AoI-optimal Scheduling for Arbitrary K-channel Update-Through-Queue Systems

Won Jun Lee and Chih-Chun Wang
School of ECE, Purdue; Email: {lee4108,chihw}@purdue.edu

***Abstract*—This work generalizes the Age-of-Information (AoI) minimization problem of *update-through-queue* systems such that in addition to deciding the *waiting time*, the sender also chooses over which "channel" each update packet will be served. Different channels have different costs, delays, and quality characteristics that reflect the scheduler's selections of routing, communications, and update modes. Instead of considering only two channels with restricted parameters as in the existing works, this work studies the general K-channel problem with arbitrary parameters. The results show that *both the optimal waiting time and the optimal channel-selection policies admit an elegant water-filling structure*, and can be efficiently computed by the proposed low-complexity fixed-point-based numerical method.**

I. Introduction

Modern networks have ushered in numerous practical applications that require up-to-date data. To reduce data staleness, a source may transmit information to a destination as frequently as possible. At the same time, sending too many packets could congest the network and lead to outdated data. The need to holistically consider data staleness and network dynamics has led to a new performance metric, *Age-of-Information* (AoI) that directly measures data freshness at the destination [1]–[5]. One canonical model of AoI minimization problems is the *update-through-queue* system [6]–[12], where a source sends update packets to a destination through a queue (see our discussion in Sec. II.) With instantaneous delivery acknowledgement, the scheduler varies the *waiting time* to minimize the average AoI. Important variants of this setting include remote sampling [5], [10], distribution oblivious optimal adaptive solutions [13]–[16], delayed feedback [1], [9]–[11], [17], [18], and energy/cost reduction [11], [19].

This work generalizes the update-through-queue model in the following way. In addition to deciding the *waiting time*, the sender also chooses *over which "channel" each update packet will be served.* Different channels have different costs, delays, and quality characteristics that reflect the scheduler's selections of routing, communications, and update modes. For example, in Fig. 1 a scheduler has four possible options to send updates: fetching the update from a cloud server through multi-hop Wi-Fi/5G, or from an edge server through single-hop Wi-Fi/5G. Updates from the edge server may be less fresh but are faster and cheaper, whereas updates from the cloud are the freshest but incur longer delays and higher costs. Each of these four choices can be modeled as an abstract "channel" to transmit the update packets. The question to answer is: in terms of data freshness (minimizing AoI), how to dynamically schedule the four "channels" in this system.

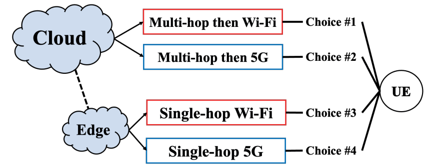


Figure 1. Cloud vs Edge, and Wi-Fi vs 5G

Existing works in this important multi-channel setting are still nascent.[1] E.g., [23] discovered an aging control policy that chooses between Wi-Fi (a low-cost, random on/off channel) and 5G (a high-cost, always-on channel). Similarly, [24] minimized the AoI while jointly considering sub-6GHz (a *deterministic-delay* channel) and mmWave channels (a Gilbert-Elliot channel). Since modern networks generally support many transmission options with diverse characteristics, e.g., Wi-Fi, 5G, Zigbee, Bluetooth, this work considers general K-channel systems with arbitrary delay distributions, cost, and quality characteristics. Our main contributions are:

(i) We prove that in a multi-channel system, the optimal waiting time follows the same water-filling structure [3], [8] as in the single-channel case.

(ii) We prove that the optimal channel selection policy is *in the reverse order of the expected delays.* That is, the optimal scheduler would choose a channel with longer *expected delay* if the current AoI is small, and would switch to a channel with shorter expected delay if the current AoI is large. "When to switch" depends on the cost differences between competing channels, which is formally characterized in Sec. III.

(iii) We further strengthen (ii) by proving that *the optimal channel selection policy also admits a water-filling structure.*

(iv) Leveraging on the findings in (i) and in (iii), we design a low-complexity fixed-point-based method that efficiently computes the optimal scheduler for arbitrary K-channel systems.

II. Problem Formulation

We model the multi-choice information update scenario in Sec. I as a single-source/single-destination update system with different *channel choices*. Each (abstract) channel represents one combination of the *origin* of the update packet, e.g., cloud vs edge, and the communication mode being used, e.g., 5G

[1]This paper focuses exclusively on the single-source system. For the multiple-source systems [20]–[22], the goal is to balance the AoI across coexisting source-destination pairs, which is very different from our goal of minimizing AoI of a single (and only) source-destination pair.

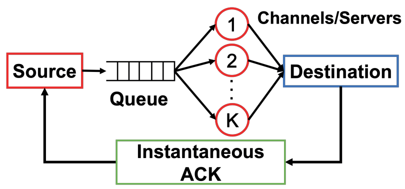


Figure 2. The K-channel Update-Through-Queue System

vs WiFi. It is single-source, since the "source" represents the central scheduler, not the different possible origins of the packets. It is single-destination since it represents a (single) user that consumes all the update information. The detailed analytical model is formally defined below.

Consider the system in Fig. 2 and assume *continuous* time axis. At any time $t \in \mathbb{R}^+$, the source can inject an information update packet P_i to a First-In First-Out (FIFO) queue. In addition to deciding when to inject an update packet, i.e., deciding the *waiting time*, the scheduler must also specify which server will be used to service the packet. To avoid confusion with the terms "cloud and edge servers" in our motivating scenario, we change the term "server" to "channel" when discussing our model. That is, all packets share a common FIFO queue, and each packet has a channel-index k in its header and will be served by CH_k when it is its turn. The choice of channels is made separately for each packet P_i.

When the service of an update packet is complete (i.e., upon delivery), the destination will feedback an instantaneous ACK. Our goal is to characterize the jointly optimal waiting time and channel selection policy that minimizes the AoI.

To provide further details, we assume there are K channels. Each channel CH_k, $k \in \mathcal{C} \triangleq \{1, 2, \cdots, K\}$ is characterized by the following three attributes that are known globally:

- **Delay distribution:** The service time (delay) of each CH_k is i.i.d. with bounded support. We denote the (marginal) distribution by $P(Y^{[k]})$. We assume the channels are indexed by the ascending order of the expected delays. Namely,

$$\mathbb{E}(Y^{[1]}) \leq \mathbb{E}(Y^{[2]}) \leq \ldots \leq \mathbb{E}(Y^{[K]}) < \infty. \tag{1}$$

- **Update quality degradation:** Recall that each "channel" represents a combination of (a) the origin of the update packets and (b) the medium of communications, see Fig. 1. For each CH_k, we use $\mathsf{Lag}_k \geq 0$ to represent the *quality degradation* of the update packets due to (a). That is, the larger the Lag_k, the more outdated the update packet is, the worst the quality. We assume the values of Lag_k is given and known. Also see the discussion around Eq. (2) for the intuition of Lag_k.
- **Transmission cost:** Since our "channels" also represent (b) the medium of communications, we use Cst_k to denote the (monetary or energy) cost of sending one packet over CH_k.

