2017 29th International Teletraffic Congress

Latency Reduction In Communication Networks Using Redundant Messages

Joseph Hollinghurst
University of Bristol

Ayalvadi Ganesh
University of Bristol

Timothy Baugé
Thales UK Ltd.

***Abstract*—In this paper, we study the use of redundancy to reduce latency and increase reliability in communication networks. The work is motivated in particular by wireless networks, where channels may be unreliable and channel characteristics may vary over time. The objective is to provide an analysis of the gains achievable through the use of redundant messages. Redundancy increases the network load and hence the delay of each packet, but can reduce overall delay by exploiting independent randomness across multiple paths. One of the goals addressed in this paper is the optimisation of this trade-off. We consider both average delay minimisation and probabilistic guarantees on delay exceeding some tolerance threshold. We study a number of different stochastic models for packet transmission times, and uses techniques from queueing theory and large deviations to analyse their performance. The analytical results are complemented by simulations.**

I. Introduction

Wireless networks have many advantages over wired including flexibility, cost and convenience. However, one major drawback of wireless networks is the increased latency caused by packet loss and retransmissions. While much is done to combat these problems in current networks some of the proposed applications of 5G and future networks such as sensor networks, virtual reality and V2V or V2x communications are said to require an end-to-end latency of a millisecond or less. This is a significant decrease in latency compared to 4G/LTE networks where one way delay is measured to be between 20ms and 60ms [1]. Therefore, in order to reduce latency by 20-60 times new methods for latency reduction must be considered.

One technique that has been studied to reduce latency in wired networks is redundancy. This was first investigated by Maxemchuk [2] [3], but redundancy has more recently been studied in [4] [5] [6]. By sending multiple copies of the same data the latency may be reduced, as it is the minimum time to receive any one of the transmitted replicates. The downside is the increased load on the network. This leads to a trade-off between the number of replicates and the delay. It should be noted that redundancy is not the same as retransmission since the redundant packets are transmitted at the same time as the original.

The current literature mainly focuses on wired networks and on data centres, and much of it, especially analytical results, is restricted to exponential service times. In addition, temporal variability in channel conditions, which is common in wireless networks, is not considered. Our contributions in this paper are as follows. After presenting results for exponential service times as a baseline, we first generalise the framework to consider phase-type service distributions. These provide a rich modelling framework as they can be used to approximate the distribution of any non-negative random variable. The Pollaczek-Khinchin (P-K) formula gives an expression for the Laplace-Stieltjes transform (LST) of the delay distribution in this case. However, we are interested in the minimum delay over parallel independent channels, and this leads to complicated expressions once inverted. To get around this, we suggest a simplification based on approximating the delay distribution by its dominant exponential, and evaluate its performance by comparing it to the exact distribution. Secondly, we explicitly consider models of time-varying channels, and show how these can be modelled using queues with phase-type service distributions. They may, in addition, require an exceptional service distribution for the first customer entering an empty system. We present results for such queueing models, and again apply and evaluate our approximation based on the dominant exponential. Throughout our analysis we see a trend that the optimal replication policy depends on the load. We analyse this further by finding an optimal replication factor that minimises the expected delay for any given load and develop a simple dynamic replication policy that uses this information. Finally, in some applications where the emphasis is on reliability, it is not the mean latency that is most relevant. Rather, there is a moderately large threshold of tolerable latency, but it is important that the overwhelming majority of packets be delivered within this threshold. Our third contribution is an analysis of the tail of the delay distribution, motivated by considerations of reliability.

A. Related Work

Related work on redundancy dates back to Maxemchuk [2], who first proposed dispersity routing. The original analysis assumes exponentially distributed interarrival and service times, but the work is extended in [3] by looking at blocking probabilities of switches. We study a different model motivated by wireless networks where, instead of packets being blocked, they might require retransmission which may trigger an exceptional first service.

Vulimiri *et al.* [4] study the use of redundant requests to multiple servers in data centres. A variety of service distributions are considered. They restrict the analysis to replicating each request exactly twice, and the main metric considered is

978-0-9883-0453-6/17 $31.00 © 2017 ITC
DOI 10.1109/ITC.2017.37

the threshold value of load at which replication is no longer beneficial. An important contribution of their work is the empirical studies performed, and particular benefit observed in the tail of the delay distributions. We formalise the benefit of redundancy in the tail of the distributions and link it to the reliability of transmissions.

[6] looks at the average latency in memoryless, heavy tailed and light tailed service distributions. The analysis performed aims to give the optimal policy, however it is limited to a binary answer of send to all or no redundancy. We show redundancy policies are not this simple, as depending on the load, there are intermediary cases where double redundancy can be of more benefit than triple.

The paper by Gardner et al. [5] analyses the behaviour of redundancy when there are also independent non-redundant arrivals in the system. Another novel feature of the paper is the comparison of redundancy with the opt-split and JSQ (join-Shortest-Queue) policies. The paper gives exact analysis for the models presented but, unfortunately, only exponential service times are considered. By providing a framework for analysing a system with redundancy and introducing phase type service distributions we believe our results are more general, and can therefore be applied to a larger class of systems. Furthermore, we complement our findings with approximations. This enables us to find results to much more complicated systems that do not have a palatable closed form.

II. Models and General Approach

We have in mind a general network, wired or wireless, with a specified traffic demand. The demand is composed of mutually independent Poisson arrival processes corresponding to distinct source-destination pairs. We focus on a single source-destination pair in such a network, and assume that we are provided with some number ℓ of vertex-disjoint paths between them. Furthermore, we have the choice of either routing all traffic between this source and destination along one of these paths, or replicating it across $k \leq \ell$ of them; in either case, the routes to be used are fixed in advance and cannot be changed based on dynamic traffic information. If replication is used, then we assume that all traffic in the network, between all source-destination pairs, is replicated by the same factor, k; the special case $k = 1$ corresponds to no replication.

