The Status Update Problem: Optimal Strategies for Pseudo-Deterministic Systems

Jared Tramontano

under the direction of
Dr. Shan-Yuan Ho
Mr. Siddharth Venkatesh
Department of Mathematics
Massachusetts Institute of Technology

Research Science Institute
July 29th, 2015

Abstract

We investigate First-Come-First-Served (FCFS) systems receiving update packets from a general integer number of sources. We calculate precisely when these sources should be sending update packets, relative to the rate at which the system can deterministically process update packets, to keep the average age of information, Δ, minimal. We devise strategies specifically for a single source, so that the server is never idle, so that the server has some idle time after every update packet, and so that the strategy is optimal in general. Finally, we discuss the case in which the server processes packets as a random process.

Summary

In this paper, we discuss queuing systems in which sources send update packets to a system that processes these packets and sets them to a monitor. These update packets could be anything from Facebook status updates to self-driving cars sending information to nearby vehicles about, say, tire pressure and acceleration. In this work, we look at deterministic systems, and devise strategies that tell sources when exactly they should be sending update packets, relative to the rate at which the server can process update packets.

1 Introduction

In the information age the value of a monitor possessing the most recent information regarding some entity has grown significantly [1]. For example, if the monitor is a human user, and the source, a stock market server, fractions of a second can be worth millions of dollars. Although of less pecuniary value, human desire to have access to the most recent weather updates or various Facebook statuses, or perhaps environmental sensors, is still quite relevant [2]. Perhaps one of the more prominent applications of real-time status updates is in the driving of intelligent systems [4]; i.e., sensors detecting speed, acceleration, tire pressure, relative position of obstacles, may send information to an internal sensor which in turn sends information to nearby intelligent vehicles.

As detailed in [2], these examples share a common description: a source generates time-stamped updates, sends them to some system, and the system then relays the updates to a monitor. In the case of the intelligent driving systems, given $n+1$ cars, each car receives information from the other n cars. The objective of real-time status updating is to ensure that the information received by the monitor from the n sources is *in as timely a manner as possible.*

In [1], it is noted that the problem of *timely updating* is neither an issue of the rate at which the system can process updates, nor the rate at which the sources send update packets. Namely, a rapid system processing unit does not ensure timely updating, nor does a rapid flow of update packets. The most timely process of update occurs at a ratio of the update packet flow to system process time. In [2], this ratio was precisely calculated for first-come-first-served (FCFS) systems. A FCFS system is a simple queue in which a packet is received from a source, and serviced in the order in which it was received. For example, if source 1, denoted Ω_1 sends update packets at t_1, t_2 and source 2, Ω_2, sends an update packet at $t_3 > t_1, t_2$, then the update packet from source 2 must wait until both update packets t_1, t_2 are serviced.

In [2], FCFS systems are investigated in which the generation of real-time update packet

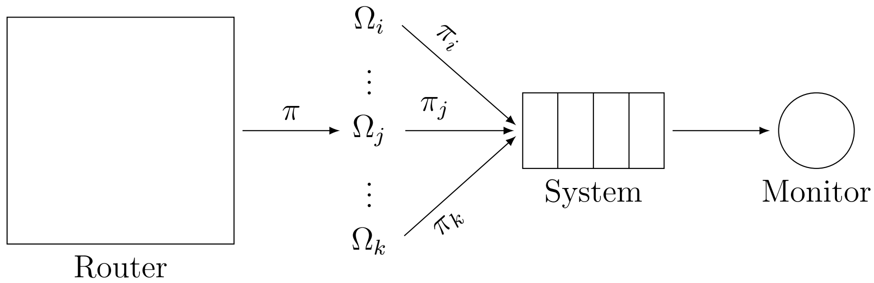

Figure 1: Intelligent Router Relaying When to Send Update Packets

updates, as well as the system service time are random Poisson processes. The optimal ratio of generation to service time is calculated a system in which both the flow of update packets and processing times are Poisson processes (M/M/1), a system in which the flow of update packets is a Poisson process with a deterministic processing time (M/D/1), and a system in which the flow of update packets is deterministic and the processing time is a Poisson process (D/M/1). In [3], an M/M/1 FCFS is analysed for two sources, described as Poisson processes [4].

Now suppose that we are given a source allowed to update at unit time at an integer time m. How often this source should update is a function of the deterministic rate at which the server can process update packets. We shall denote this deterministic rate by μ and call it the *service time*. In a sense, this is similar to a D/D/1 problem, though it is quite a bit more complex. Namely, there is an intelligent router that looks at the system, determines the value of μ, and relays an explicit strategy to the source of precisely when to send update packets.

The general problem is to investigate an FCFS system with n sources, for some integer n. We represent Δ_k as the *average time of information* of source k, and

$$\Delta = \sum_{j=1}^{n} \Delta_k$$

represents the total age of the information from all n sources. If we define

$$\delta = \max_{i,j=1,\ldots,n} \{\Delta_i - \Delta_j\},$$

then we wish minimize

$$\Delta + \delta,$$

eliminating network bias, whilst maintaining network efficiency. The problem of an intelligent router relaying explicit strategies to an integer number of sources may be seen in Figure 1.

