

Learning Optimal Scheduling Policy for Remote State Estimation Under Uncertain Channel Condition

Shuang Wu, Xiaoqiang Ren, Qing-Shan Jia, Karl Henrik Johansson, Ling Shi

Abstract—We consider optimal sensor scheduling with unknown communication channel statistics. We formulate two types of scheduling problems with the communication rate being a soft or hard constraint, respectively. We first present some structural results on the optimal scheduling policy using dynamic programming and assuming that the channel statistics is known. We prove that the Q -factor is monotonic and submodular, which leads to threshold-like structures in both problems. Then we develop a stochastic approximation and parameter learning frameworks to deal with the two scheduling problems with unknown channel statistics. We utilize their structures to design specialized learning algorithms. We prove the convergence of these algorithms. Performance improvement compared with the standard Q -learning algorithm is shown through numerical examples, which also discuss an alternative method based on recursive estimation of the channel quality.

Index Terms—State estimation, scheduling, threshold structure, learning algorithm.

I. INTRODUCTION

The development of precision manufacturing enables massive production of small-sized wireless sensors. These sensors are deployed to collect data and transmit information for monitoring, feedback control and decision making [1]. As the sensor nodes are often battery powered and the communication channel is shared by a large amount of devices, it is critical to optimize the transmission schedule of the sensors to systematically tradeoff the system performance with the sensor communication overhead [2].

In the last few decades, numerous studies have been dedicated to optimize the communication rate v.s. the estimation

error of sensor nodes [3]–[9]. The general scheduling problems impose significant computation challenges due to its combinatorial nature. Sensor scheduling problems, however, usually possesses special structures and computation overhead can be reduced. One common idea is that the sensor only transmits when the recently obtained information is important with respect to a certain criterion. For example, the work in [5], [9] chose the criterion to be the certain norms of the innovation of a Kalman filter. The work in [6], [7] chose the criterion to be the variance of the estimation error.

The literatures in sensor scheduling can be categorized according to whether the underlying communication channel is idealized [7]–[9], lossy [3]–[6] or noisy [10]. The assumption of idealized channel condition ignores the underlying communication channel and simplifies the scheduling policy design. The design of the optimal transmission protocol for a non-ideal channel treats the channel as a part of the whole system and requires information of the communication channel conditions. The packet dropout process is often treated as Bernoulli process or a two-state Markov chain, while the channel noise is treated as an additive Gaussian white noise. Based on the channel model and its parameters, the optimal scheduling policy can be derived. However, acquiring information of the channel condition may be costly or even impossible [11].

This paper considers optimal sensor scheduling over a packet-dropping channel with packet dropout rate unknown. We consider two scenarios. In the first scenario, the communication is costly. In the second scenario, there is an explicit communication rate constraint. We first prove monotonicity and submodularity of the Q -factor for these two types of problems, which leads to threshold-like structures in the optimal scheduling policy. We then design iterative algorithms to obtain the optimal solution without knowing the packet dropout rate. The major contribution of this work is as follows.

- 1) We show threshold-like structures (Theorems 1 and 2) of the optimal policy in the considered sensor scheduling problems. Specifically, the optimal policy for the costly communication problem (Problem 1) is a threshold policy and the optimal policy for the constrained communication problem (Problem 2) is a randomized threshold policy. These results are significant for scheduling problems as they leads to easy implementations and they have been reported in other papers under different setups (Discussions are in Section III. In this work, we further utilize these properties to improve the standard Q -learning algorithm.
- 2) We develop iterative algorithms based on stochastic approximation and parameter estimation, and compare them in the two different types of scheduling problems. Based on the structure of the Q -factor, we devise structural learning methods, which impose the transient Q -factor

S. Wu and L. Shi are with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong (e-mail: swuak@ust.hk, eesling@ust.hk).

X. Ren is with School of Mechatronic Engineering and Automation, Shanghai University, Shanghai, China (e-mail: xqren@shu.edu.cn). *Corresponding author*: Xiaoqiang Ren, Tel:+8619821829619.

K. H. Johansson is with EECS, KTH Royal Institute of Technology, Stockholm, Sweden (e-mail: kallej@kth.se).

Q.-S. Jia is with the Center for Intelligent and Networked Systems, Department of Automation, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing 100084, China (e-mail: jiaqs@tsinghua.edu.cn).

The work by S. Wu and L. Shi is supported by a Hong Kong RGC General Research Fund 16208517.