Each link in the network is modelled as a queue with a single server, which serves packets in their order of arrival. The queue multiplexes traffic from a number of different *flows* (source-destination pairs). We are interested in the delays incurred by packets from an index flow that we focus on, and treat all other packets as cross-traffic. The total delay along a route is the sum of link delays across each link in the route. Finally, the latency of a packet is its minimum delay over the k different routes it has been assigned.

Our main modelling assumption is now the following: we assume that each route can be modelled as a single-server FIFO queue, and that these queues are *mutually independent.* The first part of this assumption basically says that the delay along a route is dominated by that on a single bottleneck link, so it suffices to consider that single link. While it is a fairly standard assumption in network modelling, it is not essential, but is more for computational convenience. The second part of the assumption, of independence between queues, is essential for the analysis. While this is a strong assumption, we would expect it not to be too unrealistic in large and well-connected networks. If the number of nodes and links is large, and there is a large number of available vertex-disjoint routes between any pair of source and destination nodes, then we can expect that distinct routes carry traffic multiplexed from mostly non-overlapping subsets of source-destination pairs, and hence are largely independent. An implicit second assumption is that each route carries traffic from a large number of flows, and that the contribution of any single one of these flows to the total traffic is negligible. Thus, even though the index traffic stream is common to the routes along which it is replicated, this introduces no dependence because its own contribution to the total traffic along any of these routes is negligible; the waiting times it incurs are almost entirely due to the service times of cross-traffic.

Given the above assumptions, we obtain a relatively simple and tractable abstraction for the system model. Packets arrive according to a Poisson process. Each packet is replicated across k parallel single-server FIFO queues, where k is a specified parameter; the case $k = 1$ corresponds to no replication. By the PASTA property, the packets see these queues in their invariant distribution. Consequently, the waiting time seen incurred by a typical packet in the j^{th} queue, which we denote $W^{(j)}$, is a random variable drawn from the invariant waiting time distribution in this queue, and is independent across different queues. The latency suffered by a typical packet is given by the random variable $W = \min_{j=1}^{k} W^{(j)}$, where the minimum is taken over all queues in which the packet is replicated. Next, each of the queues is itself modelled as an $M/G/1$ queue; in other words, it sees Poisson arrivals at rate λ_j, the customers (packets) have independent and identically distributed (iid) service times denoted $S_i^{(j)}, i \in \mathbb{Z}$, and they are served in their order of arrival. The LST of the invariant waiting time distribution in such a queue is given by the well-known P-K formula (see [7], for example), which we now state following some definitions.

Consider a queue into which customers arrive according to a Poisson process of rate λ. There is a single server who serves customers in their order of arrival, i.e., follows a first-in first-out (FIFO) service policy. The time required to serve customer i is random, and is denoted S_i; the random variables S_i are assumed to be iid, with cumulative distribution function (cdf) $G(\cdot)$. We denote by $g(\cdot)$ the LST of the service time distribution, defined as follows:

$$g(x) = \mathbb{E}\left[e^{-xS_1}\right] = \int_0^\infty e^{-xt} dG(t). \tag{1}$$

We define the traffic intensity to be $\rho = \lambda \mathbb{E}[S_1]$, where $\mathbb{E}[S_1]$ is the mean service time and could be infinite. However, waiting times grow to infinity whenever $\rho \geq 1$, so we shall restrict

ourselves to the case $\rho < 1$. In that case, the queue has an invariant distribution under which waiting times are finite almost surely. Letting $W^*(\cdot)$ denote the LST of the invariant waiting time distribution, we have the P-K formula

$$W^*(s) = \frac{(1-\rho)s}{s - \lambda(1-g(s))}. \tag{2}$$

We now outline our approach to obtaining the distribution of the latency under replication. Using the notation above, let $W^{(j)}$ denote a random variable with the distribution of the steady-state waiting time in the j^{th} queue to which a packet from the index flow has been replicated. Given the total arrival rate $\lambda^{(j)}$ into this queue, and the service time distribution $G^{(j)}$, we can use the P-K formula to obtain the LST for the distribution of the waiting time $W^{(j)}$ in this queue. This could be inverted, either analytically or numerically, to obtain an explicit expression for the waiting time distribution. Given the waiting time distributions at each of the queues to which the packet has been replicated, and using the independence of these queues, we can compute the distribution of its latency as follows:

$$\begin{aligned} \mathbb{P}(W > x) &= \mathbb{P}\Big(\bigcap_{j=1}^{k} W^{(j)} > x\Big) \\ &= \prod_{j=1}^{j} \mathbb{P}(W^{(j)} > x). \end{aligned} \tag{3}$$

While the approach is straightforward in principle, in practice the LST of the waiting time will typically either not be invertible in closed form, or the expression will be very complicated. In either case, it might be hard to obtain much insight from the result. Therefore, after sketching the general approach, we focus attention on a few simple and stylised special cases which, nevertheless, shed some light on the gains achievable for replication.

We obtain results for two classes of service time distributions, namely exponential and phase-type distributions. A phase-type distribution is defined as the distribution of the time to absorption in a finite-state absorbing Markov process started in a specified distribution. In more detail, a phase-type (PH) distribution is parametrised by a sub-probability vector $\boldsymbol{\alpha} \in \mathbb{R}^m$ (a vector with non-negative elements whose sum is no bigger than 1) and a subgenerator matrix $S \in \mathbb{R}^{m\times m}$, namely an $m\times m$ matrix all of whose off-diagonal elements are positive, none of whose row sums is positive, and at least one of whose row sums is negative. The phase-type distribution with these parameters, denoted $PH(\boldsymbol{\alpha}, S)$, is defined as the time to absorption of the Markov process with generator

$$Q = \begin{pmatrix} 0 & \mathbf{0} \\ \mathbf{s^0} & S \end{pmatrix},$$

and initial distribution $(\alpha_0, \boldsymbol{\alpha})$, whose unique absorbing state is the first state. Here $\alpha_0 \geq 0$ is such as to make the elements of the row vector $(\alpha_0, \boldsymbol{\alpha})$ sum to 1, and the column vector $\mathbf{s^0}$ is such as to make the rows of Q sum to zero, so that it is a generator matrix. It is possible to explicity write down the cdf $G(\cdot)$, and its LST $g(\cdot)$, for the phase-type distribution. We have

$$\begin{aligned} G(t) &= 1 - \boldsymbol{\alpha} e^{tS} \mathbf{1}, \\ g(x) &= \alpha_0 + \boldsymbol{\alpha}(xI - S)^{-1}\mathbf{s^0}, \end{aligned} \tag{4}$$

where I is the $m\times m$ identity matrix. Note that the distribution $G(\cdot)$ has an atom of size α_0 at zero, and is continuous on $(0, \infty)$.

