Research Article

A Fluid Model for a Relay Node in an Ad Hoc Network: Evaluation of Resource Sharing Policies

Michel Mandjes[1,2,3] and Werner Scheinhardt[2,4]

[1] *Korteweg-de Vries Institute, University of Amsterdam, Plantage Muidergracht 24, 1018 Amsterdam, The Netherlands*
[2] *CWI, P.O. Box 94079, 1090 Amsterdam, The Netherlands*
[3] *EURANDOM, Eindhoven, The Netherlands*
[4] *University of Twente, P.O. Box 217, 7500, Enschede, The Netherlands*

Correspondence should be addressed to Michel Mandjes, mmandjes@science.uva.nl

Fluid queues offer a natural framework for analyzing waiting times in a relay node of an ad hoc network. Because of the resource sharing policy applied, the input and output of these queues are coupled. More specifically, when there are n users who wish to transmit data through a specific node, each of them obtains a share $1/(n + \text{w})$ of the service capacity to feed traffic into the queue of the node, whereas the remaining fraction $\text{w}/(n + \text{w})$ is used to serve the queue; here $\text{w} > 0$ is a free design parameter. Assume now that jobs arrive at the relay node according to a Poisson process, and that they bring along exponentially distributed amounts of data. The case $\text{w} = 1$ has been addressed before; the present paper focuses on the intrinsically harder case $\text{w} > 1$, that is, policies that give more weight to serving the queue. Four performance metrics are considered: (i) the stationary workload of the queue, (ii) the queueing delay, that is, the delay of a "packet" (a fluid particle) that arrives at an arbitrary point in time, (iii) the flow transfer delay, (iv) the sojourn time, that is, the flow transfer time increased by the time it takes before the last fluid particle of the flow is served. We explicitly compute the Laplace transforms of these random variables.

1. Introduction

Ad hoc networks are self-configuring networks of mobile routers, connected by wireless links. They enable infrastructure-free communication: no fixed equipment is needed, but instead each client acts as a hub. When information needs to be transmitted across the network, it is sent from the sender to the receiver by relaying the packets along the intermediate hubs. An excellent survey on ad hoc networks, with special emphasis on quality-of-service aspects, is [1].

On a somewhat more abstract level the nodes in ad hoc networks can be regarded as *queues*: information packets arrive and are relayed, and when the arrival rate (temporarily) exceeds the departure rate, the buffer content of the queue builds up. These queues, however, have the interesting modelling feature that the available transmission capacity at any specific node is used both to (i) "pull" information packets from the "predecessor hubs" into the queue and (ii) "push" information packets from the queue towards "successor hubs" (and eventually the destination client).

Now, consider the situation that at some point in time there are n stations that transmit traffic via the same relay node. If the standard resource sharing policy is used, then the relay-node is assigned a share $1/(n+1)$ of the available medium capacity, which is the same fraction as is allocated to each of the n "sending nodes." In other words, as soon as $n > 1$, the node's input rate exceeds its output rate, and hence the excess traffic accumulates in the node's buffer; only when $n = 0$ the queue drains. This explains why relay nodes are prone to becoming bottlenecks.

The above observations have motivated the development of alternative resource sharing policies, that assign more "weight" to serving the relay node; see for instance [2] and references therein. With the so-called *enhanced distributed channel access* (EDCA) protocol, it will be possible to set a parameter W > 0, such that each of the n sending nodes obtains a fraction $1/(n + W)$ of the capacity, while the remaining W$/(n + W)$ is allocated to serving the queue. Clearly, when W > 1 this has a benign effect on the buffer content of the queue, compared to the standard resource sharing policy described above: the queue drains for all $n <$ W (rather than just for $n = 0$). The price paid is that the traffic remains longer at the sending nodes.

There is one important modelling feature, that applies to the case W > 1, that needs to be mentioned here. When the queue is empty and the number of sending nodes n is below W, it does not make sense to assign each of the sending notes just a share $1/(n + W)$: it would imply that the (available) service rate is strictly larger than the input rate, and that a fraction $(W - n)/(n + W) > 0$ is left unused. For that reason, the EDCA protocol (IEEE 802.11E) was augmented with an "idle mode": if the queue is idle and $n <$ W, then half of the capacity is allocated to serving the queue of the relay node, whereas the other half is shared equally among the n sending nodes (such that the input and output rates are equal, the queue remains empty, and all available capacity is used). Notice that when W ≤ 1 this special rate allocation (during periods in which the buffer is empty) is not required, as it cannot be that both the buffer is empty and that there are $n <$ W jobs in the system.

The case W $= 1$, corresponding to the standard resource sharing policy, was proposed and analysed in detail in [3, 4]. Seen from a queueing-theoretic perspective, this is a model "with coupled input and output," in that the capacity is shared between the input and output process. It was assumed that flows (or jobs) arrive at the relay node according to a Poisson process and initiate a data transfer. For the special case of exponentially distributed flow sizes, [4] explicitly gave the Laplace transforms (and tail asymptotics) of several performance measures.

Importantly, the case W $= 1$ (and in fact also W < 1) has nice features that are lost for W > 1. As a consequence, the analysis of [4] for W $= 1$ does not carry over to W > 1. The main differences are the following.

(1) In the first place, as we explained, the case W $= 1$ does not require the "idle mode" rate allocation that was introduced above (during periods in which the buffer is empty). As a consequence, the number of flows present evolves independently of the buffer content process

(and hence the distribution of the number of flows present can be computed independently of the distribution of the buffer content of the queue). This nice property is lost in the case w > 1; one could say that there is then some sort of "feedback" from the buffer content to the flows, in the sense that the buffer content has impact on the flow behaviour, and hence the distributions of the number of sources present and the buffer content cannot be determined separately.

(ii) In the second place, suppose we wish to analyse the flow transfer delay, defined as the time between the arrival of the flow and the epoch that its last fluid particle has been transmitted into the queue. We know that for w = 1 the queue cannot become empty during this flow transfer. Therefore, all the "state information" that we have to keep track of is the number of flows present when entering (and *not* the buffer content); for w > 1 the queue *can* become empty, and therefore we have to take into account the buffer content, as seen upon arrival, as well.

(iii) In case w = 1, the buffer content decreases only during periods that there are no flows in the system, and these intervals are exponentially distributed; it turned out in [4] that this property entails that a direct translation is possible in terms of a related classical M/G/1 queueing model. For w > 1, the periods of net output are not exponentially distributed, and therefore we lose this nice feature.

The above explains that the analysis for w > 1 is considerably more involved than for w = 1.

For various values of w, the model described above has been extensively validated in [2], by ad hoc network simulations that include all the details of the widely used IEEE 802.11 MAC-protocol. As indicated above, the alternative resource sharing policies can be enforced in real systems by deploying the recently standardised IEEE 802.11E protocol; [2] also indicates how to map the parameter settings of IEEE 802.11E on our model parameters.

The goal of the present paper is to extend the results of [4] to the case w > 1. As in [4], four performance metrics are considered: (i) the stationary workload of the queue, (ii) the queueing delay, that is, the delay of a "packet" (a fluid particle) that arrives at the queue at an arbitrary point in time, (iii) the flow transfer delay, that is, the time elapsed between arrival of a flow and the epoch that all its traffic has been put into the queue, and (iv) the sojourn time, that is, the flow transfer time increased by the time it takes before the last fluid particle of the flow is served.

We introduce the model (including a graphical illustration) and notation in Section 2— it is noted that in our model an admission control policy is in force. We also present some preliminaries as well as the stability condition. In Section 3 the steady-state workload (i.e., buffer content) distribution of the queue is characterised. This is done by relying on techniques from [5–7]. Unfortunately, we cannot exploit the resemblance a related M/G/1 queueing model, as was possible in [4]. We find the distribution function of the steady-state workload, in terms of the solution of an eigensystem and a set of linear equations. In Section 4, we study the queueing delay of a tagged fluid particle that arrived at time 0. A full characterisation of its Laplace transform can be given; the computations turn out to be relatively straightforward, based on the observation that during the queueing delay the queue cannot enter the "idle mode."

Above we already indicated that the analysis of the flow transfer delay is much more involved than for w = 1, mainly due to the fact that the buffer can become empty during the flow transfer, so that the allocation gets into the "idle mode." The derivation of the Laplace transform requires the solution of various complex systems of equations. The results can be
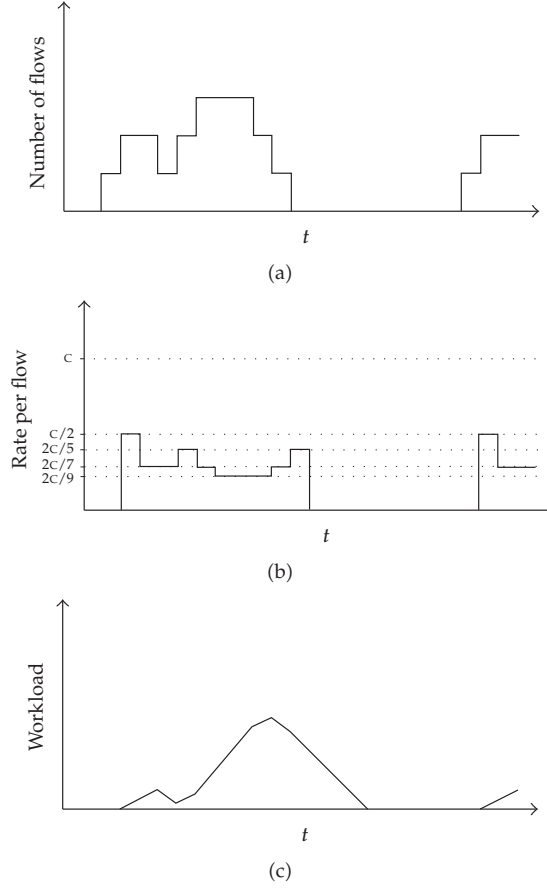
found in Section 5. In order to prove that the number of equations matches the number of unknowns, we need to show that a certain eigensystem has sufficiently many eigenvalues in the right half-plane; this we showed by using an elegant and powerful lemma of Sonneveld [8].

Section 6 concentrates on the so-called sojourn time, which is defined as the flow transfer time increased by the time it takes before the last fluid particle of the flow has left the queue; in other words, the sojourn time is the time it takes for the flow to go through the relay node. Relying on the results of Section 5, the sojourn time can be decomposed into a number of known components. Section 7 concludes and identifies a number of topics for future research.

2. Model and preliminaries

In this section, we first give a detailed description of our model and introduce notation. Then, we derive the stability condition.

