Status Updates Through Queues

Sanjit K. Kaul
IIIT-Delhi
email: *skkaul@iiitd.ac.in*

Roy D. Yates and Marco Gruteser
WINLAB, ECE Dept., Rutgers University
email: {*ryates, gruteser*}*@winlab.rutgers.edu*

***Abstract*—Anytime, anywhere network connectivity, together with portable sensing and computing devices have led to applications in which sources, for example people or environmental sensors, send updates of their status, for example location, to interested recipients, say a location service. These applications desire status updates at the recipients to be *as timely as possible*; however, this is typically constrained by limited network resources. We employ a time-averaged age metric for characterizing performance of such status update systems. We use system abstractions consisting of a source, a service facility and monitors, with the model of the service facility (physical constraints) a given. While prior work examined *first-come-first-served (FCFS)* queues, this paper looks at the queue discipline of *last-come-first-served (LCFS)*. We explore LCFS systems with and without the ability to preempt the packet currently in service. For each we derive a general expression for system age and solve for the average age a Poisson source can achieve given memoryless service. Specifically, when preemption is allowed, we evaluate how the source would share the service facility with other independent Poisson sources.**

I. Introduction

The *information age* [2] has witnessed huge improvements in computing, access and storage of information. More recently, fueled by ubiquitous connectivity and advancements in portable devices, *real-time status* updates have become increasingly popular. These range from news and weather reports and updates by individuals on Twitter about what is keeping them busy, to updates by environmental sensors [3].

Real-time status updates can enable a variety of applications. Temperature and humidity updates from a forest can help better predict and control forest fires, energy utilization information can help make a smart-home energy efficient, knowledge of the velocity, acceleration of a car can assist drivers in an intelligent transportation system to make safe maneuvers [4].

In the above examples, the goals are to ensure that the agency that monitors fires stays current about conditions in the forest and drivers stay current about status of vehicles in their vicinity, respectively. These examples share a common description: a source generates time-stamped status update messages that are transmitted through a communication system to a monitor. The goal of real-time status updating is to ensure that the *status* of interest, is *as timely as possible* at each monitor. When the monitor's most recently received update at time t is time-stamped $u(t)$, the *status update age*, which we will refer to as simply the *age*, is $t - u(t)$. The monitor's requirement of timely updating corresponds to a small average status update age.

In our work we model complex systems using simple queue-theoretic abstractions, in which one or more sources queue their packets to receive service from a single server. On completion of service a packet is received by one or more monitors. We want to minimize the age of status updates generated by a given source at the monitors.

In [1] we looked at the *first-come-first-served* (FCFS) queue discipline, under which the latest status update packet waited in queue till all previous packet updates had received service. We learned for a variety of FCFS systems that while utilization may be maximized by making the sensor send updates as fast as possible, this strategy may lead to the monitor receiving delayed statuses because the status messages become backlogged in the communication system. In this case, delay suffered by the stream of status updates could be reduced by reducing the rate of updates. Alternatively, reducing the update rate could also lead to the monitor having unnecessarily *outdated* status information because of a lack of updates.

On generation of a status packet, ideally, we would want it to receive service immediately, as reception of a newer update will set the age of status at the monitors to a smaller value. Also, under the assumption that the status is Markovian, having received an update, the monitors do not benefit from the reception of older status updates. This motivates exploring a *last-come-first-served* (LCFS) queue. We will explore two possibilities under LCFS. First, under LCFS *without* preemption, the new status packet replaces any older status packet *waiting* in the queue. It, however, has to *wait* for the packet currently under service to finish. Second, under LCFS *with* preemption, we allow the new packet to preempt the packet currently in service.

Specifically, we will derive the expression for system age for the queue discipline of LCFS, with and without preemption, under very general assumptions about the source and service. We will also calculate the age for an example system in which the source is Poisson and the service is memoryless ($M/M/1$). For the case of LCFS with preemption we will allow other independent Poisson sources to share the service facility with the source of our interest. Finally, we will show that, for a memoryless service facility, LCFS without preemption can achieve the lower bound on age that FCFS achieves.

This paper is organized as follows. Overview of related work is in Section II. LCFS queues for the case when preemption is *not* allowed are analyzed in Section III. In Section IV we show that LCFS with preemption can achieve the FCFS lower bound on achievable age for memoryless service. The

978-1-4673-3140-1/12/$31.00 ©2012 IEEE

case when preemption is allowed is analyzed in Section V. We conclude the paper in Section VI.