Define the send time and arrival time of the i-th packet P_i by S_i and A_i, respectively. At any time t, define $i^*(t) = \arg\max\{i : A_i \leq t\}$ as the index of the most recently delivered packet. The AoI $\Delta(t)$ is then defined by

$$\Delta(t) \triangleq t - S_{i^*(t)} + \mathsf{Lag}_{k(i^*(t))} \tag{2}$$

where $k(i^*(t))$ is the channel that delivered packet $P_{i^*(t)}$.

The best way to interpret (2) is to first assume $\mathsf{Lag}_k = 0, \forall k$. Eq. (2) is then equivalent to the traditional definition of AoI in [25], [26]. Recall that if we fetch the update from an edge server, its quality would be worse than if fetching it directly from a cloud server. As a result, if the latest packet $P_{i^*(t)}$ is from $\mathsf{CH}_{k(i^*(t))}$, we "penalize" the *effective AoI* $\Delta(t)$ by the quality degradation term $\mathsf{Lag}_{k(i^*(t))}$.

The scheduler at the source must decide the send time S_i of each update, and over which channel $k(i)$ it will be served, with the goal of solving the following minimization problem:

$$\beta^* \triangleq \inf_{\text{all policies}} \limsup_{T\to\infty} \frac{\mathbb{E}\left\{\int_0^T \Delta(t)\,dt + \sum_{i=1}^{i^*(T)} \mathsf{Cst}_{k(i)}\right\}}{T} \tag{3}$$

Limitations of our model: While our single-queue, multi-channel model is unambiguously defined, it does not fully capture the motivating scenario in Sec. I. For example, as will be shown later, the optimal policy under our model will never send a new packet before the older packet is delivered. (Existing works [23], [24] do not allow for parallel transmission either.) However, in the scenario of Fig. 1, the UE may requests parallel updates simultaneously to further improve the timeliness. Further generalization of our model is needed to fully reflect the motivating scenario of Fig. 1.

III. Main Results

A. Conversion to an ACPS-Semi-MDP Problem

We first show that (3) is a semi-Markov Decision Process (semi-MDP) with continuous state $s \in \mathbb{R}^+$, where s represents the AoI $\Delta(t)$ when making the decision at time t.

Since we use *generate-at-will* model [8], and because it is AoI-suboptimal [8] to let any update packet wait in the queue, an optimal scheduler only needs to make the decision of the send time S_i at time $t = A_{i-1}$, i.e., at the instant when the previous packet was delivered.

The action space $\mathcal{A}$ for any state s is defined by

$$\mathcal{A} \triangleq \{\,(k, w) : k \in \mathcal{C}, w \in \mathbb{R}^+\,\} \tag{4}$$

where w is the *waiting time*, i.e., the send time being $S_i = A_{i-1} + w$; and k is the channel that serves packet P_i.

If action (k, w) is chosen for state s at time $t = A_{i-1}$, then the state transition probability becomes[2]

$$p_{s\tilde{s}}^{(k,w)} \triangleq P\left(Y^{[k]} + \mathsf{Lag}_k = \tilde{s}\right) \tag{5}$$

Namely, at the delivery time $t = A_i = S_i + Y_i^{[k]}$, we have the new state $\tilde{s} = \Delta(A_i) = Y_i^{[k]} + \mathsf{Lag}_k$ by (2).

Our problem is a semi-MDP instead of a regular MDP because the *sojourn time* from state s to the new state $\tilde{s}$

[2]Because we consider continuous time axis, a more accurate but more burdensome description should be $p_{s\to d\tilde{s}}^{(k,w)} = P\left(Y^{[k]} + \mathsf{Lag}_k \in d\tilde{s}\right)$. For notational simplicity, we deliberately avoid using $d\tilde{s}$ in our formulation.

under action (k, w) is a random variable characterized by (see Chapter 5 of [27])

$$\tau(k, w) \triangleq A_i - A_{i-1} = w + Y_i^{[k]}. \tag{6}$$

For convenience, we define the expectation of $\tau(k, w)$ by

$$\overline{\tau}(k, w) \triangleq \mathbb{E}\{\tau(k, w)\} = w + \mathbb{E}(Y^{[k]}) \tag{7}$$

We now quantify the AoI+cost per action. Specifically, the cost of action (k, w) at state s is

$$c(s, k, w) \triangleq 0.5\left(\left(s + w + Y^{[k]}\right)^2 - s^2\right) + \mathsf{Cst}_k \tag{8}$$

where the first term is the AoI area due to sending P_i, see Fig. 3, and the last term Cst_k is the cost of using CH_k. Taking the expectation of $c(s, k, w)$ in (8) and simplifying it, we have

$$\overline{c}(s, k, w) \triangleq s\cdot\left(\mathbb{E}(Y^{[k]}) + w\right) + 0.5\cdot\mathbb{E}\left\{(Y^{[k]} + w)^2\right\} + \mathsf{Cst}_k \tag{9}$$

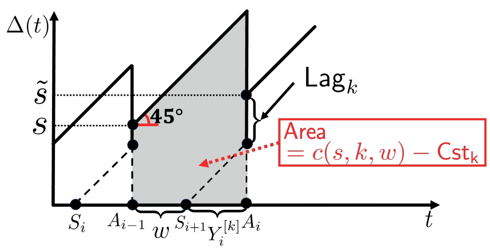


Figure 3. Evolution of The AoI

Using the above semi-MDP definitions, a scheduling policy $\pi : \mathbb{R}^+ \mapsto \mathcal{C} \times \mathbb{R}^+$, which maps the state value s to the corresponding channel and waiting-time choices (k, w), will have its average total cost per unit time being[3]

$$J^\pi = \lim_{I\to\infty} \frac{\mathbb{E}\left\{\sum_{i=0}^{I-1} \overline{c}\left(\Delta(A_i), \pi(\Delta(A_i))\right)\right\}}{\mathbb{E}\left\{\sum_{i=0}^{I-1} \overline{\tau}\left(\pi(\Delta(A_i))\right)\right\}} \tag{10}$$

The optimal π^* that minimizes J^π satisfies the following Bellman equation for all $s \in \mathbb{R}^+$ (see Chapter 5 of [27]):

$$h(s) = \min_{\substack{k\in\mathcal{C}\\ w\in\mathbb{R}^+}} \left\{\overline{c}(s, k, w) - \overline{\tau}(k, w)\cdot\beta^* + \int_{\tilde{s}=0}^{\infty} p_{s\tilde{s}}^{(k,w)} h(\tilde{s}) d\tilde{s}\right\} \tag{11}$$

where $\overline{\tau}(k, w)\cdot\beta^*$ is the adjustment term when computing the average cost per unit time.