2.1. Model

The following model was verbally motivated in the Introduction. Consider a queueing system at which flows arrive according to a Poisson process transmits traffic into a queue (which is served in a FIFO manner) and leaves when ready. When there are n flows active and the queue is nonempty, each flow transmits traffic into the queue at rate $C/(n + W)$, while a rate $WC/(n+W)$ is used to serve the queue; as a consequence, the queue drains when the number of flows present is below W. For ease, we assume that W is noninteger; we come back to this issue later in Section 7. When there are n flows active and the queue is empty, all flows transmit at rate $C/(2n)$, while the queue is served at rate $C/2$, so that the queue remains empty.

Suppose that we impose the admission control policy that the system accommodates maximally $N \in \mathbb{N}$ flows simultaneously; in this way, each active flow is guaranteed at least a transmission rate $C/(N + W)$, and the queue at least $WC/(N + W)$. We assume that $N > W$ (as otherwise the queue remains empty).

The above dynamics define a queueing process, for any given initial buffer level and initial number of flows present; we denote the buffer content at time t by W_t. We let N_t denote the number of flows present (i.e., feeding traffic into the queue) at time t. A pictorial illustration is given in Figure 1.

We introduce the following notation.

(i) $W_t > 0$: the "busy mode." It is not hard to see, under the assumption of exponentially distributed flow sizes (with mean μ^{-1}) and interarrival times (with mean λ^{-1}), that, *during periods that $W_t > 0$, the process N_t behaves as a Markov chain* on $\{0, \ldots, N\}$, with generator matrix

$$Q_b := \begin{pmatrix} -\lambda & \lambda & & & \\ \mu_1 C & -\mu_1 C - \lambda & \lambda & & \\ & \mu_2 C & -\mu_2 C - \lambda & \lambda & \\ & & \ddots & \ddots & \ddots \\ & & & \mu_N C & -\mu_N C \end{pmatrix}, \tag{2.1}$$

where $\mu_n := \mu n/(n + W)$; the subscript "b" stands for "busy." We define $\nu_n := \mu_n C + \lambda 1\{n < N\}$.

Figure 1: Illustration of the model. The top panel depicts the number of flows N_t as a function of time, the middle panel the rate allocated to each source as a function of time, and the bottom panel the workload W_t as a function of time; we have chosen $\mathrm{W} = 3/2$. Note that when $N_t = 1$ the rate per source equals $\mathrm{C}/2$ when the buffer is empty, and $2\mathrm{C}/5$ when the buffer is nonempty. Also note that the queue builds up when $N_t \geq 2$.

When $N_t = n$, the aggregate traffic rate generated by the flows is $r_{\mathrm{I},n} := \mathrm{C}n/(n+\mathrm{W})$, while the queue's output rate is $r_{\mathrm{O},n} := \mathrm{W}\mathrm{C}/(n + \mathrm{W})$, such that the net rate of change of the queue is

$$r_{\mathrm{A},n} := r_{\mathrm{I},n} - r_{\mathrm{O},n} = \mathrm{C}\,\frac{n - \mathrm{W}}{n + \mathrm{W}}. \tag{2.2}$$

Define $R_{\mathrm{I}} := \mathrm{diag}\{r_{\mathrm{I}}\}$, $R_{\mathrm{O}} := \mathrm{diag}\{r_{\mathrm{O}}\}$, and $R_{\mathrm{A}} := R_{\mathrm{I}} - R_{\mathrm{O}}$.

Busy periods are periods in which W_t is positive all the time. With $n_+ := \lceil \mathrm{W} \rceil$, it is evident that the number of active flows at the beginning of a busy period equals n_+. The number of active flows at the end of a busy period is in $\{0, \ldots, n_-\}$, with $n_- := \lfloor \mathrm{W} \rfloor$.

(ii) $W_t = 0$: the "idle mode." Let *idle periods* be periods in which $W_t = 0$ all the time. An idle period ends as soon as $N_t = n_+$. During the idle period necessarily $N_t \in \{0, \ldots, n_-\}$. One could say that N_t behaves as a Markov chain on $\{0, \ldots, n_-\}$ until N_t jumps from

n_- to n_+ (i.e., the start of a new busy period). The corresponding rate matrix (which is not a *bona fide* generator matrix) is

$$Q_{\mathrm{i}} := \begin{pmatrix} -\lambda & \lambda & & & \\ \mu\mathrm{c}/2 & -\mu\mathrm{c}/2 - \lambda & \lambda & & \\ & \mu\mathrm{c}/2 & -\mu\mathrm{c}/2 - \lambda & \lambda & \\ & & \ddots & \ddots & \ddots \\ & & & \mu\mathrm{c}/2 & -\mu\mathrm{c}/2 - \lambda \end{pmatrix}; \qquad (2.3)$$

the subscript "i" stands for "idle."

2.2. Stability condition

To make sure that the steady-state workload $W^\star$ is finite a.s., the mean drift should be negative when W_t is large. Since N_t behaves essentially like a stationary Markov process with generator Q_{b} when W_t is large, it follows that W_t can only escape to ∞ when $\sum_{n=0}^{\mathrm{N}} \pi_n r_{\mathrm{A},n} \geq 0$, denoting by $\vec{\pi} \equiv (\pi_0, \ldots, \pi_{\mathrm{N}})^{\mathrm{T}}$ the invariant measure of Q_{b}. Hence, we should require that $\sum_{n=0}^{\mathrm{N}} \pi_n r_{\mathrm{A},n} < 0$. Elementary Markov chain analysis yields that, with $\varrho := \lambda/(\mu\mathrm{c})$,

$$\pi_n = \frac{\rho^n \mathbb{B}(n + \mathrm{W}, n)}{\sum_{m=0}^{\mathrm{N}} \varrho^m \mathbb{B}(m + \mathrm{W}, m)}, \qquad (2.4)$$

where, for $k \geq \ell \geq 0$, $\mathbb{B}(\cdot, \cdot)$ can be regarded as a "generalised binomial coefficient":

$$\mathbb{B}(k, \ell) := \frac{\Gamma(k + 1)}{\Gamma(\ell + 1)\Gamma(k - \ell + 1)}. \qquad (2.5)$$

The stability condition becomes $\sum_{n=0}^{\mathrm{N}} r_{\mathrm{A},n} \pi_n < 0$.

It is instructive to show how this condition simplifies in the situation of $\mathrm{N} \to \infty$. Due to (recognize the probability density function of the negative binomial distribution)

$$\sum_{m=0}^{\infty} \varrho^m \mathbb{B}(m + \mathrm{W}, m) = \frac{1}{(1 - \varrho)^{\mathrm{W}+1}}, \qquad (2.6)$$

we have to verify whether

$$\sum_{n=0}^{\infty} \pi_n \cdot \frac{n - \mathrm{W}}{n + \mathrm{W}} = \sum_{n=0}^{\infty} \rho^n \mathbb{B}(n + \mathrm{W}, n)(1 - \varrho)^{\mathrm{W}+1} \cdot \frac{n - \mathrm{W}}{n + \mathrm{W}} < 0. \qquad (2.7)$$

Using identity (2.6) again, writing

$$\frac{n - \mathrm{W}}{n + \mathrm{W}} = \frac{2n}{n + \mathrm{W}} - 1, \qquad (2.8)$$

and observing that

$$\begin{aligned} \sum_{n=0}^{\infty} \varrho^n \mathbb{B}(n + \mathrm{W}, n) \frac{2n}{n + \mathrm{W}} &= \sum_{n=1}^{\infty} \varrho^n \mathbb{B}(n + \mathrm{W}, n) \frac{2n}{n + \mathrm{W}} \\ &= 2\sum_{n=1}^{\infty} \varrho^n \mathbb{B}(n + \mathrm{W} - 1, n - 1) = 2\varrho \sum_{n=0}^{\infty} \varrho^n \mathbb{B}(n + \mathrm{W}, n) = \frac{2\varrho}{(1 - \varrho)^{\mathrm{W}+1}}, \end{aligned} \qquad (2.9)$$

it is readily verified that the stability requirement reduces to $2\varrho - 1 < 0$. In other words, for the system to be stable, it is required that $\varrho < 1/2$ (irrespective of the value of W). This result makes sense as essentially all traffic has to be "served" twice: first it has to be transmitted from the sources to the queue, and then it has to be served by the queue.

3. Buffer content distribution

In this section, we study the steady-state workload $W^\star$ of the queue introduced in the previous section (jointly with the steady-state number $N^\star$ of sources present). We do this by relating the workload of our model (to which we refer as Model I) to the workload in a slightly different system (Model II): a model in which the generator Q_b and the traffic rate matrices R_I, R_O, and R_A apply also when the buffer is empty (so Model II has no "idle mode").

The procedure of relating a feedback system (Model I, in which the sources react to the buffer content) to an (easier) nonfeedback system (Model II, in which the flows behave independently of the buffer content) resembles that of [7, Section 2].

The distribution of the steady-state workload is characterised in terms of the solution of a certain eigensystem (and a number of additional linear equations). It also enables us to compute the corresponding Laplace transform, which we use several times in the next sections.

3.1. Preliminary results

In this subsection, we consider the model *without* feedback, that is, Model II: the generator matrix Q_b and the traffic rate matrices R_I, R_O, and R_A apply not only when the buffer content is positive, but also when the buffer is empty. We assume that the stability criterion derived above (which reduces to $\varrho < 1/2$ when $N \to \infty$) applies. Denote by V_t the buffer content of this system at time t (where $V^\star$ is its stationary version), and by M_t the number of flows present at time t (where $M^\star$ is its stationary version). Define also $\vec{F}(x) := (F_0(x), \ldots, F_N(x))^T$, where

$$F_n(x) := \mathbb{P}\left(V^\star \leq x, \ M^\star = n\right). \tag{3.1}$$

Model II has been studied extensively in the literature; we now recall a number of basic properties, which turn out to become useful when analysing Model I, see Section 3.2.

Buffer content distribution

It is well known from the literature how the $F_n(x)$ can be determined; they obey the system of linear differential equations:

$$R_A \vec{F}'(x) = Q_b^T \vec{F}(x). \tag{3.2}$$

Owing to the special birth-death structure, we can use explicit results obtained by van Doorn et al. [9].

A central role in the analysis is played by the eigensystem

$$z\vec{v}R_A = \vec{v}Q_b, \tag{3.3}$$

with eigenvectors $\vec{v}_0$ up to $\vec{v}_N$ and corresponding eigenvalues $z_0, \ldots, z_N$. Notice that $r_{A,n} \neq 0$, for all $n = 0, \ldots, N$, so R_A is invertible. Then, [9, Theorem 1] says that all eigenvalues z are real and simple.