II. Related Work

The requirement of ensuring freshness occurs in various fields, including that of networks, and real time databases. In [5] we look at minimizing the age of status updates over a large network of cars in which each car broadcasts its delay sensitive state periodically. The minimum, however, is not known in general and is seen to exist in simulations that we conducted. In [6] we explore piggybacking other cars' updates to achieve a smaller age.

In [7] the authors want to maximize the freshness of data in warehouses to meet user demands. They use networks of queues and assumption of Poisson arrivals and memoryless service to estimate the queue length and delay at the staging area of a warehouse, which is where updates wait before they are committed to the warehouse database. They do not consider the LCFS discipline, however.

Web caching reduces the latency in returning a web page to a client. Unless refreshed often enough, a cache will return stale web pages. However, the rate of refreshing is limited by the finite time it takes for a cache to be updated after the page has been updated at the server. In [8] the authors propose an architecture that limits the "*degree of staleness*" of a cache.

In [9] the authors look at periodic transactions updating real time databases. Each transaction updates the database with data that is associated with a deadline relative to when it is generated. The objective is to find the combination of update period and deadline that ensure that all transactions complete before their deadlines, ensuring the freshness of data and minimizing the CPU utilization.

Ad hoc networking protocols typically use a route cache to forward packets to their destinations. In [10] the authors propose a mechanism that avoids propagation of stale route information through the network. They do not want to broadcast new route information periodically, however, to avoid the associated overheads. In [11] the authors consider the issue of frequency of hello messages in ad-hoc networks. The frequency must not be so large as to congest the network but also not too small that the nodes have stale information.

Finally, in [1] we look at minimizing age for the queue discipline of FCFS.

III. LCFS without preemption

We will derive the average age of status updates from a source updating a monitor, for the queue discipline of LCFS without preemption. We will use a graphical argument similar to the one we used in [1].

Figure 1 shows an example change in age $\Delta(t)$ of updates from the source with time t, at a monitor. Without loss of generality, assume that we begin observing at $t = 0$ when the queue is empty and the age is $\Delta(0) = \Delta_0$.

The time instants, t_i, for $i = 1, 2, \ldots, n$ correspond to packets i generated by the source that complete service. Packet i completes service at t'_i. Packets generated between time t_i and t_{i+1} may include packets from other sources that may or may not complete service and packets from the source under consideration that do not receive service as they are preempted by a new packet arrival. The time of generation of such packets is denoted by t_{ik}, $k = 1, 2, \ldots$. In the figure, such packets are generated at t_{21}, t_{22} and t_{31}.

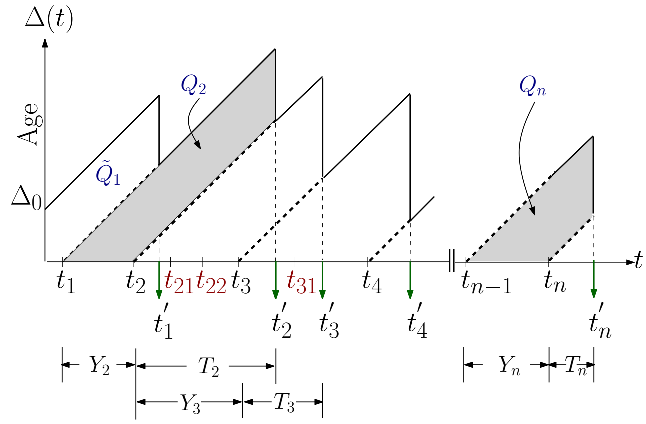

Fig. 1: Example change in age at a monitor for a system using LCFS *without* preemption.

Observe that the given source's update age at the monitor increases linearly in time in the absence of any updates (that is arrivals that complete service) and is reset to a smaller value when an update is received. At t'_i, the age $\Delta(t'_i)$ at the monitor is reset to the age $T_i = t'_i - t_i$ of the received status update. The age T_i is also the system time of the update packet i and is the sum of the time the packet waited in the queue and the time it spent in service. Thus the age function $\Delta(t)$ exhibits the sawtooth pattern shown in Figure 1.

Note that in LCFS older status updates of a source are discarded on a new arrival. When there are more than one sources, a new arrival from a source may or may not lead to an earlier arrival from another source, which is not yet under service, to be discarded. While a new arrival will always lead to an older queued update from the source to be discarded, the new arrival may take the position of the discarded update in the queue or it may be queued behind previous arrivals from other sources. The analysis that follows is not affected by the above considerations.

