Optimal Constant Splitting for Efficient Routing over Unreliable Networks

Bin Li and Atilla Eryilmaz

Abstract— **We study the question of routing for minimum average drop rate over unreliable servers that are prone to random buffer failures. Such a generic setup can be used to model scenarios of interest in unreliable data or manufacturing networks. Interestingly, we first reveal that the traditional Join-the-Shortest-Queue (JSQ) or optimal Randomized Splitting (RS) strategies are consistently outperformed by the Constant Splitting Rule (CSR) where the incoming traffic is split with a constant fraction towards the available servers.**

This finding motivates us to obtain the optimal splitting fraction under CSR. However, the objective function to be minimized depends on the mean queue length of the servers, whose closed-form expression is not available and often intractable for general arrival and service processes. Thus, we use non-derivative methods to solve this optimization problem by approximately evaluating the objective value at each iteration. To that end, we explicitly characterize the approximation error by utilizing the regenerating nature of unreliable buffers. By adaptively controlling the precision of this approximation, we show that our proposed algorithm converges to an optimal splitting decision in the almost sure sense.

I. Introduction

The design and analysis of routing decisions for unreliable networks has received a lot of research interest [16][11]. In this work, we study the problem of efficient routing for forwarding the arrivals to parallel unreliable queues, where all data in a queue will be dropped when a failure happens. Our goal is to design an efficient routing policy which has a small average drop rate under any arrival rate. One application of this problem in the field of manufacturing systems is the wafer distribution to the parallel production pipelines. If the power of one production pipeline stops even for $0.07s$, all wafers in that pipeline break down. Similarly, in data networks serving delay-sensitive traffic, any unexpected setback in the service causes the dropping of all awaiting packets. In both scenarios, we need to make intelligent routing decisions to distribute the incoming traffic to unreliable servers, which is the focus of this work.

Join the Shortest Queue (JSQ) policy [18], where all arrivals are forwarded to the shortest queue at each slot, has been widely used as a basic routing mechanism in wired or wireless communication networks. When all queues are reliable (e.g., no failure happens), the JSQ policy is shown to have minimum delay in the symmetric case [14], or under heavy-traffic [5][4], and exhibits good performance in the general cases. However, when there are some unreliable queues, the JSQ policy may perform poorly. To see this, consider a system consisting of one reliable queue and one totally unreliable queue (e.g., failure happens all the time), all arrivals are routed to the unreliable queue under the JSQ policy and thus the average drop rate is 100%. In [11], the authors studied the Randomized Splitting (RS) policy that forwards all arrivals to a queue with a certain probability in the similar scenario. They obtained the optimal RS for the Poisson arrivals and exponential services. Yet, to the best of our knowledge, there does not exist a work that systematically treats this problem under general arrival and service processes and proposes an efficient routing policy.

In this work, we propose a constant splitting rule (CSR) that forwards a constant fraction of incoming traffic to each of the available (unreliable) servers. We show that the optimal CSR minimizes the average drop rate among all routing policies when both arrivals and services are deterministic. For the general arrival and service processes, the optimal CSR outperforms, based on numerical investigations, the well-known policies, e.g., JSQ and RS. To obtain the optimal splitting fraction, we need to solve the optimization problem with the objective function depending on the mean queue length. Since the formula for the mean queue length is hard to obtain under general arrival and service processes, it is difficult to get the exact expression for the objective function, let alone its derivative. Hence, it is almost impossible to use first or higher order numerical optimization methods to solve this optimization problem and thus we use non-derivative methods to get the optimal splitting fraction.

The most popular non-derivative method includes Patterned Search (PS) algorithms [12][6][7], which construct the set of points based on the step size varying according to a certain rule: when no improvement point is obtained on this set in the current iteration, then the step size reduces and the process is repeated. However, all these works require the exact functional value for the given point, which can not be achieved in practice. In [13], the authors presented a modified PS algorithm which adaptively adjusts the precision of the functional evaluations for the deterministic system where the accuracy of the functional value improves by increasing the evaluation time.

In our setup, the functional evaluation includes estimating the mean queue length, which can be approximated by the time average queue length. However, the approximate error is in the probabilistic form and thus it is unclear how to control the precision to guarantee the convergence to the stationary point almost surely. The earlier works [15][8] attacked this

Bin Li and Atilla Eryilmaz ({lib,eryilmaz}@ece.osu.edu) are with the Department of Electrical and Computer Engineering at The Ohio State University, Columbus, Ohio 43210 USA.

problem when the objective function is continuously differentiable, which is not the case in our setup. We generalized the previous results to the case when the objective function is locally Lipschitz continuous. The following items list our main contributions along with references on where they appear in the text:

• In Section III, we reveal the advantage of optimal CSR over JSQ or RS in the presence of buffer breakdowns, both by showing its optimality under deterministic processes and also by providing numerical results under general processes. This motivates us to obtain the optimal constant splitting fraction for the general stochastic system by using the aforementioned PS algorithm.