Our motivation for studying phase-type service distributions is two-fold. The first is that we want to be able to model a wide variety of service time distributions. The service time of a packet includes not only the physical transmission time, which would only depend on the packet size and the link bandwidth, but also a random "contention time". The contention time might either be the time to be scheduled at a switch in a wired network, or the time to access the channel under a medium access control (MAC) protocol in a wireless network. These times could have rather general distributions, and phase-type distributions have the advantage of being able to approximate a very wide class of probability distributions of non-negative random variables. The second motivation is that wireless channels are inherently dynamic and suffer from conditions such as fading which make their bandwidth vary over time. Consequently, the service time over such a channel would also exhibit corresponding variability. We will describe how to model this using phase-type distributions in the next section.

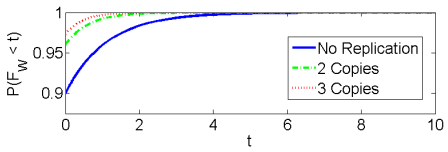
III. Replication and Mean Latency

In this section, we study the impact of replication on mean latency in three different queueing models. We assume throughout that packets of the index flow arrive into the network according to a Poisson process of rate λ, and are replicated across k routes. We approximate the total packet delay along a route by that at a bottleneck queue. We consider different models for the service time at a queue.

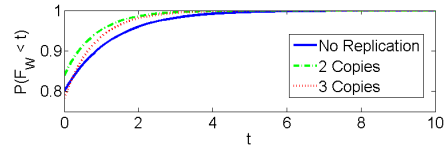
A. The $M/M/1$ queue

Service times are assumed to be iid with an Exp(μ) distribution. This can be justified on the basis that the service time is comprised mostly of the time to serve packets of all queued cross traffic; the latter is known to have exponentially decaying tails in considerable generality (see, e.g, [8]). Thus, the model is not as unrealistic as it might first appear, especially for high-bandwidth wired networks multiplexing a large number of independent flows, each of which individually contributes a small fraction of the total traffic. The service rates μ_i in different botteleneck queues would typically be different. For notational convenience, and to facilitate graphical display of the results, we reduce the number of parameters in our models by taking $\mu_i = \mu$, a constant. This is also the most interesting parameter setting as the benefits of replication are greatest when delays are similar across the available routes. If mean delays differ vastly across the routes, then the same route will usually achieve the minimum, and the benefits of diversity will be minimal.

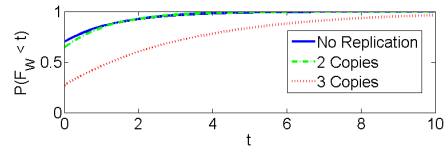
The LST of an Exp(μ) service time distribution is $g(s) = \frac{\mu}{\mu+s}$. Substituting this in the P-K formula (2), and recalling



(a) $\lambda_k = 0.1$*



(b) $\lambda_k = 0.2$*



(c) $\lambda_k = 0.3$

Fig. 1. Wait time CDF with exponential service under varied load, **scale changes on vertical axis*

that $\rho = \lambda/\mu$ denotes the traffic intensity, we obtain the LST of the waiting time to be

$$W^*(s) = (1-\rho) + \rho\frac{\mu(1-\rho)}{\mu(1-\rho)+s}. \tag{5}$$

Inverting $W^*(s)$, the probability density of the waiting time is given by

$$f_W(t) = (1-\rho)\delta(t) + \rho(\mu-\lambda)e^{-(\mu-\lambda)t},$$

where δ denotes the Dirac delta. In other words, the waiting time distribution is a mixture of an atom at zero, and an Exp$(\mu-\lambda)$ distribution. The corresponding cdf is

$$F_W(t) = 1 - \rho e^{-(\mu-\lambda)t}. \tag{6}$$

We now consider the effect of replicating every packet across k edge-disjoint routes. This will increasing the total arrival rate into each queue by the same factor k. Substituting (6) in (3), and noting that the latency, which we denote L_k, is the minimum of k iid waiting times, we get

$$\mathbb{P}(L_k > t) = \prod_{i=1}^{k} \rho e^{-(\mu-k\lambda)t}.$$

Thus, the cdf and mean of the latency are given by

$$\begin{aligned} F_{L_k}(t) &= 1 - (k\rho)^k e^{-k(1-k\rho)\mu t}, \\ \mathbb{E}[L_k] &= \frac{(k\rho)^k}{k(1-k\rho)\mu}. \end{aligned} \tag{7}$$

The cdfs are plotted in Figure 1 for $\lambda = 0.1$, 0.2 and 0.3, and $k = 1$, 2 and 3. Observe that the cdf plots can cross each other, meaning that no single value of the replication factor k may be optimal for all t. But for a wide range of t, the cdf plots for $k = 2$ and 3 are well above those for $k = 1$, demonstrating that replication significantly increases the probability of meeting a specified delay bound t.

B. The $M/PH/1$ queue

The exponential distribution is a special case of a much larger class known as phase-type distributions, which were described in Section II. Our motivations for studying them include considering routes with multiple bottleneck queues, or describing systems in which service times exhibit much greater variability than is captured by an exponential distribution, as observed in some studies. High variability can also arise due to unreliable servers, as we shall see later. We now present a result for the LST of the waiting time distribution in such a queue.