Only those packets generated by the source that *enter service* update the age. Since packets cannot be preempted in service, packets that enter service are received by the monitors. Let Y_i be the time elapsed between generation/arrival of two such packets. We have

$$Y_i = t_i - t_{i-1}. \tag{1}$$

From Figure 1, we have

$$Q_i = \frac{(Y_i + T_i)^2}{2} - \frac{T_i^2}{2}. \tag{2}$$

Our interval of observation is $(0, \mathcal{T})$. The time-average of age

$\Delta_{\mathcal{T}}$ is given by

$$\Delta_{\mathcal{T}} = \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} \Delta(t)dt. \tag{3}$$

For simplicity of exposition, the length of the observation interval is chosen to be $\mathcal{T} = t'_n$, as depicted in Figure 1. We decompose the area defined by the integral in (3) into a sum of disjoint geometric parts. Starting from $t = 0$, the area can be seen as the concatenation of the polygon area $\tilde{Q}_1$, the trapezoids Q_i for $i \geq 2$ (Q_2 and Q_n are highlighted in the figure), and the triangular area of width T_n over the time interval (t_n, t'_n). With $N(\mathcal{T}) = \max\{n|t_n \leq \mathcal{T}\}$ denoting the number of arrivals by time $\mathcal{T}$, this decomposition yields

$$\Delta_{\mathcal{T}} = \frac{\tilde{Q}_1 + T_n^2/2 + \sum_{i=2}^{N(\mathcal{T})} Q_i}{\mathcal{T}}. \tag{4}$$

Let $N(\mathcal{T}) = \max\{n|t_n \leq \mathcal{T}\}$. Substituting Q_i from (2) in (4), we get

$$\Delta_{\mathcal{T}} = \frac{\tilde{Q}_1}{\mathcal{T}} + \frac{T_n^2}{2} + \frac{N(\mathcal{T})-1}{\mathcal{T}} \frac{1}{N(\mathcal{T})-1} \sum_{i=2}^{N(\mathcal{T})} \left[\frac{Y_i^2}{2} + Y_i T_i\right]. \tag{5}$$

We have

$$\frac{N(\mathcal{T})-1}{\mathcal{T}} = \frac{N(\mathcal{T})-1}{t_1 + \sum_{i=2}^{N(\mathcal{T})} Y_i + T_{N(\mathcal{T})}}. \tag{6}$$

Since t_1 and $T_{N(\mathcal{T})}$ are finite with probability 1, (6) implies

$$\lim_{\mathcal{T}\to\infty} \frac{N(\mathcal{T})-1}{\mathcal{T}} = \frac{1}{E[Y]}. \tag{7}$$

From equations (5) and (7), we can obtain the steady state time-average age

$$\Delta = \lim_{\mathcal{T}\to\infty} \Delta_{\mathcal{T}} = \frac{1}{E[Y]} \left[\frac{E[Y^2]}{2} + E[YT]\right], \tag{8}$$

where $E[\cdot]$ is the expectation operator, and Y and T are the random variables that correspond to the interarrival time between updates from the source that complete service and system time of an update packet, respectively. Finally, we note that the average update age in (8) holds under weak assumptions on the ergodicity of the service system.

A. $M/M/1$ *LCFS Service without Preemption (Single Source)*

We will derive the expression of average age for LCFS service without preemption with status update arrivals under the assumption that the system *sees arrivals from just one source*. The arrivals are described by a Poisson process of rate λ and the service times are exponentially distributed. We will see that, given a service rate μ, this system can achieve the minimum age achieved by the system described in Section IV, in which a new status packet generation was designed to occur at the moment the previously generated packet finished service. The average age Δ is given by (8). We need to calculate the terms $E[Y]$, $E[Y^2]$ and $E[YT]$.

