The Math Citadel Review
Original Research Article

Extensions of the Single Server Efficiency Model

Rachel Traylor*, Ph.D.

Abstract
This paper comprises the third chapter of the PhD dissertation by Rachel Traylor. Herein we further generalize the single server model presented in [3]. In particular, we consider a multichannel server under the cases of both singular and clustered tasks. In the instance of singular tasks, we present a load balancing allocation scheme and obtain a stochastic breakdown rate process, as well as derive the conditional survival function as a result. The model of a multichannel server taking in clustered tasks gives rise to two possibilities, namely, independent and correlated channels. We derive survival functions for both of these scenarios.

Keywords
reliability theory — efficiency — service life distribution — probability theory

Contents

1. Load Balancing Allocation for a Multichannel Server

1.1 Model Description

Previously, we had assumed that a web server functions as a single queue that attempts to process jobs as soon as they arrive. These jobs originally brought a constant stress η to the server, with the system stress reducing by η at the completion of each job.

Now, suppose we have a server partitioned into K channels. Denote each channel as Q_k, $k = 1, \ldots, K$. Jobs arrive via a nonhomogenous Poisson process with rate $\lambda(t)$. Upon arrival, each job falls (or is routed)

*The Math Citadel

to the channel with the shortest queue length. If all queue lengths are equal or multiple channels have the shortest length, the job will enter one of the appropriate queues with equal probability.

We retain the previous notation for the baseline breakdown rate, or hazard function. This is denoted by $r_0(t)$ and is the hazard function under an idle system. We also retain the assumption that the arrival times $\mathbf{T}$ are independent. In addition, the service times $\mathfrak{W}$ are i.i.d. with distribution $G_W(w)$. We assume that all channels are serving jobs at the same time, i.e. a job can be completed from any queue at any time. We do not require load balancing for service. In other words, any queue can empty with others still backlogged. We also retain the FIFO service policy for each queue.

Since we have now "balanced", or distributed, the load of jobs in the server, not all jobs will cause additional stress to the system. Suppose all jobs bring the same constant stress η upon arrival. Under load balancing, we will define the additional stress to the system as $\eta \max_k |Q_k|$. Figure 1 shows an example server with current stress of 4η.

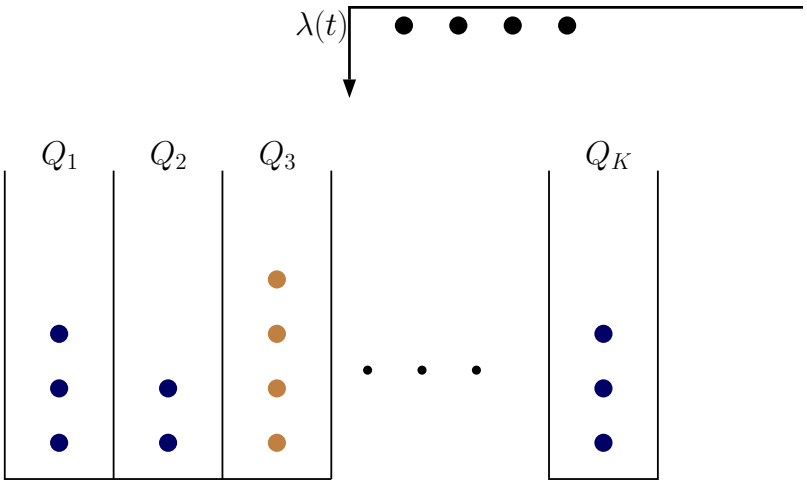
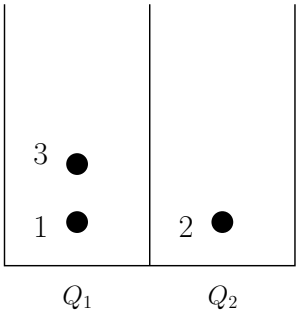


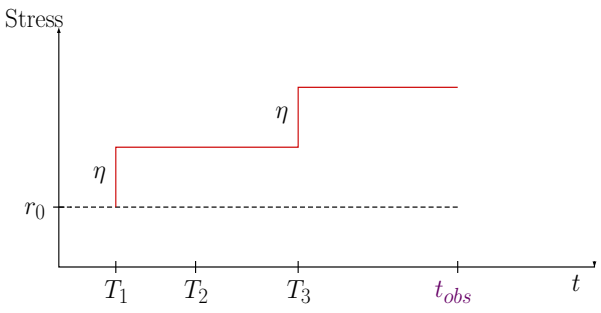
Figure 1. Partitioned Server with Load Balancing

1.2 Examples

Due to the dynamic nature of arrival times, allocation to queues, and service times, we have many possible configurations of jobs at any point in time. Therefore, the allocation scheme adds an additional layer of variation to the service times and order of service. The placement of jobs in the various queues (and thus the order of service and service times) is wholly dependent on all arrival times and service times of the prior arrivals. The following examples illustrate the effect on the workload stress added to the system in various scenarios.