• In Section IV, we first characterize the probabilistic error between the approximate value and the true objective value. Then, we present the PS algorithm that adaptively adjusts the probabilistic error in each iteration, which is shown to guarantee the convergence to the optimal point almost surely.

II. System Model and Problem Statement

We consider the classic system consisting of a router and L servers with associated unreliable queues (see Fig. 1). At

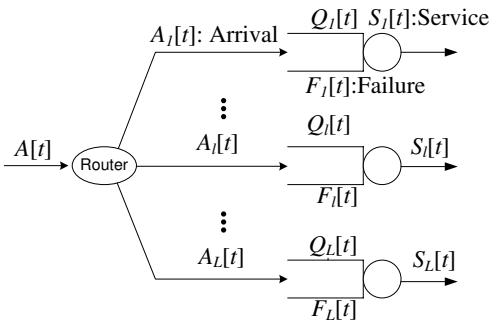

Fig. 1: System model for routing over unreliable queues

each time slot, the router needs to determine how to forward the arrivals. We assume that all data in the queue will be dropped if a failure happens. Let $F_l[t]$ denote whether a failure happens at queue l at slot t, where $F_l[t] = 1$ if a failure happens and $F_l[t] = 0$ otherwise. We assume that $F_l[t], \forall l, t$, are independently distributed over queues and identically distributed over time, with $p_l := \Pr\{F_l[t] = 1\} > 0, \forall l$. Let $Q_l[t]$ be the queue length of queue l at the beginning of slot t. Let $A[t]$ denote the amount of data arriving at router in slot t with $\mathbb{E}[A[t]] = \lambda$, and $\mathbb{E}[A^2[t]] \leq \nu$ for some $\nu < \infty$. We assume that $A[t], \forall t$, are identically distributed over time. Let $S_l[t]$ denote the maximum amount of data that can be served by server l at slot t with $\mathbb{E}[S_l[t]] = \mu_l$ and $\mathbb{E}[S_l^2[t]] \leq \kappa_l$ for some $\kappa_l < \infty$. We assume that $S_l[t], \forall l, t$, are independently distributed over queues and identically distributed over time. If the router forwards $A_l[t]$ amount of data to queue l at slot t, then the evolution of queue l is shown as follows [1]:

$$Q_l[t+1] = ((Q_l[t] + A_l[t])(1 - F_l[t]) - S_l[t])^+, \forall l, \quad (1)$$

where $(x)^+ := \max\{x, 0\}$. Our goal is to find the routing policy $\{(A_l[t])_{l=1}^L\}_{t\geq 0}$ that minimizes the average drop rate. At each slot t, the amount of dropped data at queue l is equal to $(Q_l[t] + A_l[t])F_l[t]$ and thus its expected value is $p_l\mathbb{E}[Q_l[t] + A_l[t]]$. Hence, our goal is to solve the following stochastic control problem (SCP):

Definition 1: (SCP)

$$\underset{\{(A_l[t])_{l=1}^L\}_{t\geq 0}}{\text{Minimize}} \quad \limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{l=1}^{L} p_l \mathbb{E}\left[Q_l[t] + A_l[t]\right] \quad (2)$$

$$\text{Subject to} \quad \sum_{l=1}^{L} A_l[t] = A[t], \forall t. \quad (3)$$

It is very tough to solve SCP directly. Instead, we consider the following efficient routing policy.

Definition 2: (**Constant Splitting Rule (CSR)**) Forward a_l fraction of data to queue $l, \forall l = 1, ..., L-1$ and the remaining $a_L := 1 - \sum_{l=1}^{L-1} a_l$ fraction of data to queue L at each time slot, where $a_l, \forall l = 1, ..., L-1$, are non-negative and satisfy $\sum_{l=1}^{L-1} a_l \leq 1$.

In Section III, we will show that the optimal CSR is an optimal routing policy to the above SCP problem for the system with symmetric failure probabilities under constant arrivals and constant services. Moreover, through simulations, we can observe that the optimal CSR outperforms the well-known routing policies, e.g., JSQ and optimal RS. Thus, our main task is to obtain the optimal CSR policy in the rest of the paper.

Since the mean queue length is a convex function of the arrival rate for a single queue (by Proposition 4 in Section IV) and the mean arrival rate for queue l is $a_l\lambda$, let $f_l(a_l\lambda)$ be the mean queue length for queue l. Let $\mathbf{a} \triangleq (a_l)_{l=1}^{L-1}$. To get the optimal CSR, we need to solve the following optimization problem:

$$\underset{\mathbf{a}}{\text{Minimize}} \quad g(\mathbf{a}) := \sum_{l=1}^{L} p_l f_l(a_l\lambda) + \sum_{l=1}^{L} p_l a_l \lambda \quad (4)$$

$$\text{subject to} \quad \sum_{l=1}^{L-1} a_l \leq 1 \quad (5)$$

$$a_l \geq 0, \forall l = 1, 2, ..., L-1. \quad (6)$$

It is difficult to get the exact expression for $g(\mathbf{a})$ under the general arrival and service processes, let alone to obtain its derivative. Thus, it is almost impossible to use first or higher order numerical optimization methods to solve this optimization problem. Hence, the only option for us is to use non-derivative methods, which only evaluates the functional value at each iteration. However, it is worth emphasizing that it is also hard to get the exact mean queue length for general arrival and service processes. Thus, we use the

[1]This particular evolution assumes arrivals into queues before failures. Other variations would not change the essential features of the subsequent discussion.

time average queue length to approximate the mean queue length with the error characterized in the probabilistic form. By controlling the probabilistic error in each iteration, the proposed algorithm can guarantee almost sure convergence to the optimal point.Next, we will show the optimality of CSR in the deterministic system with symmetric non-zero failure probabilities and point out its robustness in the general stochastic system.