Since in our assumed system a packet waiting in queue is replaced by a newly generated packet, a packet i enters service immediately or it waits for W_i for packet $i-1$ to finish its remaining time in service. Let S_i be the service time of packet i. Thus packet i has respective waiting and system times

$$W_i = (S_{i-1} - Y_i)^+, \tag{9}$$

$$T_i = S_i + W_i. \tag{10}$$

We need

$$E[T_i Y_i] = E[W_i Y_i] + E[S_i Y_i]. \tag{11}$$

The expectation of $W_i Y_i$ can be calculated as

$$E[W_i Y_i] = E\big[(S_{i-1} - Y_i)^+ Y_i\big] \tag{12}$$

$$= \int_{s=0}^{\infty} f_{S_{i-1}}(s) \int_{y=0}^{s} (s-y) y f_{Y_i|S_{i-1}}(y|s)\, dy\, ds. \tag{13}$$

To find $f_{Y_i|S_{i-1}}(y|s)$, we first derive the conditional probability $P\{Y_i \leq y | S_{i-1} = s\}$. For $y \leq s$, the event $Y_i \leq y$ occurs iff one or more arrivals occurs during the first y units of service of update $i-1$ but zero arrivals take place during the remaining $s-y$ units of service. This implies

$$P\{Y_i \leq y|S_{i-1} = s\} = (1 - e^{-\lambda y}) e^{-\lambda(s-y)}, \quad y \leq s. \tag{14}$$

For $y > s$, the event $Y_i > y$ occurs when there are no arrivals for a time interval of length y that consists of the length s service time of update i followed by an idle time of duration $y - s$. This implies

$$P\{Y_i \leq y|S_{i-1} = s\} = 1 - e^{-\lambda y}, \quad y > s. \tag{15}$$

Since $f_{Y_i|S_{i-1}}(y|s) = dP\{Y_i \leq y|S_{i-1} = s\}/dy$, it follows from (14) and (15) that

$$f_{Y_i|S_{i-1}}(y|s) = \begin{cases} \lambda e^{-\lambda(s-y)} & y \leq s, \\ \lambda e^{-\lambda y} & y > s. \end{cases} \tag{16}$$

Using (16), and the exponential service time PDF $f_{S_{i-1}}(s) = \mu e^{-\mu s}$, we can write (13) as

$$E[W_i Y_i] = \int_{s=0}^{\infty} \mu e^{-\mu s} \int_{y=0}^{s} (s-y) y \lambda e^{-\lambda(s-y)}\, dy\, ds \tag{17}$$

$$= \frac{1}{\lambda\mu} - \frac{\mu + 2\lambda}{\lambda(\mu+\lambda)^2}. \tag{18}$$

Using (16) and the memoryless service times, the pdf of Y_i, $f_{Y_i}(y)$ can be calculated as

$$f_{Y_i}(y) = \int_0^{\infty} f_{Y_i|S_{i-1}}(y|s) f_{S_{i-1}}(s) ds$$

$$= \lambda e^{-\lambda y}(1 - e^{-\mu y}) + \frac{\lambda\mu}{\lambda+\mu} e^{-\mu y}. \tag{19}$$

The PDF (19) can be used to obtain

$$E[Y_i] = \frac{1}{\lambda} + \frac{\lambda^2}{\mu(\mu+\lambda)^2}, \tag{20}$$

$$E\big[Y_i^2\big] = \frac{2}{\lambda^2} - \frac{2\lambda}{(\mu+\lambda)^3} + \frac{2\lambda}{(\mu+\lambda)\mu^2}. \tag{21}$$

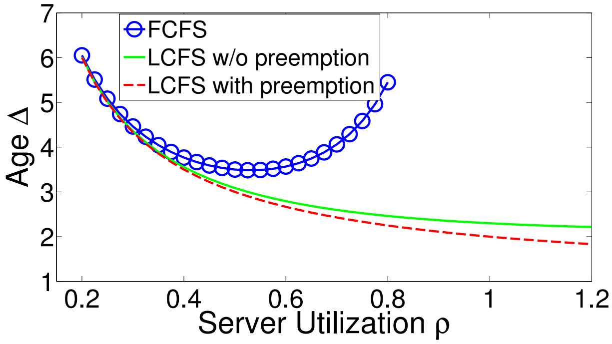

Fig. 2: A comparison of change of age (at a monitor) with server utilization for $M/M/1$ FCFS and LCFS with and without preemption. Service rate is $\mu = 1$. As $\lambda = \rho$ increases, age goes to $2/\mu = 2$ when preemption is not allowed. It goes to $1/\mu = 1$ when it is allowed.

