54}$$

As a result, the limiting variance function of the departure process is where

$$V_{d,h}^*(t) = w^*(\lambda_h t/c_{x,h}^2)c_{a,h}^2\lambda_h t + (1 - w^*(\lambda_h t/c_{x,h}^2))c_{s,h}^2\lambda_h t, \tag{55}$$

where $w^*(t)$ is the weight function in (24)*. The variability parameter is $c_{x,h}^2 = c_{a,h}^2 + c_{s,h}^2$ with $c_{s,h}^2$ being the service scv and $c_{a,h}^2$ being the limiting variability of the total arrival at station h, given by $c_{a,h}^2 \equiv Var(\tilde{A}_h^*)/\lambda_h t$.*

Proof. Start by claiming that

$$e_h'\hat{P}'_{\mathcal{H}^c,\mathcal{H}}e_h = \frac{1}{1-\hat{P}_h}, \quad \lambda_h = \frac{\hat{\lambda}_{0,h}}{1-\hat{P}_h}$$

and that

$$\frac{1}{1-\hat{P}_h}\left(e_h' + \hat{P}'_{\mathcal{H}^c,\mathcal{H}}e'_{\mathcal{H}^c}\right) = e_h'(I-P')^{-1}.$$

In fact, all three assertions can be check by writing the transition matrix in blocks according to two sets of indices $\{\mathcal{H}, \mathcal{H}^c\}$.

Now, (54) follows from dividing both sides of the limiting queue length process in Corollary 4.2 of [52] by $(1-\hat{P}_h')$ and the fact that $\psi(f/c) = \psi(f)/c$ for any function f and constant c.

The limiting variance function is derived in the exact same way as in Theorem 5.3 of [51] by noting that $\tilde{A}_h^*$ and S_h^* are two independent Brownian motions. The only change here is that we have an additional tuning function $h(\rho)$. This, however, does not change the argument, since we require that $\lim_{\rho\uparrow 1} h(\rho) = 1$. □

The approximation (22) is then justified by the exact same procedure as described in §6.2 of [51].

C.2. Heavy-Traffic Limits for Splitting

We now provide additional theoretical support for the splitting approximation in S 4.3.2. For that purpose, let

$$\Theta_i(n) \equiv (\Theta_{i,1}(n), \ldots, \Theta_{i,K}(n)) = \sum_{l=1}^{n}\theta_i^l$$

denote the splitting decisions up to the n-th decision at station i. Consider the diffusion-scaled processes indexed by ρ

$$\begin{aligned} D_{i,\rho}^*(t) &= (1-\rho)\left[D_i((1-\rho)^{-2}t) - \lambda_i(1-\rho)^{-2}t\right], \\ \Theta_{i,\rho}^*(t) &= (1-\rho)\left[\sum_{l=1}^{\lfloor(1-\rho)^{-2}t\rfloor}\theta^l - \mathbf{p}_i(1-\rho)^{-2}t\right] \in \mathcal{D}^K, \\ \mathbf{A}_{i,\rho}^*(t) &= (1-\rho)\left[\mathbf{A}_i((1-\rho)^{-2}t) - \lambda_i\mathbf{p}_i(1-\rho)^{-2}t\right] \in \mathcal{D}^K, \end{aligned} \tag{56}$$

for $t \geq 0$, where $\mathbf{p}_i \equiv E[\theta_i^l]$ is the i-th row of the routing matrix and $\mathbf{A}_{i,\rho} = (A_{i,j,\rho} : j = 1, 2, \ldots, K)$ is the vector consists of all the streams after splitting. The following result rephrases Theorem 9.5.1 in Whitt (2002).