(a) Queue Configuration



(b) Breakdown Rate Process Trajectory

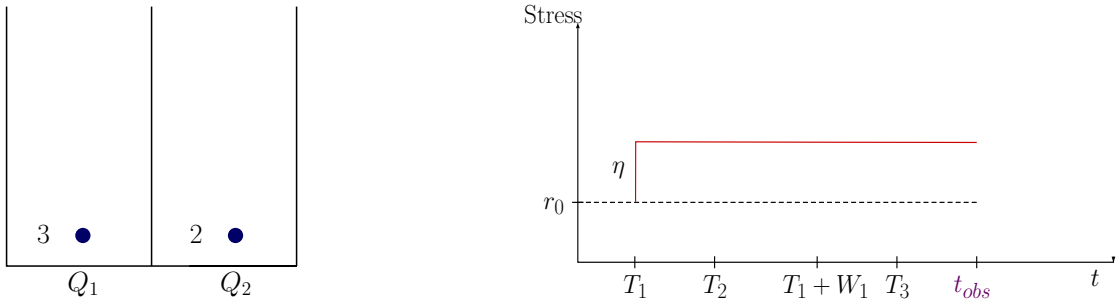
Figure 2. Example 4.1

Example 1. *Suppose for simplicity we have 2 channels. Suppose at the time of observation of the system, 3 jobs*

have arrived and none have finished. WLOG, suppose job 3 fell into Q_1. See Figure 2a. The stress to the system at $t = t_{obs}$ is $r_0(t_{obs}) + 2\eta$, as shown in Figure 2b.

Note in example 1 that Job 2 does not add any additional stress to the system. Job 1 sees an empty queue upon arrival, and $\max_K |Q_K| = 1$ when it falls into any particular queue. Job 2 arrives as Job 1 is still being processed, and thus the placement of Job 1 forces Job 2 into the empty channel. Since $\max_K |Q_K|$ is still 1, the stress to the system doesn't change. Job 3 arrives as Jobs 1 and 2 are in service, and thus its choice of queue is irrelevant due to the configuration of the two queues at T_3. Regardless of which queue Job 3 falls into, $\max_K |Q_K| = 2$. Thus the arrival of Job 3 increases the breakdown rate by η again.

The next example shows the change in system stress Job 1 from Example 1 when one job has finished processing before T_3.



(a) Queue Configuration **(b)** Breakdown Rate Process Trajectory

Figure 3. Example 4.2

Example 2. *Consider the same two-channel system from Example 1. However, now suppose WLOG that $T_3 < T_1 + W_1$. In other words, service for Job 1 was completed before Job 3 arrived. Hence Job 3 will fall into the opposite queue as Job 2. The stress to the system at the time of observation would be $r_0(t) + \eta$. See Figures 3a and 3b.*

In this scenario, the workload due to Job 3 does not contribute any additional stress to the server. Also observe that upon completion of Job 1, the workload stress to the server does not decrease, as Job 2 still resides in the system and is being served.

Contrast this behavior with the breakdown rate process given in [3]. In the single-channel, single-server model described in both [1] and [3], each job adds stress to the server upon arrival. Under the load balancing allocation scheme, the additional stress to the server depends on the arrival and service times of all prior jobs. From a stochastic perspective, this breakdown rate process has full memory.

The examples above illustrate that $\max_K |Q_K|$ depends on the intersection of the intervals $I_j = [T_j, T_j + W_j]$, $j = 1, \ldots, N(t)$. The next section details the methodology to obtain the configuration of jobs in the server at time t by deccomposition of $\bigcup_{j=1}^{N(t)} I_j$ into disjoint atoms and derives the stochastic breakdown rate process under the load balancing allocation scheme.