Using the independence of Y_i and S_i, we can write

$$E[T_i Y_i] = E[W_i Y_i] + E[S_i]E[Y_i]. \tag{22}$$

Furthermore, when the system reaches steady state $T =^{st} T_i$ and $Y =^{st} Y_i$. Using equations (18), (20), (21) and (22) to make substitutions in (8), we can calculate Δ to be

$$\Delta = \frac{1}{\mu}\left[1 + \frac{\rho^3(1+\rho)^2 + \rho^4 + (1+\rho)^3}{\rho^4(1+\rho) + \rho(1+\rho)^3}\right]. \tag{23}$$

If we bombard the server with new status packets at a very large rate, that is if $\lambda \to \infty$, the age Δ is obtained to be

$$\lim_{\lambda\to\infty} \Delta = \frac{2}{\mu}. \tag{24}$$

Specifically, in (8), the term $(E[TY]/E[Y]) \to E[S] = 1/\mu$ and the term $(E[Y^2]/(2E[Y])) \to 1/\mu$, giving an age of $\Delta = 2/\mu$. Figure 2 plots the age given by (23) for $\mu = 1$. As is seen in the plot, the $M/M/1$ LCFS system without preemption always performs better than the $M/M/1$ FCFS system. This is expected as packets under LCFS will be newer, would have waited lesser on an average than under FCFS, when they enter service. The age approaches $2/\mu = 2$ as $\rho = (\lambda/\mu) = \lambda$ increases.

IV. Minimum FCFS Achievable Age and LCFS

For the queue discipline of FCFS we observed in [1] that the lower bound on the age is achievable by a FCFS system in which the source observes the state of the packet update queue so that a new status update is generated the very moment the previous update finishes service. In this setting, the server is always busy and the waiting time of every update packet is zero. Since each delivered update packet is as young as possible, the average status update age obtained for this system is a lower bound to the age for any FCFS queue in which updates are generated as a stochastic process independent of the current state of the queue.

The lower bound was found to be

$$\Delta^* = \frac{1}{E[S]}\left[\frac{E[S^2]}{2} + (E[S])^2\right], \tag{25}$$

where S is the service time distribution. For a system with memoryless service at rate μ, the minimum average age is therefore $\Delta^* = 2/\mu$. From equation (24) we know that this minimum can be approached by a LCFS system without preemption and with arrivals independent of the current status of the queue, as the arrival rate becomes very large.

V. LCFS With Preemption

What if every packet entered service immediately after generation? A packet arrival preempts the packet currently in service, if any. Packets arrive from one or more independent sources. The new arrival and the packet being preempted may not belong to the same source. The number of packets in such a system is at most 1. Figure 3 shows an example progression of age for a given source, say u, in such a system. The packets generated by this source u at time instants t_i, for $i = 1, 2, \ldots, n$, indexed by i, complete service. Let Y_i denote the time between such arrivals $i-1$ and i. The interval Y_i begins with a busy period that ends in departure of $i-1$. The interval ends with arrival of update packet i, which will complete a service of duration Z_i.

Let $Z =^{st} Z_i$ and $Y =^{st} Y_i$ for any i. From Figure 3 and using arguments similar to those in Section III, the steady state average age Δ_u of user u can be obtained as

$$\Delta_u = \frac{1}{E[Y]}\left[\frac{E[Y^2]}{2} + E[YZ]\right]. \tag{26}$$

In order to calculate Δ_u, let D_i (see D_3 in Figure 3) be the time interval between the departure of $i-1$ and i. This interval starts with an idle period and may see zero or more arrivals of other sources, some of which may complete service, while others are preempted. Any arrivals of the given source during D_i, other than arrival i, are preempted. Thus the interval D_i consists of one or more blocks of server being idle followed by it being busy. Note that if the system consists of just one source, then D_i consists of just one block, which starts with the idle period that follows the departure of $i-1$. This idle period is followed by the server busy period that ends in departure i. Figure 3 shows D_3, which contains a random L number of blocks. The figure shows block 1 and block L. A block $1 \le k \le L$, say of length B_k, consists of an idle period of length X'_k followed by a busy period of length S_k. We have

$$D_i = \sum_{k=1}^{L} B_k = \sum_{k=1}^{L} (X'_k + S_k). \tag{27}$$

Note that packet i arrives during S_L and then spends Z_i amount of time in service.

