Age-of-Information in Tandem Queues with Delayed Feedback: Zero-Wait vs. Pipelining

Mahsa Noroozi, Markus Fidler
Institute of Communications Technology, Leibniz University Hannover
Jaya Prakash Champati, Joerg Widmer
IMDEA Networks Institute, Madrid

***Abstract*—An established policy for updating systems is zero-wait: a source immediately sends a new sample as soon as the sink acknowledges the receipt of the previous one. The rationale of zero-wait is that with instantaneous feedback, the transmission of samples can fully utilize the forward link without ever causing a queue. However, this ideal behavior does not extend to multi-hop networks and two-way delay. One approach to generalize zero-wait for use in larger networks is message pipelining, where there is a fixed number of samples and acknowledgments $k \geq 1$ in the network at any time.**

We analyze the peak age-of-information of updating systems with pipelining in multi-hop networks with arbitrarily many queues in the forward and feedback paths. While pipelining improves network utilization, it also increases queuing delays, and the optimal degree k must strike a balance between the two. We show how this depends on the diameter and topology of the network, the presence of bottlenecks, and the statistical distribution of service times. In an a priori unknown and changing network, it is beneficial to adjust the pipelining adaptively. We demonstrate how basic delay-based congestion control can be effectively used to achieve this goal.

I. Introduction

We consider an information updating system in which a source samples a sensor and transmits the samples as messages via a network to a sink. A variety of such cases can be found in cyber-physical systems, e.g., in the remote estimation of a physical process [1], [2], and in networked feedback control [3], [4]. Applications arise in industrial automation, robotics, and intelligent transportation systems [5], [6]. An important performance metric for updating systems is the age-of-information, or in short 'age', which quantifies the freshness of information at the sink. Considering the latest sample available at the sink, the age-of-information is defined as the difference between the generation time of this sample and the current time. Whenever a new sample is received at the sink, the age is reduced, resulting in a characteristic saw-tooth function with peaks in the age immediately before the reception of each new update message. Generally, the goal is to find policies that minimize the (peak) age-of-information.

In recent years, age-of-information has been a very active area of research and today analytical solutions for a wide range of queuing models [6]–[11], wireless channel models [12], [13], tandem queues [14], [15], and parallel channels [16]–[18], to name a few, are known. Recent, comprehensive surveys are [19], [20]. Next to the queuing and scheduling discipline, the sampling process also has a major impact on the age. Typically periodic or random sampling such as Poisson is used, e.g., [6]. In addition to these open-loop models, there is significant interest in closed-loop systems, where the generation of a new sample is controlled by feedback about the transmission of the previous sample.

This work was supported in part by DFG grant FI 1236/9-1.

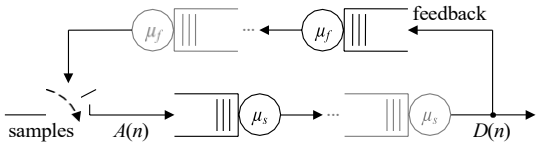


Fig. 1. Updating system with zero-wait policy and two-way delays.

A closed-loop system comprises a forward path for the transmission of samples and an additional feedback path. Feedback can be generated by the transmission system in the forward path or as acknowledgment messages from the sink. An accepted closed-loop policy is zero-wait, where feedback about the delivery of one sample immediately triggers the transmission of a new sample. Typically, the forward path is assumed to be a single link characterized by a random transmission time, while feedback is instantaneous. Because there is always exactly one message in the loop, zero-wait achieves full utilization of the forward link without creating a queue, i.e., no buffer is required. Interestingly, it turns out that the zero-wait policy is not optimal. Instead, an update-or-wait policy is better, which in certain cases when the transmission time of a sample is short, adds an extra waiting time before taking the next sample [21], [22].

In general, closed-loop policies are studied under the assumption that feedback is not delayed. Only recently, two-way delays have been included and both the forward and feedback paths have been characterized by a link model [23]–[28]. Fig. 1 shows an example with one queuing system in the forward path and one in the feedback path (only consider the black queues and not the gray ones). Most closely related to our work is [28] which proposes a zero-wait-2 policy that maintains a pipeline of two active messages to deal with two-way delays.

Compared to [28], ours is the first work to consider networks with an arbitrary number of buffered links in the forward and the feedback path (including any number of gray queues in Fig. 1) and an arbitrary degree of pipelining k, where k messages are in the network at any time. The contributions

of our work are as follows. We map the updating system in Fig. 1 with pipelining to a closed queuing network and derive the peak age-of-information. The peak age is composed of the inter-sample time and the delay of the forward path. Both are determined by the pipelining degree k. Particularly, the generation of messages, and hence the network utilization is not controlled externally but arises intrinsically through self-clocking via the feedback. It depends on network parameters, such as diameter, statistical distribution of the service times, propagation times, and on the degree of pipelining k. The optimal degree of pipelining achieves a message rate that balances the inter-sample times and the queuing delays in the forward path. Stop-and-wait $k = 1$ is the preferable option if the network diameter is small and if feedback is fast. Otherwise, in most practical cases, pipelining $k \geq 2$ performs better. The optimal degree of pipelining is larger, if the feedback path is the bottleneck, since queuing delays in the feedback path do not contribute to the age. Conversely, the optimal degree of pipelining is smaller, if the forward path is the bottleneck. Further, if the variability of the service times is larger, a smaller degree of pipelining k performs better. This is also a consequence of self-clocking, which transfers the variability of service times to the burstiness of message generation. For a given but unknown network, we adjust k adaptively. Our adaptation logic uses the basic approach of delay-based congestion control which proves to be effective.