In section 2, we discuss previous work as a means of quantifying the average age of information as a geometric interpretation of area, as well as terminology and general behaviour of the age function. In section 3, we begin to discuss an explicit model in finite time, giving a method for calculating the update time of any update packet, as well as the average time over a finite interval of observation.

In section 4, we precisely define a strategy, and discuss the properties of cyclic strategies. We discuss the special case in which the server is never idle, and investigate optimal strategies under these constraints. Contrariwise, we discuss the special case in which the server is idle after each update packet, and investigate optimal strategies under these constraints. We provide a conjecture for the optimal strategy for one source, given any μ.

2 FCFS Status Update Age

In this section we shall introduce notation, as well as derive the average status update age for a FCFS system. We begin with an initial observation time $t_0 = 0$ for sake of simplicity. At t_0, the information has aged to some initial constant Δ_0; as time progresses, the information ages linearly. A source Ω generates its first status update at t_0, which is sent to an empty queue, processed, and received by the monitor at time u_0. In general, Ω sends update i at time t_i, which is received by the monitor at time u_i. The function representing the average

age of the information at time t shall be denoted $\Delta(t)$. For instance, $\Delta(u_i) = u_i - t_i$ due to the time t_i takes to be received by the monitor. We shall denote $T_i = u_i - t_i$, the *system time of packet* t_i, as it represents the sum of the waiting time of the packet in the queue and the time it spent in service. Therefore, we see that $\Delta(t)$ behaves in a sawtooth manner, as shown in Figure 2.

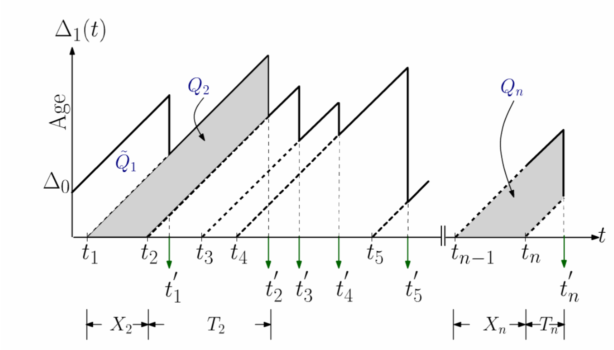

Figure 2: An Aging Function Δ in a FCFS Queue System [2]

As $\Delta(t)$ is a piecewise continuous function, we may find its average age at time $\mathcal{T}$ using the integral over the interval $(0, \mathcal{T})$. Thus,

$$\Delta_{\mathcal{T}} = \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} \Delta \, dt.$$

If we denote $X_i = t_i - t_{i-1}$, called the i^{th} *inter-arrival time*, we may find a simple geometric representation of the average age over an interval. For simplicity, let $\mathcal{T} = u_n$, so that the interval of observation is simply $(0, u_n)$. The integral can be seen as the concatenation of the polygon $\tilde{Q}_1$, the trapezoids Q_i for $i \geq 2$, and the triangular area of width T_n. If we set $N(\mathcal{T}) = \max\limits_{t_n \leq \mathcal{T}} \{n\}$ then we obtain

$$\Delta_{\mathcal{T}} = \frac{\tilde{Q}_1 + \frac{T_n^2}{2} + \sum\limits_{i=2}^{N(\mathcal{T})} Q_i}{\mathcal{T}}. \tag{1}$$

In [2] it is shown that (1) is equivalent to

$$\Delta_{\mathcal{T}} = \frac{\tilde{Q}}{\mathcal{T}} + \frac{\sum_{i=2}^{N(\mathcal{T})} X_i T_i + \frac{X_i^2}{2}}{\mathcal{T}}.$$

Often, we will split the total system time into two parts: the waiting time and the service time. Formally, we write $T_i = W_i + S_i$ to denote the total system time of update packet i. In fact, we may characterize this recursively if we see that $W_i = 0$ if t_{i-1} has been serviced before t_i arrives or $W_i = T_{i-1} - X_i$ if t_{i-1} has not been serviced when t_i arrives. In symbols, $W_i = 0$ if $T_{i-1} \leq X_i$ and $W_i = T_{i-1} - X_i$ if $T_{i-1} > X_i$. Thus, we may write $W_i = (T_{i-1} - X_i)^+$ in which $(a - b)^+$ is 0 if $a < b$ and $a - b$ otherwise. Ergo, we may write $T_i = (T_{i-1} - X_i)^+ + S_i$. Finally, if $u_{i-1} < t_i$, we shall call $t_i - u_{i-1}$ the *idle time* of the server.

3 Model in Finite Time

In the practical world, we need only concern ourselves with finite observation intervals. We begin first by deriving the information age functions. Consider first a simple case in which we initialize the system with a packet $t_0 = 0$ with initial age $\Delta(0) = \Delta_0$, and proceed to send an update packet at $t_1 = 1$, ending our interval of observation at some time $\mathcal{T} > u_1$. In a simple example such as this, we may write the age function rather simply as two cases. In the first, for $\mu \in (0, 1]$ we have

$$\Delta(t) = \begin{cases} \Delta_0 + t & : t \in [0, \mu) \\ t & : t \in [\mu, 1 + \mu) \\ t - 1 & : t \in [1 + \mu, \mathcal{T}] \end{cases}$$

and for $\mu \in (1, \infty)$ we have

$$\Delta(t) = \begin{cases} \Delta_0 + t & : t \in [0, \mu) \\ t & : t \in [\mu, 2\mu) \\ t - 1 & : t \in [2\mu, \mathcal{T}] \end{cases}$$