Lemma III.1. *Consider an $M/PH/1$ queue with Poisson(λ) arrival process, and iid service times with a $PH(\boldsymbol{\alpha}, S)$ distribution of order m. The LST of the waiting time distribution is given by:*

$$W^*(s) = \frac{1 + \lambda\boldsymbol{\alpha}S^{-1}\mathbf{1}}{1 - \lambda\boldsymbol{\alpha}(sI-S)^{-1}\mathbf{1}}, \tag{8}$$

where I denotes the $m \times m$ identity matrix, and $\mathbf{1}$ a column vector of all ones of length m.

Proof. Observe that $S\mathbf{1} + \mathbf{s^0} = \mathbf{0}$, as $\mathbf{s^0}$ was chosen to make the rows of the rate matrix describing the phase type distribution sum to zero. Consequently,

$$\begin{aligned} (sI-S)\mathbf{1} &= s\mathbf{1} + \mathbf{s^0}, \\ (sI-S)^{-1}\mathbf{s^0} &= \mathbf{1} - s(sI-S)^{-1}\mathbf{1}. \end{aligned}$$

Also, $\alpha_0 + \boldsymbol{\alpha}\mathbf{1} = 1$, as $(\alpha_0, \boldsymbol{\alpha})$ is a probability vector. Substituting these into the LST of the service time distribution given in (4), we obtain after simplification that

$$1 - g(s) = s\boldsymbol{\alpha}(sI+S)^{-1}\mathbf{1}.$$

Substituting this into the P-K formula (2), and noting that $\rho = \lambda\mathbb{E}[S]$, where the mean service time is given by $\mathbb{E}[S] = -\boldsymbol{\alpha}S^{-1}\mathbf{1}$, we obtain the claim of the lemma after some straightforward manipulations. □

The lemma shows that $W^*(s)$ is a rational function (ratio of polynomials) in s. For generic values of the matrix S, the denominator polynomial has distinct roots z_i, $i = 1, 2, \ldots, m$, and so $W^*(\cdot)$ admits the partial fraction expansion

$$W^*(s) = c_0 + \sum_{i=1}^{m} \frac{c_i}{s - z_i}.$$

The roots z_i and the coefficients c_i could be complex in general. But if they are all real, all c_i are positive and all z_i negative, then the LST is easily inverted to yield the waiting time density

$$f_W(t) = c_0\delta(t) + \sum_{i=1}^{m} c_i e^{z_i t},\ t \geq 0.$$

In words, the waiting time is a mixture of an atom at zero (which must have mass $1-\rho$, as this is the probability of

the queue being empty) and exponential distributions with parameters $-z_i$. This observation motivates us to propose the following approximation for the waiting time distribution. Define

$$\eta_\lambda = -\max_{i=1}^{m} \mathrm{Re}(z_i).$$

We have made the dependence of the roots, and hence of η, on the arrival rate λ explicit in the notation. Assume that $\eta_\lambda > 0$. The η_λ captures the dominant term, or more precisely the slowest decaying exponential, in the expression for the waiting time density. This leads us to approximate the waiting time density by

$$f_W(t) \approx (1-\rho)\delta(t) + \rho\eta_\lambda e^{-\eta_\lambda t},$$

which has the same form as in the $M/M/1$ case. It is easy to calculate the latency using this approximation. As the arrival rate increases to $k\lambda$ for k-fold replication, we see that the latency has cdf and mean given by

$$F_{L_k}(t) \approx 1 - (k\rho)^k e^{-k\eta_{k\lambda} t}, \quad \mathbb{E}[L_k] \approx \frac{(k\rho)^k}{k\eta_{k\lambda}}.$$

a) Example: We now illustrate this approach in the special case of a hyper-exponential service time distribution, namely, a mixture of exponentials. Consider a two-component mixture with density

$$f(t) = p\mu_1 e^{-\mu_1 t} + (1-p)\mu_2 e^{-\mu_2 t}.$$

This is a $PH(\boldsymbol{\alpha}, S)$ distribution with

$$\boldsymbol{\alpha} = \begin{pmatrix} p & 1-p \end{pmatrix}, \quad S = \begin{pmatrix} -\mu_1 & 0 \\ 0 & -\mu_2 \end{pmatrix},$$

and can be used to model highly variable service time distributions by taking μ_1 and μ_2 to be very far apart. Straightforward but tedious calculations now yield that the LST of the waiting time distribution is given by

$$W^*(s) = \frac{(1-\rho)(s+\mu_1)(s+\mu_2)}{(s+\mu_1)(s+\mu_2) - \lambda s - \lambda(p\mu_2 + (1-p)\mu_1)}, \tag{9}$$

where

$$\rho = \lambda\mathbb{E}[S] = \lambda\Big(\frac{p}{\mu_1} + \frac{1-p}{\mu_2}\Big).$$

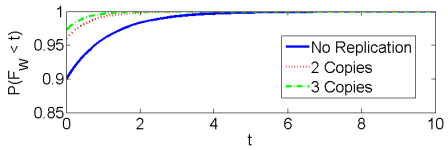
This can be inverted explicitly to get the waiting time density

$$f_W(t) = (1-\rho)\delta(t) + \frac{e^{-x_1 t} - e^{-x_2 t}}{z_1 - z_2},$$

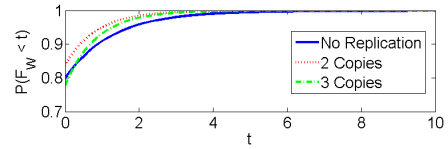
where z_1 and z_2 are the roots of the denominator polynomial in (9), and

$$x_i = -(1-\rho)z_i(z_i+\mu_1)(z_i+\mu_2).$$

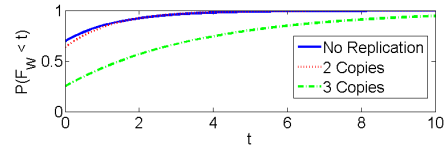
Note that $\eta_\lambda = \min\{x_1, x_2\}$ is the dominant exponential term in the waiting time distribution. To demonstrate the validity of the approximation Figure 2(d) shows the CDFs of the dominant exponential approximation and the exact density. The CDFs are very close in value and this is quantified in Table I which shows the mean and variation of the absolute difference between the two.



