Quality of Information Aware Scheduling in Task Processing Networks

Rahul Urgaonkar*, Ertugrul Necdet Ciftcioglu†, Aylin Yener†, and Michael J. Neely*

*Dept. of Electrical Engineering   †Dept. of Electrical Engineering

University of Southern California   The Pennsylvania State University

Los Angeles, CA 90089   University Park, PA 16802

{urgaonka, mjneely}@usc.edu   enc118@psu.edu, yener@ee.psu.edu

Abstract—We investigate *Quality of Information* (QoI) aware scheduling in task processing networks. Specifically, we consider the scenario where a network sequentially receives tasks from an end user, utilizes its resources to process them, and sends back its response. The utility derived by the end user from this response depends on both the *accuracy* and the *freshness* of the information. There is often a trade-off between these two attributes and we present a model that quantifies this dependence. Using dynamic programming and optimal stopping theory, we characterize the optimal scheduling policy that maximizes the time average utility delivered by the network. We show that for many scenarios of practical interest, the optimal policy has a simple threshold structure. We also propose a method to approximately compute the threshold in closed-form. This work takes a step towards incorporating application aware objectives in making optimal scheduling decisions.

Index Terms—Quality of Information, Scheduling, Optimal Stopping, Dynamic Programming

I. INTRODUCTION

Traditional performance metrics for communication networks include throughput, fairness, and delay. From the point of view of many applications in different types of networks, these may not be sufficient indicators of acceptable performance. For example, for an intrusion detection application running on a wireless sensor network, a high throughput and low delay data stream may provide little or no utility to the end user if it is unable to correctly detect the phenomenon of interest. Similarly, in a mission-critical tactical network, a low volume of reliable data may be more useful than a less accurate high volume of data. In light of this, there is growing interest in defining and quantifying new higher layer metrics that can capture application-specific objectives more adequately. To date, several such metrics have been proposed. Some examples include provenance, accuracy/fidelity, reliability, corroboration/credibility, freshness, etc. A notion of *Quality of Information* (QoI) is being proposed to formally describe this class of attributes [1] [2]. For example, [1] discusses the significance of QoI-aware networking in tactical military networks and describes several types of attributes that affect QoI.

The QoI associated with a piece of information can be defined as the *value* provided by it to the end user/application. Thus, by definition, it is application dependent. It should be noted that this is different from the notion of Quality of Service (QoS), which is defined based on traditional metrics such as delay, jitter or packet loss ratio, and where the specific information content is irrelevant to determine its utility to the end user. Recent work is beginning to explore the impact of QoI on fundamental networking operations such as rate control, scheduling, and routing. For example, [3] considers the design of information quality aware routing in sensor networks. Specifically, it addresses a least-cost routing tree problem that satisfies a given accuracy of detection constraint and proposes a greedy heuristic based on least-cost path approximation and pruning. Similarly, the QoI attribute of accuracy and its impact on rate control at the transport layer has been considered in [4], [5] in the context of event detection applications in wireless sensor networks. Reference [6] considers network utility maximization for sensor networks capable of in-network processing, while [7] focuses on the effect of corroboration, where the belief in information is reinforced by several sources reporting the same (or similar) information in social swarming applications. [10], [11] consider rate allocation and scheduling for distortion-sensitive compressible data.

The QoI attributes in these works have been studied in isolation and the trade-offs that result when these are jointly considered have not been investigated. Given that some of these can be traded off with one another, it is important to investigate the question of how to make control decisions that optimize performance with respect to multiple attributes. In this paper, we take a step in this direction by analyzing a model where the QoI depends on both the *accuracy* and the *freshness* of the information, and characterizing optimal decision strategies for various network scenarios.