If we denote the average age at time $\mathcal{T}$ by $\Delta_{\mathcal{T}}$, then, for $\mu \in (1, \infty)$, we have

$$\Delta_{\mathcal{T}} = \frac{1}{\mathcal{T}} \left(\mu(\Delta_0 + \mu) + \frac{3\mu^2}{2} + (\mathcal{T} - 2\mu) \left(2\mu - 1 + \frac{\mathcal{T} - 2\mu}{2} \right) \right).$$

In general, the endpoints of the intervals on which the age function is truncated are simply the value of u_0 and u_1, where u_1 is dependent on whether or not the update packet t_1 is waiting in the queue. In other words, the update times u_i are dependent on whether or not the i^{th} inter-arrival time is greater than the total system time of the $(i-1)^{th}$ update packet. This gives the following lemma:

Lemma 3.1. *Given a sequence of service times, $\{\mu_i\}_{i=0}^n$, the update time, u_n, of the n^{th} update packet is given by*

$$u_n = \max_{i=0,\ldots,n} \left\{ t_i + \sum_{j=i}^{n} \mu_j \right\}. \tag{2}$$

Proof. Suppose $n = 0$, then, since the queue is empty, $u_0 = t_0 + \mu_0$, so (2) holds. Assume true for $n - 1$, then, if we let

$$M = \max_{i=0,\ldots,n-1} \left\{ t_i + \sum_{j=i}^{n-1} \mu_j \right\}$$

we see that $T_{n-1} = M - t_{n-1}$. We have two cases.

Case (i): If $T_{n-1} \leq X_n$, then $W_n = 0$. In this case, $T_n = \mu_n$, so $u_n = t_n + \mu_n$.

Case (ii) If $T_{n-1} > X_n$, then $W_n = M - t_n$. In this case, $T_n = M - t_n + \mu_n$, so $u_n = M + \mu_n$.

Now, given case (i), $T_{n-1} \leq X_n$ implies $M \leq t_n$ so that $M + \mu_n \leq t_n + \mu_n$. Likewise, given case (ii), $M + \mu_n > t_n + \mu_n$. Ergo,

$$\begin{aligned}
u_n &= \max\{t_n + \mu_n, M + \mu_n\} \\
&= \max\{t_n + \mu_n, \max_{i=0,\ldots,n-1} \left\{ t_i + \sum_{j=i}^{n-1} \mu_j \right\} + \mu_n\} \\
&= \max_{i=0,\ldots,n} \{t_i + \sum_{j=i}^{n} \mu_j\}.
\end{aligned}$$

□

If instead, we are given a single deterministic μ, this lemma may be simplified.

Corollary 3.1. For a constant deterministic service time μ, the update time, u_n, of the n^{th} packet is given by

$$u_n = \max_{i=0,\ldots,n} \{t_{i-1} + (n + 1 - i))\mu\}$$

Thus, we have an easy method for characterizing the update times of individual packets. This, in turn, allows us to succinctly express the age and average age.

Theorem 3.2. *Given a deterministic service time μ, and a sequence of update packets sent at $\{t_i\}_{i=1}^{n}$, the age of the information at time t may be written as*

$$\Delta(t) = \begin{cases} \Delta_0 + t & : t \in [0, u_0) \\ t - t_0 & : t \in [u_0, u_1) \\ \quad\vdots & \\ t - t_{n-1} & : t \in [u_{n-1}, u_n] \end{cases}$$

Moreover, if we set $N'(\mathcal{T}) = \max\limits_{u_n \leq \mathcal{T}} \{n\}$, and $\Delta_{u_0} = (u_0)\left(\Delta_0 + \frac{u_0}{2}\right)$ we may write

$$\Delta_{\mathcal{T}} = \frac{1}{\mathcal{T}} \left(\Delta_{u_0} + \frac{(u_n^2 - u_0^2)}{2} - \sum_{i=1}^{N'(\mathcal{T})} t_{i-1}(u_i - u_{i-1}) + (\mathcal{T} - u_{N'(\mathcal{T})}) \left(u_{N'(\mathcal{T})} - t_{N'(\mathcal{T})} + \frac{\mathcal{T} - u_{N'(\mathcal{T})}}{2} \right) \right).$$

Proof. The proof of $\Delta(t)$ is trivial, using Figure 2. For the average age of information at time $\mathcal{T}$, note that $\Delta(t)$ is a piecewise linear function. Thus, the average age is simply the midpoint of each piecewise function, weighted by the measure of the interval on which the piece is defined, divided by the total measure of the interval of observation.