III. The Advantage of Optimal CSR

In this section, we first show that the optimal CSR minimizes the average drop rate among all routing policies for the system with symmetric failure probabilities under constant arrivals and services. Moreover, for general arrival and service processes, we numerically observe that the optimal CSR outperforms the well-known routing policies, i.e., JSQ and optimal RS.

Lemma 1: For a single queue with constant arrival λ and constant service μ under the non-zero failure probability $p \in (0,1)$, the mean queue length is $\frac{1-p}{p}(\lambda-\mu)^+$.

Proof: See our technical report [10] for the proof. □

Remarks: Note that the mean queue length is not a continuously differentiable function of the arrival rate.

Proposition 1: For the system consisting of L unreliable queues with the constant service μ_l and same non-zero failure probability p under the constant arrival λ, the optimal CSR minimizes the average drop rate among all routing policies.

Proof: Under the above setup, our goal (4) is equivalent to minimizing

$$\limsup_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\sum_{l=1}^{L}\mathbb{E}[Q_l[t]]. \tag{7}$$

We prove this proposition by first establishing the fact that under any routing policy, the term (7) in the original system is lower bounded by that in a single queue with the failure probability p under the constant arrival λ and constant service $\sum_{l=1}^{L}\mu_l$. Then, we show that the optimal CSR can achieve this lower bound and thus is optimal among all routing policies.

Let $Q_e[t]$ be the queue length in the introduced single queue at time t. In our technical report [10], we show that there exists a random variable $\bar{Q}_e$ with $\mathbb{E}[\bar{Q}_e]<\infty$ such that $\lim_{t\to\infty}\mathbb{E}[Q_e[t]]=\mathbb{E}[\bar{Q}_e]$. Hence, by Cesaro's lemma, we have

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[Q_e[t]]=\mathbb{E}[\bar{Q}_e]. \tag{8}$$

(1) If $\lambda\le\sum_{l=1}^{L}\mu_l$, then by Lemma 1, we have $\mathbb{E}[\bar{Q}_e]=0$. Thus, it is obvious that

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\sum_{l=1}^{L}Q_l[t]]\ge\limsup_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[Q_e[t]]. \tag{9}$$

(2) If $\lambda>\sum_{l=1}^{L}\mu_l$, by taking the expectation on both sides of (1), we have

$$\mathbb{E}[Q_l[t+1]]=(1-p)\mathbb{E}[(Q_l[t]+A_l[t]-\mu_l)^+],\forall l.$$

Then, we have

$$\begin{aligned}\mathbb{E}\left[\sum_{l=1}^{L}Q_l[t+1]\right]&=(1-p)\sum_{l=1}^{L}\mathbb{E}[(Q_l[t]+A_l[t]-\mu_l)^+]\\&\ge(1-p)\mathbb{E}\left[\sum_{l=1}^{L}Q_l[t]+\lambda-\sum_{l=1}^{L}\mu_l\right].\end{aligned} \tag{10}$$

On the other hand,

$$\mathbb{E}[Q_e[t+1]]=(1-p)\mathbb{E}[Q_e[t]+\lambda-\sum_{l=1}^{L}\mu_l]. \tag{11}$$

Thus, if $\sum_{l=1}^{L}Q_l[0]=Q_e[0]$, then by combining (10) and (11), we have $\mathbb{E}[\sum_{l=1}^{L}Q_l[1]]\ge\mathbb{E}[Q_e[1]]$. If $\mathbb{E}[\sum_{l=1}^{L}Q_l[t]]\ge\mathbb{E}[Q_e[t]]$, then we have

$$\begin{aligned}&\mathbb{E}\left[\sum_{l=1}^{L}Q_l[t+1]\right]\\&\ge(1-p)\mathbb{E}\left[\sum_{l=1}^{L}Q_l[t]+\lambda-\sum_{l=1}^{L}\mu_l\right]\text{(by equation (10))}\\&\ge(1-p)\mathbb{E}\left[Q_e[t]+\lambda-\sum_{l=1}^{L}\mu_l\right]=\mathbb{E}[Q_e[t+1]].\end{aligned}$$

Thus, by induction, we have $\mathbb{E}\left[\sum_{l=1}^{L}Q_l[t]\right]\ge\mathbb{E}[Q_e[t]],\forall t$ and thus (9) still holds. By Lemma 1, we have $\mathbb{E}[\bar{Q}_e]=\frac{1-p}{p}(\lambda-\sum_{l=1}^{L}\mu_l)^+$. Thus, we have

$$\limsup_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T}\sum_{l=1}^{L}\mathbb{E}\left[Q_l[t]\right]\ge\frac{1-p}{p}(\lambda-\sum_{l=1}^{L}\mu_l)^+. \tag{12}$$

This shows that the introduced single queue system is in fact a lower bound to the original system in terms of (7).