The work of X. Ren is supported by Shanghai Key Laboratory of Power Station Automation Technology, and MOST Major Project for New Generation Artificial Intelligence (2018AAA0102804).

The work of Qing-Shan Jia is supported by the National Natural Science Foundation of China (61673229), MOST Major Project for New Generation Artificial Intelligence (2018AAA0101600), and National Key Research and Development Program of China (2016YFB0901900), and the 111 International Collaboration Project of China (BP2018006).

The work of K. H. Johansson is supported in part by the Knut and Alice Wallenberg Foundation, the Swedish Strategic Research Foundation, and the Swedish Research Council.

to satisfy certain properties (Theorem 3). In addition, we develop a synchronous learning algorithm by utilizing the fact that the randomness of the state transition is independent of the particular state. By using that the optimal scheduling policy of the constrained communication problem can be written in a closed form, we show that an adaptive control method can be directly used to obtain the optimal scheduling policy (Theorem 5).

In this work, we consider optimal scheduling with unknown channel conditions. We aim to adapt the scheduling policy to the real-time estimate of the channel condition. To yield an accurate estimate for the channel condition, it is necessary to utilize the history of transmission successes and failures to determine the scheduling policy. An intuitive method is to compute the optimal scheduling policy based on the estimate of the channel condition, which is obtained by keeping track of some sufficient statistics of the channel state. By taking the scheduling decision as control actions and the remote state estimation error as system states, the optimal scheduling problem can be formulated as an optimal control problem. The computation of the optimal control law usually involves solving a Bellman optimality equation [12], which is computationally intense. In this work, we develop iterative algorithms which are relatively easy to implement and reduce significant computation overhead compared with the intuitive method.

There are two main streams of research in the area of optimal control of unknown dynamic systems. One stream, termed as reinforcement learning [13]–[15], combines the stochastic approximation and dynamic programming to iteratively solve the Bellman optimality equation. The basic idea is to iteratively “learn” the value of each control decision at each state and take control actions based on the “learned” values. The major drawback is that every state-action pairs are required to be visited comparably often so that the estimate of the value of the state-action pairs are accurate. The transient performance may not be desirable as suboptimal actions are taken to estimate the values.

The other stream uses an adaptive control approach which combines the parameter estimation and the optimal control. Under certain conditions, the “certainty equivalence” holds, which implies a separation between parameter estimation and the optimal control. It is then optimal to take the parameter estimate as its actual value and take control actions based on the estimated parameters [16]–[18]. A major problem with the adaptive control is that computing the optimal control for a given parameter is computationally intense. We illustrate this with a numerical example in Section V. In this work, we utilize structures of the optimal policy to reduce the computation burden.

Both the reinforcement learning and the adaptive control frameworks guarantee that the iterative process converges to the optimal control policy under certain conditions. However, these works are quite generic. In specific problems, the special structure may be used to improve the transient performance. The sensor scheduling problem in this work possesses some structures in the optimal policy. We devise a learning scheme which takes advantage of these structures to improve transient performance and reduce computation overhead.

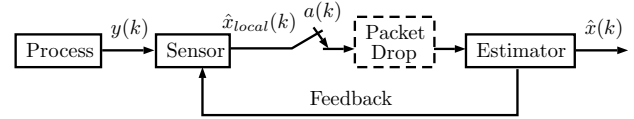


Fig. 1: System architecture.

The remainder of this paper is organized as follows. In section II, we provide the mathematical model of the sensor scheduling problem and two related optimization problems. In section III, we use a dynamic programming approach to show structural results. In section IV, we present two learning frameworks to solve the optimal scheduling policy when the channel condition is unknown. We summarize the paper in section V. Proofs are given in the appendix.

Notations: The bold symbol letter stands for a vector which aggregates all its components, e.g., $\mathbf{x} = [x_1, \dots, x_n]^T$. For a matrix X , $\rho(X)$, X^T and $\text{Tr}(X)$ stands for the spectral radius of the matrix, the matrix transpose and the trace of the matrix. The operation $[x]_{\mathcal{X}}$ denotes the projection of vector x into the constrained set \mathcal{X} . The probability and the conditional probability are denoted by $\Pr(\cdot)$ and $\Pr(\cdot|\cdot)$, respectively. The expectation of a random variable is $\mathbb{E}[\cdot]$. The set of nonnegative integers are represented by \mathbb{N} .