Formally, the average age on the interval $[u_i, u_{i-1})$ for $i > 0$, is given by

$$(u_i - u_{i-1})\left(u_{i-1} - t_{i-1} + \frac{u_i - u_{i-1}}{2}\right) = \frac{(u_i^2 - u_{i-1}^2)}{2} - t_{i-1}(u_i - u_{i-1}).$$

Moreover, the average age on the interval $[0, u_0)$ is given by $u_0(\Delta_0 + \frac{u_0}{2})$. Since $\mathcal{T}$ is arbitrary, it may happen that some t_j is sent before $\mathcal{T}$, yet $u_j > \mathcal{T}$, or $u_j = \mathcal{T}$. Thus, if $N'(\mathcal{T}) = \max\limits_{u_n \leq \mathcal{T}}\{n\}$, then the average age on the interval $[t_{N'(\mathcal{T})}, \mathcal{T}]$ is given by

$$(\mathcal{T} - u_{N'(\mathcal{T})})\left(u_{N'(\mathcal{T})} - t_{N'(\mathcal{T})} + \frac{\mathcal{T} - u_{N'(\mathcal{T})}}{2}\right).$$

Adding these, writing $\frac{(u_i^2 - u_{i-1}^2)}{2}$ as a telescoping sum, and dividing by $\mathcal{T}$ yields the theorem.

□

4 One Source

Definition 4.1. By a *strategy*, we mean a finite ordered sequence of inter-arrival times, $\pi = \{X_i\}_{i=1}^n$. By an ∞-*strategy*, we mean an ordered infinite sequence of inter-arrival times of update packets, denoted $\pi = \{X_i\}_{i\in\mathbb{N}}$. We say that a strategy is an *integral strategy* if $X_i \in \pi$ implies $X_i \in \mathbb{N}$. We call π *k-cyclic* if $X_j = X_{j+k}$ for all j. We say that π is *generated by a k-cycle* if given a sequence of k elements of $\mathbb{N}$, ψ_k, then $\pi = \{\psi_k, \psi_k, \ldots, \psi_k, \ldots\}$. For sake of simplicity, we write $\pi = \overline{\psi}_k$ for a strategy of this sort.

For example, the strategy $\pi = \{1, 1, 2, 3, 5, \ldots\}$ is a strategy in which X_j is the j^{th} Fibonacci number. Moreover, the strategy $\pi = \{1, 1, 1, 2, 1, 1, 1, 2, \ldots\}$ is 4-cyclic with $\psi_4 = \{1, 1, 1, 2\}$.

Remark 4.1

The relation between π and the explicit times at which update packets are sent should be mentioned. Namely, π is the sequence of first differences of the update packet times $\{t_i\}_{i\in\mathbb{N}}$. Thus, given any π, the time when the n^{th} update packet is sent may be written as a partial sum of the elements of π. We work with first differences as it reveals information about the cyclic nature of strategies that may otherwise be shrouded.

We have the following basic lemma regarding k-cyclic strategies.

Lemma 4.1. *Suppose $\pi = \overline{\psi}_k$ is a k-cyclic strategy such that the first update packet after a complete cycle has waiting time* 0 *i.e.* $W_{j+k+1} = 0$. *Then,*

$$\Delta = \Delta_{\mathcal{T}(k)} - \frac{\tilde{Q}}{\mathcal{T}(k)},$$

where $\mathcal{T}(k)$ is the time needed to complete one k-cycle.

Proof. We have

$$\Delta = \lim_{\mathcal{T}\to\infty} \frac{\tilde{Q}}{\mathcal{T}} + \frac{\sum_{i=2}^{N(\mathcal{T})} X_i T_i + \frac{X_i^2}{2}}{\mathcal{T}}$$

We may justify a change of variables, $\mathcal{T} = n\mathcal{T}(k)$, since $\lim_{\mathcal{T}\to\infty} \frac{1}{\mathcal{T}} = \lim_{n\to\infty} \frac{1}{n\mathcal{T}(k)}$ and since $\lim_{\mathcal{T}\to\infty} N(\mathcal{T}) = \lim_{n\to\infty} N(n\mathcal{T}(k))$. Moreover, because limits are linear operations, and because $\frac{\tilde{Q}}{\mathcal{T}}$ vanishes as $\mathcal{T}$ tends to ∞, we have

$$\begin{aligned}
\Delta &= \lim_{n\to\infty} \frac{\tilde{Q}}{\mathcal{T}(k)} + \frac{n\left(\sum_{i=2}^{k+1} X_i T_i + \frac{X_i^2}{2}\right)}{n\mathcal{T}(k)} - \frac{\tilde{Q}}{\mathcal{T}(k)} \\
&= \Delta_{\mathcal{T}(k)} - \frac{\tilde{Q}}{\mathcal{T}(k)}
\end{aligned}$$

□

Lemma 4.1 may be interpreted as stating that, for a k-cyclic strategy, the steady state average time converges to the average time of a single cycle minus the boundary terms of that cycle, at the rate which the boundary terms $\frac{\tilde{Q}}{\mathcal{T}}$ vanishes.

Thus, in the case of a single source Ω, allowed to update at integer unit times, we have the following situation: A highly intelligent router looks at the system, which has deterministic system time μ, and relays back to the single source an optimal strategy. Namely, a strategy is *optimal*, denoted π^*, if the average age of information is minimized.

4.1 Server Without Idle Time

There are two specific cases which should be acknowledged, as they shall provide insight into the optimal strategy for a single source. The first, handled here, is the case in which the server is never idle. If the server is never idle, the queue will never completely empty. We want to investigate the best strategy for which the server is never idle. That is, the strategy in which each update packet has strictly positive waiting time, or in which the next update packet arrives precisely when the previous packet finishes processing, but has minimal average information age. For example, if μ is large, $\pi = \{1, 1, 1, \ldots, \}$ is a strategy in which the server is never idle, yet the queue will become arbitrarily full.