Proposition 1. The optimal value β^*, defined in (3), can be found by solving the β^* satisfying the Bellman equation (11).

[3]Without loss of generality, we assume the initial state $s_0 = \Delta(0) = 0$.

B. Waiting Time versus Channel Selection

In this subsection, we further simplify (11). Define the expression in (11) without the min operation by

$$Q_k(s, w, \beta^*) \triangleq \overline{c}(s, k, w) - \overline{\tau}(k, w)\beta^* + \int_{\tilde{s}=0}^{\infty} p_{s\tilde{s}}^{(k,w)} h(\tilde{s}) d\tilde{s}$$

By plugging in the expressions of $\overline{c}(s, k, w)$, $\overline{\tau}(k, w)$ and $p_{s\tilde{s}}^{(k,w)}$ in (9), (7), and (5), respectively, we can easily simplify $Q_k(s, w, \beta^*)$ as follows:

$$Q_k(s, w, \beta^*) = Q_k^\circ(s, w, \beta^*) - \frac{(s - \beta^*)^2}{2} \tag{12}$$

where

$$Q_k^\circ(s, w, \beta^*) \triangleq \frac{\left((s + w) - \left(\beta^* - \mathbb{E}(Y^{[k]})\right)\right)^2}{2} + \mathcal{H}_k \tag{13}$$

and we use the *variance* of $Y^{[k]}$ to define

$$\mathcal{H}_k \triangleq \frac{\mathsf{var}(Y^{[k]})}{2} + \mathbb{E}\{h(\mathsf{Lag}_k + Y^{[k]})\} + \mathsf{Cst}_k. \tag{14}$$

Note that we can disregard $\frac{(s-\beta^*)^2}{2}$ in (12),which is independent of characterizing the optimal policy.

Proposition 2. Given any fixed k, the waiting time w^* that minimizes $Q_k^\circ(s, w, \beta^*)$ follows a water-filling structure:

$$\begin{aligned} w^* &= \max(\beta^* - \mathbb{E}(Y^{[k]}) - s, 0) && (15)\\ &= \max\{w \geq 0 : w + s + \mathbb{E}(Y^{[k]}) \geq \beta^*\}. && (16)\end{aligned}$$

Furthermore, if we plot $\min_{w\geq 0} Q_k^\circ(s, w, \beta^*)$ as a function of s, see Fig. 4, it consists of two halves: the left-hand side of the vertex ($s = \beta^* - \mathbb{E}(Y^{[k]})$) is a flat line while the right-hand side being identical to the quadratic curve $Q_k^\circ(s, 0, \beta^*)$.

Namely, the optimization over $w \geq 0$ essentially "bends" the left-hand side of the quadratic curve $Q_k^\circ(s, 0, \beta^*)$ downwards to a flat line. See [?] for additional details.

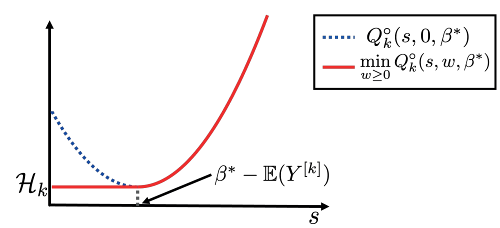


Figure 4. $Q_k^\circ(s, 0, \beta^*)$ versus $\min_{w\geq 0} Q_k^\circ(s, w, \beta^*)$

Because the optimal waiting time in (15)–(16) for our multi-channel setting is of an identical form as the single-channel case [3], [28], [29], it implies that ***an overall optimal policy can be done sequentially: Firstly, we choose the optimal*** CH_k***, and then we choose the optimal waiting time of*** P_i ***using*** (16) ***as if we are in a single-channel scenario.***

We now describe the optimal channel selection for the $K = 2$ case. Specifically, Fig. 5 plots two curves defined as follows:

$$Q_k^\star(s, \beta^*) \triangleq \min_{w\geq 0} Q_k^\circ(s, w, \beta^*), \quad \forall k \in \{1, 2\} \tag{17}$$

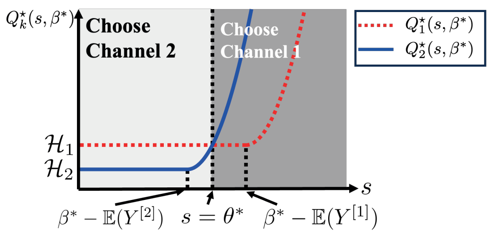


Figure 5. Optimal channel selection for the $K = 2$ channel system

Following the discussion right after Eq. (14), the optimal policy will solve $\min_{(k,w)} Q_k^\circ(s,w,\beta^*) = \min_k Q_k^\star(s,\beta^*)$, which can be solved by analyzing the relative positions between $Q_1^\star(s,\beta^*)$ and $Q_2^\star(s,\beta^*)$. Specifically, there is a threshold θ^*, see Fig. 5, such that we choose $k = 2$ if $s < \theta^*$ and choose $k = 1$ if $s \geq \theta^*$. The θ^* is the intersecting point of the two curves $Q_1^\star(s,\beta^*)$ and $Q_2^\star(s,\beta^*)$, assuming we know the values of $\beta^*, \mathbb{E}(Y^{[k]})$, and $\mathcal{H}_k$.

We now generalize the above discussion for arbitrary K:

Proposition 3. The optimal channel selection policy is of the following *water-filling* structure. Each CH_k has a *water-level value* γ_k where $\gamma_1 \leq \gamma_2 \leq \cdots \leq \gamma_K = \infty$ and each value can be positive or negative. The corresponding channel selection rule under a given state s is described by

$$k^*(s) = \min\{k \in \{1,\cdots,K\} : \gamma_k + s \geq \beta^*\}. \tag{18}$$

The proof is derived by jointly comparing the relative positions of all K curves $Q_k^\star(s,\beta^*)$. We thus omit the details.

A few remarks are in order. Firstly, (18) has the same water filling structure as in (16). Secondly, the selection rule (18) is *monotonic*, i.e., when s is small, we would select the channel with a larger k (since γ_k is non-decreasing), and vice versa Finally, Proposition 3 implies implicitly that if $\gamma_k = \gamma_{k+1}$, then, by definition, the selection rule (18) will *never* select CH_{k+1}. It means that CH_{k+1} is "dominated" by other channels, a phenomenon frequently encountered in our simulations.