1.3 Breakdown Rate Process and Conditional Survival Function

Let $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_{N(t)})$ be a $N(t)-$tuple whose components $\varepsilon_j \in \{\emptyset, c\}$, where $\emptyset$ denotes the empty set, and c denotes the complement of the set. Let $E = \{\varepsilon : \varepsilon_j \in \{\emptyset, c\}\}$ denote the set of all possible ε, excepting $\varepsilon = (c, \ldots, c)$. Then by Lemma 1 (*see Section 4*),

$$\bigcup_{j=1}^{N(t)} I_j = \bigcup_{\varepsilon \in E} \bigcap_{j=1}^{N(t)} I_j^{\varepsilon_j} \tag{1}$$

Remark: $\cap_{j=1}^{N(t)} I_j^{\varepsilon_j}$ indicates which jobs are still in the server at time t. The union is disjoint; thus only one ε will describe the server configuration at any given time t. For example, if 3 jobs have arrived to the server at time t_{obs}, $|E| = 3 \times 2 - 1 = 5$. These may be enumerated:

$\cdot I_1 \cap I_2 \cap I_3$ $\quad$ $\cdot I_1^c \cap I_2 \cap I_3$
$\cdot I_1^c \cap I_2^c \cap I_3$ $\quad$ $\cdot I_1^c \cap I_2 \cap I_3^c$
$\cdot I_1 \cap I_2^c \cap I_3^c$

As an illustration, refer to Example 1. All three jobs are in the system at $t = t_{\text{obs}}$ (that is, none have completed service), and thus $t_{\text{obs}} \in I_1 \cap I_2 \cap I_3$. Expanding, $t_{\text{obs}} \in [T_1, T_1 + W_1], [T_2, T_2 + W_2]$, and $[T_3, T_3 + W_3]$.

Compare the case with that of Example 2. In this case, three jobs have arrived at $t = t_{\text{obs}}$, but Job 1 has finished by t_{obs}. Thus $t_{\text{obs}} \notin I_1$, but since Jobs 2 and 3 are still in the system, $t_{\text{obs}} \in I_2 \cap I_3$. Thus $t_{\text{obs}} \in I_1^c \cap I_2 \cap I_3$.

Now, since the additional workload stress to the server is a multiple of $\eta \max_K |Q_K|$, it remains to derive the appropriate multiplier that accounts for the number of jobs that contribute additional stress to the system.

Let $n = \sum_{j=1}^{N(t)} \mathbb{1}(\varepsilon_j = \emptyset | \varepsilon_j \in \varepsilon)$ for a particular ε, and let α_ε be the multiplier that indicates the number of jobs that contribute stress η to the system. Under [1] and the generalization in [3], every uncompleted job in the system contributes stress, thus $\alpha_\varepsilon = n$.

Under the load balancing scheme, $\alpha_\varepsilon = \lfloor \frac{n+1}{K} \rfloor$, where K is the number of channels in the server. This is due to the allocation scheme's attempts to evenly distribute jobs across channels. Thus, for Example 1, $n = 3$, and $K = 2$, meaning $\alpha_\varepsilon = 2$, as illustrated in Figure 2b and for Example 2, $\alpha_\varepsilon = \lfloor \frac{3+1}{2} \rfloor = 1$, as in Figure 3b.

Then, the stochastic breakdown rate process under the load balancing allocation scheme is given by

$$\mathcal{B}(t) = r_0(t) + \eta \sum_{\varepsilon \in E} \alpha_\varepsilon \mathbb{1}_{I_1^{\varepsilon_1} \cap I_2^{\varepsilon_2} \cap I_{N(t)}^{\varepsilon_{N(t)}}}(t)$$

Under this expression, only one indicator function will be nonzero at any given point in time, since all atoms are disjoint. Now, $I_1^{\varepsilon_1} \cap I_2^{\varepsilon_2} \cap \ldots \cap I_{N(t)}^{\varepsilon_{N(t)}}$ may be expressed as one interval $[L_\varepsilon, R_\varepsilon]$, where

$$L_\varepsilon = \max\left(\{T_j : \varepsilon_j = \emptyset\}_{j=1}^{N(t)}\right);$$
$$R_\varepsilon = \min\left(\{T_j + W_j : \varepsilon_j = \emptyset\}_{j=1}^{N(t)}, \{T_j : \varepsilon_j = c\}_{j=1}^{N(t)}\right)$$

Thus, for a server with K channels under a load balancing routing scheme with all jobs bringing constant stress η, the breakdown rate process $\mathcal{B}(t)$ may be expressed as

$$\mathcal{B}(t) = r_0(t) + \eta \sum_{\varepsilon \in E} \alpha_\varepsilon \mathbb{1}_{[L_\varepsilon, R_\varepsilon]}(t) \tag{2}$$

Thus, the conditional survival function under the load balancing scheme is given by

$$\begin{aligned} S_{Y|\mathfrak{T},\mathfrak{W},N(t)}(t|\mathfrak{t},\mathfrak{w},n) &= e^{-\int_0^t \mathcal{B}(s)ds} \\ &= \bar{F}_0(t)\exp\left(-\eta\int_0^t \sum_{\varepsilon\in E}\alpha_\varepsilon \mathbb{1}_{[L_\varepsilon,R_\varepsilon]}(s)ds\right) \\ &= \bar{F}_0(t)\exp\left(-\eta\sum_{\varepsilon\in E}\alpha_\varepsilon \min(t-L_\varepsilon,R_\varepsilon)\right) \end{aligned} \tag{3}$$

1.4 Remarks

Finding the survival function of the single-channel environment relied on the independence of the set of arrival times and service times. From (3), the independence is clearly lost. As noted before, the random breakdown process has full memory, and thus is completely dependent upon the entire trajectory up to $t = t_{\text{obs}}$.