Specifically, we consider a task processing network such as a sensor network or a team of mobile nodes. The network is sent jobs sequentially from an end user, i.e., central command or the sink node, and it uses its resources such as energy, sensing, computation, etc. to process this task. The utility derived by the end user, i.e. the QoI, depends on the quality of the response generated by the network, and is a function of both the accuracy and the freshness of the information provided by the network. Accuracy can be increased by a longer processing time for a particular task. For example, if a task corresponds to a certain event that is sensed, its accuracy

can be increased by waiting for additional observations about the event, or by running more extensive computations on these observations. However, the utility associated with the response also diminishes the longer it takes for the network to respond. This introduces a trade-off. Should we respond with less accurate but more timely information? Or should we incur more delay while making the information more accurate? We cast this as a sequential decision making problem and make use of dynamic programming and optimal stopping theory to characterize the structure of the optimal solution. Specifically, in Sec. III, we show that for many scenarios of practical interest, the optimal policy has a simple threshold structure. Further, in Sec. IV, we provide a method to approximately compute the threshold in closed-form. Numerical results suggest that the approximate threshold performs close to optimal.

II. BASIC MODEL

We consider an information processing network that sequentially receives tasks from an end user and employs its resources to process them. When it is finished processing a task, it reports the result back to the end user. The utility derived by the end user from this response depends on its Quality of Information (QoI). This QoI is a function of both the *accuracy* and the *freshness* of the information, to be made precise below.

We assume that at most one task can be processed by the network at any time. The network can continue to work on the task as long as there is no new task at hand. However, once a new task arrives, the network must stop processing the current task, send its response, and move on to the new task. Thus, the tasks are not queued in the basic model and the deadline for any task is given by the (random, potentially unknown) time until the next arrival. Within this duration, the network may decide to stop at any time and return its response, for example, if it determines that the processing thus far has resulted in a sufficient amount of QoI. When this happens, the network stays idle until the next task arrival. Similar models for task processing networks have been considered in [6]–[9].

We assume a time-slotted system. In the basic model, we assume that the interarrival times of the tasks are independent identically distributed (i.i.d.) according to a fixed distribution function with finite mean. We also assume that the tasks themselves, the QoI associated with them and their responses, are all independent of each other. Under these assumptions, the arrival of each new task can be thought of as a *renewal* event. Later, we extend the model to a scenario where the interarrival times depend on the control decisions made by the network.

Within the lifetime of a task, the system evolves as follows. Let $X(t)$ denote the QoI state corresponding to the current task in slot t. The QoI state $X(t)$ is assumed to be a non-negative real valued number and represents the amount of utility that the end user derives if the network sends back its response in the current slot.[1] We assume that for each task, the QoI state

---

[1]More generally, the utility derived could be a functional of the QoI state $f(X(t))$, where $f(X)$ is increasing in X. For simplicity, we assume $f(X) = X$.
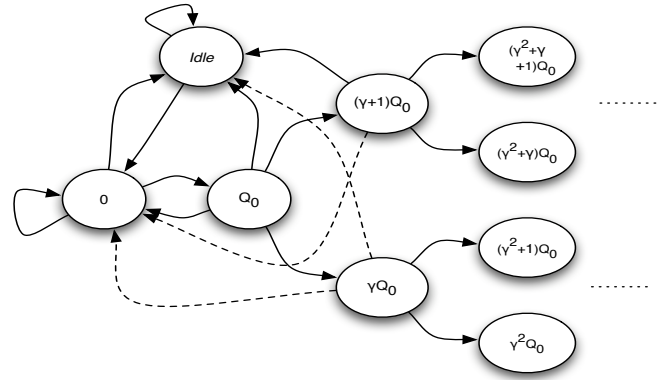

Fig. 1. State transition diagram for the case with i.i.d. Bernoulli $Q(t)$ process. For simplicity, all transitions are not shown.

starts with value 0 and transitions to an idle state I whenever the network sends back its response. However, if the network continues to process the task in the current slot, the QoI state in the next slot is given by:

$$X(t+1) = \gamma X(t) + Q(t) \qquad (1)$$

where $0 \leq \gamma < 1$ is a discounting factor that captures the effect of the attribute of freshness on the QoI. A lower value of γ implies higher dependence of the QoI on the attribute of freshness and vice versa. The second term $Q(t)$ represents the increase in QoI as a result of the continued processing, and this can be thought of as the gain in total QoI due to potentially higher accuracy of information. We assume that $Q(t)$ in (1) takes values in an i.i.d. fashion from a finite set of non-negative values and has a finite mean $\mathbb{E}\{Q(t)\} = \overline{Q}$. Note that the new QoI state is assumed to be the sum of the past discounted QoI and the increase $Q(t)$.