Note that given β^* and $\mathbb{E}(Y^{[k]})$, the optimal $w^*(s)$ is fully described by (16), but $k^*(s)$ in (18) still depends on the values of γ_k. We now describe how to compute γ_k assuming we know the values of β^* and $\mathcal{H}_k$, the latter of which can easily computed via (14) if we know the function $h(s)$ and the distribution of $Y^{[k]}$. Once we have computed γ_k, the description of the optimal channel selection policy is complete.

Lemma 1. Consider any given two channels CH_i and CH_j. We then have two cases.

Case 1: $\mathbb{E}(Y^{[i]}) = \mathbb{E}(Y^{[j]})$. If $\mathcal{H}_i = \mathcal{H}_j$, then the two curves $Q_i^\star(s,\beta^*)$ and $Q_j^\star(s,\beta^*)$ are identical, see (13). If $\mathcal{H}_i \neq \mathcal{H}_j$, then $Q_i^\star(s,\beta^*)$ and $Q_j^\star(s,\beta^*)$ are *parallel*, and do not intersect.

Case 2: $\mathbb{E}(Y^{[i]}) < \mathbb{E}(Y^{[j]})$. We have three subcases. **Case 2.1:** If $\mathcal{H}_i < \mathcal{H}_j$, then $Q_i^\star(s,\beta^*)$ and $Q_j^\star(s,\beta^*)$ have no intersecting point; **Case 2.2:** If $\mathcal{H}_i = \mathcal{H}_j$, then the two curves fully overlap for the range of $s \leq \beta^* - \mathbb{E}(Y^{[j]})$; **Case 2.3:** If $\mathcal{H}_i > \mathcal{H}_j$, then the two curves have exactly one intersecting point at $s = \theta_{i,j}$ regardless of the β^* value. The intersecting $s = \theta_{i,j}$ value can be expressed by

$$\theta_{i,j} = \beta^* - f_\gamma\left(\mathbb{E}(Y^{[i]}), \mathbb{E}(Y^{[j]}), \mathcal{H}_i - \mathcal{H}_j\right) \tag{19}$$

and the description of $f_\gamma(\cdot,\cdot,\cdot)$ is provided in Appendix C.

Using Lemma 1, Algorithm 1 computes the values of $\gamma_k, \forall k \in \mathcal{C}$. We omit the proof due to space constraints.

Algorithm 1 Computing the water-level values γ_k

Require: $\mathbb{E}(Y^{[k]}), \mathcal{H}_k$, $\forall k \in \mathcal{C}$, and the $f_\gamma(\cdot,\cdot,\cdot)$ in (19).

1: $\mathcal{P}$ is a set of *ordered pairs*; initialize $\mathcal{P} \leftarrow \emptyset$.
2: **for** all $i, j \in \mathcal{C}$ and $i < j$ **do**
3:     Consider the two curves $Q_i^\star(s,\beta^*)$ and $Q_j^\star(s,\beta^*)$.
4:     **if** they have exactly one intersecting point **then**

$$\tilde{\gamma}_{i,j} \leftarrow f_\gamma\left(\mathbb{E}(Y^{[i]}), \mathbb{E}(Y^{[j]}), \mathcal{H}_i - \mathcal{H}_j\right); \quad \mathcal{P} \leftarrow \mathcal{P} \cup \{(i,j)\}$$

5:     **end if**
6: **end for**
7: $j_0 \leftarrow \arg\min_{k \in \mathcal{C}} \mathcal{H}_k$.
8: $\gamma_k \leftarrow -\infty, \forall k \in (0, j_0)$; and $\gamma_k \leftarrow \infty, \forall k \in [j_0, K]$.
9: **while** $j_0 \geq 2$ **do**
10:     $i_0 \leftarrow \arg\max_{(i,j_0) \in \mathcal{P}} \tilde{\gamma}_{i,j_0}$
11:     **if** $i_0 \geq 1$ **then**
12:         $\gamma_k \leftarrow \tilde{\gamma}_{i_0,j_0}, \forall k \in [i_0, j_0)$.
13:     **end if**
14:     $j_0 \leftarrow i_0$.
15: **end while**

Note 1: If there a tie in Line 7, choose the smallest such j_0. If there is a tie in Line 10, choose the smallest such i_0.
Note 2: In Line 10, if no $(i, j_0) \in \mathcal{P}$, then $i_0 \leftarrow -\infty$.

C. Numerical Computation via Fixed-Point Iteration

This subsection discusses how to compute the optimal β^* and the value function $h(s)$, which can then be used to compute $w^*(s)$ using (15), compute $\mathcal{H}_k$ using (14), compute γ_k using Algorithm 1, and compute $k^*(s)$ using (18). Our method is very efficient since it utilizes the optimal water-filling structures during the fixed-point iteration computation.

Lemma 2. When solving the Bellman equation (11), we only need to consider a bounded range of $s \in [0, y_{\max} + \max_k \mathsf{Lag}_k]$ instead of the unbounded range of $s \in \mathbb{R}^+$.

The proof is provided in [?]. By Lemma 2, we quantize the interval $[0, y_{\max} + \max_k \mathsf{Lag}_k]$ with N grid points

$$\mathcal{S}_N \triangleq \left\{ n \cdot \frac{y_{\max} + \max_k \mathsf{Lag}_k}{N} : n \in \{0, 1, \cdots N-1\} \right\} \tag{20}$$

for some sufficiently large N. Namely, we only solve the β^* and the $h(s)$ values for a finite number of $s \in \mathcal{S}_N$. The functions $\bar{c}(s,k,w)$ and $\bar{\tau}(k,w)$ do not change during quantization and are still specified by (9) and (7), respectively. However, $p_{s\tilde{s}}^{(k,w)}$ in (5) will change slightly since the next state

Table I
SIMULATION CHANNEL PARAMETERS

	Example	Distribution $P(Y^{[k]})$	Lag_i	Cst_i
CH_1	**Edge+5G**	uniform over $\{1, 2, 4, 5\}$ $\mathbb{E}(Y^{[1]}) = 3$	5	150
CH_2	**Cloud+5G**	uniform over $\{1, 4, 8, 11\}$ $\mathbb{E}(Y^{[2]}) = 6$	0	100
CH_3	**Edge+WiFi**	uniform over $\{1, 7, 9, 15\}$ $\mathbb{E}(Y^{[3]}) = 8$	5	50
CH_4	**Cloud+WiFi**	uniform over $\{1, 11, 13, 23\}$ $\mathbb{E}(Y^{[4]}) = 12$	0	0

$\tilde{s}$ is also quantized to grid points in $\mathcal{S}_N$. We thus need to reassign the probability from the continuous pdf $P(Y^{[k]} \in dy)$ to their discrete pmf counterpart. Such a reassignment is standard when quantizing any continuous random variable.