Next, we will show that the optimal CSR can achieve this lower bound. Since for CSR policy, the router forwards a_l fraction of arrivals to queue l at each slot, there exists a random variable $\bar{Q}_l$ with $\mathbb{E}[\bar{Q}_l]<\infty$ such that $\lim_{t\to\infty}\mathbb{E}[Q_l[t]]=\mathbb{E}[\bar{Q}_l]$. Hence, by Cesaro's lemma, we have

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[Q_l[t]]=\mathbb{E}[\bar{Q}_l]. \tag{13}$$

By Lemma 1, we have $\mathbb{E}[\bar{Q}_l]=\frac{1-p}{p}(a_l\lambda-\mu_l)^+$. Thus, the optimization problem (7) becomes

$$\underset{(a_l)_{l=1}^{L}}{\text{Minimize}}\quad\frac{1-p}{p}\sum_{l=1}^{L}(a_l\lambda-\mu_l)^+ \tag{14}$$

$$\text{Subject to}\quad\sum_{l=1}^{L}a_l=1 \tag{15}$$

$$a_l\ge0,\forall l=1,...,L. \tag{16}$$

It is easy to see that if $\lambda \leq \sum_{l=1}^{L} \mu_l$, then $a_l^* = \frac{\mu_l}{\sum_{l=1}^{L} \mu_l}, \forall l$. In this case, the optimal value is 0. If $\lambda > \sum_{l=1}^{L} \mu_l$, then

$$a_l^* = \frac{\mu_l}{\lambda}, \forall l = 1, ..., L-1, a_L^* = \frac{\lambda - \sum_{l=1}^{L-1} \mu_l}{\lambda}. \quad (17)$$

In this case, the optimal value is $\frac{1-p}{p}(\lambda - \sum_{l=1}^{L} \mu_l)$, which is exactly the lower bound in (12). Thus, the optimal CSR is indeed optimal among all routing policies. □

Next, we will show that the optimal CSR also outperforms the well-known routing policies, i.e., JSQ and optimal RS through simulations.

In the simulation, there are $L = 2$ unreliable queues. The amount of arrivals and services in each slot follows Poisson and exponential distribution in Fig. 2 and 3, respectively. We use brute-force search method to get the optimal RS and optimal constant splitting fraction. We compare the average drop rate among optimal CSR, optimal RS and JSQ under both symmetric and asymmetric cases. From Fig. 2a, 2b, 3a and 3b, we can observe that JSQ exhibits good performance when the failure probability is small and the optimal CSR is quite robust for both low and high failure probability. From Fig. 2c, 2d, 3c and 3d, we can see that JSQ performs better than the optimal RS in the high arrival rate and shows worse performance in the low arrival rate, while the optimal CSR has the best performance in all cases. Thus, the optimal CSR is quite robust when the highly unreliable queues exist. However, it is hard to solve the optimization problem (4) without the exact expression for the objective function by the first or higher order numerical optimization methods. Instead, we introduce the non-derivative method in the next section.

IV. Non-derivative method for the optimal CSR

In Section III, we observed the advantage of using optimal CSR over JSQ or optimal RS. In this section, we turn to the question of finding the optimal splitting fraction that solves the problem (4). This calls for the use of non-derivative methods since the objective function is not known in closed form, and can only be evaluated approximately. To that end, in Section IV-A, we estimate the error between the approximate objective function $g(\mathbf{a})$ in (4) and its true value at the given splitting fraction $\mathbf{a}$ in the probabilistic form. Then, in Section IV-B, we use non-derivative methods to solve the optimization problem (4) and show its almost sure convergence to an optimal point.

A. Approximation Error

To use the non-derivative method, we need to know the functional value at each point. However, for a general stochastic system, it is almost impossible to get the exact value of $g(\mathbf{a})$ for each splitting $\mathbf{a}$ due to its dependence on the mean queue length of each queue. In this subsection, we obtain the approximation of $g(\mathbf{a})$ by using the time average queue length to estimate the mean queue length. To analyze the performance of the non-derivative method, we need to get the estimation error for each approximation. To that end, we first give the convergence rate of time average queue length to the mean queue length for a single queue (see Proposition 2). Then, we estimate the error between the approximation of $g(\mathbf{a})$ and its true value in the probabilistic form (see Proposition 3).

For a single unreliable queue, let N be the number of failures happening since $t = 0$ and Y_N be the time at which N^{th} failure happens for a single unreliable queue. The following proposition characterizes the rate at which the time average queue length converges to the mean queue length.