We will now calculate the terms $E[Y]$, $E[Y^2]$ and $E[YZ]$ in the expression for Δ_u (equation (26)) in terms of D_i and Z_i. Consider the interval Y_i, for any i. We can write (the case for $i = 3$ is shown in Figure 3),

$$Y_i = Z_{i-1} + D_i - Z_i. \tag{28}$$

Further, because $Z =^{st} Z_{i-1} =^{st} Z_i$, $Y =^{st} Y_i$, and $D =^{st}$

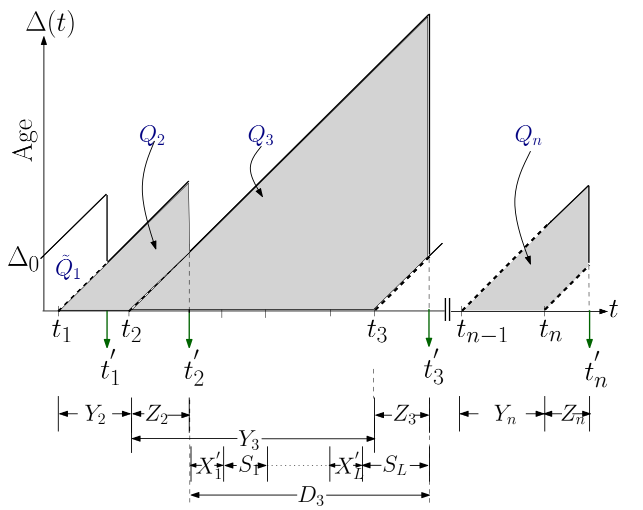

Fig. 3: Example change in age for a system using LCFS *with* preemption.

D_i we get

$$E[Y] = E[Y_i] = E[D_i] = E[D]. \tag{29}$$

Note that D_i and Z_i are dependent but are each independent of Z_{i-1}. Using this fact and (28) we get

$$E\big[Y^2\big] = E\big[D^2\big] + 2\,\mathrm{Var}[Z] - 2\,\mathrm{Cov}[DZ]. \tag{30}$$

Finally, note that Y_i and Z_i are mutually independent. Using this and (29) we can write

$$E[Y_i Z_i] = E[Y_i]E[Z_i] = E[Y]E[Z] = E[D]E[Z]. \tag{31}$$

Using (29), (30) and (31), equation (26) can be written as

$$\Delta_u = E[Z] + \frac{E\big[D^2\big]}{2E[D]} + \frac{\mathrm{Var}[Z] - \mathrm{Cov}[DZ]}{E[D]} \tag{32}$$

Next we calculate Δ_u for a $M/M/1$ system.

A. M/M/1 and LCFS with preemption (One or more Sources)

Let's assume that one or more independent Poisson sources contribute status update arrivals to the system such that the cumulative status update arrival rate is λ. Further let λ_u be the rate of arrival of update packets from source u. We will now derive Δ_u.

The general expression for Δ_u is given by (32). Thus we need to calculate the terms $E[Z] = E[Z_i]$, $E\big[Z^2\big] = E\big[Z_i^2\big]$, $E[D] = E[D_i]$, $E\big[D^2\big] = E\big[D_i^2\big]$, and $E[DZ] = E[D_i Z_i]$.

Note that Z_i is the time arrival i, from the given source u, spends in service. Arrival i completes service (is not preempted) and hence is the last arrival from any source during S_L. Thus Z_i is the random time interval the server is busy after i arrives, and no new arrivals occur during it. Note that Z_i is independent of the fraction of interval S_L that had elapsed before arrival of i. The distribution of Z_i is that of the time to service completion, say X_μ, after packet i arrives, conditioned on X_μ being smaller than the time to the next packet arrival, say X_λ, from any source. Thus $P[Z_i > z] = P[X_\mu > z | X_\mu < X_\lambda]$.

Since service and packet arrival times in our system are memoryless with means $1/\mu$ and $1/\lambda$ respectively, X_μ is exponential with mean $1/\mu$ and X_λ is exponential with mean $1/\lambda$. From earlier observations we can write

$$\begin{aligned} P[Z_i > z] &= \frac{\int_z^\infty P[z < X_\mu < y | X_\lambda = y] f_{X_\lambda}(y)\,dy}{P[X_\mu < X_\lambda]} \\ &= \frac{\int_z^\infty (e^{-\mu z} - e^{-\mu y})\lambda e^{-\lambda y}\,dy}{\mu/(\mu+\lambda)} \qquad (33) \\ &= e^{-(\lambda+\mu)z}. \qquad (34) \end{aligned}$$

Equation (33) is obtained by noting that X_μ and X_λ are independent and exponentially distributed. Equation (34) implies that Z_i is an exponentially distributed random variable with

$$E[Z_i] = \frac{1}{\lambda+\mu}, \text{ and } E\big[Z_i^2\big] = \frac{2}{(\lambda+\mu)^2}. \tag{35}$$