II. PROBLEM SETUP

A. System Model

The architecture of the system is depicted in Fig. 1. We consider the following LTI process.

$$\begin{aligned} x(k+1) &= Ax(k) + w(k), \\ y(k) &= Cx(k) + v(k), \end{aligned}$$

where $x(k) \in \mathbb{R}^n$ is the state of the process at time k and $y(k) \in \mathbb{R}^m$ is the noisy measurement taken by the sensor. We assume, at each time k , that the state disturbance noise $w(k)$, the measurement noise $v(k)$, and the initial state $x(0)$ are mutually independent random variables, which follow Gaussian distributions as $w(k) \sim \mathcal{N}(0, \Sigma_w)$, $v(k) \sim \mathcal{N}(0, \Sigma_v)$, and $x(0) \sim \mathcal{N}(0, \Pi)$. We assume that the covariance matrices Σ_w and Π are positive semidefinite, and Σ_v is positive definite. We assume that the pair (A, C) is detectable and that $(A, \sqrt{\Sigma_w})$ is stabilizable.

The sensor measures the process states and computes its local state estimates $\hat{x}_{local}(k)$ using a Kalman filter. After that, the sensor decides whether it should or not transmit the estimate through the packet-dropping communication channel to a remote state estimator. We use $a(k) = 1$ to denote transmitting local estimate $\hat{x}_{local}(k+1)$ at time $k+1$ and $a(k) = 0$ to denote no transmission. Let $\eta(k) = 1$ denote that the packet is successfully received by the remote estimator at time k and $\eta(k) = 0$ otherwise. The successful transmissions are assumed to be independent and identically distributed as

$$\Pr(\eta(k+1)|a(k) = 1) = \begin{cases} r_s, & \text{if } \eta(k+1) = 1, \\ 1 - r_s, & \text{if } \eta(k+1) = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Meanwhile, it is straightforward that $\Pr(\eta(k+1) = 0|a(k) = 0) = 1$.

The remote state estimator will either synchronize the remote state estimate with the local state estimate if the updated data is received, or use process dynamics to predict the state if no data is received. We assume that the local state estimate of the Kalman filter is in steady state. Define the remote state estimate as

$$\hat{x}(k) = \mathbb{E}[x(k)|\eta(0), \eta(0)\hat{x}_{local}(0), \dots, \eta(k), \eta(k)\hat{x}_{local}(k)].$$

The mean square estimation error covariance of the remote estimator at time k , which is defined as

$$P(k) = \mathbb{E}[(x(k) - \hat{x}(k))(x(k) - \hat{x}(k))^\top | \eta(0), \eta(0)\hat{x}_{local}(0), \dots, \eta(k), \eta(k)\hat{x}_{local}(k)],$$

can be computed as follows:

$$P(k) = \begin{cases} \bar{P}, & \text{if } \eta(k) = 1, \\ AP(k)A^\top + \Sigma_w, & \text{if } \eta(k) = 0, \end{cases}$$

where \bar{P} is the steady state of the state estimation error covariance of the Kalman filter.

The remote estimator will feed back a one-bit signal to the sensor to acknowledge its successful reception of the packet. The information of the remote state estimate available to the sensor for transmission decision is

$$\tau(k) = \min\{0 \leq t \leq k : \eta(k-t) = 1\},$$

which is the time elapsed since the last successful transmission. The temporal relation among $a(k)$, $\eta(k)$ and $\tau(k)$ is illustrated in Fig. 2. Notice that $\tau(k)$ and $\eta(k)$ are equivalent

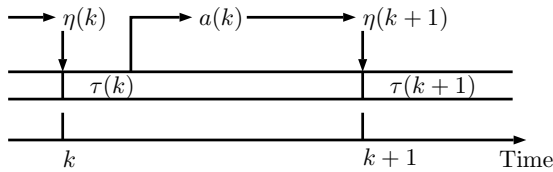


Fig. 2: Relation among state $\tau(k)$, action $a(k)$ and transmission result $\eta(k)$.

in the sense that both of them can be used to compute the estimation error covariance at the remote estimator, which can be written as

$$P(k) = \begin{cases} \bar{P}, & \tau(k) = 0, \\ A^{\tau(k)}\bar{P}(A^\top)^{\tau(k)} + \sum_{t=0}^{\tau(k)-1} A^t \Sigma_w (A^\top)^t, & \tau(k) \geq 1. \end{cases} \quad (1)$$