2. Clustered Tasks in a Multichannel Server

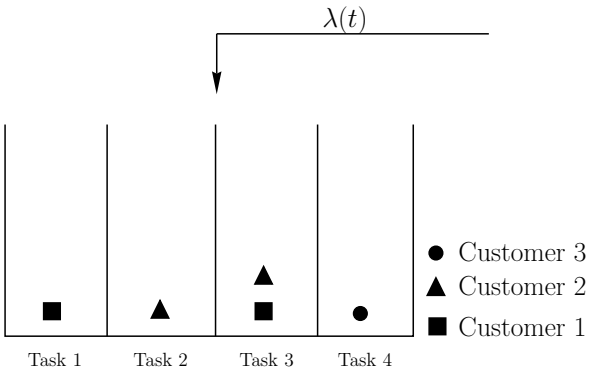


Figure 4. Illustration of Clustered Tasks in a Multichannel Server

The previous multichannel server model in Section 1 implicitly assumed each job comes with one task, and all channels are identical in their ability to serve any task brought by a job. A classic illustration is a block of registers at a retail establishment. Each customer will survey the length of the various queues at each register before choosing the shortest queue. Viewing each of these separate registers as a channel in a single server under these conditions gave rise to the load balancing allocation model detailed in the previous section. This section presents a different interpretation of a multichannel, single-server model.

Suppose a server has multiple channels $Q_1,\ldots,Q_K$, but each channel serves a different type of task. A customer arrives to the server and may select any number from 0 to K tasks for the server to perform. Said customer will select each possible task j with probability p_j. Figure 4 illustrates an example of such a situation in which three customers visit the server and each customer picks a different number and set of tasks at random. A customer is considered fully serviced (i.e. the job is complete) upon completion of the last task belonging to that particular customer.

2.1 Model Assumptions

The following mathematical assumptions are made for the multichannel server with clustered tasks:

(i) Customers arrive to the server with K channels via a nonhomogenous Poisson process (NHPP) with intensity $\lambda(t)$.

(ii) The breakdown rate of the idle server is given by $r_0(t)$.

(iii) Each channel corresponds to a different task the server can perform.

(iv) The selection of each task is a Bernoulli random variable with probability p_k. Thus the number of tasks selected by each customer is a binomial random variable.

(v) The workload stress to the server is a constant multiple η of the number of tasks requested by the customer, i.e. the additional stress is given by ηN, where N is the number of tasks requested.

(vi) The PDF of each channel's service time is given by $g_i(w)$, $i = 1, \ldots, K$. Since the customer's service is not complete until all requested tasks have finished, the service life distribution for the customers is given by $\max_i G_i(w)$.

Under these assumptions, this model is a special interpretation of the random stress environment developed in [3]. In this case, the random workload stress is ηN, where N is a binomial random variable, and the service life distribution $G_W(w) = \max_i G_i(w)$, which may be easily obtained through the mathematical properties of order statistics. Two variations are considered in this section: independent channels and correlated channels.

2.2 Independent Channels in a Clustered Task Server

Suppose the selection probabilities for each task in the server are identical, that is, $p_1 = p_2 = \ldots = p_K = p$. Then $N \sim \text{Bin}(K, p)$. Using Theorem 3 in [3], the survival function of the multichannel server is given in the following theorem:

Theorem 1 (Survival Function of Multichannel Server with Clustered Tasks and Independent Channels)**.** *Suppose conditions (i)-(vi) above are satisfied. In addition, assume $p_1 = p_2 = \ldots = p_K = p$. Then the survival function of the server is given by*

$$S_Y(t) = \bar{F}_0(t) \exp\left(-K\eta \left[e^{-\eta t} \left(1 - p + p e^{-\eta t}\right)^{K-1} - p(1-p)^{K-1} \right] \int_0^t m(t-w) \bar{G}_W(w) dw \right)$$

where $m(x) = \int_0^x \lambda(s) ds$, $\bar{F}_0(t) = e^{-\int_0^t r_0(s) ds}$, $\bar{G}_W(w) = 1 - G_W(w)$, and $G_W(w) = \max_i G_i(w)$.