Moreover, observe that the number of states of M_t in which V_t drains (or remains empty) is n_+; in the other $N - n_-$ the buffer level increases. Provided that the stability condition is satisfied, [9, Theorem 1] entails that there are $N - n_-$ negative eigenvalues, one eigenvalue that

equals 0, and n_- positive eigenvalues. Put the eigenvalues in increasing order; let $(\vec{v}_m)_n$ refer to the nth component of $\vec{v}_m$. Then the above, in conjunction with the fact that $F_n(\infty)$ lies between 0 and 1 for all n, implies that in the representation

$$F_n(x) = \sum_{m=0}^{N} c_m \cdot (\vec{v}_m)_n \cdot e^{z_n x}, \tag{3.4}$$

the terms $N - n_+ + 1$ up to N are 0. As $z_{N-n_-} = 0$ and $\vec{v}_{N-n_-} = \pi$, it follows that the requirement $\vec{F}(\infty) = \pi$ implies $c_{N-n_-} = 1$. We obtain

$$F_n(x) = \pi_n + \sum_{m=0}^{N-n_+} c_m \cdot (\vec{v}_m)_n \cdot e^{z_m x}. \tag{3.5}$$

Now, only the c_m (for $m = 0$ up to $N - n_+$) need to be determined. These follow from the fact that $F_n(0) = 0$ for $n = n_+, \ldots, N$. These are $N - n_-$ equations in the same number of unknowns, and can be determined explicitly in terms of $z_0, \ldots, z_{N-n_+}$, as described in [9], Section 4 .

Busy and idle periods

Elwalid and Mitra [10] give explicit expressions for a number of quantities that are related to the busy and idle periods of the queue. A busy period is, as before, defined as a period in which the buffer content is positive, whereas an idle period is a period in which the buffer is empty. It is easily seen that at the beginning of a busy period the number of flows present is equal to n_+; at the end of the busy period the number of flows present is at most n_-.

   Denote by $\vec{P}$ the distribution of the number of flows present at the end of the busy period. Let the matrices Q_b^{DD}, Q_b^{DU}, Q_b^{UD}, Q_b^{UU} be the submatrices that are obtained by partitioning Q_b into down-states (i.e., states n such that $r_{A,n} < 0$) and up-states ($r_{A,n} > 0$); similarly, $\vec{F}(x)$ is partitioned in $\vec{F}^D(x)$ and $\vec{F}^U(x)$.

   Then, it is not hard to prove that

$$\vec{P} = \frac{1}{\langle \vec{F}^D(0)Q_b^{DD}, \mathbf{1} \rangle} \cdot \vec{F}^D(0)Q_b^{DD}, \tag{3.6}$$

see [10, Equation (5.9)]; $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors. The mean idle period $\mathbb{E}I$ is given by

$$\mathbb{E}I = -\frac{\sum_{n=0}^{n_-} F_n(0)}{\langle \vec{F}^D(0)Q_b^{DD}, \mathbf{1} \rangle}. \tag{3.7}$$

Finally, the mean busy period $\mathbb{E}B$ can be calculated. According to "renewal reward" $\sum_{n=0}^{n_-} F_n(0) = \mathbb{E}I/(\mathbb{E}I + \mathbb{E}B)$, so that

$$\mathbb{E}B = \mathbb{E}I \cdot \frac{1 - \sum_{n=0}^{n_-} F_n(0)}{\sum_{n=0}^{n_-} F_n(0)}. \tag{3.8}$$

3.2. Analysis of buffer content distribution

Now, we turn back to Model I, as described in Section 2.1. Our goal is to show that the steady-state buffer content $W^\star$ of Model I (in which there are different queueing dynamics when the buffer is empty or nonempty) is intimately related to the steady-state buffer $V^\star$ content of Model II (in which there is no distinction between an empty and nonempty buffer).

Let us start by making a number observations. First, observe that in both Models I and II a busy period starts with n_+ flow present. Also, the distribution of the length of the busy period B is the same for both models, as well as the distribution $\vec{P}$ of the number of flows present at the end of the busy period. In other words, the difference between the models lies just in the duration of the idle periods. In Section 3.1, we already found the mean idle period $\mathbb{E}I$ of Model II. Let us therefore consider the mean idle period $\mathbb{E}J$ of Model I.

As in [7, Lemma 2.3], we have that the mean idle period of Model I equals

$$\mathbb{E}J = \langle - \vec{P}Q_i^{-1}, \mathbf{1} \rangle; \tag{3.9}$$

the expected amount of time during this idle time in which there are n flows present, say $\mathbb{E}(J_n)$, is $(-\vec{P}Q_i^{-1})_n$, that is, the nth entry of $-\vec{P}Q_i^{-1}$. This follows from the fact that the mean time $\mathbb{E}_m(J_n)$ spent in n during the idle time, given that at the beginning of the idle time m flows were present, satisfies the linear system, for $m = 1, \dots, n_-,$

$$\mathbb{E}_m(J_n) = \left(\frac{1}{\lambda + \mu C/2} \right) 1\{m = n\} + \frac{\lambda}{\lambda + \mu C/2} \cdot \mathbb{E}_{m+1}(J_n) + \frac{\mu C/2}{\lambda + \mu C/2} \cdot \mathbb{E}_{m-1}(J_n), \tag{3.10}$$

with $\mathbb{E}_{n_+}(J_n) = 0$, and

$$\mathbb{E}_0(J_n) = \frac{1}{\lambda} 1\{n = 0\} + \mathbb{E}_1(J_n). \tag{3.11}$$

We now have collected all the required elements to determine the distribution of the steady-state buffer content $W^\star$. Analogously to [7, Theorem 2.4] we obtain the following result for $G_n(x) := \mathbb{P}(W^\star \le x, N^\star = n)$.

Theorem 3.1. *For all $x \ge 0$,*

$$G_n(x) = (F_n(x) - F_n(0)) \cdot \frac{\mathbb{E}I + \mathbb{E}B}{\mathbb{E}J + \mathbb{E}B} + \frac{\mathbb{E}(J_n)}{\mathbb{E}J + \mathbb{E}B}. \tag{3.12}$$

Proof. This is proven as follows. We first condition on $W^\star$ being positive or zero:

$$G_n(x) = \mathbb{P}(W^\star \le x, N^\star = n \mid W^\star > 0)\mathbb{P}(W^\star > 0) + \mathbb{P}(W^\star = 0, N^\star = n). \tag{3.13}$$

By applying the renewal-reward theorem, the latter probability can be rewritten as

$$\mathbb{P}(W^\star = 0, N^\star = n) = \frac{\mathbb{E}(J_n)}{\mathbb{E}J + \mathbb{E}B}; \tag{3.14}$$

notice that these probabilities equal 0 for $n = n_+, \dots, N$. Also, from "renewal-reward,"

$$\frac{\mathbb{P}(W^\star > 0)}{\mathbb{P}(V^\star > 0)} = \frac{\mathbb{E}I + \mathbb{E}B}{\mathbb{E}J + \mathbb{E}B}. \tag{3.15}$$

Hence, we are left with determining the first probability in the right-hand side of (3.13). We first rewrite it as

$$\mathbb{P}(N^\star = n \mid W^\star > 0)\mathbb{P}(W^\star > 0) - \mathbb{P}(W^\star > x, N^\star = n \mid W^\star > 0)\mathbb{P}(W^\star > 0). \qquad (3.16)$$

Now, recall that the distribution of $(W^\star, N^\star)$, conditional on $W^\star > 0$, is the same as the distribution of $(V^\star, M^\star)$, conditional on $V^\star > 0$. Combining this with (3.15), we obtain

$$\mathbb{P}(M^\star = n \mid V^\star > 0)\mathbb{P}(W^\star > 0) - \mathbb{P}(V^\star > x, M^\star = n \mid V^\star > 0)\mathbb{P}(W^\star > 0)$$

$$= (\mathbb{P}(V^\star > 0, M^\star = n) - \mathbb{P}(V^\star > x, M^\star = n)) \times \frac{\mathbb{E}I + \mathbb{E}B}{\mathbb{E}J + \mathbb{E}B} \qquad (3.17)$$

$$= (F_n(x) - F_n(0)) \cdot \frac{\mathbb{E}I + \mathbb{E}B}{\mathbb{E}J + \mathbb{E}B}.$$

This proves the claim.   □

Upon combining the above theorem with representation (3.5), we find the following useful result.

Corollary 3.2. *For all $x \geq 0$, Theorem 3.1, in conjunction with (3.5), defines numbers α_n and β_{nm} (with $n = 0, \dots, \mathrm{N}$ and $m = 0, \dots, \mathrm{N} - n_+$) such that*

$$G_n(x) = \alpha_n + \sum_{m=0}^{\mathrm{N}-n_+} \beta_{nm} e^{z_m x}. \qquad (3.18)$$

Here, $G_n(0) > 0$ if and only if $n \in \{0, \dots, n_-\}$; $G_n(0)$ is given by

$$G_n(0) = \mathbb{P}(W^\star = 0, N^\star = n) = \alpha_n + \sum_{m=0}^{\mathrm{N}-n_+} \beta_{nm}. \qquad (3.19)$$

The probability of n flows in the system is given by $\mathbb{P}(N^\star = n) = G_n(\infty) = \alpha_n$ (where $\sum_{n=0}^{\mathrm{N}} \alpha_n = 1$). The Laplace transform of $W^\star$ reads, for $s > 0$,

$$\mathbb{E}(e^{-sW^\star} 1\{N^\star = n\}) = \alpha_n + \sum_{m=0}^{\mathrm{N}-n_+} \frac{s\beta_{nm}}{s - z_m}. \qquad (3.20)$$

4. Queueing delay distribution

It is clear that it is a nontrivial step to translate the steady-state workload distribution into the queueing delay distribution. Importantly, to study the delay of a fluid particle arriving at time, say, 0, the arrivals and departures of flows *after* 0 have impact. In Sections 4.1 and 4.2, we analyse the so-called *virtual queueing delay*, that is, the delay experienced by a fluid particle arriving at a random point in time (i.e., a "time average"); this is done through a direct approach in Section 4.1 and through so-called "double transforms" is Section 4.2. Section 4.3 characterises the queueing delay of an arbitrary fluid particle (i.e., a "traffic average").