The following is an example scenario that fits in this model. Consider a sensor network where consecutive queries are issued and the network can handle one query at a time. To respond to any query, the network must make observations that incur some cost and potentially generate stochastic outcomes. While more observations tend to increase the quality of the response, the end user also cares about its timeliness, so that delay in responding diminishes its value to the end user. Thus, the overall QoI can be captured by (1).

Before proceeding, we note an underlying assumption of the model. It is assumed that the QoI or utility associated with data can be computed in the network/fusion center. Doing this would require the network to be aware of a model and its associated parameters that enables the evaluation. For example, in (1), it would require a knowledge of γ. This could be specified by the end user based on application characteristics.

From the description of the basic model, denoting the system idle state with I, it can be seen that in any state $X \neq I$, there are two possible control actions: *Stop*: A response is sent back to the end user, and *Continue*: The network continues to process the current task. In the following, we assume that

stopping in any state $X \neq I$ yields a QoI utility of X while continuing incurs a processing cost of β. For simplicity, we assume that β is constant in the rest of the paper while noting that the results can be easily generalized to the case where β takes values randomly from a set in an i.i.d. fashion.[2] If there is no new task arrival in the next slot, then the control action of *Stop* transitions the state to I whereas with a new task arrival, the state transitions to 0. Likewise, if there is no new task arrival in the next slot, the control action of *Continue* transitions the state according to (1). However, with a new task arrival, the network must send its response for the last task first and then the state transitions to 0. Once the response has been sent for a task, the system stays in the idle state $X = I$ until the arrival of a new task.

Fig. 1 shows the state transition diagram for a sample scenario where $Q(t)$ is i.i.d. Bernoulli taking values from $\{0, Q_0\}$. Given this model, our goal is to design a control algorithm that maximizes the time-average QoI delivered by the network. This is made precise in the next section.

A. Control Objective

Let X_r denote the QoI delivered to the end user for task r under any policy (where $r \in \{1, 2, 3, \ldots\}$). Similarly, let C_r denote the total processing cost incurred on task r. This is equal to β times the number of slots in which the control action of *Continue* was chosen for task r. Then, our control objective is to maximize the time average QoI minus the average processing cost. Since the length of a renewal period is independent of the control actions, this is equivalent to maximizing the average QoI minus the total processing cost per task. This can be expressed as:

$$\text{Maximize: } \lim_{R \to \infty} \frac{1}{R} \sum_{r=1}^{R} \mathbb{E}\left\{ X_r - C_r \right\} \qquad (2)$$

Since the tasks are independent of each other, this maximization can be achieved by *separately* maximizing the expected QoI minus the total processing cost in each renewal period. This problem can be cast as a *stochastic shortest path* (SSP) problem [13]. Specifically, we always start in the QoI state $X = 0$ and terminate whenever the next task arrives. Since the expected interarrival time of the tasks is finite, this SSP problem always terminates under any control policy. In the next section, we present the dynamic programming formulation for this problem.

B. Dynamic Programming Formulation

Let (X, N) represent the current state of the system where X denotes the QoI state and N denotes the number of slots that have elapsed since the last renewal. Let $p(N)$ be the probability that a new task arrives in the next slot given that N slots have elapsed since the last renewal. Finally, let $J(X, N)$ denote the optimal cost-to-go value functions. Then,

---

[2]The structural results of the paper also hold if there is a constant or random i.i.d. communication cost associated with sending the response to the end user.