We then solve the quantized Bellman equation (11) over $s \in \mathcal{S}_N$ in an iterative fashion: We initialize $\beta^{(0)} = 0$ and $h^{(0)}(s) = 0, \forall s \in \mathcal{S}_N$. Then for $l \geq 1$, we do the following.

Step 1: Use $h^{(l-1)}(s)$ to compute $\mathcal{H}_k^{(l-1)}$ using (14) and the (quantized) distribution of $P(Y^{[k]})$.

Step 2: Use $\beta^{(l-1)}$ and $\mathcal{H}_k^{(l-1)}$ to compute the optimal channel selection rule $k^{(l)}(s)$ using (18) and Algorithm 1. Use $\beta^{(l-1)}$ and $\mathbb{E}(Y^{[k]})$ to compute the optimal waiting time $w_k^{(l)}(s)$ for each CH_k. Collectively, $k^{(l)}(s)$ and $w_k^{(l)}(s)$ form a scheduling policy, which we denote as $\pi^{(l)}$.

Step 3: Replace the $\min_{(k,w)}$ operator in (11) by the policy $\pi^{(l)}$. Namely, $\forall s \in \mathcal{S}_N$, we have a simple linear equation.

$$h(s) = \bar{c}(s, k(s), w(s)) - \bar{\tau}(k(s), w(s))\beta + \sum_{\tilde{s} \in \mathcal{S}_N} p_{s\tilde{s}}^{(k(s),w(s))} h(\tilde{s}) \tag{21}$$

where $k(s)$ and $w(s)$ are the action choices under policy $\pi^{(l)}$. We then solve the β and $h(s)$ values that satisfy (21) for all $s \in \mathcal{S}_N$ while hardwiring $h(0) = 0$ when finding the solution. We denote the end results by $\beta^{(l)}$ and $h^{(l)}(s)$, respectively.

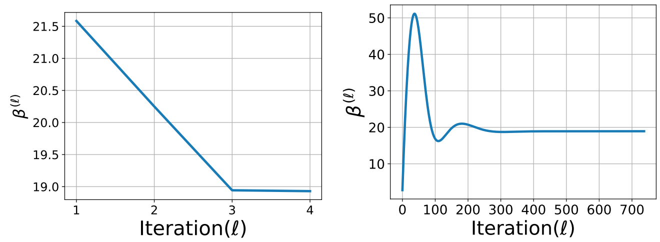
Step 4: Repeat Steps 1 to 3 until $\beta^{(l)}$ and $h^{(l)}(s)$ converge.

Lemma 3. In the above 4-step process, the resulting $\beta^{(l)}$ is a non-increasing function when $l \geq 1$ (excluding $l = 0$), and $\lim_{l \to \infty} \beta^{(l)} = \beta^*$.

IV. NUMERICAL EVALUATIONS

Consider a 4-ch system described in Table I. The example scenario is related to our discussion of Fig. 1, i.e., an update from the edge server will experience an AoI degradation $\mathsf{Lag} = 5$. We assume 5G and single-hop having shorter delay than Wi-Fi and multihop communications. We assign Cst_k in the reverse order of $\mathbb{E}(Y^{[k]})$ to avoid the less interesting cases that one channel is dominated by other channels.

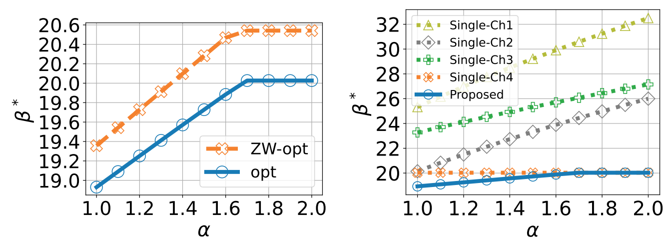
Fig. 6(a) tracks the convergence of $\beta^{(l)}$ in our proposed method. In Fig. 6(b), we use a generic Q-function value iteration [30] to solve the semi-MDP in (11). While convergence is guaranteed for both methods, both outputting the same $\beta^* = 18.93$, the Q-value iteration took hundreds



(a) Proposed Iterative Algorithm (b) Q Value Iteration [30]

Figure 6. Convergence behavior of $\beta^{(l)}$

of iterations to converge. In contrast, our low-complexity algorithm converged in just a few iterations ($l \approx 4$).



(a) Versus Zero-Wait Policy (b) Versus Single-Channel Policy

Figure 7. Achievable AoI compared other non-trivial solutions.

We also compare with some other non-trivial solutions. Specifically, we designed separately a policy that hardwires $w = 0$, i.e., zero-wait, but optimally switches between different CH_k depending on the state $s = \Delta(t)$. We also designed "dumb" deterministic channel selection policies (say always choose CH_k) but with optimal waiting time w^* [8], [9]. We call them "single-ch$k$ policies". In Figs. 7(a) and 7(b), we multiply the cost Cst_k in Table I of each channel by a common factor $\alpha \in [1, 2]$. We then plot the optimal β^* versus different α. Fig. 7(a) reaffirms that varying the waiting time is crucial to achieve β^* since our scheme significantly outperforms ZW-opt, which optimizes only channel selection but not waiting time. The same setup is repeated in Fig. 7(b) but this time we focus on single-ch$k$ policies. By dynamically utilizing the best of the 4 channels for different state s, our optimal scheme consistently outperforms any single-ch$k$ policy. Note that when $\alpha \geq 1.7$, single-ch4 becomes optimal as all other channels becomes too costly with the new cost $\alpha\mathsf{Cst}_k$. In sum, our algorithm takes full advantage the heterogeneity of K channels and employs only the best channel and best waiting time at any given state s, the key to its optimal performance.

V. CONCLUSION

We have studied AoI minimization for heterogeneous K-channel systems and fully characterized the optimal scheduler. New *water-filling* structures of optimal policies and efficient computation methods have been discovered. This work was supported in parts by NSF Grants CNS-2008527, CNS-2107363, CCF-2309887 and also by under grant National Spectrum Consortium (NSC) W15QKN-15-9-1004.

References

[1] C.-C. Wang, "How useful is delayed feedback in AoI minimization — a study on systems with queues in both forward and backward directions," in *2022 IEEE International Symposium on Information Theory (ISIT)*, (Espoo, Finland), pp. 3192–3197, IEEE, Jun. 2022.

[2] J. Zhang and C.-C. Wang, "On the rate-cost of gaussian linear control systems with random communication delays," in *2018 IEEE International Symposium on Information Theory (ISIT)*, (Vail, CO, USA), p. 2441–2445, IEEE Press, Jun. 2018.

[3] Y. Sun and B. Cyr, "Sampling for data freshness optimization: Nonlinear age functions," *Journal of Communications and Networks*, vol. 21, no. 3, pp. 204–219, 2019.