We begin by noting that Lemma (3.1) can be simplified if the server is never idle.

Lemma 4.2. *Given a sequence of update packets and a deterministic μ so that $T_{j-1} \geq X_j$ for each j, then $u_n = t_0 + (n+1)\mu$.*

Proof. Suppose $n = 0$, then, since the queue is empty, $u_0 = t_0 + \mu$.

Assume true for $n - 1$. Then, $u_{n-1} = t_0 + n\mu$. Since the server is never idle, $T_{n-1} \geq X_n$, implying $t_0 + n\mu + \mu \geq t_n + \mu$. Thus, $u_n = t_0 + (n+1)\mu$. □

Moreover, we may also write the second half of Theorem (3.2) in a simple form.

Theorem 4.2. *Let $\mathcal{T} = u_n$. Suppose $\{t_i\}_{i=1}^n$ and μ are such that the server is never idle.*

Then, if we let $\Delta_{u_0} = (u_0)\left(\Delta_0 + \frac{u_0}{2}\right)$ *the average age of information may be written as*

$$\Delta_{u_n} = \frac{1}{t_0 + (n+1)\mu}\left(\Delta_{u_0} + \frac{\mu^2(n^2-1)}{2} + (n\mu)t_0 - \left(\mu\sum_{i=1}^{n} t_{i-1}\right)\right) \tag{3}$$

Proof. Using lemma 4.2,

$$\begin{aligned}(u_n)\Delta_{u_n} &= \Delta_{u_0} + \frac{(u_n^2 - u_0^2)}{2} - \sum_{i=1}^{n} t_{i-1}(u_i - u_{i-1}) \\ &= \Delta_{u_0} + \frac{((t_0+n\mu)^2 - (t_0+\mu)^2)}{2} - \left(\mu\sum_{i=1}^{n} t_{i-1}\right) \\ &= \Delta_{u_0} + \frac{\mu^2(n^2-1)}{2} + (n\mu)t_0 - \left(\mu\sum_{i=1}^{n} t_{i-1}\right)\end{aligned}$$

□

Using this theorem, we may devise an optimal strategy so that the server is never idle.

Theorem 4.3. *Let* Ω *be such that* π *is generated by* $G = \mathbb{N}$*. Then,*

$$\pi^* = \{\lfloor\mu\rfloor, \lfloor 2\mu\rfloor - \lfloor\mu\rfloor, \ldots, \lfloor k\mu\rfloor - \lfloor(k-1)\mu\rfloor, \ldots, \lfloor n\mu\rfloor - \lfloor(n-1)\mu\rfloor\}$$

is the optimal finite strategy for which the server is never idle, for $\mu \in (1,\infty)$*. If* $\mu \in (0,1]$*, no such strategy exists in which the server is never idle.*

Proof. We first show that this strategy does indeed ensure that the server is never idle. That is, $T_{j-1} \geq X_j$ for $j > 0$. We may assume $t_0 = 0$, as this is just a shift of our interval of observation, to simplify computations.

We proceed by induction. For $j = 1$, we have that $X_1 = \lfloor\mu\rfloor$, while $T_0 = \mu$. Since $\mu \in (1,\infty)$, we have that $T_0 < X_1$.

Now, assume this is true for $n-1$. Then, $T_{n-2} \geq X_{n-1}$, so Lemma 4.2 applies, ergo $u_{n-1} = n\mu$. This gives that $T_{n-1} = n\mu - \lfloor(n-1)\mu\rfloor$. Also, $X_n = \lfloor n\mu\rfloor - \lfloor(n-1)\mu\rfloor$. Therefore, $T_{n-1} \geq X_n$ since $n\mu \geq \lfloor n\mu\rfloor$.

We show that π^* is the optimal strategy so that the server is never idle. Suppose we have a different strategy so that the server is never idle, call it π. We shall write t_j^* and t_j to denote the times when update packets are sent for π^* and π, respectively.

We first show that this strategy may never have $t_j > t_j^*$ for some j.

We proceed inductively. Suppose for $j = 1$, we have $t_1 = t_1^* + k$, for some $k \in \mathbb{N}$. Then, $X_1 > T_0$, since $\lfloor \mu \rfloor + k > \mu$. Now, suppose $t_i \leq t_i^*$ for $i = 1, \ldots, j-1$. At t_j, we have $t_j = t_j^* + k$ for some $k \in \mathbb{N}$. Then, $X_j > T_{j-1}$, since $\lfloor j\mu \rfloor + k - t_{j-1} > j\mu - t_{j-1}$. This contradicts the supposition that the server is never idle.

Next, we proceed to show that if $t_i < t_i^*$ for some collection of indices, then the average age of π will be greater than that of π^*.