$\forall N \geq 0, \forall X \neq I$, Bellman's equation for this system is:

$$J(X, N) = \max \Big[X, -\beta + p(N)(\gamma X + \overline{Q}) $$
$$+ (1 - p(N))\mathbb{E}\left\{ J(\gamma X + Q, N + 1) \right\} \Big] \qquad (3)$$

This can be interpreted as follows. In any state $(X, N), X \neq I$, we have two options: *Stop*, or *Continue*. The first term within the max operator on the right hand side denotes the utility derived if we stop in state (X, N). The second term denotes the expected utility derived if we continue. This involves a processing cost of β. Then, if a new task arrives in the next slot, we must stop processing the current task and send its response. This yields an expected utility given by $p(N)(\gamma X + \overline{Q})$. Finally, if there is no new task in the next slot, we transition to the state $(\gamma X + Q, N + 1)$ where Q is the random increase in QoI due to the continued processing. The optimal value function in state (X, N) is given by the maximum of these two terms. Note that when $X = I$, we trivially have $J(I, N) = 0, \forall N \geq 0$. From (3), it is optimal to stop in state (X, N) if:

$$X \geq \frac{-\beta + p(N)\overline{Q} + (1 - p(N))\mathbb{E}\left\{ J(\gamma X + Q, N + 1) \right\}}{1 - \gamma p(N)}$$

Else, it is optimal to continue. In the next section, we analyze the above property in detail for specific types of task arrival processes.

III. CHARACTERIZING THE OPTIMAL POLICY

We consider three types of task arrival processes: geometric task arrivals, periodic task arrivals, and infinite backlog.

A. Geometric Arrivals

Suppose in each slot, a new task arrives according to an i.i.d. Bernoulli process with rate $0 < \lambda \leq 1$. Then, the interarrival times are geometrically distributed with mean $\frac{1}{\lambda}$, and we have that $p(N) = \lambda$ for all N. Using the memoryless property of the geometric distribution, (3) can be simplified:

$$J(X) = \max \Big[X, -\beta + \lambda(\gamma X + \overline{Q}) $$
$$+ (1 - \lambda)\mathbb{E}\left\{ J(\gamma X + Q) \right\} \Big] \qquad (4)$$

Thus, it is optimal to stop in any state X if $X \geq \frac{-\beta + \lambda\overline{Q} + (1-\lambda)\mathbb{E}\{J(\gamma X + Q)\}}{1 - \lambda\gamma}$. We now show that the optimal control policy is of threshold type with a unique threshold.

Theorem 1: For geometric interarrival times, the optimal control policy is of threshold type with a unique threshold α^g such that it is optimal to stop in any state $X \geq \alpha^g$, and continue otherwise.

Proof: It is known that the value iteration algorithm converges to the optimal value function [13]. Specifically, consider the following iteration:

$$J_0(X) = 0 \qquad \forall X \qquad (5)$$
$$J_{k+1}(X) = \max \Big[X, -\beta + \lambda(\gamma X + \overline{Q}) $$
$$+ (1 - \lambda)\mathbb{E}\left\{ J_k(\gamma X + Q) \right\} \Big] \qquad (6)$$

Fig. 2. Figure illustrating the threshold behavior of the optimal policy for geometric interarrival times.
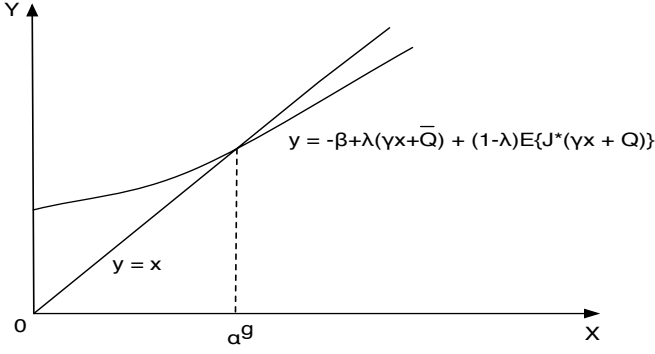

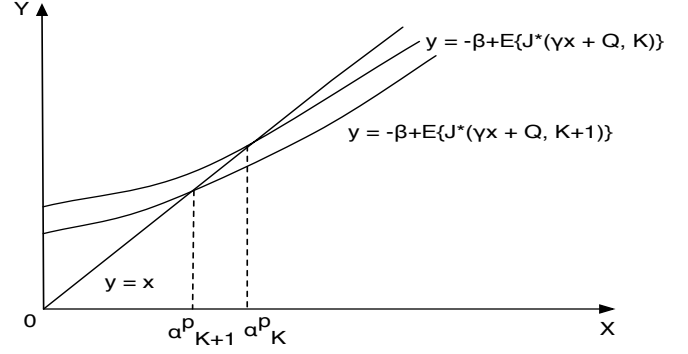Fig. 3. Figure illustrating the stage dependent threshold behavior of the optimal policy for periodic interarrival times.