4.1. Virtual queueing delay

Let $D^\star$ denote the delay experienced by a fluid particle arriving at the queue in steady state, say for ease at time 0; this type of delay is sometimes referred to as *virtual queueing delay*. Let $O(0, t)$ denote the amount of output capacity available in the interval $[0, t)$. If the fluid particle arrives at an empty queue, then the virtual delay is clearly zero; if the fluid particle arrives at a nonempty queue, then the queue is drained according to the rates $r_{\mathrm{O}, n}$ until the particle has been served (in fact even until the queue is empty). Define, for $z \geq 0$, the random variable τ_z as the time until z units of service have become available:

$$\tau_z := \inf\left\{t \geq 0 : O(0, t) = z\right\} = \inf\left\{t \geq 0 : \int_0^t r_{\mathrm{O}, N_s} \mathrm{d}s = z\right\};\tag{4.1}$$

notice that $O(0, t)$ is increasing in t. Then, analogously to [4, Section 4.1], with some abuse of notation,

$$\mathbb{E}e^{-sD^\star} = \sum_{n=0}^{\mathrm{N}} \int_0^\infty \mathbb{E}\left(e^{-s\tau_z} \mid N^\star = n\right) \mathbb{P}(W^\star = z, N^\star = n) \mathrm{d}z.\tag{4.2}$$

Hence, to further compute this expression, we need to evaluate $\mathbb{E}(e^{-s\tau_z} \mid N^\star = n)$. Here, we can use [4, Proposition 4.1]:

$$\mathbb{E}\left(e^{-s\tau_z} \mid N^\star = n\right) = \left(\exp\left((R_{\mathrm{O}}^{-1} Q_{\mathrm{b}} - s R_{\mathrm{O}}^{-1}) z\right) \mathbf{1}\right)_n,\tag{4.3}$$

where $\mathbf{1}$ denotes an $(\mathrm{N}+1)$-dimensional vector with 1s. As the same proposition entails that the eigenvalues are simple and negative (hence real numbers), it allows us to write, for constants γ_{mn}, with $m, n = 0, \ldots, \mathrm{N}$,

$$\mathbb{E}\left(e^{-s\tau_z} \mid N^\star = n\right) = \sum_{m=0}^{\mathrm{N}} \gamma_{mn} e^{\delta_m(s)z}.\tag{4.4}$$

We thus obtain the following result.

Theorem 4.1. *For $s > 0$,*

$$\mathbb{E}e^{-sD^\star} = \sum_{n=0}^{\mathrm{N}} \sum_{m=0}^{\mathrm{N}} \gamma_{mn} \mathbb{E}\left(e^{\delta_m(s)W^\star} 1\{N^\star = n\}\right),\tag{4.5}$$

where the γ_{mn} are as in (4.4). The $\delta_n(s)$, for $n = 0, \ldots, \mathrm{N}$, are the eigenvalues of $R_{\mathrm{O}}^{-1} Q_b - s R_{\mathrm{O}}^{-1}$ (which are negative). An expression for $\mathbb{E}(e^{-sW^\star} 1\{N^\star = n\})$, with $s > 0$, is available from Corollary 3.2.

4.2. A second approach: double transforms

We now proceed with demonstrating a second approach, which relies on the concept of "double transforms." We feel that this is instructive, as this approach is used extensively in the remainder of the paper (when analysing the flow transfer delay and the sojourn time).

Let us first condition on the buffer content ($W_0 = x$) that the fluid particle sees (say that it arrives at time 0), and the number of flows that are then present ($N_0 = n$). Define, for given $n = 1, \ldots, N$ and $x \geq 0$, the transform of the queueing delay:

$$\xi_n(s \mid x) := \mathbb{E}\big(e^{-sD} \mid W_0 = x, N_0 = n\big). \tag{4.6}$$

Then, we also introduce the transform of $\xi_n(s \mid x)$ with respect to the workload x:

$$K_n(s,t) = \int_0^\infty e^{-tx} \xi_n(s \mid x) \mathrm{d}x; \tag{4.7}$$

we say that $K_n(s,t)$ is a "double transform." Below, we show how to use these double transforms to derive $\mathbb{E}e^{-sD^\star}$.

Our first goal is to characterise the $K_n(s,t)$, $n = 1, \ldots, N$, for fixed $s \geq 0$ and $t > 0$. We do this by expressing $K_n(s,t)$ in terms of $K_m(s,t)$ (with $m \neq n$) as follows. Condition on the time until the service rate changes; this time has an exponential distribution with mean ν_n^{-1}. Hence,

$$K_n(s,t) = \int_0^\infty e^{-tx} \left(\int_0^{x/r_{O,n}} \nu_n e^{-\nu_n y} e^{-sy} \left(\frac{\lambda \mathbf{1}\{n < n\}}{\nu_n} \xi_{n+1}(s \mid x - r_{O,n}y) + \frac{\mu_n C}{\nu_n} \xi_{n-1}(s \mid x - r_{O,n}y) \right) \mathrm{d}y \right.$$
$$\left. + \int_{x/r_{O,n}}^\infty \nu_n e^{-\nu_n y} e^{-sx/r_{O,n}} \mathrm{d}y \right) \mathrm{d}x. \tag{4.8}$$

A straightforward change of variable ($x - r_{O,n}y = z$) then yields that

$$K_n(s,t) = \frac{1}{\nu_n + s + t r_{O,n}} \big(\lambda \mathbf{1}\{n < N\} K_{n+1}(s,t) + \mu_n C\, K_{n-1}(s,t) + r_{O,n} \big). \tag{4.9}$$

For given s and t, these are $N+1$ linear equations in the same number of unknowns. It is easy to see that the corresponding linear system is diagonally dominant, and hence there is a unique solution. This enables us to find the $K_n(s,t)$.

Our second goal is to show how these $K_n(s,t)$ yield an expression for $\mathbb{E}e^{-sD^\star}$. At an arbitrary point in time, the distribution function of the workload (jointly with the number of flows present) is given by $G_n(\cdot)$, as given by (3.18). But, as the corresponding density is the weighted sum of exponentials, it entails that knowledge of the $K_n(s,t)$ gives an expression for the Laplace transform of the virtual delay:

$$\mathbb{E}e^{-sD^\star} = \sum_{n=0}^N \int_{[0,\infty)} \xi_n(s \mid x) \mathrm{d}G_n(x)$$
$$= \sum_{n=0}^N \left(\xi_n(s \mid 0) G_n(0) + \int_{(0,\infty)} \sum_{m=0}^{N-n_-} \xi_n(s \mid x) z_m \beta_{nm} e^{z_m x} \mathrm{d}x \right) \tag{4.10}$$
$$= \sum_{n=0}^{n_-} G_n(0) + \sum_{n=0}^N \sum_{m=0}^{N-n_-} z_m \beta_{nm} K_n(s, -z_m);$$

recall that $z_m < 0$ for $m = 0, \ldots, N - n_-$.

Theorem 4.2. *For $s > 0$,*

$$\mathbb{E}e^{-sD^\star} = \sum_{n=0}^{n_-} G_n(0) + \sum_{n=0}^N \sum_{m=0}^{N-n_-} z_m \beta_{nm} K_n(s, -z_m). \tag{4.11}$$

4.3. "Packet average" queueing delay

The previous subsections presented expressions for the Laplace transform of the queueing delay "at an arbitrary point in time" (a "time average"). Clearly, there is a bias between the delay $D^\star$ "at an arbitrary point in time" and delay $\overline{D}^\star$ "seen by an arbitrary fluid particle." The correction to be made is analogous to [4, Section 4.2] and rather straightforward:

$$\mathbb{E}e^{-s\overline{D}^\star} = \sum_{n=0}^{N} \left(\frac{r_{i,n}}{\sum_{k=0}^{N} \pi_k r_{I,k}} \right) \sum_{m=0}^{N} \gamma_{mn} \mathbb{E}\left(e^{\delta_m(s)W^\star} 1\{N^\star = n\} \right), \tag{4.12}$$

compared to [11, Proposition 7.2].

5. Flow transfer delay distribution

In this section, we focus on the time F it takes for an arbitrary arriving flow to transmit its traffic. We define the flow transfer delay as the time between arrival and the epoch that its last fluid particle has been transmitted into the queue. Realise that the flow transfer time depends on the buffer content and number of flows that the tagged flow sees upon arrival. Due to the PASTA-property, these coincide with the corresponding time-averages. Recall that the case $W = 1$, as addressed in [4] is simpler, as the buffer content seen upon arrival does not play a role.

Let us first condition on the buffer content ($W_0 = x$) and the number of flows ($N_0 = n$). Define, for $s \geq 0, t > 0$, the Laplace transform of the flow transfer time (conditional on $W_0 = x$ and $N_0 = n$) and its transform with respect to x:

$$\eta_n(s \mid x) := \mathbb{E}\left(e^{-sF} \mid W_0 = x, N_0 = n\right);$$

$$L_n(s,t) = \int_0^\infty e^{-tx} \eta_n(s \mid x)\mathrm{d}x. \tag{5.1}$$

For later reference, we also introduce, for $n = 1, \dots, N$ and $m > W$,

$$\kappa_n(s,t) := v_n + s - \mathrm{tr}_{A,n}; \quad t_n(s) := \frac{v_n + s}{r_{A,n}}; \quad \ell_{n,m}(s) := L_n(s, t_m(s)). \tag{5.2}$$

Notice that, for $n > W$, $t_n(s)$ is positive.

In Section 5.1, we find the $L_n(s,t)$; this is in terms of auxiliary transforms that are determined in Section 5.2. We conclude this section by presenting the transform of the flow transfer delay; see Section 5.3.

5.1. A system of equations for the double transform

We now deduce a system of equations for the $L_n(s,t)$, $n = 1, \dots, N$. We do so by distinguishing between "down-states" (n with $r_{A,n} > 0$) and "up-states" (with $r_{A,n} < 0$). The idea is that for up-states during the time till the first event (new arrival or departure) the buffer content cannot become 0, while for down-states this is possible. As a consequence, these cases have to be dealt with differently.