The remainder of this work is organized as follows. Sec. II presents related works. In Sec. III we model the updating system with pipelining as a closed queuing network and derive the age-of-information thereof. Sec. IV evaluates how the degree of pipelining affects the age under different network topologies and parameters. In Sec. V we show how the source can adaptively adjust the degree of pipelining to the network. Sec. VI presents brief conclusions.

II. Related Works

Only recently, a few papers have addressed the difficulties of closed-loop updating systems under two-way delays [23]–[28]. In [23], the authors study the update-or-wait policy with constant feedback delay. For general service time distribution, they characterized the optimal sampling policy for minimizing the mean peak age by allowing service preemptions.

In [24], [25], the authors study a generalization of the update-or-wait policy under two-way delays, where samples in the forward channel, as well as acknowledgments in the feedback channel, have a random transmission time. In [25], the decision when to take a new sample and when to wait is a joint decision that is distributed among the source, monitoring a Wiener process, and the sink: upon reception of an acknowledgment, the source may decide to wait before taking a new sample; similarly, when receiving the sample the sink may decide to wait before sending the acknowledgment. The authors of [26] study remote estimation of a sensor process in an updating system with an unreliable forward channel and a reliable feedback channel, both with random transmission times. In their solution, a new sample is sent immediately if the previous transmission fails. Otherwise, the source waits until the expected estimation error exceeds a threshold. Different from our work, the papers [23]–[26] use variants of a stop-and-wait policy that ensures that there is only one message in the loop at any time. Hence, queuing delays do not occur and channels only cause a random transmission time.

Systems with queues, one in the forward and one in the feedback path are studied in [27], [28], both using a discrete-time model. The author of [27] compares variants of update-and-wait, zero-wait, and open-loop with periodic sampling and derives a lower and an upper bound of the age-of-information of an optimal policy under two-way delay. Following our results in Fig. 3, the authors conclude that it is advantageous to switch to the open-loop system if the mean service times in the feedback path are larger than in the forward path. The authors in [28] consider a source that sends samples at will and a sink that generates control packets to trigger new samples. Service times in the forward and the feedback channels are geometric. They phrase optimal control of the sampler as a Markov decision process and derive the average age of three policies wait-1, zero-wait-1, and zero-wait-2, that use either stop-and-wait, i.e., $k = 1$, or pipelining of degree $k = 2$. Depending on the forward and feedback service rates, either zero-wait-1 or zero-wait-2 is shown to achieve a smaller age. In contrast to [27], [28] we consider arbitrary numbers of queues in the forward and feedback paths, and an arbitrary degree of pipelining $k \geq 1$, using closed queuing networks.

In the Internet, congestion control adjusts the degree of pipelining at the transport layer, e.g., the Transmission Control Protocol (TCP) uses feedback provided by acknowledgments. While throughput is a primary focus of TCP, the recently proposed Age Control Protocol ACP+ develops an algorithm that adjusts the rate of update messages to control network backlog and therefore the age-of-information [29]. In our work, we contribute an analysis of a queuing model of the feedback loop that is part of congestion control.

III. Queuing Analysis

We define a queuing model of an updating system with two-way delays and message pipelining and analyze the age-of-information in steady-state. First numerical results illustrate how pipelining benefits the age.

A. System Model

We investigate the queuing network shown in Fig. 1 as a generic model of an updating system with two-way delays and message pipelining. Samples $n \in \mathrm{N}$ are generated and transmitted as messages at times $A(n)$. The messages arrive at the sink at times $D(n)$. The sink immediately confirms the receipt of each sample through feedback, i.e., acknowledgment messages. When a feedback message is received at the source, a new sample is generated and transmitted. The system uses pipelining of degree $k \in \mathrm{N}$, i.e., there are exactly k messages (samples plus feedback messages = k) in the system at any given time. The system starts with k messages in the first

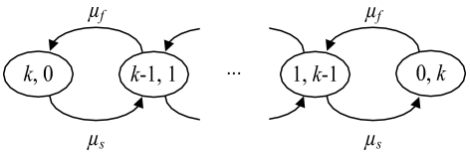


Fig. 2. State transition diagram.

queue and any new message is sent to the first queue only when a feedback message is received from the last queue.

Optionally, feedback messages can contain additional information beyond the acknowledgments. An example is networked feedback control, where the feedback messages also contain an actuator signal.

The queuing network consists of a variable number of work-conserving, lossless, first-come first-served queues $l \in \mathrm{N}$, of which $l_s \in \{1, 2, \ldots, l\}$ queues are in the forward path and $l_f = l - l_s$ queues are in the feedback path. We model the service times of messages carrying samples in the forward path as exponential random variables with mean $1/\mu_s$ and the service times of feedback messages as exponential with mean $1/\mu_f$. Different mean service times for the individual queues in the forward and feedback path are possible at the expense of additional notation.