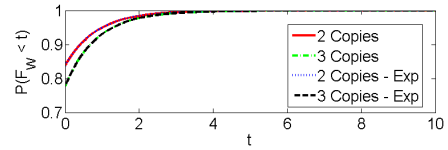
(a) $\lambda_k = 0.1$



(b) $\lambda_k = 0.2$



(c) $\lambda_k = 0.3$



(d) Dominant exp. approx. - $\lambda_k = 0.2$

Fig. 2. CDF of the wait time with Hyper-Exponential service

TABLE I
ABSOLUTE DIFFERENCE - HYPER-EXPONENTIAL

Mean	No Replication	2 Copies	3 Copies
$\lambda_k = 0.1$	0.00055	0.000014	0.0000044
$\lambda_k = 0.2$	0.0013	0.00014	0.00025
$\lambda_k = 0.3$	0.0023	0.0011	0.072
Variance	No Replication	2 Copies	3 Copies
$\lambda_k = 0.1$	$0.63\text{x}10^{-7}$	$0.0017\text{x}10^{-7}$	$0.00078\text{x}10^{-7}$
$\lambda_k = 0.2$	$3.4\text{x}10^{-7}$	$0.11\text{x}10^{-7}$	$1.0\text{x}10^{-7}$
$\lambda_k = 0.3$	$11\ \text{x}10^{-7}$	$3.5\text{x}10^{-7}$	$15000\text{x}10^{-7}$

It is straightforward to calculate the latency when replication is employed, using the explicit waiting time density given above. The corresponding cdfs are plotted in Figure 2, for the parameters $\mu_1 = 1$, $\mu_2 = 0.1$, $p = 0.999$. The results are similar to those in the exponential case, and show that the benefits of replication are most apparent at low loads.

C. Channel or server variability

We now return to the model of exponential service times but, motivated by wireless channels, whose capacity varies over time due to phenomena such as fading, assume that the server works at a variable rate. We model the service rate as a Markov process that evolves independently of the arrival process. Let R denote the rate matrix (generator) of this Markov process, on a finite state space $\{1, 2, \ldots, m\}$, and let μ_i denote the service

rate when the Markov process is in state i. This model shares many similarities with the phase-type service distribution. In fact, conditional on the initial state of the Markov process, the service time is of phase type. However, the initial state itself is dependent on the length of the busy period that has elapsed, and so service times are not iid and the P-K formula cannot be applied. The model can be analysed using matrix-geometric methods, but rather than introduce those here, we consider a special case that is of interest in its own right, and which can be handled by transform methods.

This is the case of an On-Off channel or server. We denote by α the rate at which the channel goes from the Off to the On state, and by β the rate for going from On to Off. Thus, the invariant or steady-state probability of being in the On state is $\alpha/(\alpha+\beta)$. The service rate in the On state is denoted μ; it is 0 in the Off state. Note that the channel necessarily has to be in the On state at a service completion time. Hence, a customer that is at the head of the queue when a service completes starts its service immediately thereafter. We can work out the service time of this customer easily. The minimum of the times to service completion or the channel going Off has an Exp($\mu+\beta$) distribution. With probability $\mu/(\mu+\beta)$, this time corresponds to a service completion. With the residual probability, the customer has to wait a further Exp(α) time for the channel to return to the On state, and resume service. On resumption, the customer needs a further Exp(μ) service time, due to the memoryless property of the exponential distribution. This description lends itself to an easy calculation of the LST of the service time distribution, which turns out to be

$$g(s) = \frac{\frac{\mu}{\mu+\beta}\frac{\mu}{\mu+s}}{1-\frac{\beta}{\mu+\beta+s}\frac{\alpha}{\alpha+s}}. \tag{10}$$

Detailed derivations are omitted for space reasons.

Now, the above description applies to all customers served during a busy cycle except the one that initiated it. That customer, which entered an empty queue, had a non-zero probability of arriving when the channel was Off, and having to wait to begin service. Consequently, its service time LST is given by

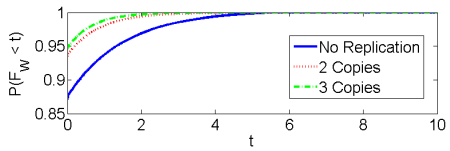
$$\begin{aligned} g_0(s) &= (1-q)g(s) + \frac{\alpha q}{\alpha+s}g(s), \\ \text{where } \; q &= \frac{\beta}{\alpha+\beta+\lambda}. \end{aligned} \tag{11}$$

Here, q is the probability that the first customer to arrive after the queue becomes empty finds the channel in the Off state.

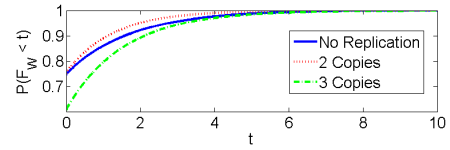
This model is known as the $M/G/1$ queue with Exceptional First Service, and an analogue of the P-K formula is given in [9] for the LST of the waiting time distribution:

$$\begin{aligned} W^*(s) &= A\frac{s-\lambda g_0(s)+\lambda g(s)}{s-\lambda+\lambda g(s)}, \\ \text{where } \; A &= \frac{\alpha(\mu-\lambda)(\mu+\beta)-\lambda\beta(\mu+\alpha+\beta)}{\mu(\mu+\beta)(\alpha+q\lambda)}, \end{aligned} \tag{12}$$

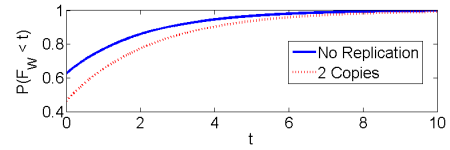
for q given in (11).