Proof. Since $p_1 = \ldots = p_K = p$, the number of tasks selected by any particular customer $N \sim \text{Bin}(K, p)$. Then the $\mathcal{H}$ from Theorem 3 in [3] is given by $\mathcal{H} = \eta N$. Thus

$$S_Y(t) = \bar{F}_0(t) \exp\left(-E_{\mathcal{H}} \left[\mathcal{H} \int_0^t e^{-\mathcal{H}w} m(t-w) \bar{G}_W(w) dw \right] \right)$$

In this case,

$$\begin{aligned}
&E\left[\mathcal{H} \int_0^t e^{-\mathcal{H}w} m(t-w) \bar{G}_W(w) dw \right] \\
&\qquad = E\left[\eta N \int_0^t e^{-\eta N w} m(t-w) \bar{G}_W(w) dw \right] \\
&\qquad = \sum_{n=0}^{K} \left[\eta n \int_0^t e^{-\eta n w} m(t-w) \bar{G}_W(w) dw \right] \cdot P(N = n) \\
&\qquad = \sum_{n=0}^{K} \left[\eta n \int_0^t e^{-\eta n w} m(t-w) \bar{G}_W(w) dw \right] \binom{K}{n} p^n (1-p)^{K-n} \\
&\qquad = \eta \int_0^t m(t-w) \bar{G}_W(w) \left(\sum_{n=0}^{K} n e^{-\eta n w} \binom{K}{n} p^n (1-p)^{K-n} \right) dw
\end{aligned}$$

Now,

$$\begin{aligned}\sum_{n=0}^{K} ne^{-\eta nw}\binom{K}{n}p^n(1-p)^{K-n} &= \sum_{n=0}^{K}\frac{K!}{(K-n)!n!}ne^{-\eta nw}p^n(1-p)^{K-n}\\ &= \sum_{n=0}^{K}\frac{K(K-1)!}{(n-1)!(K-1-(n-1))!}e^{-\eta nw}p^n(1-p)^{K-n}\\ &= \sum_{n=0}^{K}K\binom{K-1}{n-1}e^{-\eta nw}p^n(1-p)^{K-n}\end{aligned}$$

Making a change of indices, let $j = n-1$. Then

$$\sum_{n=0}^{K}K\binom{K-1}{n-1}e^{-\eta nw}p^n(1-p)^{K-n} = K\sum_{j=0}^{K-1}\binom{K-1}{j}p^{j+1}(1-p)^{K-(j+1)}e^{-\eta(j+1)w}$$

Note the above resembles a scaled and shifted moment generating function of a binomial random variable. Let $X \sim \text{Bin}(K-1,p)$. Then

$$\begin{aligned}K\sum_{j=0}^{K-1}\binom{K-1}{j}p^{j+1}(1-p)^{K-(j+1)}e^{-\eta(j+1)w} &= K\left(E\left[e^{-\eta(X+1)t}\right] - P(X=0)\right)\\ &= K\left(e^{-\eta t}E\left[e^{-\eta Xt} - p(1-p)^{K-1}\right]\right)\\ &= K\left(e^{-\eta t}\left[1-p+pe^{-\eta t}\right]^{K-1} - p(1-p)^{K-1}\right)\end{aligned}$$

Thus,

$$S_Y(t) = \bar{F}_0(t)\exp\left(-K\eta\left[e^{-\eta t}\left(1-p+pe^{-\eta t}\right)^{K-1} - p(1-p)^{K-1}\right]\int_0^t m(t-w)\bar{G}_W(w)dw\right)$$

□

2.3 Correlated Channels in a Cluster Server

Now suppose the server tasks are correlated, in that the selection of one particular task may affect the selection of any or all of the other tasks. Thus the channels are a sequence of dependent Bernoulli random variables. The construction of dependent Bernoulli random variables is given in [2], and a summary is given.