Then, it is known that $\lim_{k \to \infty} J_k(X) = J^*(X)$ where $J^*(X)$ is optimal value function [13]. Using induction, we first show that $J^*(X)$ is convex and increasing in X. Further, we show that the maximum slope of $J^*(X)$ is 1.

Using the above iteration with $k = 0$, $J_1(X)$ is given by:

$$J_1(X) = \begin{cases} -\beta + \lambda(\gamma X + \overline{Q}) & \text{if } X < \frac{\lambda \overline{Q} - \beta}{1 - \lambda \gamma} \\ X & \text{else} \end{cases}$$

Thus, $J_1(X)$ is convex and increasing in X. Further, since $\lambda \gamma < 1$, the maximum slope of $J_1(X)$ is 1.

Now assume that $J_k(X)$ is convex and increasing in X and that the maximum slope of $J_k(X)$ is 1. Then $\mathbb{E}\{J_k(\gamma X + Q)\}$ is also convex and increasing and has maximum slope given by $\gamma < 1$. Therefore, using (6), it can be seen that $J_{k+1}(X)$, which is the maximum of X and $-\beta + \lambda(\gamma X + \overline{Q}) + (1 - \lambda)\mathbb{E}\{J_k(\gamma X + Q)\}$, is also convex and increasing. Further, it has a maximum slope of $\max[1, \lambda \gamma + (1 - \lambda)\gamma] = 1$. Since this holds for all k and since $\lim_{k \to \infty} J_k(X) = J^*(X)$, we have that $J^*(X)$ is convex and increasing in X and has maximum slope 1.

Since the maximum slope of $J^*(X)$ is 1, the maximum slope of the line $y = -\beta + \lambda(\gamma X + \overline{Q}) + (1 - \lambda)\mathbb{E}\{J^*(\gamma X + Q)\}$ is $\lambda \gamma + (1 - \lambda)\gamma = \gamma < 1$. Together with the fact that[3] $J^*(0) > 0$, this implies that there is *exactly* one point of intersection between the curves $y = X$ and $y = -\beta + \lambda(\gamma X + \overline{Q}) + (1 - \lambda)\mathbb{E}\{J^*(\gamma X + Q)\}$. This point is given by α^g (see Fig. 2). ∎

B. Periodic Arrivals

Next we consider the case where tasks arrive periodically, once every D slots. Thus, each task has a fixed deadline of D slots. $p(N)$ in (3) becomes $p(D - 1) = 1$ and $p(N) = 0$ for all $N \neq D - 1$, and we have:

$$J(X, D) = X, \tag{7}$$

$$J(X, D - 1) = \max\left[X, -\beta + \gamma X + \overline{Q}\right], \tag{8}$$

$$J(X, K) = \max\left[X, -\beta + \mathbb{E}\{J(\gamma X + Q, K + 1)\}\right], \tag{9}$$

[3] We ignore the degenerate case $J^*(0) = 0$.

where $0 \leq K \leq D - 2$ in (9). Thus, it is optimal to stop in any state $(X, D - 1)$ if $X \geq \frac{\overline{Q} - \beta}{1 - \gamma}$. Similarly, it is optimal to stop in any state (X, K) if $X \geq -\beta + \mathbb{E}\{J(\gamma X + Q, K + 1)\}$. We will show that the optimal policy is described by stage dependent thresholds α_K^p.

We first establish the following property of the value functions that allows us to characterize the relationship between the thresholds.

Lemma 1: $J(X, K) \geq J(X, K + 1) \quad \forall X, K$

Proof: This follows by noting that by choosing the same sequence of control actions starting in state (X, K) as the optimal policy would choose starting in state $(X, K + 1)$, we can get at least the same utility as $J(X, K + 1)$. ∎

Then, we have the following:

Theorem 2: For periodic interarrival times, the optimal control policy is of threshold type with stage dependent thresholds α_K^p such that in any stage K, it is optimal to stop in any state $X \geq \alpha_K^p$ and continue otherwise. Further, these thresholds satisfy the relation $\alpha_K^p \geq \alpha_{K+1}^p$ for all K.