Proposition 2: For a single unreliable queue with non-zero failure probability p, where both arrivals $A[t]$ and services $S[t]$ are identically and independently distributed over time, and $\mathbb{E}[A[t]] = \lambda$ and $\mathbb{E}[A^2[t]] = \nu < \infty$, we have

$$\Pr\left\{\left|\frac{1}{Y_N}\sum_{t=1}^{Y_N} Q[t] - \mathbb{E}[\bar{Q}]\right| > \epsilon\right\} \leq h_N(\epsilon|\lambda, \nu, p), \quad (18)$$

where

$$h_N(\epsilon|\lambda,\nu,p) := h_{1,N}\left(\frac{\sqrt{\epsilon}}{2}\middle|p\right) + h_{1,N}\left(\frac{\epsilon p^2}{2\lambda(1-p)}\middle|p\right) + h_{2,N}\left(\frac{\epsilon}{4p}\middle|\lambda,\nu,p\right) + h_{2,N}\left(\frac{\sqrt{\epsilon}}{2}\middle|\lambda,\nu,p\right),$$

$$h_{1,N}(\epsilon|p) := \begin{cases} \frac{(1-p)(p+\epsilon)^2}{N\epsilon^2}, \text{if } \epsilon \geq p \\ \frac{(1-p)(p+\epsilon)^2}{N\epsilon^2} + \frac{(1-p)(p-\epsilon)^2}{N\epsilon^2}, \text{if } \epsilon < p \end{cases},$$

and

$$h_{2,N}(\epsilon|\lambda,\nu,p) := \frac{(\nu - 2\lambda^2)p^3 + (10\lambda^2 - 3\nu)p^2 + (2\nu - 14\lambda^2)p + 6\lambda^2}{N\epsilon^2 p^4}.$$

Proof: The proof is based on the observation that this system can be regarded as a renewal reward process. See our technical report [10] for details. □

Remarks: 1. In a given stochastic system, for any given $\epsilon > 0$, we have $\lim_{N\to\infty} h_N(\epsilon|\lambda,\nu,p) = 0$. Thus, we can use the time average queue length to approximate the mean queue length at arbitrary accuracy by observing sufficiently many failures N.

2. We note that, if all moments of $A[t]$ are bounded, the convergence rate of time average queue length to the mean queue length is exponentially fast, but is hard to characterized due to the complexity of queue length evolution. Nevertheless, (18) is enough for us, since we focus on the convergence of the proposed algorithm rather than its convergence rate.

Next, we give the approximation of $g(\mathbf{a})$ and obtain its error in the probabilistic form. Let N_l be the number of failures occurring in queue l. Let $\mathbf{N} := (N_l)_{l=1}^{L}$. The next proposition gives the approximation of $g(\mathbf{a})$ with the error in the probabilistic form.

Proposition 3: For a system with L unreliable queues, we have

$$\Pr\{|\hat{g}_{\mathbf{N}}(\mathbf{a}) - g(\mathbf{a})| > \epsilon\} \leq \sum_{l=1}^{L} h_{N_l}\left(\frac{\epsilon}{Lp_l}\middle|a_l\lambda, \nu_l, p_l\right),$$

where $\hat{g}_{\mathbf{N}}(\mathbf{a}) := \sum_{l=1}^{L} p_l \hat{f}_{N_l}(a_l\lambda) + \lambda\sum_{l=1}^{L} p_l a_l$, and $\hat{f}_{N_l}(a_l\lambda)$ is the time average queue length during the interval $[1, Y_{N_l}]$ at queue l, that is, $\hat{f}_{N_l}(a_l\lambda) := \frac{1}{Y_{N_l}}\sum_{t=1}^{Y_{N_l}} Q_l[t]$.

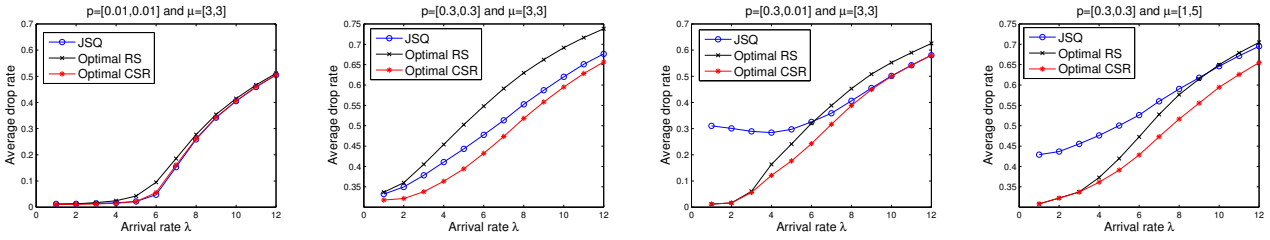

Fig. 2: Average drop rate under Poisson distribution: (a) Symmetric case: small failure probability; (b) Symmetric case: large failure probability; (c) Asymmetric case: same service rate; (d) Asymmetric case: same failure probability.