Now we will calculate $E[D_i]$. From (27) we know that D_i is a random sum of random variables B_j, $1 \le j \le L$. Also, D_i ends with the departure of an update packet of u. Since the arrival rate of λ is the sum of rates of independent Poisson sources, the probability that any block B_j ends in the departure of status packet of source u is λ_u/λ. Thus, the probability that D_i consists of $L = l$ blocks is the probability of the event that $l-1$ consecutive blocks end in departures of packets not of user u, followed by block l that ends in a user u departure. It is given by

$$P[L = l] = (1-q)^{l-1} q, \tag{36}$$

where $q = \lambda_u/\lambda$. Note that $B_j = X_j' + S_j$, where X_j' is an idle period and S_j is the server busy period. During the busy period a random number of packet arrivals may be preempted. Note that the service rate for all packet arrivals is μ. Also the busy period S_j is memoryless in nature and is independent of the number of arrivals during it that get preempted and the user whose packet departs at its end. The above observations and given the Poisson arrivals of rate λ, we can write

$$E\big[X_j'\big] = \frac{1}{\lambda},\; E[S_j] = \frac{1}{\mu}, \text{ and } E[B_j] = \frac{1}{\lambda} + \frac{1}{\mu}. \tag{37}$$

The memoryless nature of the arrival and service processes also implies that each B_j is independent of L. Using this fact and equations (27), (36) and (37), we can write

$$E[D_i] = E[L]E[B_j] = \frac{\mu+\lambda}{\lambda_u \mu}. \tag{38}$$

Now we will calculate $E\big[D_i^2\big]$. Let the random variable B be stochastically identical to block lengths B_j, $j = 1, \ldots, L$. Using arguments we used to calculate $E[D_i]$, and noting that B_i and B_j, for $i \ne j$ are independent random variables, we can write

$$E\big[D_i^2\big] = E[L]E\big[B^2\big] + E[L(L-1)](E[B])^2, \tag{39}$$

Also note that the idle period X_j' and busy period S_j that constitute B_j are mutually independent. This allows us to write

$$E[B^2] = \frac{2}{\lambda^2} + \frac{2}{\mu^2} + \frac{2}{\lambda\mu}. \tag{40}$$

Using equations (36), (37) and (40), we can write (39) as

$$E[D_i^2] = 2\frac{\lambda}{\lambda_u}\left(\frac{\lambda}{\lambda_u}\left[\frac{1}{\lambda}+\frac{1}{\mu}\right]^2 - \frac{1}{\lambda\mu}\right). \tag{41}$$

Finally, we calculate $E[DZ] = E[D_i Z_i]$. We can write

$$\begin{aligned} E[D_i Z_i|L] &= E\left[Z_i \sum_{j=1}^{L} B_j | L\right] \\ &= E[Z_i B_1 + \ldots + Z_i B_L] \quad (42)\\ &= (L-1)E[Z_i]E[B] + E[Z_i B_L]. \quad (43)\end{aligned}$$

We argued when we calculated $E[D_i]$ that the block lengths B_j are independent of L. This gives us (42). Also, note that Z_i is the time that the departing packet of user u spent in service and is a part of block L. Thus Z_i is independent of the length of the $L-1$ blocks that preceded block L and that gives us (43). We can write $E[Z_i B_L]$ as

$$\begin{aligned} E[Z_i B_L] &= E[Z_i(X'_L + S_L)] \\ &= E[Z_i]E[X'_L] + E[Z_i S_L]. \quad (44)\end{aligned}$$

Equation (44) can be obtained by noting that the time Z_i the departure i spends in service is independent of the idle period X'_L that precedes the busy period S_L.

Note that S_L is the sum of two disjoint intervals, the first of which ends with arrival i of user u. Let's call it $\hat{S}_L$. All arrivals during $\hat{S}_L$ are preempted and do not complete service. Interval Z_i, during which no arrivals take place, follows $\hat{S}_L$. We have $S_L = \hat{S}_L + Z_i$. Further $\hat{S}_L$, which is the sum of packet inter-arrival times during S_L, is independent of Z_i. Using this fact and equations (35) and (37) we can write

$$\begin{aligned} E[Z_i S_L] &= E\left[Z_i(Z_i + \hat{S}_L)\right] \\ &= E[Z_i^2] + E[Z_i]E[S_L - Z_i] = \frac{\lambda + 2\mu}{(\lambda+\mu)^2\mu}. \quad (45)\end{aligned}$$