Suppose $t_i = t_i^*$ for $i = 0, \ldots, j-1$. Then, at t_j we must have $t_j = t_j^* - k_j$ for $k_j < X_j$ and $k_j \in \mathbb{N}$. If $k_j > X_j$, then $t_{j-1} < t_{j-1}^*$, a contradiction. For t_ℓ with $\ell = j+1, \ldots, n$, either $t_\ell = t_\ell^*$, or $t_\ell = t_\ell^* - k_\ell$ with $k_\ell < X_\ell$. Call L_1 the union of the set of indices such that $t_\ell = t_\ell^*$ and $i = \{0, \ldots, j-1\}$. Call L_2 the set of indices such that $t_\ell < t_\ell^*$, including t_j, so that $L_1 \cup L_2 = \{1, \ldots, n\}$. Then, using Theorem (4.2),

$$\mu\left(\sum_{i=1}^{n} t_{i-1}^*\right) > \mu\left(\sum_{\ell \in L_1} t_\ell^* + \sum_{\ell \in L_2} t_\ell^* - k_\ell\right)$$

which implies that (3) is larger for π. Since π was arbitrary, π^* is the optimal strategy for which the server is never idle. A similar analysis shows that $\lim\limits_{n\to\infty} \pi^*$ is the optimal ∞-strategy for which the server is never idle.

Finally, if $\mu \in (0, 1]$, then no such strategy exists, as the total system time will always be less than the smallest inter-arrival time. $\square$

For this strategy, it should be noted that π^* is cyclic if and only if $\mu \in \mathbb{Q}$, by simple properties of the floor function.

4.2 Server With Idle Time

Contrariwise, we may discuss the case in which there exists some idle time after every update. That is, for an update packet t_j, we have $W_j = 0$ and $t_j \geq u_{j-1}$. Intuitively, this means that the queue is always empty. Again, we want to discuss the optimal strategy for which the server is always idle. For example, the trivial strategy for which the queue is always empty is to only send a single update packet.

We begin by noting that Lemma (3.1) can be simplified if the queue is always empty.

Lemma 4.3. *Given a sequence of update packets and a deterministic μ so that $W_j = 0$ for each t_j, then $u_n = t_n + \mu$.*

Proof. Suppose $n = 0$, then, since the queue is empty, $u_0 = t_0 + \mu$. Assume true for $n - 1$. Then, $u_{n-1} = t_{n-1} + \mu$. Since the queue is always empty, $T_{n-1} < X_n$, implying $t_0 + (n-1)\mu + \mu < t_n + \mu$. Thus, $u_n = t_n + \mu$.

□

Let us define a recursive sequence as follows:

Let $f_0 = 0$ and

$$\begin{aligned} f_k &= \lceil f_{k-1} + \mu \rceil \\ &= k \lceil \mu \rceil . \end{aligned}$$

Letting $t_k = f_k$, we have the following theorem:

Theorem 4.4. *Let Ω be such that π is an integral strategy for $\mu \in (1, \infty)$. In this case,*

$$\pi^* = \{\lceil \mu \rceil , \lceil \mu \rceil , \ldots , \lceil \mu \rceil\}$$

is the optimal finite strategy for which the queue is always empty.

Proof. We first show that π^* does ensure that there is an idle time after each update packet, so $T_{i-1} \leq X_i$. This may be seen inductively, imitating the first part of Theorem 4.3 by the fact that $k\mu \leq k\lceil \mu \rceil$. We show that π^* is the optimal strategy so that the queue is always empty. Suppose we have a different strategy so that the queue is always empty, call it π. We shall write t_j^* and t_j to denote the times when update packets are sent for π^* and π, respectively.

We first show that this strategy may never have $t_j < t_j^*$ for some j.

Suppose $t_1 = t_1^* - k_1$ for $k \in \mathbb{N}$ and $k < X_1$. Then, we see that $T_0 > X_1$. Inductively, if $t_i = t_i^*$ for $i = 0, \ldots, j-1$, and $t_j = t_j^* - k_j$ for $k_j < X_j$, then we see $T_{j-1} > X_j$, contradicting the assumption that the queue is always empty.

Thus, π must be such that there is some collection of indices L so that $t_\ell = t_\ell^* + k_\ell$, for $k_\ell \in \mathbb{N}$. Clearly then, the average age is minimized by shifting t_ℓ down by k_ℓ, precisely the definition of π^*. A similar analysis shows that $\lim\limits_{n\to\infty} \pi^*$ is the optimal ∞-strategy for which the queue is always empty.

□

4.3 Optimal Strategy

In the following, assume Ω is such that π is an integral strategy. We first discuss lemmata relevant to finding the optimal strategy for one source.

Lemma 4.4. *If π^* is an optimal strategy (∞-strategy) for deterministic service time μ. Then,*

(i) $W_j < 1$ for all j.

(ii) $t_j - u_{j-1} < 1$ for all j.

Proof. (i) Suppose not, so that there exists some j so that $W_j \geq 1$. Since $W_j \geq 1$, we have $T_{j-1} - X_j \geq 1$, so by Lemma 4.2 $u_j = t_0 + (j+1)\mu$. Then, if we let $t_j' = t_j + 1$, we see that

$X_j^{'} = X_j + 1$. This implies that $T_{j-1} \geq X_j^{'}$, so, by Lemma 4.2, $u_j^{'} = u_j$. Using the equation for average age given in Theorem 3.2,

$$\sum_{i=1}^{n} t_{i-1}(u_i - u_{i-1}) < \sum_{i=1}^{j} t_{i-1}(u_i - u_{i-1}) + (t_j + 1)(u_{j+1} - u_j) + \sum_{i=j+2}^{n} t_{i-1}(u_i - u_{i-1}),$$

which shows that the average age is smaller for the strategy with $t_j^{'}$, contradicting optimality. If more than one such j exists, we apply this argument multiple times.