THEOREM 3. (*Theorem 9.5.1 of [50]*) *Suppose that*

$$(D^*_{i,\rho}, \Theta^*_{i,\rho}) \Rightarrow (D^*_i, \Theta^*_i) \quad as \quad \rho \uparrow 1 \quad in \quad D^{K+1} \tag{57}$$

and that almost surely D^* *and* $\Theta^* \circ \lambda \mathbf{e}$ *have no common discontinuities of opposite sign. Then*

$$\mathbf{A}^*_{i,\rho} \Rightarrow \mathbf{A}^*_i \quad in \quad D^K,$$

with

$$A^*_{i,j} \equiv p_{i,j} D^* + \Theta^*_{i,j} \circ \lambda_i e, \quad for \quad 1 \le j \le K, \tag{58}$$

where $e(t) = t$ *is the identity mapping.*

REMARK 6. (splitting the departures from a $G/GI/1$ queue) If we split the departure process from the $GI/GI/1$ model with Markovian routing, then D^* is independent of Θ^* and Θ^* is a zero-drift K-dimensional Brownian motion with covariance matrix $\Sigma = (\sigma_{i,j}) \in \mathbb{R}^{K \times K}$, where $\sigma^2_{i,i} = p_i(1-p_i)$ and $\sigma^2_{i,j} = -p_i p_j$ for $i \neq j$. Hence, from (58) we obtain

$$\mathbf{A}^* = \mathbf{p} D^* + \Theta^* \circ \lambda e, \tag{59}$$

which is consistent with (30) and thus (31). $\square$

Theorem 3 assumes only a joint FCLT for the flow to split and the splitting decision process, so dependence is allowed. Thus it provides support for the general splitting equation in (32) and (33) for the case where $D_{i,j}$ and $\Theta_{i,j}$ are correlated. Furthermore, define the HT-scaled correction term as

$$\alpha^*_{i,j,\rho}(t) \equiv \alpha_{i,j}((1-\rho)^{-2}t). \tag{60}$$

Finally, define the limiting correction term as

$$\alpha^*_{i,j}(t) \equiv 2\mathrm{cov}(p_{i,j} D^*_i(t), \Theta^*_{i,j}(\lambda_i t))/p_{i,j}\lambda_i t. \tag{61}$$

The following corollary follows from Theorem 3.

COROLLARY 1. *Under the assumptions in Theorem 3 plus the uniform integrability conditions, we have* $\alpha^*_{i,j,\rho}(t) \Rightarrow \alpha^*_{i,j}(t)$ *as* $\rho \uparrow 1$.

Proof. By the definitions of the correction term in (33) and HT-scaled processes, we write

$$\begin{aligned}
\alpha^*_{i,j,\rho}(t) &= \alpha_{i,j}((1-\rho)^{-2}t) \\
&= I_{a,i,j}((1-\rho)^{-2}t) - p_{i,j} I_{d,i}((1-\rho)^{-2}t) - (1-p_{i,j}) \\
&= \frac{\mathrm{Var}((1-\rho)A_{i,j}((1-\rho)^{-2}t))}{p_{i,j}\lambda_i t} - p_{i,j}\frac{\mathrm{Var}((1-\rho)D_i((1-\rho)^{-2}t))}{\lambda_i t} - (1-p_{i,j}) \\
&= \frac{\mathrm{Var}(A^*_{i,j,\rho}(t))}{p_{i,j}\lambda_i t} - p_{i,j}\frac{\mathrm{Var}(D^*_{i,\rho}(t))}{\lambda_i t} - (1-p_{i,j}) \\
&\Rightarrow \frac{\mathrm{Var}(A^*_{i,j}(t))}{p_{i,j}\lambda_i t} - p_{i,j}\frac{\mathrm{Var}(D^*_i(t))}{\lambda_i t} - (1-p_{i,j}) = \alpha^*_{i,j}(t). \quad \square
\end{aligned}$$

This corollary supports the following approximation for the correction term $\alpha_{i,j}$ in

$$\alpha_{i,j}(t) \approx \alpha^*_{i,j}((1-\rho)^2 t) \tag{62}$$

with $\alpha^*_{i,j}$ defined in (61).

C.3. An Approximation Scheme for General Correction Terms

In a general open queueing network with feedback and superposition of dependent flows, the correction terms $\alpha_{i,j}$ and β_i can be non-trivial. The key idea is that, for each correction term, we select a suitable queue and assume it to be the bottleneck queue. Then we apply Corollary 4.2 of [52] to obtain HT approximation of the correction terms and utilize Corollary 5.1 of [51] to obtain explicit form of the correction term. We now discuss the two types of correction terms in turn.