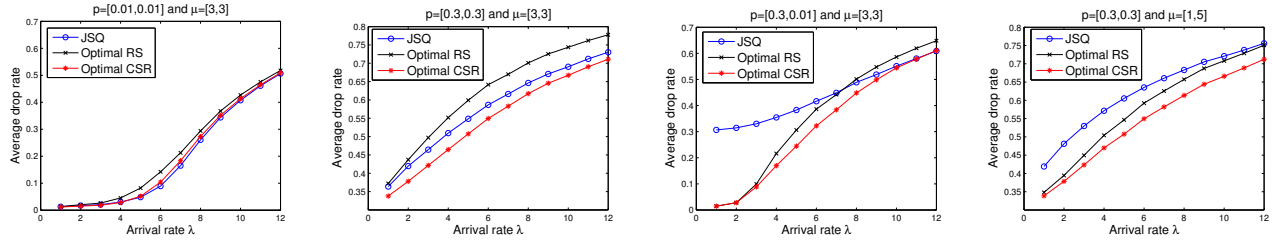

Fig. 3: Average drop rate under exponential distribution: (a) Symmetric case: small failure probability; (b) Symmetric case: large failure probability; (c) Asymmetric case: same service rate; (d) Asymmetric case: same failure probability.

Proof:

$$\begin{aligned}
&\Pr\{|\hat{g}_{\mathbf{N}}(\mathbf{a}) - g(\mathbf{a})| > \epsilon\} \\
&= \Pr\left\{\left|\sum_{l=1}^{L} p_l \hat{f}_{N_l}(a_l\lambda) - \sum_{l=1}^{L} p_l f_l(a_l\lambda)\right| > \epsilon\right\} \\
&\leq \Pr\left\{\sum_{l=1}^{L} p_l \left|\hat{f}_{N_l}(a_l\lambda) - f_l(a_l\lambda)\right| > \epsilon\right\} \\
&\leq \sum_{l=1}^{L} \Pr\left\{p_l \left|\hat{f}_{N_l}(a_l\lambda) - f_l(a_l\lambda)\right| > \frac{\epsilon}{L}\right\} \\
&\leq \sum_{l=1}^{L} h_{N_l}\left(\frac{\epsilon}{Lp_l}\middle| a_l\lambda, \nu_l, p_l\right), \qquad (19)
\end{aligned}$$

where the last inequality follows from Proposition 2. □

Remark: By Proposition 2, we have $\lim_{N_l\to\infty} h_{N_l}\left(\frac{\epsilon}{Lp_l}\middle| a_l\lambda, \nu_l, p_l\right) = 0$. Given any $\delta > 0$, if we want $\sum_{l=1}^{L} h_{N_l}\left(\frac{\epsilon}{Lp_l}\middle| a_l\lambda, \nu_l, p_l\right) < \delta$, we can choose N_l such that $h_{N_l}\left(\frac{\epsilon}{Lp_l}\middle| a_l\lambda, \nu_l, p_l\right) < \frac{\delta}{L}$. Thus, we can get the approximation of $g(\mathbf{a})$ at arbitrary accuracy by observing sufficient many failures N_l for each queue l.

B. Non-Derivative method

Let $\Omega := \{\mathbf{a} : \sum_{l=1}^{L-1} a_l \leq 1, a_l \geq 0, \forall l = 1, 2, ..., L-1\}$. Note that Ω is an intersection of linearly constraints. In the following proposed algorithm, we need to construct the positive spanning sets $\mathbf{B}$ and $\mathbf{D}(\mathbf{a}_k) \subseteq \mathbf{B}$ at each point $\mathbf{a}_k$ that conforms to Ω, that is, for some $\tau > 0$, if for each $\mathbf{x}$ in the boundary of Ω for which $\|\mathbf{x} - \mathbf{a}_k\| < \tau$, the tangent cone $T_\Omega(\mathbf{x}) := \text{closure}\{\mu(\mathbf{y} - \mathbf{x}) : \mu \geq 0, \mathbf{y} \in \Omega\}$ can be generated by nonnegative linear combination of the columns of $\mathbf{D}(\mathbf{a}_k)$. Paper [9] introduced the method to construct $\mathbf{D}(\mathbf{a}_k)$. For example, if $L = 2$, we can choose $\mathbf{D}(\mathbf{a}_k) \equiv [1, -1]$; if $L = 3$, we can select $\mathbf{D}(\mathbf{a}_k) \equiv [\mathbf{I}, -\mathbf{I}, \mathbf{F}]$, where $\mathbf{I}$ is an 2×2 identity matrix and $\mathbf{F} = [1\ -1; -1\ 1]$. Let N_l^k be the number of failures happening at queue l during the k^{th} iteration. Let $\mathbf{N}^k := (N_l^k)_{l=1}^L$.

Pattern Search (PS) method:

Requirement: $\rho \in (0,1)$, $\lim_{k\to\infty} \sum_{n=k}^{\infty} \delta_n = 0$ and $\lim_{k\to\infty} \frac{\epsilon_k}{\Delta_k} = 0$, as $\Delta_k \to 0$.