Proof: Using the same technique as in the geometric case, it can be shown that $J(X, K)$ is convex and increasing in X for all K. Further, the maximum slope of $J(X, K)$ is 1.

Since the maximum slope of $J(X, K)$ is 1 for all K, the maximum slope of the line $y = -\beta + \mathbb{E}\{J(\gamma X + Q, K + 1)\}$ is given by $\gamma < 1$. Together with the fact that $J(0, K) > 0$, these facts imply that for each K, there is *exactly* one point of intersection between the curves $y = X$ and $y = -\beta + \mathbb{E}\{J(\gamma X + Q, K + 1)\}$. This point is given by α_K^p.

Using Lemma 1, we have that $\mathbb{E}\{J(\gamma X + Q, K + 1)\} \geq \mathbb{E}\{J(\gamma X + Q, K + 2)\}$. Thus, the point of intersection of $y = X$ and $y = -\beta + \mathbb{E}\{J(\gamma X + Q, K + 1)\}$, i.e., α_K^p, cannot be smaller than the point of intersection of $y = X$ and $y = -\beta + \mathbb{E}\{J(\gamma X + Q, K + 2)\}$, i.e., α_{K+1}^p (See Fig. 3). This shows that $\alpha_K^p \geq \alpha_{K+1}^p$. ∎

This result also captures the intuitive observation that as the deadline approaches, the thresholds to stop decrease as there are fewer opportunities to increase the QoI. For large D, the optimal policy is well approximated by a stationary policy that uses a single threshold for all stages. This can be thought of as the case where the network is lightly loaded with tasks.

Further, in the limit as $D \to \infty$, this reduces to a one-shot problem with only one task and thus we have the following:

Theorem 3: For the one-shot problem with only one task, the optimal control policy is of threshold type with a unique threshold α^{os} such that it is optimal to stop in any state $X \geq \alpha^{os}$ and continue otherwise.

C. Infinite Backlog

Next, we consider the scenario where an infinite number of tasks are waiting in the system. This can be thought of as the case where the network is heavily loaded with tasks. In contrast to the cases in Sec. III-A and III-B, the interarrival times for the tasks are now a function of the control decisions. Specifically, a new task arrives whenever the network stops processing the current task and sends its response. This makes the problem structurally different from the ones treated earlier.

This problem can be cast as an optimal stopping time problem [12] with the goal of maximizing the rate of return. Specifically, we want to maximize the ratio:

$$\frac{\mathbb{E}\{Y_N\}}{\mathbb{E}\{T_N\}} \tag{10}$$

for a stopping rule N, where Y_N is the QoI delivered minus total processing cost and T_N is the duration of the renewal period resulting from using the stopping rule N. Note that unlike the cases treated in Sec. III-A and III-B, it is not sufficient to simply maximize the numerator $\mathbb{E}\{Y_N\}$, as the length of the renewal period T_N depends on the control policy used.

Such problems can be approached by transforming the rate of return problem into a stopping time problem. Specifically, (10) can be maximized by solving a stopping problem that maximizes $\mathbb{E}\{Y_N - \xi T_N\}$ where ξ is the optimal rate of return. Note that with this transformation, the total per stage cost to continue becomes $\beta + \xi$. Since the total per stage cost to continue $\beta + \xi > 0$, it can be shown that this problem satisfies the framework of Assumption P in [13, Section 3.4]. Then, the DP algorithm is given by:

$$J(X) = \max\left[X, -\beta - \xi + \mathbb{E}\{J(\gamma X + Q(t))\}\right] \tag{11}$$

Further, as $k \to \infty$, the following value iteration algorithm converges to the optimal value function:

$$J_0(X) = 0 \qquad \forall X \tag{12}$$

$$J_{k+1}(X) = \max\left[X, -\beta - \xi + \mathbb{E}\{J_k(\gamma X + Q(t))\}\right] \tag{13}$$

Comparing this with (6), it can be seen that the iterates have similar structure. Using similar arguments, for this case also, we have the following:

Theorem 4: For the case with infinite backlog, the optimal control policy is such that it is optimal to stop in any state $X \geq \alpha^{inf}$ and continue otherwise. Furthermore, the threshold α^{inf} is unique.