C.3.1. Dependent Splitting: the Correction Term $\alpha_{i,j}$ Unfortunately, the covariance in (61) is complicated. We do obtain a useful approximation under the extra condition that only queue i enters heavy traffic.

For any $\alpha_{i,j}$, the relevant routing flow is $A_{i,j}$ while the relevant departure flow is D_i. Naturally, we choose station i to be the HT station. So we let $\rho_i = \rho \uparrow 1$ and keep $\rho_j < 1$ for $j \neq i$. Define the HT scaled processes as in §3.2 of [52] and apply Lemma 1 with $h = i$, we have

$$D^*_{i,\rho} \Rightarrow D^*_i = \tilde{A}^*_i + \tilde{Q}^*_i(0) - \tilde{Q}^*_i. \tag{63}$$

For the routing flow $A_{i,j}$, we apply Theorem 3 so that

$$A^*_{i,j,\rho} \Rightarrow A^*_{i,j} = p_{i,j} D^*_i + \Theta_{i,j} \circ \lambda_i e \quad \text{as} \quad \rho \uparrow 1. \tag{64}$$

Define the correction term $\alpha^*_{i,j}$ as in (62), then Corollary 4.2 of [52] implies the following corollary, which leads to the correction term in (34).

THEOREM 4. *Under the assumptions in Corollary 4.2 of [52] and Theorem* 3 *plus the uniform integrability conditions, we have*

$$\begin{aligned} \alpha^*_{i,j,\rho}(t) &\Rightarrow 2\text{cov}(p_{i,j} D^*_i(t), \Theta^*_{i,j}(\lambda_i t))/(p_{i,j}\lambda_i t) \\ &= 2\xi_{i,j} p_{i,j}(1-p_{i,j}) w^*(\lambda_i t / c^2_{x,i}), \quad \textit{as} \quad \rho \uparrow 1, \end{aligned} \tag{65}$$

where $\xi_{i,j}$ is the $(i,j)^{\text{th}}$ entry of the matrix $(I - P')^{-1}$, $c^2_{x,i} = c^2_{a,i} + c^2_{s,i}$ and $c^2_{a,i}$ is the limiting variability parameter as solved from (40) *and $c^2_{s,i}$ is the scv of the service distribution at station i.*

Proof. Apply Corollary 4.2 of [52] to obtain expression for $D^*_i(t)$, then apply Corollary 5.1 of [51] for the explicit covariance in (65). □

As a direct result of Theorem 4, we propose to define the correction term as

$$\alpha_{i,j,\rho}(t) = 2\xi_{i,j} p_{i,j}(1-p_{i,j}) w^*((1-\rho)^{-2}\lambda_i t/(\rho c^2_{x,i})), \tag{66}$$

which is asymptotically exact as $\rho \uparrow 1$.

C.3.2. Dependent Superposition: the Correction Term β_i Next, we consider the correction term β_i associated with dependent superposition. From (36), it suffices to specify $\beta_{k,i;j,i}$ for any station i and any pair of sub-flows $(A_{j,i}, A_{k,i})$ at that station. We assume without loss of generality that (i) $\rho_j \geq \rho_k$, or (ii) $\rho_j = \rho_k$ and $\lambda_{j,i} \geq \lambda_{k,i}$. In the case (ii), we break the tie by picking the index that gives the larger rate $\lambda_{j,i}$. In both cases, we consider station j to be the HT station while keep all other stations unsaturated.

By Corollary 4.2 of [52], we have

$$\begin{aligned} A^*_\rho \Rightarrow A^* &= \tilde{A}^* + \gamma_j \left(\tilde{Q}^*_j(0) - \tilde{Q}^*_j\right) \\ D^*_{j,\rho} \Rightarrow D^*_j &= \tilde{A}^*_j + \tilde{Q}^*_j(0) - \tilde{Q}^*_j, \\ D^*_{l,\rho} \Rightarrow D^*_l &= A^*_l, \quad \text{for} \quad l \neq j, \end{aligned}$$

where

$$\tilde{A}^* = (I - P')^{-1} \left(A^*_0 + (\Theta^*)'\mathbf{1}\right),$$

$\tilde{Q}^*_j$ is defined in Lemma 1 with $h = j$ and $\gamma_j \in \mathbb{R}^K$ is defined as

$$\gamma_j = P'(I - P')^{-1} e'_j (1 - \hat{P}_j)$$

with $\hat{P}_j$ defined as in (3.9) of [52] with $\mathcal{H} = \{j\}$.