(1) Initialization: choose any $\mathbf{a_0} \in \Omega$ and $\Delta_0 > 0$. Given any $\epsilon_0 > 0$ and $\delta_0 \in (0,1)$, compute $\mathbf{N}^0$ such that $\Pr\{|\hat{g}_{\mathbf{N}^0}(\mathbf{a_0}) - g(\mathbf{a_0})| > \epsilon_0\} \leq \delta_0$.

(2) Poll step: In the k^{th} iteration, construct $\mathcal{M}_k \triangleq \{\mathbf{a_k} + \Delta_k\mathbf{d} : \mathbf{d} \in \mathbf{D}(\mathbf{a}_k)\}$. Choose $\epsilon_k > 0$ and $\delta_k \in (0,1)$, and sequentially evaluate the functional value $\hat{g}_{\mathbf{N}^k}(\mathbf{a}')$ for any $\mathbf{a}' \in \mathcal{M}_k$ satisfying $\Pr\{|\hat{g}_{\mathbf{N}^k}(\mathbf{a}') - g(\mathbf{a_k})| > \epsilon_k\} \leq \delta_k$ until some $\mathbf{a}' \in \mathcal{M}_k$ satisfying $\hat{g}_{\mathbf{N}^k}(\mathbf{a}') < \hat{g}_{\mathbf{N}^k}(\mathbf{a_k})$ is obtained, or until all points in $\mathcal{M}_k$ are evaluated.

(3) Step size update: If the poll step produced an improved point, i.e., $\hat{g}(\mathbf{a_{k+1}}) < \hat{g}(\mathbf{a_k})$, then $\Delta_{k+1} = \Delta_k$; Otherwise, $\hat{g}(\mathbf{a_k}) \leq \hat{g}(\mathbf{a_k} + \Delta_k\mathbf{d})$ for all $\mathbf{d} \in \mathbf{D}(\mathbf{a}_k)$, set $\mathbf{a}_{k+1} = \mathbf{a}_k$ and update $\Delta_{k+1} = \rho\Delta_k$. Increase $k \leftarrow k+1$, and go back to the poll step.

Next, we will establish the convergence property of the PS algorithm.

Lemma 2: The sequence of step sizes $\{\Delta_k\}_{k=0}^{\infty}$ produced by the PS algorithm satisfy $\lim_{k\to\infty} \Delta_k = 0, a.s.$.

Proof: We show $\Pr\left(\lim_{k\to\infty}\Delta_k > 0\right) = 0$ by using probabilistic argument. See report [10] for details. □

Next, we will show that the mean queue length is a convex function of the arrival rate, which implies that $g(\mathbf{a})$ is convex and thus is directional differentiable [2].

Proposition 4: For a single queue with the failure probability p, the mean queue length is a convex function of the arrival rate under general arrival and service processes.

Proof: The proof is similar to [17] and is by introduction on $Q[t]$ using the following identities.

$$\begin{aligned} Q[t+1] =& ((Q[t]+\lambda x_t)(1-z_t)-\mu y_t)^+ \\ =& \begin{cases} \max\{0, Q[t]+\lambda x_t-\mu y_t\} & \text{, if } z_t=0; \\ 0 & \text{, if } z_t=1. \end{cases} \end{aligned}$$

If we assume that $Q[t]$ is convex in λ, we can see that $Q[t+1]$ is also convex in λ. Since the theorem is true for any values of x_t, y_t and z_t, it is also true when these are realizations of random variables. Thus, the expected queue length is a convex function of the arrival rate. □

By Proposition 4, it is easy to show the convexity of $g(\mathbf{a})$ over Ω. Before stating the main convergence result, we need the concept of refining subsequence introduced in [1].

Definition 3: (**Refining subsequence**) Consider a sequence $\{\mathbf{a}_k\}_{k=0}^{\infty}$ constructed by PS algorithm. We define the subsequence $\{\mathbf{a}_k\}_{k\in\mathbf{K}}$ as the refining subsequence, if $\Delta_{k+1} < \Delta_k$ for all $k \in \mathbf{K}$, and $\Delta_{k+1} = \Delta_k$ for all $k \notin \mathbf{K}$.

Proposition 5: Let $\mathbf{a}^*$ be a limit point of a refining subsequence $\{\mathbf{a_k}\}_{k\in\mathbf{K}}$, constructed by PS algorithm. Let $\mathbf{d}$ be any column of positive spanning set $\mathbf{B}$ along which $\hat{g}(\cdot)$ was evaluated for infinitely many iterates in the subsequence $\{a_k\}_{k\in\mathbf{K}}$. Then, we have

$$\Pr\left\{ g'(\mathbf{a}^*;\mathbf{d}) = \limsup_{\mathbf{a}\to\mathbf{a}^*, t\downarrow 0} \frac{g(\mathbf{a}+t\mathbf{d})-g(\mathbf{a})}{t} \geq 0 \right\} = 1.$$

Proof: In our technical report [10], we show that

$$\begin{aligned} &\Pr\left\{ 0 > g'(\mathbf{a}^*;\mathbf{d}) = \limsup_{\mathbf{a}\to\mathbf{a}^*, t\downarrow 0} \frac{g(\mathbf{a}+t\mathbf{d})-g(\mathbf{a})}{t} \right\} \\ &\leq 2\lim_{n\to\infty}\sum_{k\geq n}\delta_k = 0. \end{aligned} \tag{20}$$

Hence, we have the desired result. □

Remark: In fact, this result does not require the convexity of g. Proposition 5 continues to hold if g is locally Lipschitz continuous, which guarantees the existence of its directional derivative [3].