[4] T. Z. Ornee and Y. Sun, "Sampling and remote estimation for the ornstein-uhlenbeck process through queues: Age of information and beyond," *IEEE/ACM Transactions on Networking*, vol. 29, pp. 1962–1975, Oct. 2021.

[5] Y. Sun, Y. Polyanskiy, and E. Uysal, "Sampling of the wiener process for remote estimation over a channel with random delay," *IEEE Transactions on Information Theory*, vol. 66, no. 2, pp. 1118–1135, 2020.

[6] L. Huang and E. H. Modiano, "Optimizing age-of-information in a multi-class queueing system," *2015 IEEE International Symposium on Information Theory (ISIT)*, pp. 1681–1685, Apr. 2015.

[7] V. Tripathi, R. Talak, and E. H. Modiano, "Age of information for discrete time queues," Jan. 2019.

[8] 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 Transactions on Information Theory*, vol. 63, pp. 7492–7508, Nov. 2017.

[9] C.-H. Tsai and C.-C. Wang, "Age-of-information revisited: Two-way delay and distribution-oblivious online algorithm," in *2020 IEEE International Symposium on Information Theory (ISIT)*, (Los Angeles, CA, USA), pp. 1782–1787, 2020.

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

[11] C.-H. Tsai and C.-C. Wang, "Jointly minimizing AoI penalty and network cost among coexisting source-destination pairs," in *2021 IEEE International Symposium on Information Theory (ISIT)*, (Melbourne, Australia), pp. 3255–3260, 2021.

[12] R. D. Yates and S. K. Kaul, "The age of information: Real-time status updating by multiple sources," *IEEE Trans. Inform. Theory*, vol. 65, no. 3, pp. 1807–1827, 2019.

[13] C.-H. Tsai and C.-C. Wang, "Distribution-oblivious online algorithms for age-of-information penalty minimization," *IEEE/ACM Transactions on Networking*, vol. 31, no. 4, pp. 1779–1794, Aug. 2023.

[14] C. Kam, S. Kompella, and A. Ephremides, "Learning to sample a signal through an unknown system for minimum AoI," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, (Paris, France), pp. 177–182, IEEE, 2019.

[15] E. U. Atay, I. Kadota, and E. H. Modiano, "Aging bandits: Regret analysis and order-optimal learning algorithm for wireless networks with stochastic arrivals," Dec. 2020.

[16] H. Tang, Y. Chen, J. Sun, J. Wang, and J. Song, "Sending timely status updates through channel with random delay via online learning," in *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications*, (London, United Kingdom), p. 1819–1827, IEEE Press, May. 2022.

[17] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff, "Optimal sampling for data freshness: Unreliable transmissions with random two-way delay," *IEEE/ACM Trans. Networking*, vol. 31, no. 1, pp. 408–420, Feb. 2023.

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

[19] G. Yao, C.-C. Wang, and N. B. Shroff, "Age minimization with energy and distortion constraints," in *In Proc. of the Twenty-fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing (MobiHoc '23).*, MobiHoc '23, (Washington, DC, USA), p. 101–110, Association for Computing Machinery, 2023.

[20] Y. Zou, K. T. Kim, X. Lin, and M. Chiang, "Minimizing age-of-information in heterogeneous multi-channel systems: A new partial-index approach," in *Proceedings of the Twenty-second International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, (Shanghai China), pp. 11–20, ACM, Jul. 2021.

[21] Y. Chen, J. Wang, X. Wang, and J. Song, "Age of information optimization in multi-channel network with sided information," *IEEE Communications Letters*, vol. 27, pp. 1030–1034, Jun. 2023.

[22] G. Chen, Y. Chen, J. Wang, and J. Song, "Scheduling to minimize age of synchronization in multi-channel time-sensitive networks," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, (Austin, TX, USA), pp. 1605–1610, IEEE, Apr. 2022.

[23] E. Altman, R. El-Azouzi, D. S. Menasche, and Y. Xu, "Forever young: Aging control for smartphones in hybrid networks," in *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 91–100, 2019.

[24] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff, "Age-optimal scheduling over hybrid channels," *IEEE Transactions on Mobile Computing*, vol. 22, no. 12, pp. 7027–7043, 2023.

[25] X. Song and J. Liu, "Performance of multiversion concurrency control algorithms in maintaining temporal consistency," in *Proceedings., Fourteenth Annual International Computer Software and Applications Conference*, (Chicago, IL, USA), pp. 132–139, IEEE Comput. Soc. Press, 1990.

[26] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *IEEE INFOCOM 2012 - IEEE Conference on Computer Communications*, (Orlando, FL, USA), pp. 2731–2735, IEEE, 2012.

[27] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena scientific optimization and computation series, Nashua, NH: Athena scientific, 4th edition ed., 2012.

[28] R. D. Yates, "Lazy is timely: Status updates by an energy harvesting source," in *2015 IEEE International Symposium on Information Theory (ISIT)*, (Hong Kong, Hong Kong), pp. 3008–3012, IEEE, June 2015.

[29] A. Arafa, J. Yang, S. Ulukus, and H. V. Poor, "Age-minimal transmission for energy harvesting sensors with finite batteries: Online policies," *IEEE Trans. Inform. Theory*, vol. 66, no. 1, pp. 534–556, Jan. 2020.

[30] A. Gosavi and V. K. Le, "Maintenance optimization in a digital twin for industry 4.0," *Annals of Operations Research, Springer*, pp. 1–25, 2022.

APPENDIX A
PROOF OF PROPOSITION 2

If the state $s = \Delta(t)$ satisfies $s < \beta^* - E(Y^{[k]})$, i.e., s is on the left-hand side of the vertex of the quadratic curve $Q_k^\circ(s, 0, \beta^*)$, see Fig. 4, then increase s value by a small $w > 0$ will lead to $Q_k^\circ(s + w, 0, \beta^*) = Q_k^\circ(s, w, \beta^*) < Q_k^\circ(s, 0, \beta^*)$, where the equality is by (13). By the same reasoning but going one step deeper, the best strategy under the starting state s is to set the waiting time $w^* = \beta^* - \mathbb{E}(Y^{[k]}) - s$ so that after waiting for w^* time the new $s' = s + w^*$ will "hit" the vertex, the lowest point of the quadratic curve $Q_k^\circ(s, 0, \beta^*)$.

On the other hand, if $s \geq \beta^* - \mathbb{E}(Y^{[k]})$, i.e., s is on the right-hand side of the vertex, we then set the waiting time $w^* = 0$ since we have already passed the vertex and any additional waiting time $w > 0$ will increase the cost: $Q_k^\circ(s + w, 0, \beta^*) = Q_k^\circ(s, w, \beta^*) > Q_k^\circ(s, 0, \beta^*)$.