Up-state

First, assume that n is an "up-state": $n \in \{n_+, \ldots, N\}$. It is elementary to see that, conditioning on the first event taking place after w units of time,

$$L_n(s,t) = \int_0^\infty e^{-tx} \left(\int_0^\infty v_n e^{-v_n w} e^{-sw} \right.$$
$$\left. \times \left(\frac{\lambda 1\{n<N\}}{v_n} \eta_{n+1}(s \mid x + r_{A,n}w) + \frac{n-1}{n} \frac{\mu_n C}{v_n} \eta_{n-1}(s \mid x + r_{A,n}w) + \frac{1}{n} \frac{\mu_n C}{v_n} \right) \right) dw \, dx. \tag{5.3}$$

This is the sum of three integrals. The third equals

$$L_n^{(3)}(s,t) = \frac{\mu_n C}{n} \cdot \frac{1}{t} \cdot \frac{1}{v_n + s} =: \mu_n^+(s,t). \tag{5.4}$$

Consider the first, and perform the change of variable $x + r_{A,n}w = y$:

$$L_n^{(1)}(s,t) = \int_0^\infty e^{-tx} \left(\int_0^\infty \lambda 1\{n < N\} e^{-v_n w} e^{-sw} \eta_{n+1}(s \mid x + r_{A,n}w) \right) dw \, dx$$
$$= \int_0^\infty \int_0^{y/r_{A,n}} e^{-t(y - r_{A,n}w)} \lambda 1\{n < n\} e^{-v_n w} e^{-sw} \eta_{n+1}(s \mid y) dw \, dy$$
$$= \int_0^\infty e^{-ty} \lambda 1\{n < N\} \left(\frac{e^{(t - v_n/r_{A,n} - s/r_{A,n})y} - 1}{t r_{A,n} - v_n - s} \right) \eta_{n+1}(s \mid y) dy \tag{5.5}$$
$$= \lambda_n(s,t) \left(L_{n+1}(s,t) - L_{n+1}\left(s, \frac{v_n + s}{r_{A,n}}\right) \right), \quad \text{where } \lambda_n(s,t) := \frac{\lambda 1\{n < N\}}{\kappa_n(s,t)}.$$

Similarly,

$$L_n^{(2)}(s,t) = \mu_n(s,t) \left(L_{n-1}(s,t) - L_{n-1}\left(s, \frac{v_n + s}{r_{A,n}}\right) \right), \quad \text{where } \mu_n(s,t) := \frac{n-1}{n} \frac{\mu_n C}{\kappa_n(s,t)}. \tag{5.6}$$

We arrive at

$$L_n(s,t) = \lambda_n(s,t)(L_{n+1}(s,t) - \ell_{n+1,n}(s)) + \mu_n(s,t)(L_{n-1}(s,t) - \ell_{n-1,n}(s)) + \mu_n^+(s,t). \tag{5.7}$$

Later, it will turn out to be also useful to consider the representation

$$\kappa_n(s,t)L_n(s,t) = \lambda 1\{n < N\} \cdot L_{n+1}(s,t) + \frac{n-1}{n} \mu_n C \cdot L_{n-1}(s,t) + \frac{\mu_n C}{n} \cdot \frac{\kappa_n(s,t)}{t} \cdot \frac{1}{v_n + s}$$
$$- \lambda 1\{n < N\} \cdot \ell_{n+1,n}(s) - \frac{n-1}{n} \mu_n C \cdot \ell_{n-1,n}(s). \tag{5.8}$$

Down-state

Now, assume that n is a down-state: $n \in \{0, \ldots, n_-\}$. In this case, we must distinguish between the cases that the process remains in state n shorter, respectively, longer than $-x/r_{A,n}$ (which is

a positive number); in the former case, the buffer does not become empty before the first event, whereas in the latter case it does. In more detail, we have

$$
L_n(s,t) = \int_0^\infty e^{-tx} \bigg(\int_0^{-x/r_{A,n}} v_n e^{-v_n w} e^{-sw}
$$
$$
\times \bigg(\frac{\lambda}{v_n} \eta_{n+1}(s \mid x + r_{A,n}w) + \frac{n-1}{n} \frac{\mu_n C}{v_n} \eta_{n-1}(s \mid x + r_{A,n}w) + \frac{1}{n} \frac{\mu_n C}{v_n} \bigg) dw
$$
$$
+ \int_{-x/r_{A,n}}^\infty v_n e^{-v_n x} e^{sx/r_{A,n}} \eta_n(s \mid 0) dw \bigg) dx.
$$

(5.9)

With $\mu_n^-(s,t) := \mu_n C / (n \cdot \kappa_n(s,t) \cdot t)$, this simplifies to

$$
L_n(s,t) = \lambda_n(s,t) L_{n+1}(s,t) + \mu_n(s,t) L_{n-1}(s,t) + \mu_n^-(s,t) - \frac{r_{A,n}}{\kappa_n(s,t)} \cdot \eta_n(s \mid 0).
$$

(5.10)

As indicated, our goal is to generate a system of equations for the $L_n(s,t)$ (with s and t fixed); we therefore wish to express $\eta_n(s \mid 0)$ in terms of the $L_n(s,t)$. This can be done as follows.

(i) First, for $n \in \{0,\dots,n_-\}$,

$$
\eta_n(s \mid 0) = \frac{1}{v_n + s} \bigg(\lambda \eta_{n+1}(s \mid 0) + \frac{n-1}{n} \mu_n C \cdot \eta_{n-1}(s \mid 0) + \frac{\mu_n C}{n} \bigg),
$$

(5.11)

whereas for $n \in \{n_+,\dots,N\}$,

$$
\eta_n(s \mid 0) = \frac{\lambda \mathbf{1}\{n < N\}}{r_{A,n}} \cdot L_{n+1}\bigg(s, \frac{v_n + s}{r_{A,n}} \bigg)
$$
$$
+ \frac{n-1}{n} \cdot \frac{\mu_n C}{r_{A,n}} \cdot L_{n-1}\bigg(s, \frac{v_n + s}{r_{A,n}} \bigg) + \frac{\mu_n C}{n} \cdot \frac{1}{v_n + s}
$$

(5.12)

$$
= \frac{\lambda \mathbf{1}\{n < N\}}{r_{A,n}} \cdot \ell_{n+1,n}(s) + \frac{n-1}{n} \cdot \frac{\mu_n C}{r_{A,n}} \cdot \ell_{n-1,n}(s) + \frac{\mu_n C}{n} \cdot \frac{1}{v_n + s}.
$$

(ii) Now, consider the vector $\vec{\eta}(s) \equiv (\eta_1(s \mid 0),\dots,\eta_{n_-}(s \mid 0))^T$. Define for $n,m = 1,\dots,n_-$,

$$
a_{n,m}(s) := \begin{cases} -\lambda, & \text{if } m = n+1; \\ v_n + s, & \text{if } m = n; \\ -(n-1)/n \cdot \mu_n C, & \text{if } m = n+1. \end{cases}
$$

(5.13)

and 0 else. The corresponding matrix is called $A(s)$, that is, $A(s) \equiv (a_{n,m}(s))_{n,m=1}^{n_-}$; for $s > 0$ we have that $A(s)$ is diagonally dominant and hence invertible. Also, $\vec{u}(s) \equiv (u_1(s),\dots,u_{n_-}(s))^T$, with

$$
u_n(s) := \begin{cases} \mu_n C/n, & \text{if } n = 1,\dots,n_- - 1, \\ \mu_{n_-} C/n_- + \eta_{n_+}(s \mid 0), & \text{if } n = n_-. \end{cases}
$$

(5.14)

Then, (5.11) implies that $\vec{\eta}(s) = (A(s))^{-1}\vec{u}(s)$. In other words, once we know $\eta_{n_+}(s \mid 0)$, we can compute $\eta_1(s \mid 0),\dots,\eta_{n_-}(s \mid 0)$.

(iii) Let $a_{n,m}^{-1}(s) := (A(s))_{n,m}^{-1}$. Now, (5.12) entails that, for $n = 1, \ldots, n_-$,

$$
\eta_n(s \mid 0) = \sum_{m=1}^{n_- - 1} a_{n,m}^{-1}(s) \frac{\mu_m C}{m} + a_{n,n_-}^{-1}(s) \left(\frac{\mu_{n_-} C}{n_-} + \eta_{n_+}(s \mid 0) \right)
$$

$$
= \alpha_n(s) + \beta_n(s) \ell_{n_+ + 1, n_+}(s) + \gamma_n(s) \ell_{n_-, n_+}(s),
$$

(5.15)

where

$$
\alpha_n(s) := \sum_{m=1}^{n_-} a_{n,m}^{-1}(s) \frac{\mu_m C}{m} + a_{n,n_-}^{-1}(s) \frac{\mu_{n_-} C}{n_-} \cdot \frac{1}{v_{n_-} + s};
$$

(5.16)

$$
\beta_n(s) := a_{n,n_-}^{-1}(s) \cdot \frac{\lambda}{r_{A,n_+}}; \qquad \gamma_n(s) := a_{n,n_-}^{-1}(s) \cdot \frac{n_-}{n_+} \cdot \frac{\mu_{n_+} C}{r_{A,n_+}}.
$$

Inserting this into (5.10), we have found the following relation for down-states n:

$$
\kappa_n(s,t) L_n(s,t) = \lambda \cdot L_{n+1}(s,t) + \frac{n-1}{n} \mu_n C \cdot L_{n-1}(s,t)
$$

$$
+ \frac{\mu_n C}{nt} - r_{A,n} \cdot \left(\alpha_n(s) + \beta_n(s) \ell_{n_+ + 1, n_+}(s) + \gamma_n(s) \ell_{n_-, n_+}(s) \right);
$$

(5.17)

notice the similarity with the equation for the up-states (5.8).

5.2. Determining the auxiliary transforms

From (5.8) and (5.17), it follows that for known functions $\phi_n(\cdot, \cdot)$ and $\psi_{m,k}(\cdot)$,

$$
\vec{L}_n(s,t) = \sum_{m=1}^{N} \left((\overline{Q}(s,t))^{-1} \right)_{n,m} \left(\phi_m(s,t) + \sum_{k=n_+}^{N} \psi_{m,k}(s) \ell_{m,k}(s) \right);
$$

(5.18)

here, the $N \times N$ matrix $\overline{Q}(s,t)$ is given through

$$
(\overline{Q}(s,t))_{n,m} := \begin{cases} \lambda 1\{n < N\}, & \text{if } m = n + 1; \\ -\kappa_n(s,t), & \text{if } m = n; \\ (n-1)/n \cdot \mu_n C, & \text{if } m = n - 1. \end{cases}
$$

(5.19)

In other words, if the transforms $\ell_{n,m}(s)$, for $n = 1, \ldots, N$ and $m = n_+, \ldots, N$, would be known, then, for fixed s and t, the values of the $L_n(s,t)$ can be found directly from solving a system of linear equations. The rest of this subsection is devoted to explaining how to identify the $\ell_{n,m}(s)$, for given $s > 0$. We first prove a useful lemma.

Lemma 5.1. *Consider, for fixed $s > 0$, the $t \in \mathbb{C}$ such that $\{\text{de}\} t(\overline{Q}(s,t)) = 0$. There are $N - n_-$ such values such that $\{\text{Re}\} t > 0$.*

(a)   (b)
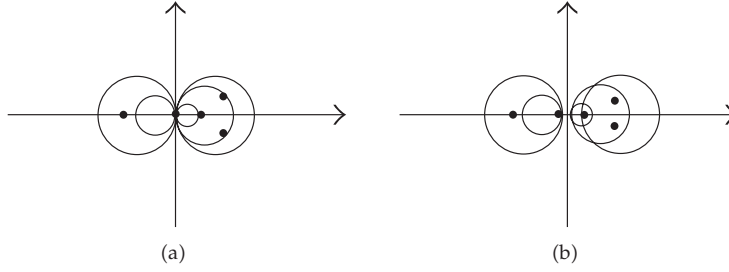
Figure 2: Eigenvalues in the complex plane. (a) Eigenvalues of $-R_A^{-1}\breve{Q}$; (b) eigenvalues of $-R_A^{-1}(\breve{Q} - \varepsilon D)$. In this example $N = 5$, and there are 3 up-states and 2 down-states.