2.3.1 Dependent Bernoulli Random Variables and the Generalized Binomial Distribution

Korzenwioski [2] constructs a sequence of dependent Bernoulli random variables using a binary tree that distributes probability mass over dyadic partitions of $[0,1]$. Let $0 \le \delta \le 1$, $0 < p < 1$, and $q = 1-p$. Then define the following quantities:

$$\begin{aligned} q^+ &:= q + \delta p & p^+ &:= p + \delta q\\ q^- &:= q(1-\delta) & p^- &:= p(1-\delta)\end{aligned} \tag{4}$$

The quantities in (4) satisfy the following conditions:

$$\begin{aligned} &q^+ + p^- = q^- + p^+ = q + p = 1\\ &qq^+ + pq^- = q, \quad qp^- + pp^+ = 1\end{aligned} \tag{5}$$

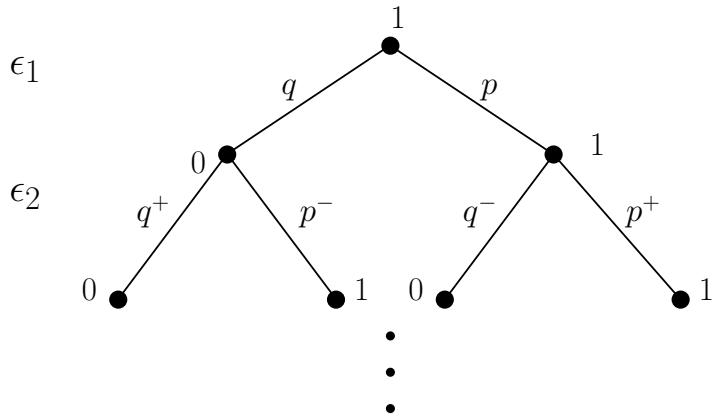


Figure 5. Construction of Dependent Bernoulli Random Variables

Figure 5 shows the construction shows the dependencies. The following examples using coin flips illustrate the effect of the dependency coefficient δ:

Example 3 ($\delta = 1$)**.** *For $\delta = 1$, $q^+ = q + p = 1, q^- = 0$, $p^+ = p + q = 1$, and $p^- = 0$. Supposing the first coin flip $\varepsilon_1 = 1$. Then every successive ε_i will also be 1. Similarly if $\varepsilon_1 = 0$. Thus the result of the first coin flip completely determines the outcomes of all the rest.*

Example 4 ($\delta = 0$)**.** *For $\delta = 0$, $q^+ = q^- = q$, and $p^+ = p^- = p$. Thus, the first coin flip (and all subsequent ones) have no effect on the ones that follow.*

Example 5 ($\delta = \frac{1}{4}$)**.** *Suppose $p = q = \frac{1}{2}$. Then $p^+ = q^+ = \frac{5}{8}$, and $p^- = q^- = \frac{3}{8}$. Then the subsequent outcomes $\varepsilon_i, i \geq 2$ are more likely to match the outcomes of ε_1 than not.*

Now suppose $p = \frac{1}{4}, q = \frac{3}{4}$. Then $p^+ = \frac{7}{16}, p^- = \frac{3}{16}, q^+ = \frac{13}{16}$, and $q^- = \frac{9}{16}$. In this example of an unfair coin, the dependency coefficient δ still attempts to skew the results following the first coin flip in favor of the outcome of ε_1. However, the dependency here heightens the effect of $\varepsilon_1 = 0$ on subsequent flips, and cannot overcome the discrepancy between the probability of success and failure to skew $\varepsilon_i, i \geq 2$ in favor of a 1 following the outcome of $\varepsilon_1 = 1$.

Using these dependent Bernoulli random variables, [2] presents a Generalized Binomial Distribution for identically distributed but dependent Bernoulli random variables.

Generalized Binomial Distribution

Let $X = \sum_{i=1}^{n} \varepsilon_i$, where $\varepsilon_i, i = 1, \ldots, n$ are identically distributed Bernoulli random variables with probability of success p and dependency coefficient δ. Then

$$P(X = k) = q\binom{n-1}{k}(p^-)^k(q^+)^{n-1-k} + p\binom{n-1}{k-1}(p^+)^{k-1}(q^-)^{n-1-(k-1)} \tag{6}$$

2.3.2 Survival Function of Correlated Channels in a Cluster Server

Suppose the selection of tasks may be modeled by the dependent Bernoulli random variables given in the previous section. That is, suppose the customer selects Tasks 1-K in sequence, and the selection or rejection of Task 1 affects all subsequent tasks by a dependency coefficient δ. From [2], the correlation between task selections $\varepsilon_i, \varepsilon_j$ is given by

$$\rho = \text{Cor}(\varepsilon_i, \varepsilon_j) = \begin{cases} \delta, & i = 1; j = 2, \ldots, K \\ \delta^2, & i \neq j; i, j \geq 2 \end{cases} \tag{7}$$

This illustrates the dependency of Tasks 2-K on the outcome of Task 1, and notes that while Tasks 2-K are still correlated with each other, the dependency is much lower. In a similar fashion to the independent channel server, the survival function is derived.