For the special case $l_s = 1$, $l_f = 0$, and $k = 1$, we have the well-known instantaneous feedback model with zero-wait policy, whereas for $l_f \geq 1$ there is feedback delay. Furthermore, for $l \geq 2$, messages may reside in any of the queues of the network, so that even in case of instantaneous feedback, i.e., $l_f = 0$, the sender does not know, how far the messages have traveled and which forward queues are idle. The problem increases with l, where we explore how pipelining can help and evaluate how the degree of pipelining k affects the age.

We consider the mean peak age-of-information that can be expressed as, e.g., [19],

$$\hat{\Delta} = \mathrm{E}[T(n-1)] + \mathrm{E}[D(n-1, n)], \tag{1}$$

where $T(n) = D(n) - A(n)$ is the system time of the forward path and $D(n-1, n) = D(n) - D(n-1)$ is the inter-departure time of messages at the sink.

B. Steady-state Analysis

The updating system that we consider, Fig. 1, is a closed queuing network. We denote the state of the network $\mathbf{k} = (k_1, k_2, \ldots, k_l)$ where $k_i \in \mathrm{N}_0$ is the number of messages at queue $i \in \{1, 2, \ldots, l\}$, the head of the line message is in service and the subsequent messages are waiting, if any. At every point in time it holds that $\sum_{i=1}^{l} k_i = k$, so that the state space can be formally written as

$$S(k, l) = \left\{ \mathbf{k} \in \mathrm{N}_0^l : \sum_{i=1}^{l} k_i = k \right\}. \tag{2}$$

State transitions occur whenever a message finishes service at queue i and moves to the next queue $i \pmod{l} + 1$. In the special case $i = l_s$, a message departs from the network to the sink and an acknowledgment message is sent in the feedback path. In the case of $i = l_s + l_f = l$, an acknowledgment message arrives at the source and triggers a new sample that is transmitted in the forward path.

The process of the state of the queues is visualized as a state transition diagram in Fig. 2 for the example $l_s = l_f = 1$. This corresponds to the network comprising the two queues depicted in black in Fig. 1. With larger networks, the dimension of the state increases. The transition rates are the service rates of the queues μ_s and μ_f and due to the memorylessness of the service times, the process is a Markov chain.

We denote by $\mathbf{Q}$ the transition matrix of the Markov chain and the state probabilities in equilibrium by $\pi(\mathbf{k})$. The state probabilities can be obtained as the solution of the global balance equations, see e.g. [30], expressed as

$$\boldsymbol{\pi}\mathbf{Q} = 0.$$

The marginal state probabilities, i.e., the probability that there are κ messages in queue $i \in \{1, 2, \ldots, l\}$, are denoted by

$$\pi_i(\kappa) = \sum_{\mathbf{k} \in S(k,l):\, k_i = \kappa} \pi(\mathbf{k}). \tag{3}$$

For the example of the Markov chain for $l = 2$ queues in Fig. 2 and a pipelining degree of $k = 2$, we have a two-dimensional state space $S(2, 2) = \{(2, 0), (1, 1), (0, 2)\}$ that comprises three possible states. The state transition matrix is

$$\mathbf{Q} = \begin{bmatrix} -\mu_s & \mu_s & 0 \\ \mu_f & -\mu_s - \mu_f & \mu_s \\ 0 & \mu_f & -\mu_f \end{bmatrix}.$$

The state probabilities are $\boldsymbol{\pi} = (\pi(2, 0), \pi(1, 1), \pi(0, 2))$ and the marginal state probabilities are simply $\pi_1(\kappa) = \pi(\kappa, k-\kappa)$ and $\pi_2(\kappa) = \pi(k - \kappa, \kappa)$ where $\kappa \in \{0, 1, 2\}$ and $k = 2$.

The performance measures result directly from the state probabilities [30]. The utilization of queue i is $\rho_i = 1 - \pi_i(0)$. The throughput of all queues $i \in \{1, 2, \ldots, l\}$ is identical

$$\lambda = \mu_i(1 - \pi_i(0)),$$

and the mean inter-departure time is

$$\mathrm{E}[D(n-1, n)] = \frac{1}{\lambda}.$$

The mean number of messages at queue i is

$$\mathrm{E}[K_i] = \sum_{\kappa=1}^{k} \kappa \pi_i(\kappa)$$

and with Little's theorem, the mean system time at queue i is

$$\mathrm{E}[T_i] = \frac{\mathrm{E}[K_i]}{\lambda}.$$

The mean peak age follows by insertion into Eq. (1) as

$$\hat{\Delta} = \frac{1 + \sum_{i=1}^{l_s} \mathrm{E}[K_i]}{\lambda} = \frac{1 + \sum_{i=1}^{l_s} \sum_{\kappa=1}^{k} \kappa \pi_i(\kappa)}{\mu_1(1 - \pi_1(0))}. \tag{4}$$

In case of larger networks, the Markov chain of the state-transition diagram in Fig. 2 becomes cumbersome due to