An admissible scheduling policy $f = \{f_k\}_{k=0}^\infty$ is a sequence of mappings from $\tau_{0:k}$ and $a_{0:k-1}$ to the transmission decision, i.e.,

$$a(k) = f_k(\tau_{0:k}, a_{0:k-1}),$$

where $\tau_{0:k}$ and $a_{0:k-1}$ stand for $\tau(0), \dots, \tau(k)$ and $a(0), \dots, a(k-1)$, respectively. Denote \mathbb{F} as the set of all admissible policies, i.e., policies that are measurable by $\tau_{0:k}, a_{0:k-1}$.

B. Performance Metrics and Problem Formulation

Given a scheduling policy $f = \{f_k\}_{k=0}^\infty$, we define the expected average estimation error covariance of the remote estimator and the expected transmission rate. We use \mathbb{E}^f to denote the expectation under the scheduling policy f . The expected average estimation error covariance is

$$J_e(f) = \limsup_{T \rightarrow \infty} \frac{1}{T} \mathbb{E}^f \left[\sum_{k=0}^{T-1} \text{Tr}(P(k)) | P(0) = \bar{P} \right],$$

and the expected average transmission rate is

$$J_r(f) = \limsup_{T \rightarrow \infty} \frac{1}{T} \mathbb{E}^f \left[\sum_{k=0}^{T-1} a(k) | P(0) = \bar{P} \right].$$

We are interested in two optimization problems for these performance metrics.

Problem 1 (Costly Communication) *Given the communication cost for one transmission λ , solving the following minimization problem on the total cost:*

$$\inf_{f \in \mathbb{F}} J_e(f) + \lambda J_r(f).$$

Problem 2 (Constrained Communication) *Given a communication budget b , solving the following constrained minimization problem:*

$$\inf_{f \in \mathbb{F}: J_r(f) \leq b} J_e(f).$$

Remark 1 *The two problems are closely related. According to [19, Sec 11.4], if a policy f^* is a solution to Problem 2, then there exists a Lagrangian multiplier λ^* such that f^* minimizes $J_e(f) + \lambda^*(J_r(f) - b)$, which means that f^* minimizes $J_e(f) + \lambda^* J_r(f)$, i.e., f^* is a solution to Problem 1 with $\lambda = \lambda^*$. However, even if λ^* is known beforehand, an optimal policy of Problem 1 may not be an optimal policy for the corresponding Problem 2. As it will later be shown, optimal policy of Problem 1 can be found in the set of deterministic policies while optimal policies of Problem 2 are randomized in general.*

We assume for the main results of this paper that the channel condition r_s is unknown. When r_s is known, Problems 1 and 2 can be solved via dynamic programming [8], [9] or linear programming. Here, we cannot directly use the classical methods. We instead use a learning-based method. Dynamic programming approach is used to find some structures in the optimal scheduling policies. By utilizing the structures, we can accelerate the learning process.

A naive method to solve the problems is to iterate between estimating r_s and solving the corresponding mathematical programming. However, the optimization problem then needs to be solved at each time step, which is computationally intense. In this work instead, we find a simple iterative method, which does not incur much computation overhead compared to the naive method.

III. OPTIMAL SCHEDULING POLICY WITH KNOWN CHANNEL CONDITION

Before proceeding to the learning approach, we establish some structural results for the Problems 1 and 2 when assuming r_s is known. We reformulate the original two problems using Markov decision process (MDP). The costly communication problem can be directly formulated as an MDP, while the constrained communication problem is a constrained MDP (coMDP). We will show the connection between these models.

Some of the results (e.g., Theorems 1 and 2) are similar to those in the literatures. The setup in [8], [9], [20] is different from ours. Leong et al. [21] showed the optimality of a threshold policy for the costly communication problem (Problem 1), but no results were developed for the constrained communication problem (Problem 2). In addition, they showed threshold property by studying the relative value function instead of the Q -factor as we do in this work. To enable the structural learning procedure developed in the next section, we need to establish the monotonicity and submodularity of the Q -factor.