Proof. First, rewrite $\overline{Q} = \breve{Q} - D + tR_A$, with

$$\breve{Q}_{n,m} := \begin{cases} \lambda 1\{n \leq N\} & \text{if } m = n + 1, \\ -\lambda 1\{n \leq N\} - (n-1)/n \cdot \mu_n C & \text{if } m = n, \\ (n-1)/n \cdot \mu_n C & \text{if } m = n - 1, \end{cases} \tag{5.20}$$

$d_n := n^{-1} \cdot \mu_n C + s > 0$, $D := \text{diag}\{\vec{d}\}$. Observe that $\breve{Q}$ is a generator matrix. Notice also that solutions of $\det(\overline{Q}(s,t)) = 0$ are the eigenvalues of $-R_A^{-1}(\breve{Q} - D)$.

(i) We first focus on properties of the eigenvalues of $-R_A^{-1}\breve{Q}$. Recall that it follows from Theorem 2, part 3 of [8] that $-R_A^{-1}\breve{Q}$ has as many eigenvalues in the right half plane as the number of up-states in R_A, that is, $M := N - n_-$; there is also one eigenvalue of $-R_A^{-1}\breve{Q}$ equal to 0 (note that $\breve{Q}$ is singular), and the remaining $n_- - 1$ are in the left half plane. "Geršgorin" states that all eigenvalues are in at least one of the disks

$$\left\{ z \in \mathbb{C} : \left| z + \frac{\breve{Q}_{n,n}}{r_{A,n}} \right| \leq \left| \frac{\breve{Q}_{n,n}}{r_{A,n}} \right| \right\}; \tag{5.21}$$

the nth disk is a circle in the complex plane around $-\breve{Q}_{n,n}/r_{A,n}$ of radius $|\breve{Q}_{n,n}/r_{A,n}|$ (which therefore goes through 0). Notice that $\breve{Q}_{n,n} < 0$ implies that the number of disks in the right (left) half plane equals the number of up-states (down-states, resp.). These observations are illustrated in the left panel of Figure 2.

(ii) Now, consider the eigenvalues of $-R_A^{-1}(\breve{Q} - \varepsilon D)$ for small $\varepsilon > 0$. Observe that these solve the equation $\zeta(\varepsilon,t) := \det(\breve{Q} - \varepsilon D + tR_A) = 0$. As seen in the proof of [8, Theorem 2],

$$\left. \frac{\partial}{\partial t} \zeta(\varepsilon,t) \right|_{\varepsilon,t=0} \tag{5.22}$$

has the same sign as the mean drift, that is, negative. Likewise, the derivative of ζ with respect to ε is positive (use that all diagonal entries of D are positive). Hence, replacing $-R_A^{-1}\breve{Q}$ by $-R_A^{-1}(\breve{Q} - \varepsilon D)$ moves the zero eigenvalue to the left.

"Geršgorin" implies that all eigenvalues of $-R_A^{-1}(\breve{Q} - \varepsilon D)$ are in at least one of the disks

$$\left\{ z \in \mathbb{C} : \left| z + \frac{\breve{Q}_{n,n} - \varepsilon d_n}{r_{A,n}} \right| \leq \left| \frac{\breve{Q}_{n,n}}{r_{A,n}} \right| \right\}; \tag{5.23}$$

compared to the situation of $\varepsilon = 0$ this means that the disks corresponding to the up-states (that were in the right half plane) move to the right (with the same radius); likewise, the disks corresponding to the down-states move to the left. This implies that all eigenvalues in the left (right) half plane *remain* in the left (right) half plane, because of the continuity of the solutions of $\zeta(\varepsilon, t)$ in the coefficients of the characteristic polynomial.

Conclude that for small $\varepsilon > 0$, there are then $N - n_-$ in the right half plane, and the remaining n_- in the left half plane. This is illustrated in the right panel of Figure 2.

(iii) Observe that the same arguments imply this classification remains valid when increasing ε further. The special case $\varepsilon = 1$ proves our claim. □

Now, we are able to characterise the transforms $\ell_{n,m}(s)$ as follows.

Step 1. Determine linear equations for the entries of $\vec{\ell}(s)$.

(a) First, focus on (5.8); these relate to $n = n_+, \dots, N$. We introduce the notation $\kappa_{n,m}(s) := \kappa_n(s, t_m(s))$. Then, for $n \neq m$ and $m = n_+, \dots, N$, we obtain by inserting $t = t_m(s)$:

$$\kappa_{n,m}(s)\ell_{n,m}(s) = \lambda 1\{n \leq N\}(\ell_{n+1,m}(s) - \ell_{n+1,n}(s))$$
$$+ \frac{n-1}{n}\mu_n C(\ell_{n-1,m}(s) - \ell_{n-1,n}(s)) + \kappa_{n,m}(s)\mu_n^+(s, t_m(s)). \tag{5.24}$$

Notice that $\kappa_{n,n}(s) = \kappa_n(s, t_n(s)) = 0$, so that for $n = m$ both sides reduce to 0; hence, these M equations are meaningless.

(b) Now, consider (5.17); these relate to $n = 1, \dots, n_-$. Plugging in $t = t_m(s)$ for $m = n_+, \dots, N$, we obtain

$$\kappa_{n,m}(s)\ell_{n,m}(s) = \lambda \ell_{n+1,m}(s) + \frac{n-1}{n}\mu_n C \ell_{n-1,m}(s) + \frac{\mu_n C}{n t_m(s)}$$
$$- r_{A,n} \cdot (\alpha_n(s) + \beta_n(s)\ell_{n_++1,n_+}(s) + \gamma_n(s)\ell_{n_-,n_+}(s)). \tag{5.25}$$

Step 2. Reduce dimension of the vector $\vec{\ell}(s)$. The sets of (5.24) and (5.25) enable us to express the $\ell_{n,m}(s)$ for $n = 1, \dots, N$ and $m = n_+, \dots, N$, but $n \neq m$, in terms of $\vec{\ell}_+(s) := (\ell_{n_+,n_+}(s), \dots, \ell_{N,N}(s))^T$. We have thus identified functions $\varphi_m(\cdot, \cdot)$ and $\vartheta_{m,k}(\cdot)$ such that for $n = 1, \dots, N$,

$$L_n(s, t) = \sum_{m=1}^{N} ((\overline{Q}(s,t))^{-1})_{n,m}\left(\varphi_m(s,t) + \sum_{k=n_+}^{N} \vartheta_{m,k}(s)\ell_{k,k}(s)\right). \tag{5.26}$$

In other words, when the M functions $\ell_{n_+,n_+}(s), \dots, \ell_{N,N}(s)$ would be known, we would have found the $L_n(s, t)$.

Step 3. Apply Lemma 5.1. By virtue of Cramer's rule, we obtain from (5.26) that

$$L_n(s, t) = \frac{\det \overline{Q}_n(s,t)}{\det \overline{Q}(s,t)}, \tag{5.27}$$

where $\overline{Q}_n(s, t)$ is defined as $\overline{Q}(s, t)$ but with the nth column replaced by a vector of which the mth entry is $\varphi_m(s, t) + \sum_{k=n_+}^{N} \vartheta_{m,k}(s)\ell_{k,k}(s)$. For any t in the right half plane, this should have a finite norm. Now fix $s > 0$, and use Lemma 5.1. Denote the zeroes of the denominator by $\tau_1(s), \dots, \tau_{N-n_-}(s)$. Conclude that each zero of the denominator should correspond to a zero of the numerator. This yields $N - n_-$ equations that determine $\ell_{n_+,n_+}(s), \dots, \ell_{N,N}(s)$.

The above results are summarised in the following theorem.

Theorem 5.2. *For $s, t > 0$,*

$$\vec{L}(s,t) = \left(\overline{Q}(s,t)\right)^{-1}\left(\vec{\varphi}(s,t) + \Theta(s)\vec{\ell}_+(s)\right), \tag{5.28}$$

with $\vec{\phi}(s)$ and $\Theta(s)$ defined by (5.26). For any $s > 0$ there are $N - n_-$ values of t in the right half plane such that $\{\det\}\overline{Q}(s,t) = 0$, say $\tau_1(s), \ldots, \tau_{N-n_-}(s)$. The vector $\vec{\ell}_+(s)$ follows, for fixed $s > 0$, from letting $t \to \tau_m(s)$ in the numerator of (5.27) and equating this to 0, for $m = 1, \ldots, N - n_-$.

Remark 5.3. It is easily verified that, in passing, we have also found a procedure to compute the $\eta_n(s \mid 0)$, for $n = 0, \ldots, N$, compared to (5.11), (5.12), (5.15).

5.3. Flow transfer delay

It is clear that, due to PASTA, the number of customers present at (i.e., *just after*) arrival of an (accepted) flow has distribution ($n = 1, \ldots, N$)

$$H_n := \frac{G_{n-1}(\infty)}{\sum_{m=0}^{N-1} G_m(\infty)}. \tag{5.29}$$

For determining the flow transfer delay, however, it is also necessary to know the amount of work found in the buffer. The joint distribution of the number of flows and the buffer content is given by

$$H_n(x) := \frac{G_{n-1}(x)}{\sum_{m=0}^{N-1} G_m(\infty)} = A \times \left(\alpha_{n-1} + \sum_{m=0}^{N-n_-} \beta_{n-1,m} e^{z_m x}\right), \quad A := \left(\sum_{m=0}^{N-1} \alpha_m\right)^{-1}; \tag{5.30}$$

observe that indeed $\sum_{n=1}^{N} H_n(\infty) = 1$. Hence,

$$\mathbb{E}e^{-sF} = \sum_{n=1}^{N} \int_{[0,\infty)} \eta_n(s \mid x)\mathrm{d}H_n(x). \tag{5.31}$$

Mimicking the derivation of $\mathbb{E}e^{-sD^\star}$, we obtain the following result.