If we plot $\min_{w \geq 0} Q_k^\circ(s, w, \beta^*)$ versus s, see Fig. 4, the new curve is a flat line on the left-hand side of the vertex but a quadratic curve on the right-hand side of vertex.

APPENDIX B
PROOF OF PROPOSITION 2 (CONVEX OPTIMIZATION VER.)

The optimization problem of Proposition 2 is as follows.

$$\begin{aligned} \underset{w}{min} \quad & Q_k^\circ(s, w, \beta^*) \\ \text{s.t.} \quad & w \geq 0 \end{aligned} \tag{22}$$

Since $Q_k^\circ(s, w, \beta^*)$ is convex over s and the constraint w is a linear function, the optmization problem is a convex optimization problem. A Lagrangian of the optimization problem can be defined as

$$\mathcal{L}(w, \lambda) = Q_k^\circ(s, w, \beta^*) - \lambda w \tag{23}$$

A dual function is defined by

$$g(\lambda) \triangleq \inf_w \mathcal{L}(w, \lambda) \tag{24}$$

where dual problem is formulated as

$$\begin{aligned} \underset{\lambda}{max} \quad & g(\lambda) \\ \text{s.t.} \quad & \lambda \geq 0 \end{aligned} \tag{25}$$

Given that

$$\frac{\partial}{\partial w}\mathcal{L}(w, \lambda) = \Big((s + w) - \Big(\beta^* - \mathbb{E}(Y^{[k]})\Big)\Big) - \lambda \tag{26}$$

It follows that

$$w^* = \max\Big(\Big(\beta^* - \mathbb{E}(Y^{[k]})\Big) - s + \lambda, 0\Big) \tag{27}$$

Thus, we have

$$\begin{aligned} g(\lambda) &= \mathcal{L}(w^*, \lambda) && (28) \\ &= \frac{\big((s + w^*) - \big(\beta^* - \mathbb{E}(Y^{[k]})\big)\big)^2}{2} + \mathcal{H}_k && (29) \\ &= \begin{cases} -\frac{1}{2}\lambda^2 + (s - \theta_k)\lambda + \mathcal{H}_k & \lambda > s - \theta_k \\ \frac{1}{2}(s - \theta_k)^2 + \mathcal{H}_k & \lambda \leq s - \theta_k \end{cases} && (30) \end{aligned}$$

where $\theta_k = \beta^* - \mathbb{E}(Y^{[k]})$. The solution λ^* such that satisfies $\frac{\partial g}{\partial \lambda}(\lambda^*) = 0$ is

$$\lambda^* = (s - \theta_k)^+ \tag{31}$$

Thus, the solution to the dual problem is

$$g(\lambda^*) = \begin{cases} \frac{1}{2}(s - \theta_k)^2 + \mathcal{H}_k & s > \theta_k \\ \mathcal{H}_k & s \leq \theta_k \end{cases} \tag{32}$$

Since the Karush-Kuhn-Tucker (KKT) conditions are satisfied, strong duality holds. Therefore,

$$\begin{aligned} Q_k^\star(s, \beta^*) &= \min_{w \geq 0} Q_k^\circ(s, w, \beta^*) && (33) \\ &= g(\lambda^*) && (34) \\ &= \begin{cases} \frac{1}{2}(s - \theta_k)^2 + \mathcal{H}_k & s > \theta_k \\ \mathcal{H}_k & s \leq \theta_k \end{cases} && (35) \end{aligned}$$

Therefore, due to the complementary slackness, the optimal waiting time w^* is given as

$$\begin{aligned} w^* &= \max(\beta^* - \mathbb{E}(Y^{[k]}) - s, 0) && (36) \\ &= \max\{w \geq 0 : w + s + \mathbb{E}(Y^{[k]}) \geq \beta^*\}. && (37) \end{aligned}$$

APPENDIX C
PROOF SKETH OF LEMMA 1

We get a closed form of $f_\gamma(\mathbb{E}(Y^{[i]}), \mathbb{E}(Y^{[j]}), \mathcal{H}_i - \mathcal{H}_j)$ as below.

$$\begin{aligned} & f_\gamma(\mathbb{E}(Y^{[i]}), \mathbb{E}(Y^{[j]}), \mathcal{H}_i - \mathcal{H}_j) \\ &= \begin{cases} \gamma_A & \frac{(\mathbb{E}(Y^{[j]}) - \mathbb{E}(Y^{[i]}))^2}{2} < \mathcal{H}_i - \mathcal{H}_j \\ \gamma_B & \frac{(\mathbb{E}(Y^{[i]}) - \mathbb{E}(Y^{[j]}))^2}{2} \geq \mathcal{H}_i - \mathcal{H}_j \end{cases} \end{aligned} \tag{38}$$

where

$$\gamma_A = \frac{\mathcal{H}_i - \mathcal{H}_j}{\mathbb{E}(Y^{[i]}) - \mathbb{E}(Y^{[j]})} + 0.5(\mathbb{E}(Y^{[i]}) + \mathbb{E}(Y^{[j]})) \tag{39}$$

$$\gamma_B = \mathbb{E}(Y^{[j]}) - \sqrt{2(\mathcal{H}_i - \mathcal{H}_j)} \tag{40}$$

Proof. When $\mathcal{H}_i > \mathcal{H}_j$, we exactly have one intersection of $Q_i^\star(s, \beta^*)$ and $Q_j^\star(s, \beta^*)$. Let the intersection of two function be (s^*, Q^*). Then , there are two cases: (i) $Q^* > \mathcal{H}_i$, which is equivalent to $\frac{(\mathbb{E}(Y^{[j]}) - \mathbb{E}(Y^{[i]}))^2}{2} < \mathcal{H}_i - \mathcal{H}_j$. In this case, the intersection lies on the right hand side of both $Q_i^\star(s, \beta^*)$ and $Q_j^\star(s, \beta^*)$. Thus, by solving an equation of two quadratic functions, one can easily get $s^* = \beta^* - \gamma_A$. (ii) $Q^* = \mathcal{H}_i$, which is equivalent to $\frac{(\mathbb{E}(Y^{[j]}) - \mathbb{E}(Y^{[i]}))^2}{2} \geq \mathcal{H}_i - \mathcal{H}_j$. In this case, the intersection lies on the left hand side of $Q_i^\star(s, \beta^*)$ and the right hand side of $Q_i^\star(s, \beta^*)$. We can get $s^* = \beta^* - \gamma_B$ by calculating an equation of one quadratic function and one constant function. $\square$

Appendix D
Proof Sektch of Lemma 2

By the transition probability discussion in (11), the (random) state $\tilde{s}$ that can be reached by any arbitrary action choice (k, w) and any arbitrary starting state s is less than $y_{\max} + \max_k \mathsf{Lag}_k$ with probability one. Therefore, the right-hand side of the Bellman equation only uses $h(\tilde{s})$ for those $\tilde{s} \in [0, y_{\max} + \max_k \mathsf{Lag}_k]$. As a result, any $h(s)$ with $s > y_{\max} + \max_k \mathsf{Lag}_k$ only appears in the left-hand-side of (11), which does not impose any "constraint" when solving the Bellman equation and can thus be ignored during numerical computation.