IV. Approximate Threshold

The value iteration algorithm, in principle, can be used to compute the optimal value function and the optimal threshold.

However, it can be computationally intensive. Here, we present an alternate method to determine the optimal threshold approximately. The advantage of this is that it gives an expression for the optimal threshold in closed form. This method relies on making an approximation of the optimal value function.

We focus on the geometric case. For simplicity, we assume that $Q(t)$ is a Bernoulli i.i.d. process: $Q(t) = Q_0$ w.p. q and 0 else, so that $\overline{Q} = qQ_0$. Since it is optimal to stop in any state $X \geq \alpha^g$ (Theorem 1), using (4) we have that:

$$J(\alpha^g) = \alpha^g = \max\Big[\alpha^g, -\beta + \lambda(\gamma\alpha^g + \overline{Q})$$
$$+ (1-\lambda)[qJ(\gamma\alpha^g + Q_0) + (1-q)J(\gamma\alpha^g)]\Big] \tag{14}$$

Note that at the threshold α^g, the two terms within the max operator must be equal. This is because α^g corresponds to the point of intersection between the curves $y = X$ and $y = -\beta + \lambda(\gamma X + \overline{Q}) + (1-\lambda)\mathbb{E}\{J(\gamma X + Q(t))\}$ (see Fig. 2). Next, the maximum possible value for any state X is $\frac{Q_0}{1-\gamma}$. This implies that $\gamma\alpha^g + Q_0 \geq \alpha^g$. This is because if $\gamma\alpha^g + Q_0 < \alpha^g$, we get $\alpha^g > \frac{Q_0}{1-\gamma}$, a contradiction. Next, when state $X = \gamma\alpha^g + Q_0$, the optimal decision is to stop, since $\gamma\alpha^g + Q_0 \geq \alpha^g$ and we are to the right of the threshold. Thus, $J(\gamma\alpha^g + Q_0) = \gamma\alpha^g + Q_0$.

Using these, we can get the following equation for α^g:

$$\alpha^g = -\beta + \lambda(\gamma\alpha^g + \overline{Q})$$
$$+ (1-\lambda)[q(\gamma\alpha^g + Q_0) + (1-q)J(\gamma\alpha^g)] \tag{15}$$

Next, for $0 \leq X < \alpha^g$, we approximate the value function $J(X)$ by a straight line with slope γ, i.e., $J(X) \approx \gamma X + \alpha^g(1 - \gamma)$. The intuition behind this approximation is that the maximum slope of $y = -\beta + \lambda(\gamma X + \overline{Q}) + (1 - \lambda)\mathbb{E}\{J(\gamma X + Q(t))\}$ is γ. Thus, our approximation is a lower bound on the actual value function $J(X)$ when $0 \leq X < \alpha^g$.

Using this approximation, we get $J(\gamma\alpha^g) \approx \gamma^2\alpha^g + \alpha^g(1 - \gamma) = \alpha^g(1 - \gamma + \gamma^2)$. Now we can use (15) to evaluate α^g as follows:

$$\alpha^g \approx \frac{\overline{Q} - \beta}{1 - \lambda\gamma - (1-\lambda)[q\gamma + (1-q)(1 - \gamma + \gamma^2)]} \tag{16}$$

We compare the performance of a scheme that uses this approximate threshold with the optimal scheme for different parameter settings. Specifically, we fix $Q_0 = 1, q = 0.5, \beta = 0.01$ and vary the task arrival rate λ and decay factor γ. As shown in Fig. 4, a scheme that uses the approximate threshold performs quite close to the optimal policy. This suggests the goodness of the approximation (16).