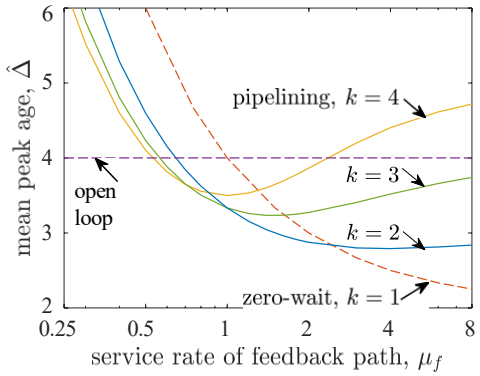


Fig. 3. Mean peak age of a network with $l = 3$ queues.

the increasing dimensionality. Therefore, we use Gordon and Newell's product-form solution for closed queuing networks, see e.g., [30]. For the updating system in Fig. 1 the state probabilities in equilibrium are

$$\pi(\mathbf{k}) = \frac{1}{G(k, l)} \prod_{i=1}^{l} \frac{1}{\mu_i^{k_i}},$$

where $\mu_i = \mu_s$ for $i \in \{1, \ldots, l_s\}$, $\mu_i = \mu_f$ for $i \in \{l_s + 1, \ldots, l_s + l_f\}$, and $G(k, l)$ is a normalization constant defined as

$$G(k, l) = \sum_{\mathbf{k} \in S(k,l)} \prod_{i=1}^{l} \frac{1}{\mu_i^{k_i}},$$

where $S(k, l)$ is the state-space defined in Eq. (2). The marginal state probabilities Eq. (3) and the mean peak age Eq. (4) follow as before.

C. Case Study

We begin our evaluation with a comparatively small network with $l = 3$ queues, $l_s = 1$ in the forward path, and $l_f = 2$ in the feedback path. We make this choice because it allows us to illustrate some relevant effects. A wider range of networks is elaborated in more detail in Sec. IV.

The mean peak age of the updating system is shown in Fig. 3. The service rate in the forward path is normalized $\mu_s = 1$, and the service rate in the feedback path μ_f is varied. Also, we show results for different degrees of pipelining k. For $k = 1$ message in the network, the updating system uses the well-known zero-wait policy. For this case the mean peak age can be easily computed as $\hat{\Delta} = 2/\mu_s + 2/\mu_f$, that is the sum of the mean service times of a sample, of a feedback message (that traverses two queues), and of the following sample. For $k > 1$ we have an updating system with pipelining. For comparison, we also show results for an open-loop M|M|1 queue that does not use the feedback channel. In this case, the arrival rate is set so that the utilization of the forward path is $\rho_s = 0.5$, which minimizes the mean peak age (cf. [8]) of the open-loop system resulting in $\hat{\Delta} = 4$.

Regarding the service rate of the feedback path μ_f, we can distinguish three regions:

a) $\mu_f \gg \mu_s$: If μ_f is large, in the limit $\mu_f \to \infty$, the case of instantaneous feedback is approached. In this case, the zero-wait policy, i.e., $k = 1$, converges towards a mean peak age of $\hat{\Delta} = 2$ and starting at $\mu_f > 2.7$ it outperforms the system with pipelining, $k \geq 2$. The reason for this is that feedback messages are delivered with negligible delay so that all $k \geq 2$ messages of the system with pipelining tend to be found in the forward path. In the given example, the forward path has only one queue and the size of this queue grows linearly with the degree of pipelining k. This generalizes an observation that is made in [28] for pipelining degree $k = 2$ compared to $k = 1$ for a network with $l = 2$ queues.

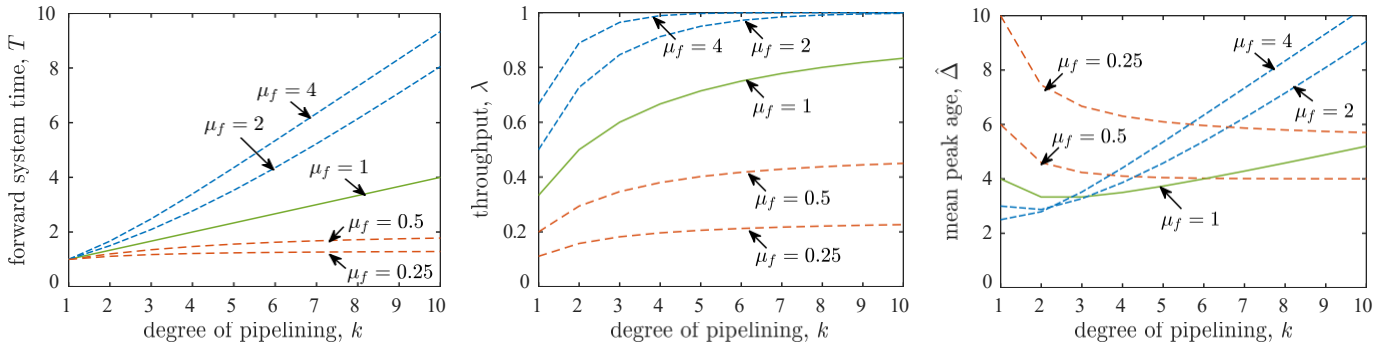
b) $\mu_f < \mu_s$: The service rate of the feedback path is smaller than that of the forward path. This may occur, e.g., if feedback messages carry additional information that increases their service time. In this case, the feedback path is the bottleneck and the forward path is under-utilized. Increasing the degree of pipelining k helps somewhat, but with decreasing μ_f even a large k cannot fix the problem. As a consequence, the open-loop system performs better starting at $\mu_f < 0.6$. Here, a different feedback strategy, e.g., fewer feedback messages with cumulative acknowledgments for several messages in the forward path, would be preferable.