Theorem 5.4. *For $s > 0$,*

$$\mathbb{E}e^{-sF} = \sum_{n=1}^{n_+} AG_{n-1}(0) \cdot \eta_n(s \mid 0) + \sum_{n=1}^{N} \sum_{m=0}^{N-n_-} \left(Az_m \beta_{n-1,m}\right) \cdot L_n(s, -z_m). \tag{5.32}$$

Expressions for $L_n(\cdot, \cdot)$ and $\eta_n(\cdot \mid 0)$ are available from the previous subsection.

6. Sojourn time distribution

In this section, we study the sojourn time S of flows in the system, which is defined as the flow transfer time, increased by the time it takes to serve the last particle of the flow. These components are not independent. Due to PASTA, the joint distribution of the workload and the number of flows that a new (accepted) flow sees upon arrival, is given by $H_n(\cdot)$, as defined through (5.30). To derive the Laplace transform of S, we first need to describe the workload increment (which can be positive or negative) during the flow transfer time F, see Section 6.1. Then, in Section 6.2 we put the components together, and derive the transform of S.

6.1. Joint transform of flow transfer time and workload increment

In the sequel it will turn out that, in order to characterise the distribution of the sojourn time, we do not just need the distribution of F, but rather its joint distribution with the workload W_{F+} and the number of flows present N_{F+} at the end of the transfer (not counting the flow that just left). To this end we introduce the counterparts of $\eta_n(s \mid w)$ and $L_n(s,t)$:

$$\eta_{n,m}(\vec{s} \mid x) := \mathbb{E}\big(e^{-s_1 F - s_2 W_{F+}} 1\{N_{F+} = m\} \mid W_0 = x, N_0 = n\big);$$

$$L_{n,m}(\vec{s},t) = \int_0^\infty e^{-tx} \eta_{n,m}(\vec{s} \mid x)\mathrm{d}x, \tag{6.1}$$

with $\vec{s} \equiv (s_1, s_2)$. Similarly to before, we also define $\ell_{n,m,k}(\vec{s}) := L_{n,m}(\vec{s}, t_k(s_1))$.

$L_{n,m}(\vec{s},t)$ can be derived essentially in the same fashion as $L_n(s,t)$ was found in Sections 5.1 and 5.2. We sketch this procedure. The counterpart of (5.7) is, for $n > \mathrm{W}$,

$$L_{n,m}(\vec{s},t) = \lambda_n(s_1,t)\big(L_{n+1,m}(\vec{s},t) - \ell_{n+1,m,n}(\vec{s})\big) + \mu_n(s_1,t)\big(L_{n-1,m}(\vec{s},t) - \ell_{n-1,m,n}(\vec{s})\big)$$

$$+ \frac{\mu_n \mathrm{C}}{n} \cdot \frac{1}{t + s_1} \cdot \frac{1}{v_n + s_1 + s_2 r_{\mathrm{A},n}} \cdot 1\{n - 1 = m\}, \tag{6.2}$$

whereas (5.10) generalises to, for $n < \mathrm{W}$,

$$L_{n,m}(\vec{s},t) = \lambda_n(s_1,t) L_{n+1,m}(\vec{s},t) + \mu_n(s_1,t) L_{n-1,m}(\vec{s},t)$$

$$+ \frac{\mu_n \mathrm{C}}{n} \cdot \frac{1}{\kappa_n(s_1,t)} \cdot \frac{1}{t + s_2} \cdot 1\{n - 1 = m\} - \frac{r_{\mathrm{A},n}}{\kappa_n(s_1,t)} \cdot \eta_{n,m}(\vec{s} \mid 0). \tag{6.3}$$

In the last equation, as before, the $\eta_{n,m}(\vec{s} \mid 0)$ can be expressed in terms of the $\ell_{n,m,k}(\vec{s})$. Then, it remains to find the transforms $\ell_{n,m,k}(\vec{s})$; these can be determined as in Section 5.2.

6.2. Sojourn time

The sojourn time can be decomposed into (i) the flow transfer delay and (ii) the time it takes to serve the traffic that is in the buffer at the end of the flow transfer delay (i.e., the time it takes to serve the last particle of the tagged flow). This allows us to write, as in [4, Section 6.3], with the usual abuse of notation

$$\mathbb{E}e^{-sS} = \mathbb{E}\big(e^{-sF - s\tau_{W_{F+}}}\big)$$

$$= \int_0^\infty \int_{[0,\infty)} \sum_{n=1}^{\mathrm{N}} \sum_{m=0}^{\mathrm{N}-1} \mathbb{E}\big(e^{-sF} 1\{W_{F+} = y, N_{F+} = m\} \mid W_0 = x, N_0 = n\big) \tag{6.4}$$

$$\times \mathbb{E}\big(e^{-s\tau_y} \mid N_0 = m\big)\mathrm{d}H_n(x)\mathrm{d}y.$$

Consider the cases $\{W_0 = 0\}$ and $\{W_0 > 0\}$ separately. The contribution due to $\{W_0 = 0\}$ amounts to

$$\int_0^\infty \sum_{n=1}^{\mathrm{N}} \sum_{m=0}^{\mathrm{N}-1} \mathbb{E}\big(e^{-sF} 1\{W_{F+} = y, N_{F+} = m\} \mid W_0 = 0, N_0 = n\big) \sum_{k=0}^{\mathrm{N}} \gamma_{km} e^{\delta_k(s)y} A G_{n-1}(0)\mathrm{d}y$$

$$= \sum_{n=1}^{\mathrm{N}} \sum_{m=0}^{\mathrm{N}-1} \sum_{k=0}^{\mathrm{N}} \gamma_{km} \mathbb{E}\big(e^{-sF + \delta_k(s) W_{F+}} 1\{N_{F+} = m\} \mid W_0 = 0, N_0 = n\big) A G_{n-1}(0) \tag{6.5}$$

$$= \sum_{n=1}^{\mathrm{N}} \sum_{m=0}^{\mathrm{N}-1} \sum_{k=0}^{\mathrm{N}} \gamma_{km} \cdot A G_{n-1}(0) \cdot \eta_{n,m}\big(s, -\delta_k(s) \mid 0\big).$$

Similarly, we find that the contribution due to $\{W_0 > 0\}$ is

$$
\int_{(0,\infty)} \sum_{n=1}^{N} \sum_{m=0}^{N-1} \sum_{k=0}^{N} \gamma_{km} \eta_{n,m} (s, -\delta_k(s) \mid x) \sum_{j=0}^{N-n_-} (Az_j \beta_{n-1,j}) e^{z_j x} dx
$$

$$
= \sum_{n=1}^{N} \sum_{m=0}^{N-1} \sum_{k=0}^{N} \sum_{j=0}^{N-n_-} \gamma_{km} \cdot (Az_j \beta_{n-1,j}) \cdot L_{n,m}(s, -\delta_k(s), -z_j).
$$

(6.6)

Theorem 6.1. *For $s > 0$, $\mathbb{E} e^{-sS}$ is given by the sum of Expressions (6.5) and (6.6).*

7. Discussion and concluding remarks

In this paper, we have considered a relay node in an ad hoc network, fed by a Poisson stream of exponentially distributed jobs. We have characterised its performance in terms of (the Laplace transforms of) the buffer content, the queueing delay, the flow transfer delay, and the sojourn time.

Integer weights

In our analysis, we throughout assumed that the weight W was noninteger. If $W \in \mathbb{N}$, the analysis is slightly more involved. We now indicate how the analysis should be adapted. In the first place, one of the coupled differential equations in (3.2) has left-hand side 0 (because $r_{A,W} = 0$); if we enumerate the equations $0, \dots, W$, then the Wth equation reads

$$
0 = \sum_{n=0}^{N} q_{n,W} F_n(x), \quad \text{or} \quad F_W(x) = -\frac{1}{q_{W,W}} \sum_{n \neq W} q_{n,W} F_n(x),
$$

(7.1)

where $q_{n,m}$ is the (n, m)-entry of Q_b. Then the nth differential equation (where $n \neq W$) becomes

$$
r_{A,n} F_n'(x) = \sum_{m \neq W} q_{m,n}^{-W} F_m(x), \quad \text{where} \quad q_{m,n}^{-W} := \left(q_{m,n} - \frac{q_{W,n}}{q_{W,W}} \cdot q_{m,W} \right).
$$

(7.2)

Interestingly, $q_{m,n}^{-W} \geq 0$ for $m \neq n$, and $\sum_{n \neq W} q_{m,n}^{-W} = 0$, and hence the $Q_b^{-W} := (q_{m,n}^{-W})_{m,n}$ (with $m, n = 0, \dots, N$, but $m, n \neq W$) correspond to an N-dimensional generator matrix. In self-evident notation, we have arrived at

$$
R_A^{-W} (\vec{F}^{-W})'(x) = (Q_b^{-W})^{\mathrm{T}} \vec{F}^{-W}(x),
$$

(7.3)

with all entries of R_A^{-W} not equal to 0. In this way, the steady-state buffer content distribution of Model II can be determined: we first find the $F_n(\cdot)$ for $n \neq W$, and then we use (7.1) to derive $F_W(\cdot)$.

The buffer content distribution follows in the same fashion as in Section 3.2. It is of crucial importance to choose a definition of busy and idle periods (i.e., one has to choose whether periods with an empty buffer and W flows in the system belong to the busy or to the idle periods), and then to consistently use this definition. It is readily verified that for the other performance measures no specific problems arise.

Limiting cases

We now consider a number of interesting limiting choices for the weight W. For ease, we lift the assumption of performing admission control; in other words: we take N = ∞. Let $N^\star(W)$ be the steady-state number of flows in the system (i.e., transmitting traffic into the queue), for a given weight W; $W^\star(W)$ is defined analogously.

As argued in [3], the *total* amount of traffic in the system has the same dynamics as an M/M/1 queue; in this M/M/1 all job sizes are inflated by a factor 2 (as they have to be processed twice). This total amount of traffic is to be understood as $2W_s^\star(W) + W^\star(W)$, $W_s^\star(W)$ denoting the traffic at the sources, and $W^\star(W)$ the traffic at the queue; the factor 2 is due to the fact that traffic at the sources still needs to be processed twice, as opposed to traffic at the queue. Importantly, the evolution of the total queue is *independent* of W; realise that the total queue is work-conserving. It follows from the Pollaczek-Khinchine formula that the mean amount of work (measured in processing time) in the system is $4\lambda^{-1}{\cdot}\varrho^2/(1 - 2\varrho)$, independently of the choice of W. It can be seen that *pathwise* the amount of traffic that is at the sources increases in W (as the $r_{1,n}$ decrease), so that the amount of traffic in the queue decreases in W.