(ii) Suppose not, so that there exists some j so that $t_j - u_{j-1} \geq 1$. Now, for $\ell = j, \ldots, n$, let $t_\ell^{'} = t_\ell - 1$. Doing so will keep the average age of the information the same for all $t_\ell^{'}$ the same as the t_ℓ, but $X_j^{'} = X_j - 1$. Thus, using the geometric characterization of Δ, the average age for this strategy will be lower, contradicting optimality.

□

These lemmata imply directly that an optimal strategy is one in which the times when update packets are sent are either the floor or ceiling of the previous update time. That is, we must have either $t_{j+1} = \lfloor u_j \rfloor$ or $t_{j+1} = \lceil u_j \rceil$. Let $\eta(x) = \lfloor x + \frac{1}{2} \rfloor$, so that η represents "rounding to the nearest integer." We may thus state a conjecture for the optimal strategy for a single source:

Conjecture 4.5. For a single source Ω, the optimal integral ∞-strategy π^* for $\mu \in (\mathbb{R}^+ \setminus \mathbb{N}) \cap (1, \infty)$ may be written as

$$\pi^* = \{\eta(u_0), \eta(u_1) - \eta(u_0), \ldots, \eta(u_k) - \eta(u_{k-1}), \ldots\}. \tag{4}$$

For $\mu \in \mathbb{N}$, the optimal integral ∞-strategy is

$$\pi^* = \{\mu, \mu, \ldots, \mu, \ldots\} \tag{5}$$

and for $\mu \in (0,1]$, the optimal integral ∞-strategy is

$$\pi^* = \{1, 1, \ldots, 1, \ldots\}. \tag{6}$$

Sketch of Proof:

By the previous lemmata, there are two possible optimal strategies for $\mu \in (\mathbb{R}^+ \setminus \mathbb{N}) \cap (1, \infty)$. The first strategy is one in which infinitely many $\lceil u_j \rceil$ occur, and the second is for which finitely many such terms occur. In the first case, since there are infinitely many $\lceil u_j \rceil$ terms, either all times when update packets are sent are of this form, or there exist periods of length n consisting of $\{\underbrace{\lfloor u_j \rfloor, \lfloor u_{j+1} \rfloor, \ldots, \lfloor u_{j+(n-1)} \rfloor}_{(n-1)\text{ times}}, \lceil u_{j+n} \rceil\}$. We wish to show that, for each μ, there exists an n so that the average age of this period is minimal. After the first cycle, which has negligible age contribution, the age from the j^{th} bequeathed to the $(j+1)^{th}$ cycle is given by

$$\int_0^{\mu} t + \lceil \mu \rceil \, dt$$

since $\lceil n\mu \rceil - \lfloor (n-1)\mu \rfloor = \lceil \mu \rceil$. Assuming a cyclic strategy of this sort of period of length n, the average age of such a strategy, denoted Δ_n is given by

$$\begin{aligned}
\Delta_n &= \frac{1}{\lceil n\mu \rceil}\left(\int_0^{\mu} t + \lceil \mu \rceil \, dt + \sum_{i=1}^{n-1} \int_{u_i}^{u_{i+1}} t - t_i \, dt + \int_{n\mu}^{\lceil n\mu \rceil} t - \lfloor (n-1)\mu \rfloor \, dt\right) \\
&= \frac{1}{\lceil n\mu \rceil}\left(\int_0^{\lceil n\mu \rceil} t \, dt - \sum_{i=1}^{n-1} \int_{u_i}^{u_{i+1}} t_i \, dt + \int_0^{\mu} \lceil \mu \rceil \, dt - \int_{n\mu}^{\lceil n\mu \rceil} \lfloor (n-1)\mu \rfloor \, dt\right) \\
&= \frac{1}{\lceil n\mu \rceil}\left(\frac{\lceil n\mu \rceil^2}{2} - \sum_{i=1}^{n-1} \int_{u_i}^{u_{i+1}} i \lfloor \mu \rfloor \, dt + \alpha(n) + \mu \lceil \mu \rceil - \lfloor (n-1)\mu \rfloor (\lceil n\mu \rceil - n\mu)\right)
\end{aligned}$$

where $\alpha(n)$ is a linear function of n, by the equation $\lfloor i\mu \rfloor = i \lfloor \mu \rfloor + \alpha_i$. Moreover, since $\lceil n\mu \rceil = n\mu + \beta$ for $0 \leq \beta < 1$, we may write

$$\begin{aligned}\Delta_n &= \frac{1}{\lceil n\mu \rceil}\left(\frac{(n\mu+\beta)^2}{2} - \frac{1}{2}(n^2-n)\mu\lfloor\mu\rfloor + \alpha_n + \mu\lceil\mu\rceil - \lfloor(n-1)\mu\rfloor(\lceil n\mu\rceil - n\mu)\right)\\ &= \frac{1}{\lceil n\mu\rceil}\left(\frac{1}{2}(n^2(\mu^2-\mu\lfloor\mu\rfloor)) + \omega(n)\right),\end{aligned}$$

where $\omega(n)$ is a function bounded above and below by linear functions in n, since

$$0 \le \lfloor(n-1)\mu\rfloor(\lceil n\mu\rceil - n\mu) < \lfloor(n-1)\mu\rfloor.$$