c) $\mu_f \gtrsim \mu_s$: In the practical case where the feedback path is neither instantaneous nor a serious bottleneck, here $0.6 \leq \mu_f \leq 2.7$, the system with pipelining outperforms the zero-wait policy. Ideally pipelining achieves a continuous flow of update messages, while a careful choice of the degree k avoids congestion in the forward path. For such a small network a moderate degree of pipelining is already sufficient. However, there is no limit to k, and in some of the networks we consider in Sec. IV, significantly larger k are required.

For the example network used for Fig. 3, Fig. 4 shows details of the different effects that occur. Here, we vary the degree of pipelining k and show curves for different feedback service rates μ_f, clustered in the three regions identified above. Since the mean peak age Eq. (1) is composed of the mean system time of the forward path T and the mean inter-departure time at the sink, that is the reciprocal of the throughput λ, we also show these quantities on their own.

The system time of the forward path shown in Fig. 4(a) increases with k. This is to be expected as more messages can cause queues to build up. However, it is important to distinguish where these queues form. When μ_f is small, queuing occurs predominantly in the feedback path and the system time of the forward path is only slightly affected by k. As μ_f increases, there is less queuing in the feedback path and more in the forward path. In this case, the increase of the forward path system time with k becomes more pronounced and it is becoming increasingly important to set k well.

Due to the feedback mechanism, the throughput λ as well as the arrival rate of new sample messages are intrinsically determined by the queuing network and the degree of pipelining k. The throughput shown in Fig. 4(b) increases as k increases and approaches the service rate of the forward path $\mu_s = 1$ when the rate of the feedback path $\mu_f \geq \mu_s$, i.e. when the



(a) Forward path system time (b) Throughput (c) Mean peak age

Fig. 4. Network with $l = 3$ queues as used for Fig. 3.

forward path is the bottleneck. Otherwise, if $\mu_f < \mu_s$, the feedback path is the bottleneck and the throughput is limited by μ_f . The throughput is reflected in the age via the mean inter-departure time $1/\lambda$.

The mean peak age depicted in Fig. 4 is composed of the mean system time of the forward path and the mean inter-departure time. If $\mu_f < \mu_s$, the forward channel is under-utilized due to slow feedback and the age is affected by large inter-departure times. This effect can be mitigated somewhat by pipelining, but cannot be prevented. If $\mu_f \geq \mu_s$, careful pipelining improves the age. A lower degree of pipelining is required when the feedback service rate is higher, i.e., when the delay in the feedback path is low.

IV. Evaluation

To evaluate further relevant service time distributions such as deterministic and Pareto and to include propagation times, we implement the model in Fig. 1 in the OMNeT++ simulator. We show the mean peak age and the network utilization for $l_s = l_f = 1$ and different degrees of pipelining k.

1) Deterministic service times: First, we use deterministic service times for the forward and feedback channels. We also add a fixed propagation time W to each channel. We evaluate the mean peak age for different k increasing from 1 to 40. We compare the results regarding the service rates of the forward and feedback paths in two different regions:

a) $\mu_f \geq \mu_s$: The service rate of the feedback channel surpasses that of the forward channel, e.g., $\mu_s = 1$, $\mu_f = 2$ in Fig. 5(a). Thus, the forward path is the bottleneck and since the system is purely deterministic, queuing occurs only in the forward path, if at all. These queuing delays directly impact the age. Depending on the propagation times W in both paths, different degrees of pipelining achieve the minimal age. If the propagation time $W = 0$, the feedback loop is almost instantaneous and increasing k quickly results in a queue in the forward path, which increases the age. When W is larger the feedback loop gets slower and more messages can be in flight at the same time. This requires a larger degree of pipelining. The optimal k that minimizes the age achieves full utilization of the forward path as can be seen in Fig. 5(c). Eventually, when k is increased beyond the optimal value, a queue begins to build in the forward path that increases the age, e.g., in the case of $W = 5ms$ a queue starts to build at $k \geq 12$.