Theorem 2 (Survival Function of Multichannel Server with Clustered Tasks and Dependent Channels)**.** *Suppose conditions (i)-(vi) above are satisfied. In addition, suppose the selection of channels $1-K$ are determined by identically distributed Bernoulli random variables with dependency coefficient δ as defined in [2]. Then the survival function of the server is given by*

$$S_Y(t) = \bar{F}_0(t) \exp\left(-\eta \int_0^t m(t-w)\bar{G}_W(w)\mathscr{S}(w)dw\right) \tag{8}$$

where $m(x) = \int_0^x \lambda(s)ds$, and

$$\begin{aligned}\mathscr{S}(w) = &\sum_{n=0}^{K} e^{-\eta n w} \sum_{j=0}^{K-n-1} \binom{K-1}{n-1, j, K-1-n-j} p^{K-1-j}(1-p)^{j+1}\delta^{K-1-n-j}(1-\delta)^n \\ &+ \sum_{n=0}^{K} n e^{-\eta n w} \sum_{i=0}^{n-1} \binom{K-1}{K-1-n, i, n-1-i} p^{i+1}(1-p)^{K-n}\delta^{n-1-j}(1-\delta)^{K-n-j}\end{aligned}$$

Proof. By Theorem 3 in [3],

$$S_Y(t) = \bar{F}_0(t) \exp\left(-E\left[\mathcal{H}\int_0^t e^{-\mathcal{H}w} m(t-w)\bar{G}_W(w)dw\right]\right)$$

Similar to the proof of Theorem 1, $\mathcal{H} = \eta X$, where this time X has the generalized binomial distribution given in (6). Then

$$\begin{aligned}&E\left[\mathcal{H}\int_0^t e^{-\mathcal{H}w} m(t-w)\bar{G}_W(w)dw\right] \\ &\quad= \sum_{x=0}^{K}\left[\eta x \int_0^t e^{-\eta x w} m(t-w)\bar{G}_W(w)dw\right] P(X=x) \\ &\quad= \sum_{x=0}^{K} \eta x \left[\int_0^t e^{-\eta x w} m(t-w)\bar{G}_W(w)dw\right]\left[q\binom{K-1}{x}(p^-)^x(q^+)^{K-1-x}\right] \\ &\qquad+ \sum_{x=0}^{K} \eta x \left[\int_0^t e^{-\eta x w} m(t-w)\bar{G}_W(w)dw\right]\left[p\binom{K-1}{x-1}(p^+)^{x-1}(q^-)^{K-x}\right] \\ &\quad= \eta \int_0^t m(t-w)\bar{G}_W(w)(\mathscr{S}_1(w)+\mathscr{S}_2(w))dw\end{aligned}$$

where

$$\mathscr{S}_1(w) = \sum_{x=0}^{K} x e^{-\eta x w} q\binom{K-1}{x}(p^-)^x(q^+)^{K-1-x};$$

$$\mathscr{S}_2(w) = \sum_{x=0}^{K} x e^{-\eta x w} p\binom{K-1}{x-1}(p^+)^{x-1}(q^-)^{K-x}.$$

Using the definitions given in (4),

$$\begin{aligned}\mathscr{S}_1(w) &= \sum_{x=0}^{K} x e^{-\eta x w}(1-p)\binom{K-1}{x}(p-\delta p)^x(1-p+\delta p)^{K-1-n} \\ &= \sum_{x=0}^{K} x e^{-\eta x w}(1-p)\binom{K-1}{x}p^x(1-\delta)^x \sum_{j=0}^{K-1-x}\binom{K-1-x}{j}(1-p)^j(\delta p)^{K-1-x-j}\end{aligned}$$

Now, $x\binom{K-1}{x}\binom{K-1-x}{j} = \frac{(K-1)!}{(x-1)!j!(K-1-x-j)!} = \binom{K-1}{x-1,j,K-1-x-j}$. Then

$$\mathscr{S}_1(w) = \sum_{x=0}^{K} e^{-\eta x w} \sum_{j=0}^{K-x-1} \binom{K-1}{x-1,j,K-1-x-j} (1-p)^{j+1}(1-\delta)^x \delta^{K-1-x-j} p^{K-1-j}.$$

Similarly,

$$\begin{aligned}
\mathscr{S}_2(w) &= \sum_{x=0}^{K} x e^{-\eta x w} p \binom{K-1}{x-1} (p+\delta(1-p))^{x-1}((1-p)(1-\delta))^{K-x} \\
&= \sum_{x=0}^{K} x e^{-\eta x w} p (1-\delta)^{K-x}(1-p)^{K-x} \sum_{i=0}^{x-1} \binom{x-1}{i} p^i (1-\delta)^i \delta^{x-1-i} \\
&= \sum_{x=0}^{K} x e^{-\eta x w} \sum_{i=0}^{x-1} x \binom{K-1}{K-1-x,i,x-1-i} p^{i+1} \delta^{x-1-i}(1-\delta)^{K-x+i}(1-p)^{K-x}
\end{aligned}$$