Further using equations (35), (36), (37), (43), (44), and (45) we can compute $E[D_i Z_i] = E[E[D_i Z_i|L]]$ to be

$$E[D_i Z_i] = \frac{1}{\mu\lambda}\left(\frac{\lambda - \lambda_u}{\lambda_u}\right) + \frac{1}{\lambda(\lambda+\mu)} + \frac{\lambda+2\mu}{(\lambda+\mu)^2\mu}. \tag{46}$$

Using equations (35), (38), (41), (46), and substituting in equation (32) we can obtain the update age of user u as

$$\Delta_u = \frac{\lambda}{\lambda_u}\left(\frac{1}{\lambda} + \frac{1}{\mu}\right). \tag{47}$$

Note that if we fix $(\lambda - \lambda_u)$ and let $\lambda_u \to \infty$, the age $\Delta_u \to 1/\mu$, that is the average update age of the source converges to the average packet service time at the facility, as the source rate of u increases, while contributions of other sources are kept fixed.

Similarly, if we have N sources and $\lambda_1, \lambda_2, \ldots, \lambda_N \to \infty$, while $\lambda_i = \lambda_j$ for all i and j, all sources' updates experience an age of N/μ each. Specifically, in (47), $\Delta_u \to N/\mu$.

Finally, note that if our system sees arrivals from only one user, that is if $\lambda = \lambda_u$, the update age Δ_u for the sole user becomes

$$\Delta_u = \frac{1}{\lambda} + \frac{1}{\mu}. \tag{48}$$

This age is plotted in Figure 2. As is expected, LCFS with preemption achieves the smallest age for any given server utilization ρ, under the assumption of independent and memoryless inter-arrival and service times. As $\lambda \to \infty$, it achieves half the age of when preemption is not allowed.

VI. Conclusions and Future Work

We have looked at the problem of keeping the status updates as new as possible at interested recipients, given a set of physical constraints. We looked at queue-theoretic abstractions for the queue discipline of last-come-first-served (LCFS). We showed that the smallest age under FCFS can be achieved by using a LCFS queue with waiting room of size 1 and the source generating updates at a very large rate. Such systems may further benefit by allowing a new packet to preempt the packet currently in service. For the $M/M/1$ system, a 50% reduction in minimum age was obtained. We plan to extend the work to systems in which multiple sources may compete with each other to minimize their update ages.

References

[1] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM*, 2012.

[2] D. S. Alberts, D. S. Papp, and et al., *The Information Age: An Anthology on Its Impacts and Consequences (Information Age Anthology, Volume I); Part 1: The Information and Communication Revolution.* Natl Defense Univ Pr, Jun. 1997.

[3] A. Mainwaring, D. Culler, and et al., "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, ser. WSNA '02. Atlanta, Georgia, USA: ACM, 2002, pp. 88–97, ACM ID: 570751.

[4] P. Papadimitratos, A. La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 84–95, Nov. 2009.

[5] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, Salt Lake City, Utah, USA, 2011.

[6] S. Kaul, R. Yates, and M. Gruteser, "On piggybacking in vehicular networks," in *Global Telecommunications Conference, 2011. IEEE GLOBECOM 2011. IEEE*, Dec. 2011.

[7] A. Karakasidis, P. Vassiliadis, and E. Pitoura, "ETL queues for active data warehousing," in *Proceedings of the 2nd international workshop on Information quality in information systems*, ser. IQIS '05. Baltimore, Maryland: ACM, 2005, pp. 28–39, ACM ID: 1077509.

[8] H. Yu, L. Breslau, and S. Shenker, "A scalable web cache consistency architecture," *SIGCOMM Comput. Commun. Rev.*, vol. 29, no. 4, pp. 163–174, Aug. 1999, ACM ID: 316219.

[9] M. Xiong and K. Ramamritham, "Deriving deadlines and periods for real-time update transactions," in *The 20th IEEE Real-Time Systems Symposium, 1999. Proceedings*. IEEE, 1999, pp. 32–43.

[10] Y. C. Hu and D. B. Johnson, "Ensuring cache freshness in on-demand ad hoc network routing protocols," in *Proceedings of the second ACM international workshop on Principles of mobile computing*, 2002.

[11] V. C. Giruka and M. Singhal, "Hello protocols for ad-hoc networks: overhead and accuracy tradeoffs," in *World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM*. IEEE, Jun. 2005.