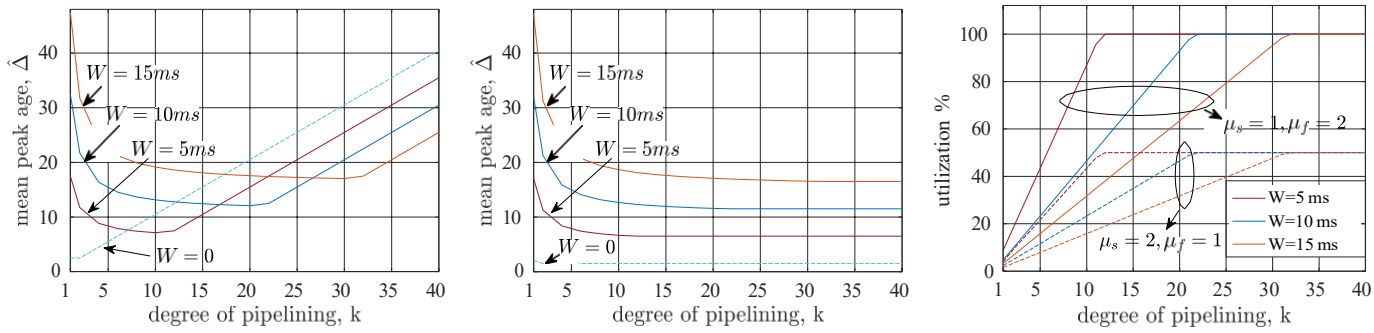
b) $\mu_f < \mu_s$: In contrast to the previous case, here, the service rate of the feedback path is lower than that of the forward path, e.g., $\mu_s = 2$, $\mu_f = 1$ in Fig. 5(b). Queuing now only occurs in the feedback path, as it is the bottleneck. As before, larger propagation times W require a higher degree of pipelining. However, since the bottleneck in the feedback path determines the sampling rate, the forward path generally cannot be fully utilized and the utilization is limited by μ_f , as shown in Fig. 5(c). Increasing k further does not affect the age, since queuing only forms in the feedback path.

2) Random service times: In Fig. 6, we consider two random service time distributions, exponential and Pareto. In case of the Pareto distribution, we set the shape parameter $a = 1.5$, which causes an infinite variance of the service times. In these experiments, the service times of the forward and the feedback path have the same distribution and the same mean rate $\mu_f = \mu_s = 1$. We omit to show results for deterministic service times for this case since they match Fig. 5(a) closely.

Compared to the case of deterministic service times, the variability of random service times results in transient queuing even when the mean network utilization is less than one. In addition, due to the nature of the closed queuing network, the variability of service times results in an increased burstiness of the arrival process. As a consequence, the degree of pipelining that minimizes the age gets smaller. Due to the high burstiness, the effect is even stronger with Pareto service times. The optimal k that minimizes the age in the different cases is summarized in the following table:

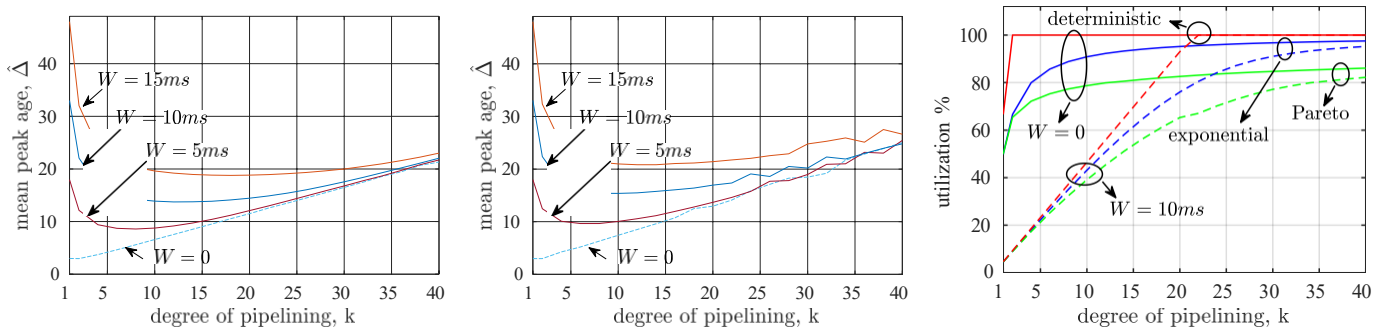
Service times	$W = 5ms$	$W = 10ms$	$W = 15ms$
Deterministic	12	22	32
Exponential	8	12	18
Pareto	8	10	14

The burstiness also impacts the average network utilization. When a large burst of messages is queued in the feedback path, the forward path can become idle. We show the effect on the utilization in Fig. 6(c) for $W = 0ms$ and $W = 10ms$ propagation time. It can be seen that the utilization is generally lower in case of random service times and converges slowly



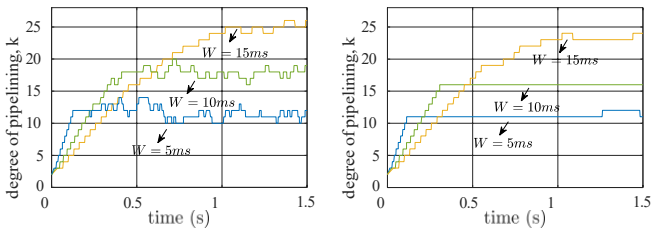
(a) $\mu_s = 1, \mu_f = 2$ (b) $\mu_s = 2, \mu_f = 1$ (c) Utilization

Fig. 5. Mean peak age of a network for different propagation times W, with different deterministic service times in (a), (b), and their utilization in (c).



(a) Exponential (b) Pareto (c) Utilization

Fig. 6. Mean peak age of a network with different service time distributions with a mean of $\mu_s = \mu_f = 1$.