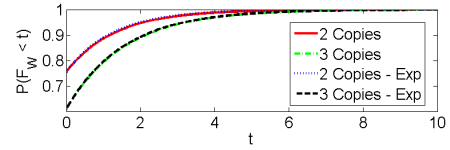
(a) $\lambda_k = 0.1$



(b) $\lambda_k = 0.2$



(c) $\lambda_k = 0.3$



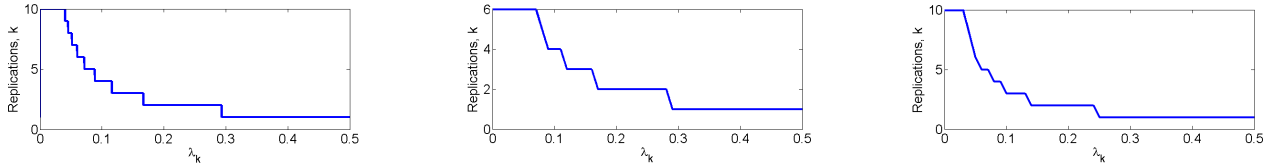
(d) Dominant exp. approx. - $\lambda_k = 0.2$

Fig. 3. CDF of the wait time with exceptional first service under

TABLE II
ABSOLUTE DIFFERENCE - EXCEPTIONAL FIRST SERVICE

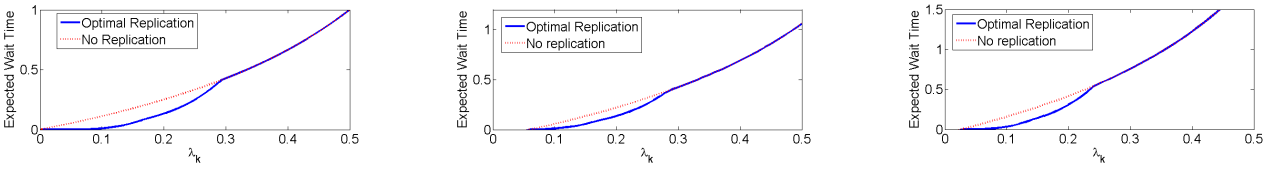
Mean	No Replication	2 Copies	3 Copies
$\lambda_k = 0.1$	0.0084	0.0009	0.0003
$\lambda_k = 0.2$	0.0119	0.0019	0.0012
$\lambda_k = 0.3$	0.0112	0.0022	N/A
Variance	No Replication	2 Copies	3 Copies
$\lambda_k = 0.1$	$3.858\text{x}10^{-7}$	$0.746\text{x}10^{-7}$	$0.225\text{x}10^{-7}$
$\lambda_k = 0.2$	$5.573\text{x}10^{-7}$	$2.228\text{x}10^{-7}$	$1.181\text{x}10^{-7}$
$\lambda_k = 0.3$	$4.899\text{x}10^{-7}$	$1.172\text{x}10^{-7}$	N/A

Substituting for g and g_0 from (10) and (11) in (12), we get an explicit, albeit complicated, expression for the LST of the waiting time. We do not display that expression here, but it can be inverted (with numerical root finding) to get an explicit expression for the waiting time distribution. This can be used in turn to obtain the latency after replication, using the same procedure as for exponential and phase-type service times. Also, similarly to the phase-type case, the inversion leads to an expression that is a sum of exponential terms, therefore a dominant exponential term can be found. The resulting CDFs for the latency and dominant exponential approximation are plotted in Figure 3, with the absolute difference between the approximation and exact distribution presented in Table II. The parameters chosen are $\alpha = 0.9$, $\beta = 0.1$ and $\mu = 1$. This corresponds to a channel that is available 90% of the time, but the Off period is ten times as long as the service time, on average. Thus, it exhibits considerable variability in the service time, like the hyperexponential model considered earlier.



(a) Exponential (b) Hyper-exponential (c) Exceptional First Service

Fig. 4. Optimal Number of Replications



(a) Exponential (b) Hyper-exponential (c) Exceptional First Service

Fig. 5. Optimal Mean vs. No Redundancy

IV. Optimal Replication Factor

A. Theory

Having developed models to represent the network an important factor to consider is by how much to replicate, since replicating by too much can cause excessive queuing caused by congestion, but not replicating enough will under utilise the available capacity. A simple way to look at the optimal replication factor is to look at the mean wait time whilst varying the load and number of replications. The optimal replication factor will then be the number of replicates that give the minimum mean wait time for a given load. This is the approach taken to produce Figures 4 and 5. Figure 4 shows a clear pattern, in that for a light load the number of replicates should be high, and for a heavy load the number of replicates should be reduced. To quantify the benefit of replication Figure 5 shows the optimal expected wait time, with the expected wait time for the unreplicated case also plotted for comparison.

A further point to consider is the limit to which replication reduces the expected wait time. Intuitively, redundancy should improve the wait time up to a load value of 0.5 since at this point replicating by the minimum amount, 2 copies, would cause the load to become 1. However, there is variance to the service time, hence the maximum load to which replication reduces the expected wait time is less than 0.5. A further point to consider is that the more variable cases of the hyper-exponential and exceptional first service models reduce the limit at which redundancy is useful.

B. Simulation

In order to utilise the information from section IV-A a simple dynamic redundancy policy can be created. In the method used in this paper an 'instantaneous arrival rate' is calculated to make the decision of by how much to replicate. A simple way to do this is to calculate the interarrival time between the current and previous arrivals, taking the inverse of this then gives the IAR (instantaneous arrival rate). Once the IAR is calculated, it is passed to a decision process that matches the IAR to a level of redundancy taken from Figure 4. In the simulation a cap of three replicates was used to represent a cap in the resources available. For the simulation, the arrivals are taken from a Poisson Process with the arrival rate set at 0.2 (0.2 was chosen as it is a more interesting case as seen in figure 1), and the service is exponential with rate 1.

The simulation results are shown in Figure 6 and agree with the theory in Section III-A. The CDF shows that the simple dynamic policy performs better in this instance than all the other policies. Furthermore, the extreme valued times are mainly seen by the no-redundancy case. Finally, it is important to note there is a consequence of replicating by too much. To see this, observe that there is a crossover between the double and triple redundancy in the CDF, this indicates the double redundancy outperforms the triple redundancy for larger values of t.