Corollary 1: If g is strictly differentiable at a limit point $\mathbf{a}^*$ of a refining subsequence, and if the selection of the positive spanning sets $\mathbf{D}(\mathbf{a}_k)$ conforms to Ω for a $\tau > 0$, then $\mathbf{a}^*$ is a KKT point almost surely, that is, $\nabla g(\mathbf{a}^*)^T x \geq 0$ for all $x \in T_\Omega(\mathbf{a}^*)$, and $-\nabla g(\mathbf{a}^*) \in N_\Omega(\mathbf{a}^*)$ hold with probability 1, where $N_\Omega(\mathbf{x}) := \{\mathbf{y} : \forall \mathbf{y} \in T_\Omega(x), \mathbf{y}^T\mathbf{x} \leq 0\}$.

Proof: The proof is similar to the argument in [1]. □

Remarks: 1. Here, we only require that the objective function g is differentiable at the limiting point rather than being continuously differentiable, which is required in [15][8].

2. Even if g is not differentiable at $\mathbf{a}^*$, we still have desirable property that $g'(\mathbf{a}^*;\mathbf{d}) \geq 0$ for all $\mathbf{d} \in \mathbf{B}$ from Proposition 5.

V. Conclusions

In this paper, we investigated the problem of efficient routing for unreliable networks that are prone to probabilistic buffer failures. We first revealed the advantage of using constant splitting rule (CSR) in such a setup over the more traditional choices of Join the Shortest Queue (JSQ) or Randomized Splitting (RS). This motivated us to obtain the optimal splitting fraction that solves the optimization problem with the objective function depending on the mean queue length.

Realizing the difficulty in getting the exact expression for the objective function under general arrival and service processes, we use non-derivative methods to solve this optimization problem by using the time average queue length to approximate the mean queue length. By adaptively controlling the approximation error, we show that the proposed algorithm can almost surely converge to an optimal splitting fraction under mild conditions.

References

[1] C. Audet and J. Dennis. Analysis of generalized pattern searches. *SIAM Journal on Optimization*, 13(3):889–903, 2003.

[2] D. Bertsekas, A. Nedic, and A. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific, Belmont, Mass., 2003.

[3] F. Clarke. *Optimization and Nonsmooth Analysis (Classics in Applied Mathematics)*. Society for Industrial Mathematics, 1987.

[4] A. Eryilmaz and R. Srikant. Asymptotically tight steady-state queue length bounds implied by drift conditions, 2011. Available online at `http://arxiv.org/abs/1104.0327`.

[5] G. Foschini and J. Salz. A basic dynamic routing problem and diffusion. *IEEE Transactions on Communications*, 26(3):320–327, 1978.

[6] R. Hooke and T. A. Jeeves. Direct search solution of numerical and statistical problems. *Journal of ACM*, 8(2):212–229, 1961.

[7] J. J. Dennis and V. Torczon. Direct search methods on parallel machines. *SIAM Journal on Optimization*, 1(4):448–474, 1991.

[8] S. Kim and D. Zhang. Convergence properties of direct search methods for stochastic optimization. In *Proc. Winter Simulation Conference (WSC)*, Baltimore, MD, Dec. 2010.

[9] R. Lewis and V. Torczon. Pattern search methods for linearly constrained minimization. *SIAM Journal on Optimization*, 10(3):917–941, 2000.

[10] B. Li and A. Eryilmaz. Optimal constant splitting for efficient routing over unreliable networks. Technical Report, 2012. Available online at `http://www.ece.osu.edu/~eryilmaz/LiEryilmaz_CDC12_Report.pdf`.

[11] I. Mitrani and P. Wright. Routing in the presence of breakdowns. *Performance Evaluation*, 20(1-3):151–164, 1994.

[12] E. Polak. *Computational Methods in Optimization: A Unified Approach*. Academic Press, 1971.

[13] E. Polak and M. Wetter. Precision control for generalized pattern search algorithms with adaptive precision function evaluations. *SIAM Journal on Optimization*, 16(3):650–669, 2006.

[14] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Transactions on Information Theory*, 39(2):466–478, 1993.

[15] M. Trosset. On the use of direct search methods for stochastic optimization. *Technica Report, Department of COmputational & Applied Mathematics, Rice University, Houston, TX.*, 2000.

[16] J. Tsitsiklis. *Optimal Dynamic Routing in an Unreliable Manufacturing System*. M.Sc. Thesis, Department of EECS, MIT, 1981.

[17] R. Weber. A note on waiting times in single server queues. *Operations Research*, 31(5):950–951, 1983.

[18] W. Whitt. Deciding which queue to join: Some counterexamples. *Operation Research*, 34(1):55–62, 1986.