(a) Current RTT (b) Using EWMA

Fig. 7. Adaptive pipelining for exponential service times and $\mu_s = \mu_f = 1$.

as k is increased. Since the Pareto distribution causes a higher burstiness, the utilization is even smaller in this case.

Since we consider the case $\mu_f = \mu_s = 1$, queuing occurs equally in both the forward and the feedback paths. The situation is different in the deterministic case: when k is increased beyond full network utilization, only the queue in the forward path grows. Since queuing delays in the forward path but not in the feedback path affect the age, we see a different slope of the age with increasing k in Fig. 5(a) for deterministic service compared to Figs. 6(a) and 6(b) for random service.

V. Adaptive Degree of Pipelining

We have shown how the correct setting of k can minimize the age-of-information of the pipelining protocol in different network scenarios. In practice, relevant network parameters including link service rates and delays are unknown a priori and may change during operation. It is important to develop methods that are adaptive, such as the Age Control Protocol ACP+ [29]. Inspired by the basic approach of delay-based congestion control of TCP Vegas and BBR, e.g., [31], we use an adaptation logic that estimates the size of queues in the network to adapt the degree of pipelining k dynamically. Different from [29], we do not explicitly calculate a target message rate, as in our case this is controlled by k.

The algorithm operates as follows. The source monitors the round-trip time RTT, measured as the time between sending a sample and receiving the corresponding acknowledgment. Using the current RTT and a $baseRTT$, which is an observation of the smallest RTT that is updated regularly, the source can estimate the queuing delay in the network as $RTT - baseRTT$. Since k/RTT is the current throughput of the pipelining protocol, an estimate of the queue size in the network is $Q = (RTT - baseRTT) \cdot k/RTT$. The goal is to maintain a small queue in the network to achieve good utilization without causing noticeable delays. Hence, the algorithm starts with a small number of k and increments/decrements k by one per RTT depending on the estimate of Q. It increases k when the condition $Q < a$ holds and it decreases k when the condition $Q > b$ is met, where $a = 2$ and $b = 5$ [31].

We show simulation results in Fig. 7 for a network with exponential service times ($\mu_s = \mu_f = 1$) and propagation times ($W = 5, 10, 15ms$). The convergence of k to a steady-state value can be seen. The value is slightly higher than the optimal value of k observed in Fig 6(a) since the algorithm strives to maintain a queue of size 2 to 5 messages in the

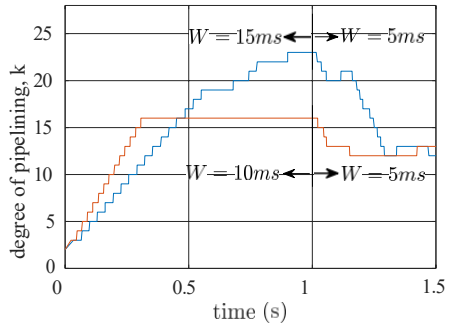


Fig. 8. Adaptive pipelining using EWMA in a network with changing delay.

network. In case of higher propagation times, the convergence takes longer since the acknowledgments are delayed.

We evaluate two different policies to update the degree of pipelining. First, we use the current RTT to estimate Q as can be seen in Fig. 7(a). Second, we use an exponential weighted moving average (EWMA) of the RTT. We compute $\text{EWMA}_n = \gamma RTT_n + (1 - \gamma)\ \text{EWMA}_{n-1}$ where RTT_n is the RTT observed by message n and $\gamma = 0.1$. We use the EWMA of the RTT to estimate Q. This makes the adaption significantly smoother, as shown in Fig. 7(b).

In another experiment, we change the network properties by reducing the propagation time after $1s$ from the beginning of the simulation. In one scenario, we start with $W = 10ms$ and reduce to $W = 5ms$, in the other $W = 15ms$ drops to $W = 5ms$. We use the EWMA procedure and exponential service times as before. In the first $1s$, Fig. 8 shows the same behavior as we saw in Fig. 7(b). After $1s$, however, the algorithm starts to adapt to the smaller delay and it reduces k, showing a good performance of the adaptation logic.

VI. Conclusions

We explored an update policy with message pipelining to deal with two-way delays and multi-hop networks. We modeled the updating system as a closed queuing network and derived the mean peak age-of-information. The degree of pipelining k determines the mean update rate and the extent of queuing delays. The optimal choice of k causes a trade-off between the two. We performed a comprehensive numerical study that showed how k can be adjusted to minimize the age-of-information. In practice, a basic delay-based congestion control protocol that monitors the round-trip time can adaptively adjust the degree of pipelining.

References

[1] Y. Sun, Y. Polyanskiy, and E. Uysal, "Sampling of the Wiener process for remote estimation over a channel with random delay," *IEEE Trans. Inf. Theory*, vol. 66, no. 2, pp. 1118–1135, Feb. 2020.

[2] T. Z. Ornee and Y. Sun, "Sampling and remote estimation for the Ornstein-Uhlenbeck process through queues: Age of information and beyond," *IEEE/ACM Trans. Netw.*, vol. 29, no. 5, pp. 1962–1975, 2021.

[3] J. P. Champati, M. Mamduhi, K. Johansson, and J. Gross, "Performance characterization using AoI in a single-loop networked control system," in *Proc. of IEEE INFOCOM AoI Workshop*, Apr. 2019, pp. 197–203.