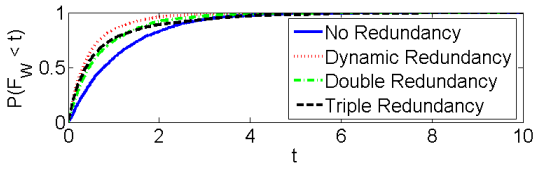


Fig. 6. CDF Comparison

V. Replication and Reliability

In some applications, the primary goal is not necessarily to reduce mean latency, but to guarantee packet delivery within a specified bound on the delay. Safety applications and virtual reality are two example situations which require delay guarantees to function correctly. In this paper, we consider probabilistic guarantees, i.e., we want to ensure that some high proportion of packets, e.g., 99% or 99.9% ($\epsilon = 0.01$

and $\epsilon = 0.001$ in the analysis), suffer delay no greater than a specified bound. We explore how redundancy can help achieve this goal.

We begin by observing that the moment generating function $M(\cdot)$ of the waiting time is closely related to its LST: we have

$$M_\lambda(\theta) := \mathbb{E}\left[e^{\theta W}\right] = W^*(-\theta),$$

for all $\theta \in \mathbb{R}$; we have made explicit the dependence of the waiting time distribution on the arrival rate λ in the notation for its mgf. Now, given a threshold τ for the acceptable latency, the probability of the waiting time exceeding τ satisfies Chernoff's bound: applying Markov's inequality to the random variable $e^{\theta W}$ for some $\theta > 0$, we get

$$\mathbb{P}(W > \tau) = \mathbb{P}\left(e^{\theta W} > e^{\theta\tau}\right) \leq e^{-\theta\tau} M_\lambda(\theta).$$

As this bound holds for all $\theta > 0$, the tightest bound is obtained by minimising over θ in this range. Taking logarithms, we can rewrite the above equation as

$$\begin{aligned} \log \mathbb{P}(W > \tau) &\leq -I_\lambda(\tau), \quad \text{where} \\ I_\lambda(\tau) &= \sup_{\theta>0} \ \theta\tau - \log M_\lambda(\theta). \end{aligned} \tag{13}$$

Thus, when k-fold replication is employed, it follows from (3) and (13) that the probability of the latency exceeding the threshold τ is bounded by

$$\log \mathbb{P}(L_k > \tau) \leq -k I_{k\lambda}(\tau). \tag{14}$$

We can now ask how large we need to choose the threshold τ in order to achieve a probabilistic guarantee, $\mathbb{P}(L_k > \tau) \leq \epsilon$, where ϵ is a specified small constant such as 0.01 or 0.001, which is the tolerance to excessive delays. Or, conversely this can be described by looking at $\mathbb{P}(L_k < \tau) \leq 1 - \epsilon$, which is the probability that the expected value L_k is less than τ for 99% ($\epsilon = 0.01$) or 99.9% ($\epsilon = 0.001$) of the time. It is immediate from (14) that the threshold τ should be chosen such that

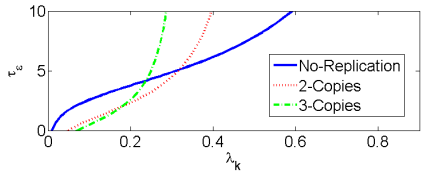
$$\log \frac{1}{\epsilon} = k I_{k\lambda}(\tau). \tag{15}$$

It is known that $I_\lambda(\cdot)$, defined in (13), is a monotone increasing function for any $\lambda > 0$, that $I_\lambda(\tau) = 0$ for all $\tau \leq \mathbb{E}[W]$ and that, if $M_\lambda(\theta)$ is finite for some $\theta > 0$, then $I_\lambda(x)$ tends to infinity as x tends to infinity(see [8]). Hence, equation (15) has a unique solution, which we denote τ_ϵ to make the dependence on ϵ explicit, provided $M_\lambda(\theta) < \infty$ for some $\theta > 0$; this condition is met in all our models, for λ small enough that the queue is stable.

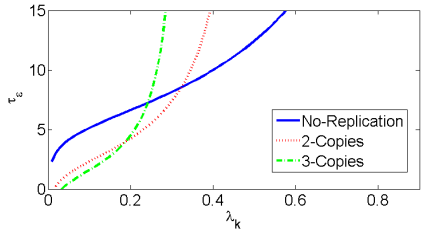
We can calculate τ_ϵ explicitly for the $M/M/1$ queue, using the exact expression for the latency cdf given in (7). Straightforward calculations yield that, for k-fold replication,

$$\tau_\epsilon = \frac{1}{k(\mu - k\lambda)} \log \frac{1}{\epsilon} - \frac{\log \frac{\mu}{k\lambda}}{\mu - k\lambda}.$$

For the other models, we obtained numerical expressions based on the LST. The results for $\epsilon = 0.01$ and 0.001 are shown in Figure 7 for the exponential case and Tables III & IV for the hyper-exponential and exceptional first service



(a) $\epsilon = 0.01$



(b) $\epsilon = 0.001$

Fig. 7. τ_ϵ, with varied λ_k

TABLE III
HYPER-EXPONENTIAL, τ_ϵ

$\tau_\epsilon, \epsilon = 0.01$	No Replication	2 Copies	3 Copies
$\lambda = 0.1$	2.6100	0.8880	**0.4920**
$\lambda = 0.2$	3.9370	**2.3930**	2.6790
$\lambda = 0.3$	5.2670	**4.7490**	N/A
$\lambda = 0.4$	**6.9010**	11.6600	N/A
$\lambda = 0.5$	**9.1470**	N/A	N/A
$\tau_\epsilon, \epsilon = 0.001$	No Replication	2 Copies	3 Copies
$\lambda_k = 0.1$	5.4440	2.3720	**1.6140**
$\lambda_k = 0.2$	8.6230	**4.4810**	4.7490
$\lambda_k = 0.3$	13.5750	**8.1490**	N/A
$\lambda_k = 0.4$	**19.0150**	19.2210	N/A
$\lambda_k = 0.5$	**24.4600**	N/A	N/A