(i) In case W = ∞, the queue has maximum weight, and never builds up. Always half of the capacity is dedicated to the flows, and the other half to the queue. It can be verified that the queue becomes a normal processor-sharing queue. The flow transfer delays and the sojourn times coincide. Elementary computations reveal that the steady-state distribution of the number of flows in the system is geometrically distributed. The probability of n flows in the system is $(1 - 2\varrho)(2\varrho)^n$; the mean number of flows is $\mathbb{E}N^\star(\infty) = 2\varrho/(1 - 2\varrho)$.

(ii) A second extreme case is W = 0. Then, the queue is only served when the flows have transmitted all their traffic into the queue; when the flows have something to transmit, the queue grows at a rate C. In this case, the probability of n flows in the system is $(1 - \varrho)\varrho^n$; the mean number of flows is $\mathbb{E}N^\star(0) = \varrho/(1 - \varrho)$.

Evidently, the mean amount of work in the queue *does* depend on W. First, observe that the mean amount of traffic (measured in processing time) that the flows still need to inject into the queue is

$$2\mathbb{E}N^\star(W){\cdot}\frac{1}{\mu},\tag{7.4}$$

where the factor 2 reflects the fact that the traffic still needs to be processed twice. We conclude that the mean amount of work (measured in processing time) at the queue is

$$\mathbb{E}W^\star(\infty) = C\left(\frac{1}{\lambda}{\cdot}\frac{4\varrho^2}{1 - 2\varrho} - \frac{2}{\mu C}{\cdot}\frac{2\varrho}{1 - 2\varrho}\right) = 0;$$

$$\mathbb{E}W^\star(0) = C\left(\frac{1}{\lambda}{\cdot}\frac{4\varrho^2}{1 - 2\varrho} - \frac{2}{\mu C}{\cdot}\frac{\varrho}{1 - \varrho}\right) = \frac{1}{\mu}{\cdot}\frac{2\varrho}{(1 - \varrho)(1 - 2\varrho)}.$$

$$\tag{7.5}$$

These arguments can be used to quantify the tradeoff between the flow transfer delay (which increases in W) and the queueing delay of the last particle (which decreases in W). The formula for W = 1 was already given in [3]

$$\mathbb{E}W^\star(1) = \frac{1}{\mu}{\cdot}\frac{4\varrho^2}{(1 - \varrho)(1 - 2\varrho)},\tag{7.6}$$

which is indeed between (7.5) for $\varrho < 1/2$.

Numerical aspects

The numerical techniques to be used in the approach presented in this paper (namely, solving eigensystems, solving linear systems, and numerically inversion of Laplace transforms) are techniques that are well established, and for which efficient and reliable computer code is available. The distributions of $W^\star$ and $D^\star$ can be found without the need of performing any Laplace inversion; as a result, their evaluation boils down to solving a linear system of differential equations, comparable to those highlighted in, for example, [5, 6]. Determining the distribution of F and S does require Laplace inversion. It is noted that recently, substantial progress has been made with respect to this type of inversion techniques. Besides the "classical" reference [12], we wish to draw attention to significant recent progress by den Iseger, as reported on in [13]; the latter reference also provides a fairly complete literature overflow.

　　When focusing on obtaining numerical output, there are several alternatives to the approach described in this paper. A first alternative is to rely on tail asymptotics, as done in [4] for W = 1, in the spirit of

$$\lim_{x\to\infty}\frac{1}{x}\log\mathbb{P}(W^\star > x) = -\vartheta^\star, \tag{7.7}$$

where the constant $\vartheta^\star > 0$ follows from the solution of the corresponding eigensystem; likewise, one could consider the logarithmic tail asymptotics of $D^\star, F$, and S. As this just provides us with decay rate (i.e., it says that $\mathbb{P}(W^\star > x) = \psi(x)e^{-\vartheta^\star x}$ for an *unknown* function $\psi(\cdot)$ such that $x^{-1}\cdot\log\psi(x)\to 0$), one could use importance sampling-based simulations to improve on this, where the twisted distribution can be computed as in [14]. For sojourn times, such importance sampling schemes can be set up as in [15].

Subjects for future research

We mention the following directions for further research.

　　(i) *Multiple bottlenecks.* In some situations, the scenario of a single bottleneck link may be an oversimplification of reality, and in such cases one could study *multiple bottleneck* links that share capacity. The complicating factor is that then the dynamics of the flows feeding into one queue will be affected by the workload process in other queues; the queues cannot be analysed separately. This gives the model the flavor of coupled-processors systems as studied in, for example, [16].

　　(ii) *Other flow-size distributions.* Another challenging extension is to consider nonexponential flow sizes; particularly the impact heavy-tailed jobs is interesting to study. Suppose for instance that the flow sizes have a regularly varying distribution of index $-\delta$, then it is an open question whether the sojourn times are regularly varying of index $1 - \delta$, as is the case in the M/G/1 FIFO queue [17, 18], or regularly varying of index $-\delta$, as is the case in the M/G/1 processor sharing (PS) queue [19] (or perhaps regularly varying of another index).

　　(iii) *Other queueing disciplines.* In this study, as well as in [3, 4], the queue was supposed to operate under the FIFO discipline. This introduces some "unfairness" in that even small jobs can incur significant delay. Put differently the sojourn time of a job of size x, say $S(x)$, is such that $\lim_{x\downarrow 0}\mathbb{E}S(x) > 0$. If the scheduling discipline in the queue would be PS (rather than FIFO), then this limit would be 0; in this sense, PS could be regarded as a remedy for unfairness.

(iv) *Weight selection.* Now, that we are able to evaluate the performance of the relay node for a given weight w, one may wonder what value of w should be chosen. As argued above, there is a tradeoff between the flow transfer delay and the queueing delay of the last particle, imposing some cost structure, and optimal value for w can be selected.

In a network setting, each node chooses its own weight. A high weight may be beneficial for the node itself, but harmful for other nodes. In view of this it may make sense to charge nodes for their weight. Pricing schemes could provide incentives for users to act as transit nodes on multihop paths, compared to [20, 21].

Acknowledgments

References

[1] T. Bheemarjuna Reddy, I. Karthigeyan, B. S. Manoj, and C. Siva Ram Murthy, "Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions," *Ad Hoc Networks*, vol. 4, no. 1, pp. 83–124, 2006.

[2] F. Roijers, H. van den Berg, and M. Mandjes, "Fluidflow modeling of a relay node in an IEEE 802.11 wireless ad-hoc network," in *Proceedings of the 20th International Teletraffic Congress on Managing Traffic Performance in Converged Networks*, pp. 321–334, Ottawa, Canada, June 2007.

[3] H. van den Berg, M. Mandjes, and F. Roijers, "Performance modeling of a bottleneck node in an IEEE 802.11 ad-hoc network," in *Proceedings of the 5th International Conference on Ad-Hoc, Mobile, and Wireless Networks*, T. Kunz and S. S. Ravi, Eds., vol. 4104 of *Lecture Notes in Computer Science*, pp. 321–336, Ottawa, Canada, August 2006.

[4] M. Mandjes and F. Roijers, "A fluid system with coupled input and output, and its application to bottlenecks in ad hoc networks," *Queueing Systems*, vol. 56, no. 2, pp. 79–92, 2007.

[5] D. Anick, D. Mitra, and M. M. Sondhi, "Stochastic theory of a data-handling system with multiple sources," *The Bell System Technical Journal*, vol. 61, no. 8, pp. 1871–1894, 1982.

[6] L. Kosten, "Stochastic theory of data-handling systems with groups of multiple sources," in *Performance of Computer-Communication Systems*, H. Rudin and W. Bux, Eds., pp. 321–331, Elsevier, Amsterdam, The Netherlands, 1984.

[7] M. Mandjes, D. Mitra, and W. Scheinhardt, "Models of network access using feedback fluid queues," *Queueing Systems*, vol. 44, no. 4, pp. 365–398, 2003.

[8] P. Sonneveld, "Some properties of the generalized eigenvalue problem $Mx = \lambda(\Gamma - cI)x$, where M is the infinitesimal generator of a Markov process, and Γ is a real diagonal matrix," Tech. Rep. Report 04-02, Department of Applied Mathematical Analysis, Delft University of Technology, Delft, The Netherlands, 2004.

[9] E. A. van Doorn, A. A. Jagers, and J. S. J. de Wit, "A fluid reservoir regulated by a birth-death process," *Stochastic Models*, vol. 4, no. 3, pp. 457–472, 1988.

[10] A. I. Elwalid and D. Mitra, "Analysis and design of rate-based congestion control of high-speed networks—I: stochastic fluid models, access regulation," *Queueing Systems*, vol. 9, no. 1-2, pp. 29–64, 1991.

[11] S. Asmussen, *Ruin Probabilities*, vol. 2 of *Advanced Series on Statistical Science and Applied Probability*, World Scientific, River Edge, NJ, USA, 2000.

[12] J. Abate and W. Whitt, "Numerical inversion of Laplace transforms of probability distributions," *ORSA Journal on Computing*, vol. 7, no. 1, pp. 36–43, 1995.

[13] P. den Iseger, "Numerical transform inversion using Gaussian quadrature," *Probability in the Engineering and Informational Sciences*, vol. 20, no. 1, pp. 1–44, 2006.

[14] M. Mandjes and A. Ridder, "Finding the conjugate of Markov fluid processes," *Probability in the Engineering and Informational Sciences*, vol. 9, no. 2, pp. 297–315, 1995.

[15] M. Mandjes and B. Zwart, "Large deviations of sojourn times in processor sharing queues," *Queueing Systems*, vol. 52, no. 4, pp. 237–250, 2006.

[16] G. Fayolle and R. Iasnogorodski, "Two coupled processors: the reduction to a Riemann-Hilbert problem," *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, vol. 47, no. 3, pp. 325–351, 1979.

[17] J. W. Cohen, "Some results on regular variation for distributions in queueing and fluctuation theory," *Journal of Applied Probability*, vol. 10, pp. 343–353, 1973.

[18] A. G. Pakes, "On the tails of waiting-time distributions," *Journal of Applied Probability*, vol. 12, no. 3, pp. 555–564, 1975.

[19] A. P. Zwart and O. J. Boxma, "Sojourn time asymptotics in the M/G/1 processor sharing queue," *Queueing Systems*, vol. 35, no. 1–4, pp. 141–166, 2000.

[20] J. Crowcroft, R. Gibbens, F. Kelly, and S. Östring, "Modelling incentives for collaboration in mobile ad hoc networks," *Performance Evaluation*, vol. 57, no. 4, pp. 427–439, 2004.

[21] J. P. Göbel, A. Krzesinski, and M. Mandjes, "Analysis of an ad hoc network with autonomously moving nodes," in *Proceedings of the Australasian Telecommunication Networks and Applications Conference (ATNAC '07)*, pp. 41–46, Christchurch, New Zealand, December 2007.