Furthermore, Theorem 3 gives

$$\begin{aligned} A^*_{j,i} &= p_{j,i} D^*_j + \Theta^*_{j,i} \circ \lambda_j e \\ &= p_{j,i} \tilde{A}^*_j + \Theta^*_{j,i} \circ \lambda_j e + p_{j,i} (\tilde{Q}^*_j(0) - \tilde{Q}^*_j) \qquad (67) \\ A^*_{k,i} &= p_{k,i} D^*_k + \Theta^*_{k,i} \circ \lambda_k e \\ &= p_{k,i} \tilde{A}^*_k + \Theta^*_{k,i} \circ \lambda_k e + p_{k,i} \gamma_{j,k} (\tilde{Q}^*_j(0) - \tilde{Q}^*_j). \qquad (68) \end{aligned}$$

We utilize the following approximations

$$A^*_{k,i} \approx p_{k,i} \tilde{A}^*_k + \Theta^*_{k,i} \circ \lambda_k e \equiv \tilde{A}^*_{k,i} \qquad (69)$$

and

$$p_{j,i} \tilde{Q}^*_j \approx \psi \left(p_{j,i} \tilde{Q}_j(0) + p_{j,i} A^*_j + \Theta^*_{j,i} \circ \lambda_j e - p_{j,i} S^*_j - p_{j,i} \lambda_j e\right) \equiv \tilde{Q}^*_{j,i}. \qquad (70)$$

By Corollary 5.1 of [51]

$$2\text{cov}\left(\tilde{A}^*_{k,i}(t), \tilde{A}^*_{j,i}(t) - \tilde{Q}^*_{j,i}(t)\right) / (\lambda_i t) = 2 \frac{\zeta_{j,i;k,i}}{\lambda_i} w^*(t/c^2_{x,j}), \qquad (71)$$

where $\tilde{A}^*_{j,i} \equiv p_{j,i} \tilde{A}^*_j + \Theta^*_{j,i} \circ \lambda_j e$ and $\zeta_{j,i;k,i}$ is the constant defined as

$$\zeta_{j,i;k,i} = \frac{1}{t} \text{cov}\left(\tilde{A}^*_{k,i}(t), \tilde{A}^*_{j,i}(t)\right). \qquad (72)$$

Note that $\zeta_{j,i;k,i}$ is a constant independent of t since $\tilde{A}^*_{k,i}(t)$ and $\tilde{A}^*_{j,i}(t)$ are Brownian motions.

Finally, we define

$$\beta_{j,i;k,i}(t) = \beta_{k,i;j,i}(t) = 2 \frac{\zeta_{j,i;k,i}}{\lambda_i} w^*((1 - \rho_j)^2 p_{j,i} \lambda_j t / (\rho c^2_{x,j,i})), \qquad (73)$$

where $c^2_{x,j,i} = p_{j,i} c^2_{a,j} + (1 - p_{j,i}) + p_{j,i} c^2_{s,j}$ and $c^2_{a,j}$ is solved from (44).

The following lemma gives explicit formula for $\zeta_{j,i;k,i}$. Let $\nu_l \equiv p_{l,i} e'_l (I - P')^{-1}$ for $l = j, k$, where e_i is the i-th unit vector.