Clearly $\mathscr{S}(w) = \mathscr{S}_1(w) + \mathscr{S}_2(w)$. □

3. Conclusion

The generalized model of a server under random workload proposed in [3] admits further expansion by way of relaxing the assumption that incoming tasks have exactly one queue to enter on arrival. In considering a server partitioned into several channels, a cost is incurred, namely that additional stress to the server is dependent upon arrival and service times of all previous jobs. However, even under these circumstances, we may obtain a breakdown rate process and satisfactory conditional survival function for the server, and the door is opened to further discussion. By examining the multichannel server, we consider the interrelations of the channels themselves, and derive survival functions to meet the case when the channels are independent as well as when they are correlated.

4. Appendix

Lemma 1 (Decomposition of a Union of Events into Disjoint Atoms)**.** *Let $A_1, \ldots, A_n$ be events. Let $F = \{\emptyset, c\}$, $C_n = \{\underbrace{c, \ldots, c}_{n}\}$, and $E_n = (F \times^n F) \setminus C_n$, where $\times^n$ denotes the $n-$fold cross product. Let $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n) \in E_n$ denote the $n-$tuple where $\varepsilon_i \in F$.*

$$\bigcup_{i=1}^{n} A_i = \bigcup_{\varepsilon \in E_n} \bigcap_{i=1}^{n} A_i^{\varepsilon_i}$$

Proof. First, it will be shown that $E_{n+1} = [(E_n \cup C_n) \times F] \setminus C_{n+1}$.

$$\begin{aligned}
E_{n+1} &= (F \times^n F) \setminus C_{n+1} \\
&= ([F \times^n F] \times F) \setminus C_{n+1} \\
&= [(E_n \cup C_n) \times \{\emptyset, c\}] \setminus C_{n+1}
\end{aligned}$$

Now, it remains to be shown that the tuples $\varepsilon \in E_n$ account for all atoms of $\bigcup_{i=1}^{n} A_i$. Consider $n = 2$.

Then $E_2 = \{(\emptyset, c), (c, \emptyset), (\emptyset, \emptyset)\}$. Then

$$\begin{aligned}(A_1 \cap A_2^c) \cup (A_1^c \cap A_2) \cup (A_1 \cap A_2) \\ &= (A_1 \setminus A_2) \cup (A_2 \setminus A_1) \cup (A_1 \cap A_2) \\ &= (A_1 \cup (A_1 \cap A_2)) \setminus (A_2 \setminus (A_1 \cap A_2)) \cup (A_2 \setminus A_1) \\ &= [A_1 \cup (A_1 \cap A_2) \cup (A_2 \setminus A_1)] \setminus [A_2 \setminus (A_1 \cap A_2) \setminus (A_2 \setminus A_1)] \\ &= (A_1 \cup A_2) \setminus \emptyset \\ &= A_1 \cup A_2\end{aligned}$$

Now assume for $k \leq n$, $\bigcup_{i=1}^{n} A_i = \bigcup_{\varepsilon \in E_n} \bigcap_{i=1}^{n} A_i^{\varepsilon_i}$. Letting $Y = [(E_{n-1} \cup C_{n-1}) \times \{\emptyset, c\}] \setminus C_n$

$$\bigcup_{i=1}^{n} A_i = \left[\bigcup_{i=1}^{n-1} A_i\right] \bigcup A_n$$

Now, $\varepsilon_n \in \{\emptyset, c\}$. Thus,

$$\begin{aligned}\bigcup_{i=1}^{n} A_i &= \bigcup_{\varepsilon \in Y} \bigcap_{i=1}^{n} A_i^{\varepsilon_i} \\ &= \bigcup_{\varepsilon \in E_n} \bigcap_{i=1}^{n} A_i^{\varepsilon_i}\end{aligned}$$

□

References

[1] Ki Hwan Cha and Eui Yong Lee. A stochastic breakdown model for an unreliable web server system and an optimal admission control policy. *Journal of Applied Probability*, 48(2):453–466, 2011.

[2] Andrzej Korzeniowski. On correlated random graphs. *Journal of Probability and Statistical Science*, 11:43–58, 2013.

[3] R. Traylor. Stochastic reliability of a server under random workload. *Academic Advances of the CTO*, 1, 2017.