V. Conclusions

In this paper, we have considered a model for Quality of Information (QoI) aware scheduling in task processing networks. Specifically, we have focused on two attributes that affect QoI: accuracy and freshness, and presented a model that quantifies the trade-off between the two. Using dynamic programming and optimal stopping theory, we showed that the optimal scheduling policy that maximizes the time average utility delivered by the network has a simple threshold structure for
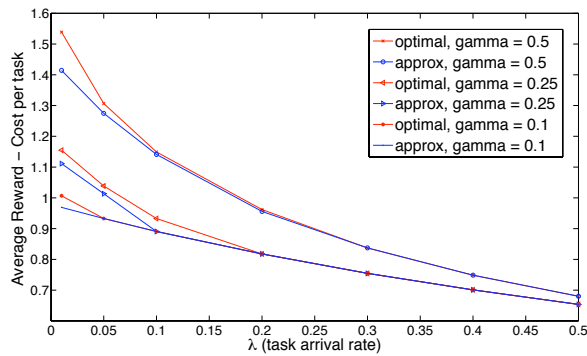
Fig. 4. Comparison of average reward minus cost per task under the optimal and approx. thresholds for different task arrival rates λ and decay factor γ.

many scenarios of practical interest. Further, we proposed a method to approximately compute the threshold. Numerical results suggest that the approximate threshold performs close to optimal. We believe that this work takes a step towards incorporating application aware objectives in making optimal scheduling decisions.

ACKNOWLEDGMENT

REFERENCES

[1] A. Bar-Noy, G. Cirincione, R. Govindan, S. Krishnamurthy, T. F. La Porta, M. J. Neely, P. Mohapatra, and A. Yener. Quality-of-Information aware networking for tactical military networks. In *The Third International Workshop on Information Quality and Quality of Service for Pervasive Computing (IQ2S 2011)*, Seattle, WA, March 2011.

[2] C. Bisdikian, L. M. Kaplan, M. B. Srivastava, D. J. Thornley, D. Verma, and R. I. Young. Building principles for a quality of information specification for sensor information. In *12th Int'l Conf. on Information Fusion (Fusion '09)*, Seattle, WA, July 2009.

[3] H.-X. Tan, M.-C. Chan, W. Xiao, P.-Y. Kong, and C.-K. Tham. Information quality aware routing in event-driven sensor networks. *Proc. of IEEE INFOCOM*, March 2010.

[4] Z. M. Charbiwala, S. Zahedi, Y. Kim, Y. H. Cho, and M. B. Srivastava. Toward quality of information aware rate control for sensor networks. In *Fourth International Workshop on Feedback Control Implementation and Design in Computing Systems and Networks*, April 2009.

[5] S. Zahedi, M. B. Srivastava, and C. Bisdikian. A framework for qoi-inspired analysis for sensor network deployment planning. In *International Workshop on Performance Control in Wireless Sensor Networks*, Austin, TX, October 2007.

[6] S. Eswaran, M. P. Johnson, A. Misra, and T. F. La Porta. Adaptive in-network processing for bandwidth and energy constrained mission-oriented multi-hop wireless networks. *Proc. of IEEE/ACM DCOSS*, Los Angeles, CA, June 2009.

[7] B. Liu, P. Terlecky, A. Bar-Noy, R. Govindan, and M. J. Neely. Optimizing information credibility in social swarming applications. *Proc. IEEE INFOCOM*, April 2011.

[8] M. J. Neely. Dynamic optimization and learning for renewal systems. *Proc. of Asilomar Conference*, Nov. 2010.

[9] C. H. Liu, C. Bisdikian, J. W. Branch, and K. K. Leung. QoI-Aware wireless sensor network management for dynamic multi-task operations. *Proc. IEEE SECON*, June 2010.

[10] M. J. Neely and A. Sharma. Dynamic data compression with distortion constraints for wireless transmission over a fading channel. *arXiv.0807.3768v1, submitted*, July 2008.

[11] E. N. Ciftcioglu, Y. E. Sagduyu, A. Yener, and R. A. Berry. Queue-based compression in a two-way relay network. *Proc. Conference on Information Sciences and Systems*, Princeton, NJ, March 2010.

[12] T. Ferguson. Optimal stopping and applications. *http://www.math.ucla.edu/~tom/Stopping/Contents.html*, 2006.

[13] D. Bertsekas. *Dynamic Programming and Optimal Control*, vol. 2, Athena Scientific, Belmont, MA, 2007.