TABLE IV
EXCEPTIONAL FIRST SERVICE, τ_ϵ

$\tau_\epsilon, \epsilon = 0.01$	No Replication	2 Copies	3 Copies
$\lambda = 0.1$	3.4700	1.5900	**1.1300**
$\lambda = 0.2$	5.3400	**4.0600**	5.7200
$\lambda = 0.3$	**7.2400**	9.3200	N/A
$\lambda = 0.4$	**9.6800**	>30	N/A
$\lambda = 0.5$	**13.3400**	N/A	N/A
$\tau_\epsilon, \epsilon = 0.001$	No Replication	2 Copies	3 Copies
$\lambda = 0.1$	5.1400	3.5600	**2.7200**
$\lambda = 0.2$	7.5000	**6.9500**	9.3200
$\lambda = 0.3$	**10.1000**	14.6800	N/A
$\lambda = 0.4$	**13.6100**	>30	N/A
$\lambda = 0.5$	**19.0500**	N/A	N/A

cases. Note that N/A is displayed in the tables when the traffic intensity exceeded 1. An important aspect of these results is that there are crossover points, where a higher level of redundancy does not necessarily provide better reliability in terms of delay (the best policy is highlighted in bold).

VI. CONCLUSION AND FURTHER WORK

In this paper we studied the effect that redundancy has on the delay in a communications network. We generalised the procedure for finding the delay distributions by utilising the P-K formula and the LST for different service models. We found that the expressions for the distributions can be very complex, but that a simple approximation using a dominant

exponential term led to satisfying results. The analysis of the distributions indicated that an optimal policy is dependent upon the network load. Simulations then showed that a simple dynamic replication policy can be used to further reduce the latency in a system. Furthermore, the analysis highlighted the benefit of redundancy in the tail of the distribution. This motivated the need for the formalisation of the reliability of the delay and led to the analysis of τ_ϵ, which gives a probabilistic guarantee that a delay, τ_ϵ, is only exceeded with probability ϵ. The importance of τ_ϵ is most prevalent in latency sensitive applications since a guarantee for the delay may be needed. We found that by altering the amount of redundancy the reliability of the delay can be influenced. However, as with the minimisation of the mean delay, it is not always advantageous to replicate by the highest amount and thus an optimal and more efficient level of replication can be realised. Overall, we believe our work highlights the advantages of redundancy and should therefore be a consideration in the future design of wireless systems.

Further work for this research includes extending the model to a more realistic scenario. For example, the work by Kristen Gardner et al. in [10] breaks the assumption that the packet size is independent across paths, but instead maintains a single variable for the packet size and uses independent variables for the service rate at each server. Another extension would be to try and model a general arrival process, but this would increase the complexity dramatically and would likely require simulated results rather than analytic. Similarly, the complexity of the phase type service distribution example could be increased by introducing a larger number of available paths. A Massive MIMO (Multiple Input Multiple Output) system would be a specific use case for this model, since a large number of paths are available. A similar research area is that of delay analysis in network coding [11] [12]. In this case the load can be controlled more finely by changing the coding rate (the ratio of data sent to non-redundant data), but the receiver must wait for multiple packets to be received to reconstruct the message. Thus, the delay will not be minimised but we leave it to further work to study the effects more closely.

ACKNOWLEDGMENT

This work was supported by the Engineering and Physical Sciences Research Council [grant numbers EP/I028153/1]; Thales UK; and the University of Bristol. We would like to thank Thales UK Research & Technology for supporting this research through an EPSRC CASE Award

REFERENCES

[1] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck, "A close examination of performance and power characteristics of 4g lte networks," in *Proc. of the 10th Int. Conf. on Mobile Systems, Applications, and Services*, ser. MobiSys '12, 2012, pp. 225–238.

[2] N. F. Maxemchuk, "Dispersity routing," in *Proc. of ICC*, vol. 75, 1975, pp. 41–10.

[3] ——, "Dispersity routing in high-speed networks," *computer networks and ISDN systems*, vol. 25, no. 6, pp. 645–661, 1993.

[4] A. Vulimiri, P. B. Godfrey, R. Mittal, J. Sherry, S. Ratnasamy, and S. Shenker, "Low latency via redundancy," in *Proc. of the 9th ACM conf. on Emerging networking experiments and technologies*. ACM, 2013, pp. 283–294.

[5] K. Gardner, S. Zbarsky, S. Doroudi, M. Harchol-Balter, and E. Hyytia, "Reducing latency via redundant requests: Exact analysis," *ACM SIGMETRICS Perf. Eval. Review*, vol. 43, no. 1, pp. 347–360, 2015.

[6] N. B. Shah, K. Lee, and K. Ramchandran, "When do redundant requests reduce latency?" *IEEE Trans. on Comms*, vol. 64, no. 2, pp. 715–722, 2016.

[7] J. N. Daigle, "The basic m/g/1 queueing system," *Queueing Theory with Applications to Packet Telecommunication*, pp. 159–223, 2005.

[8] A. J. Ganesh, N. O'Connell, and D. J. Wischik, *Big queues*. Springer, 2004.

[9] H. Takagi, *Queueing analysis: a foundation of performance evaluation, vol. 1 : vacation and priority systems*, ser. Queuing Analysis.

[10] K. Gardner, M. Harchol-Balter, A. Scheller-Wolf, and B. Van Houdt, "A better model for job redundancy: Decoupling server slowdown and job size," *MASCOTS*, 2016.

[11] D. E. Lucani, M. Medard, and M. Stojanovic, "On coding for delaynetwork coding for time-division duplexing," *IEEE Trans. on Info. Theory*, vol. 58, no. 4, pp. 2330–2348, 2012.

[12] G. Joshi, Y. Liu, and E. Soljanin, "Coding for fast content download," in *2012 50th Annual Allerton Conf. on Communication, Control, and Computing*. IEEE, 2012, pp. 326–333.