LEMMA 2.

$$\zeta_{j,i;k,i} = \nu_j' \left(\text{diag}(c_{a,0,i}^2 \lambda_i) + \sum_{l=1}^{K} \Sigma_l \right) \nu_k + \nu_k' \Sigma_j e_i + \nu_j' \Sigma_k e_i, \tag{74}$$

where $\text{diag}(c_{a,0,i}^2 \lambda_i)$ *is the diagonal matrix with* $c_{a_{0,i}}^2 \lambda_i$ *as the* i*-th diagonal entry,* Σ_l *is the covaraince matrix of Brownian limit of the splitting decision process* $(\Theta_{l,i}^*)_{i=1}^K$ *at station* l *defined as* $\Sigma_l \equiv (\sigma_{i,j}^l)$ *with* $\sigma_{i,i}^l = p_{l,i}(1-p_{l,i})\lambda_l$ *and* $\sigma_{i,j}^l = -p_{l,i}p_{l,j}\lambda_l$ *for* $i \neq j$.

Proof. By the definition of $\tilde{A}^*$ and $\tilde{A}_{j,i}^*$, we have

$$\begin{aligned}
\tilde{A}_{j,i}^* \equiv p_{j,i}\tilde{A}_j^* + \Theta_{j,i}^* &= p_{j,i} e_j'(I - P')^{-1}(A_0^* + (\Theta^*)'\mathbf{1}) + \Theta_{j,i}^* \\
&= \nu_j \left(A_0^* + \sum_{l=1}^{K} \Theta_l^* \right) + e_i' \Theta_j^*, \\
\tilde{A}_{k,i}^* \equiv p_{k,i}\tilde{A}_k^* + \Theta_{k,i}^* &= p_{k,i} e_k'(I - P')^{-1}(A_0^* + (\Theta^*)'\mathbf{1}) + \Theta_{k,i}^*, \\
&= \nu_k \left(A_0^* + \sum_{l=1}^{K} \Theta_l^* \right) + e_i' \Theta_k^*,
\end{aligned}$$

where A_0^* is the Brownian limit of the external arrival processes, i.e., $A_{0,i}^* \stackrel{d}{=} c_{a_{0,i}} B_{a_{0,i}} \circ \lambda_i e$ and $\Theta^* \equiv (\Theta_1^*, \ldots, \Theta_K^*)' \in \mathbb{R}^{K \times K}$ with $\Theta_i^* = (\Theta_{i,1}^*, \ldots, \Theta_{i,K}^*)$. Recall that Θ_i^* is the the collection of the Brownian limits of the decision processes at station i, so that

$$\text{cov}(\Theta_{i,j}^*, \Theta_{i,k}^*) = \begin{cases} p_{i,j}(1-p_{i,j})\lambda_i t, & j = k, \\ -p_{i,j}p_{i,k}\lambda_i t, & j \neq k. \end{cases}$$

Define

$$\Sigma_i \equiv \left(\text{cov}(\Theta_{i,j}^*, \Theta_{i,k}^*)/t \right)_{j,k=1}^K \in \mathbb{R}^{K \times K}$$

so that Σ_i is a constant matrix independent of t.

Notice that $A_{0,i}^*$, Θ_j^* for $1 \leq i, j \leq K$ are mutually independent, we have

$$\begin{aligned}
\zeta_{j,i;k,i} &\equiv \frac{1}{t} \text{cov}\left(\tilde{A}_{k,i}^*(t), \tilde{A}_{j,i}^*(t) \right) \\
&= \frac{1}{t} \text{cov}\left(\nu_j A_0^* + \sum_{l=1}^{K} \left(\nu_j + \delta_{l,j} e_i' \right) \Theta_l^*, \nu_k A_0^* + \sum_{l=1}^{K} \left(\nu_k + \delta_{l,k} e_i' \right) \Theta_l^* \right) \\
&= \frac{1}{t} \text{cov}\left(\nu_j A_0^*, \nu_k A_0^* \right) + \frac{1}{t} \sum_{l=1}^{K} \text{cov}\left(\left(\nu_j + \delta_{l,j} e_i' \right) \Theta_l^*, \left(\nu_k + \delta_{l,k} e_i' \right) \Theta_l^* \right) \\
&= \nu_j' \left(\text{diag}(c_{a_{0,i}}^2 \lambda_i) + \sum_{l=1}^{K} \Sigma_l \right) \nu_k + \nu_k' \Sigma_j e_i + \nu_j' \Sigma_k e_i. \quad \square
\end{aligned}$$