Appendix E
Proof sketch of the correctness of Algorithm 1

By the definition of j_0, for all $i > j_0$, we have $Q^\star_i(s, \beta^*) \geq Q^\star_{j_0}(s, \beta^*)$. Thus, line 8 $\gamma_k \leftarrow \infty$ for $k \in [j_0, K]$ is justifiable. Also by the definition of j_0, for all $i < j_0$, $Q^\star_i(s, \beta^*) \geq Q^\star_{j_0}(s, \beta^*)$ during the range of $s \leq \beta^* - \mathbb{E}(Y^{[j_0]})$. For any (i, j) such that $\tilde{\gamma}_{i,j}$ is uniquely defined by Line 4, define $\theta_{i,j} = \beta^* - \tilde{\gamma}_{i,j}$. We now run the for loop until and including Line 10 for the very first time and thus finished computing the $\gamma_{\tilde{i_0}, j_0}$ value for the very first time. For i_0 such that $i_0 = -\infty$, then we must have $Q^\star_{i_0}(s, \beta^*) \geq Q^\star_{j_0}(s, \beta^*)$ for all $i_0 < j_0$ during the range of $s > \beta^* - \mathbb{E}(Y^{[j_0]})$. Thus, if $i_0 = -\infty$, $Q^\star_{j_0}(s, \beta^*)$ is the lower envelope of $Q^\star_{i_0}(s, \beta^*)$ for all $i_0 \in \mathcal{C}$. Since the while loop stops in this case. Our algorithm is correct.

If $i_0 \geq 1$, then we must have $Q^\star_{i_0}(s, \beta^*) \geq Q^\star_{j_0}(s, \beta^*)$ during the range of $s \in (\beta^* - \mathbb{E}(Y^{[j_0]}), \theta^+_{i_0,j_0}]$. Here $\theta^+_{i_0,j_0}$ indicates a value $\theta_{i_0,j_0} + \delta$ for some sufficiently small but strictly positive $\delta > 0$. Furthermore, $Q^\star_{i_0}(s, \beta^*)$ is the lower envelope during the range $s \in [\theta_{i_0,j_0}, \theta^+_{i_0,j_0}]$. The above discussion shows that for the very first (i_0, j_0), we have characterized the lower-envelop for the range of $s \leq \theta^+_{i_0,j_0}$ after finishing Line 12 of the while loop. We now use mathematical induction.

Hypothesis: Suppose we have successfully characterized the lower envelop for the range of $s \leq \theta^+_{i_0,j_0}$ for some pair of (i_0, j_0), not necessarily the first one. And also assume that $Q^\star_{i_0}(s, \beta^*)$ is the lower[4] envelope during the range $s \in [\theta_{i_0,j_0}, \theta^+_{i_0,j_0}]$ for that particular (i_0, j_0), not necessarily the first pair. We now like to prove that *after one iteration of the while loop, the induction hypothesis still holds with the new* $(\tilde{i}_0, \tilde{j}_0)$.

Induction: If $i_0 = 1$, then the while loop stops in the next iteration since $\tilde{j}_0 = i_0 = 1$. Because no further γ_k is assigned, the our algorithm will assume the lower envelop will extend from $\theta^+_{i_0,j_0}$ to ∞. This assumption turns out to be correct since $Q^\star_1(s, \beta^*)$ is the lower envelop during $[\theta_{1,j_0}, \theta^+_{1,j_0}]$ implies that it is also the envelope during $[\theta_{1,j_0}, \infty)$

If $i_0 \geq 2$, then we have $\tilde{j}_0 = i_0 \geq 2$. Consider two cases: **Case 1:** If $\tilde{i}_0 = -\infty$ and $\theta_{\tilde{j}_0,i}$ exists for some $i > \tilde{j}_0 = i_0$, then $\theta_{\tilde{j}_0,i} \leq \theta_{i_0,j_0}$. Suppose there exists an $i' > i_0$ such that $\theta_{i_0,i'} > \theta_{i_0,j_0}$. This contradicts that $Q^\star_{i_0}(s, \beta^*)$ is the lower envelope in the range of $s \in [\theta_{i_0,j_0}, \theta^+_{i_0,j_0}]$. Thus, in the case of $\tilde{i}_0 = -\infty$, the curve $Q^\star_{i_0}(s, \beta^*)$ is also the lower envelope in the range of $s \in [\theta_{i_0,j_0}, \infty)$. Since Algorithm 1 stops in this case, it again correctly characterizes the entire lower envelope.
Case 2: $\tilde{i}_0 \geq 1$. If $\tilde{i}_0 \geq 1$, then $\theta_{\tilde{i}_0,\tilde{j}_0} > \theta_{i_0,j_0}$. Suppose there exists an $i' < i_0$ such that $\theta_{i',i_0} \leq \theta_{i_0,j_0}$. This contradicts that $Q^\star_{\tilde{i}_0}(s, \beta^*)$ is the lower envelope in the range of $s \in [\theta_{i_0,j_0}, \theta^+_{i_0,j_0}]$. Also, I claim if $\tilde{i}_0 \geq 1$ and $\theta_{\tilde{j}_0,i}$ exists for some $i > \tilde{j}_0 = i_0$, then $\theta_{\tilde{j}_0,i} \leq \theta_{i_0,j_0}$. Suppose there exists an $i' > i_0$ such that $\theta_{i_0,i'} > \theta_{i_0,j_0}$. This contradicts that $Q^\star_{i_0}(s, \beta^*)$ is the lower envelope in the range of $s \in [\theta_{i_0,j_0}, \theta^+_{i_0,j_0}]$. Therefore, we have that if $\tilde{i}_0 \geq 1$, then $Q^\star_{i_0}(s, \beta^*)$ is the lower envelope in the range of $s \in [\theta_{i_0,j_0}, \theta_{\tilde{i}_0,\tilde{j}_0}]$ and $Q^\star_{\tilde{i}_0}(s, \beta^*)$ is the lower envelope in the range of $s \in [\theta_{\tilde{i}_0,\tilde{j}_0}, \theta^+_{\tilde{i}_0,\tilde{j}_0}]$. By induction, the proof is complete.

[4]If two curves overlap, the lower envelope is defined as the unique one with the smallest channel index. Therefore, there is no tie in our definition.