[4] O. Ayan, M. Vilgelm, M. Klu¨gel, S. Hirche, and W. Kellerer, "Age-of-information vs. value-of-information scheduling for cellular networked control systems," in *Proc. of ACM/IEEE ICCPS*, Apr. 2019.

[5] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. of IEEE SECON*, Jun. 2011, pp. 350–358.

[6] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. of IEEE INFOCOM*, Mar. 2012, pp. 2731–2735.

[7] C. Kam, S. Kompella, and A. Ephremides, "Age of information under random updates," in *Proc. of IEEE International Symposium on Information Theory*, Jul. 2013, pp. 66–70.

[8] L. Huang and E. Modiano, "Optimizing age-of-information in a multi-class queueing system," in *Proc. of IEEE International Symposium on Information Theory*, Jun. 2015, pp. 1681–1685.

[9] J. P. Champati, H. Al-Zubaidy, and J. Gross, "On the distribution of AoI for the GI/GI/1/1 and GI/GI/1/2* systems: Exact expressions and bounds," in *Proc. of IEEE INFOCOM*, Apr. 2019, pp. 37–45.

[10] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Minimizing the age of information through queues," *IEEE Trans. Inf. Theory*, vol. 65, no. 8, pp. 5215–5232, 2019.

[11] A. Rizk and J.-Y. L. Boudec, "A Palm calculus approach to the distribution of the age of information," *IEEE Trans. Inf. Theory*, vol. 69, no. 12, pp. 8097–8110, Dec. 2023.

[12] R. Talak, S. Karaman, and E. Modiano, "Optimizing information freshness in wireless networks under general interference constraints," *IEEE/ACM Trans. Netw.*, vol. 28, no. 1, pp. 15–28, Feb. 2020.

[13] M. Noroozi and M. Fidler, "A min-plus model of age-of-information with worst-case and statistical bounds," in *Proc. of IEEE ICC*, 2022.

[14] A. M. Bedewy, Y. Sun, and N. B. Shroff, "The age of information in multihop networks," *IEEE/ACM Trans. Netw.*, vol. 27, no. 3, pp. 1248–1257, 2019.

[15] J. P. Champati, H. Al-Zubaidy, and J. Gross, "Statistical guarantee optimization for AoI in single-hop and two-hop FCFS systems with periodic arrivals," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 365–381, Jan. 2021.

[16] R. D. Yates, "Status updates through networks of parallel servers," in *Proc. of IEEE ISIT*, 2018.

[17] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff, "Age-optimal scheduling over hybrid channels," *IEEE Trans. Mob. Comput.*, pp. 1–17, 2022.

[18] M. Fidler, J. P. Champati, J. Widmer, and M. Noroozi, "Statistical age-of-information bounds for parallel systems: When do independent channels make a difference?" *IEEE J. Sel. Topics Inf. Theory*, vol. 4, pp. 591–606, Nov. 2023.

[19] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, pp. 1183–1210, May 2021.

[20] A. Kosta, N. Pappas, and V. Angelakis, *Age of Information: A New Concept, Metric, and Tool*. now publishers, 2017.

[21] R. D. Yates, "Lazy is timely: Status updates by an energy harvesting source," in *Proc. of IEEE ISIT*, Jun. 2015, pp. 3008–3012.

[22] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Trans. Inf. Theory*, vol. 63, no. 11, pp. 7492–7508, Nov. 2017.

[23] J. P. Champati, R. R. Avula, T. J. Oechtering, and J. Gross, "Minimum achievable peak age of information under service preemptions and request delay," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, 2021.

[24] C.-H. Tsai and C.-C. Wang, "Age-of-information revisited: Two-way delay and distribution-oblivious online algorithms," in *Proc. of IEEE ISIT*, Jun. 2020, pp. 1782–1787.

[25] ——, "Unifying AoI minimization and remote estimation – optimal sensor/controller coordination with random two-way delay," *IEEE/ACM Trans. Netw.*, vol. 30, no. 1, pp. 229–242, Feb. 2022.

[26] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff, "Optimizing sampling for data freshness: Unreliable transmissions with random two-way delay," in *Proc. of IEEE INFOCOM*, May 2022, pp. 1–10.

[27] C.-C. Wang, "How useful is delayed feedback in AoI minimization – a study of systems with queues in both forward and backward direction," in *Proc. of IEEE ISIT*, Jun. 2022, pp. 1–6.

[28] M. Moltafet, M. Leinonen, M. Codreanu, and R. D. Yates, "Status update control and analysis under two-way delay," *IEEE/ACM Trans. Netw.*, vol. 31, no. 6, pp. 2918–2933, Dec. 2023.

[29] T. Shreedhar, S. K. Kaul, and R. D. Yates, "ACP+: An age control protocol for the internet," *IEEE/ACM Trans. Netw.*, pp. 1–16, 2024.

[30] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing Networks and Markov Chains*, 2nd ed. Wiley, 2006.

[31] P. L. Dordal, *An Introduction to Computer Networks*, 2nd ed., Jul. 2023, https://intronetworks.cs.luc.edu/.