Thus, $\lim_{n\to\infty}\Delta_n = \infty$, since $\mu^2 > \mu\lfloor\mu\rfloor$ for $\mu \in (\mathbb{R}^+ \setminus \mathbb{N}) \cap (1,\infty)$. Let $M \in \mathbb{R}$ be sufficiently large, and choose $N \in \mathbb{N}$ so that $\Delta_N > M$. Thus, we may restrict our attention to the subset $J := \{0, 1, \ldots, N-1\} \subset \mathbb{N}$. Since this is a finite list, $\min_{n\in J}\{\Delta_n\}$ exists, though it need not be unique. Ergo, given an arbitrary strategy with infinitely many ceiling terms, we may write the average age as

$$\Delta = \lim_{\mathcal{T}\to\infty}\frac{1}{\mathcal{T}}\left(\left(\frac{a_i}{\sum_{i=0}^{P(\mathcal{T})} a_i}\right)\sum_{i=0}^{P(\mathcal{T})}\Delta_i\right)$$

where $P(\mathcal{T})$ is the longest cycle to complete before time $\mathcal{T}$, and a_i is the number of times Δ_i occurs. An optimal strategy is therefore the strategy in which we replace all Δ_i terms with $\min_{n\in J}\{\Delta_n\}$.

Moreover, given a strategy with finitely many $t_{j+1} = \lceil u_j \rceil$ terms, then we may truncate the sequence after the last ceiling term. The first piece of this truncation represents a finite boundary contribution, so the average time will depend on the infinite tail of floor terms ante-ceding the ceiling term. However, such a sequence is precisely $\lim_{n\to\infty}\Delta_n$. Since this is infinite, we conclude that a strategy with finitely may ceiling terms is not optimal.

Therefore, if we calculate $\min_{n\in J}\{\Delta_n\}$ precisely, we will have our optimal strategy. We conjecture that this is precisely (4). We also conjecture, that for $\mu \in \mathbb{N}$ and $\mu \in (0, 1]$, our optimal strategies will be (5) and (6).

5 Future Directions and Applications

In future work, this model can surely be extended to non-deterministic service times. Namely, interesting models would arise if μ is given as a Gaussian Distribution, or a Uniform Distribution. It should be noted that strategies generated by $\mathbb{N}$ can be scaled accordingly, depending on the capabilities of the source. For example, a rather applicable model would be one in which $\mathbb{N}$ is scaled significantly, say, a minimum inter-arrival time of $\frac{1}{n}$ for some n, and in which μ is a Gaussian Distribution. This sort of model would cover systems such as automated stock market updates, where the system processes the update packets randomly.

Moreover, we would like to generalize these models to strategies generated by $\mathbb{Q}$, with some minimal inter-arrival time $q \in \mathbb{Q}$. Given a strategy generated by $\mathbb{Q} \setminus \{r \in \mathbb{Q} : r < q\}$, we would be able to accurately model most real world systems of status updates. Though, models of this sort may become increasing complex as we relax constraints. Thus, this work may set the foundations for work in computational models that cover relaxed versions of this model. For example, if, for an integer number of sources, we wish to minimize the average age, variance, and cost, over a strategy generated by this rational set, such a model would probably best be handled computationally. Other models than can be developed from this model include modifying the definition of optimal to fit some other predetermined set of constraints.

6 Acknowledgments

I'd like to thank Dr. Shan-Yuan Ho for suggesting this problem and for her discussions on the subject. I am grateful to Mr. Siddharth Venkatesh for his guidance in regards to the problem. I'd like to thank Dr. Tanya Khovanova for reading and editing my paper, as well as suggesting future directions through discussion. Additionally, I'm grateful to Dr. John

Rickert for his discussions and support. Finally, I'd like to thank the Center for Excellence in Education, the Massachusetts Institute of Technology, and the MIT Math Department for hosting this research.

References

[1] D. Alberts, D. Papp, and et al. The information age: An anthology on its impacts and consequences (information age anthology, volume 1); part 1: The information and communication revolution. *Natl Defense Univ Pr*, 1997.

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

[3] S. Kaul and R. Yates. Real-time status updating: Multiple sources. *Proc. IEEE International Symposium on Information Theory*, 2012.

[4] A. Papoulis. Probability, random variables, and stochastic processes, 3rd ed. *New York: McGraw-Hill*, 1991.

[5] J. Kim, H. Kim, and K. Lakshmanan. Parallel scheduling for cyber-physical systems: analysis and case study on a self-driving car. *Proceedings of the ACM/IEEE 4th International Conference on Cyber-Physical Systems*, 2013.

[6] M. Costa, M. Codreanu, and A. Ephremide. Age of information with packet management. *Proceedings of ISIT*, 2014.

[7] S. Kaul, R. Yates, and M. Gruteser. tatus updates thro ugh queues. *Proc. 46th Annual Conference on Information Sciences and Systems*, 2012.

[8] A. Reddy, S. Banerjee, and A. Gopalan. On distributed scheduling with heterogeneously delayed network-state information. *Queueing Systems, Vol. 72, No. 3-4, pp 193-218*, Dec 2012.

[9] L. Ying and S. Shakkottai. On throughput optimality wit h delayed network-state information. *IEEE Trans. on Information Theory , vol. 57, pp. 51165132*, 2011.