A. Costly Communication

An MDP $(\mathbb{S}, \mathbb{A}, \mathcal{P}, c)$ consists of the state space \mathbb{S} , the action space \mathbb{A} , the state transition probability \mathcal{P} , and one stage cost c . In our formulation, the state space consists of all the possible $\tau(k) = \tau \in \mathbb{N}$. The action space consists of the transmission decision $a = a(k) \in \{0, 1\}$. If action a is taken when the current state is τ , the state in the next time step will transit to τ_+ according to the state transition probability

$$\Pr(\tau_+|\tau, a) = \begin{cases} r_s, & \text{if } \tau_+ = 0 \text{ and } a = 1, \\ 1 - r_s, & \text{if } \tau_+ = \tau + 1 \text{ and } a = 1, \\ 1, & \text{if } \tau_+ = \tau + 1 \text{ and } a = 0, \\ 0, & \text{otherwise.} \end{cases}$$

The one stage cost is

$$c(\tau, a) = \text{Tr}(P(\tau)) + \lambda a,$$

where we use $P(\tau)$ to emphasize that the estimation error can be determined by τ from (1). A policy corresponds to the scheduling policy $f := \{f_k\}_{k=0}^\infty$, which maps the history $\tau_{0:k}, a_{0:k-1}$ to the action space, i.e., $f_k(\tau_{0:k}, a_{0:k-1}) = a(k)$. By the Markovian property of the state transitions, it suffices to consider Markovian policies, the decision of which only depends on the current state. Therefore, we only need to consider the policies of the form of $a(k) = f_k(\tau(k))$.

The costly communication problem is compatible with the MDP model described above and its solution can be obtained by solving the following problem

$$\inf_{f \in \mathbb{F}^M} \limsup_{T \rightarrow \infty} \frac{1}{T+1} \mathbb{E} \left[\sum_{k=0}^T c(\tau(k), a(k)) | \tau(0) = 0 \right], \quad (2)$$

where \mathbb{F}^M is the set of all Markovian policies. Moreover, the optimal policy can be found in the set of all stationary policies \mathbb{F}^S , i.e., $\mathbb{F}^S = \{f : f_k = f_{k+1}, \forall k \geq 0\}$, if a stability condition holds.

Lemma 1 *If $\rho^2(A)(1 - r_s) < 1$, there exists a stationary policy $f^* \in \mathbb{F}^S$ such that $a = f^*(\tau)$ solves the Bellman optimality equation:*

$$V(\tau) = \min_{a \in \mathbb{A}} \left[c(\tau, a) + \sum_{\tau_+} V(\tau_+) \Pr(\tau_+|\tau, a) - \mathcal{J}^* \right], \quad (3)$$

where \mathcal{J}^* is the optimal value of the trace of the average estimation error.

The stationary solution of the unconstrained MDP (2) can be obtained by solving the Bellman optimality equation with respect to (w.r.t.) a constant \mathcal{J}^* and the relative value function $V(\tau)$. The optimal policy is to choose the action that minimizes the right hand side of (3):

$$f(\tau) = \arg \min_{a \in \mathbb{A}} \left[c(\tau, a) + \sum_{\tau_+} V(\tau_+) \Pr(\tau_+|\tau, a) - \mathcal{J}^* \right].$$

We denote the value function of a state-action pair as

$$Q(\tau, a) = c(\tau, a) + \sum_{\tau'} V(\tau') \Pr(\tau'|\tau, a) - \mathcal{J}^*.$$

Note that $V(\tau) = \min_{a \in \mathbb{A}} Q(\tau, a)$. We rewrite (3) as

$$Q(\tau, a) = c(\tau, a) + \sum_{\tau'} \min_{u \in \mathbb{A}} Q(\tau', u) \Pr(\tau'|\tau, a) - \mathcal{J}^*. \quad (4)$$

We can develop the following structural results for the V -function and the Q -factor.

Lemma 2 (Monotonicity) $V(\tau) \geq V(\tau'), \forall \tau \geq \tau'$.

Lemma 3 (Monotonicity) $Q(\tau, a) \geq Q(\tau', a), \forall \tau \geq \tau'$.

Lemma 4 (Submodularity) $Q(\tau, a) - Q(\tau, a') \leq Q(\tau', a) - Q(\tau', a'), \forall \tau \geq \tau', a \geq a'$.

Thanks to monotonicity and submodularity, we have the threshold structure on the optimal policy for Problem 1¹.

Theorem 1 (Costly communication) *The optimal policy f^* for Problem 1 with known channel condition r_s is of threshold type, i.e., there exists a constant $\theta^* \in \mathbb{S}$ such that*

$$f^*(\tau) = \begin{cases} 0, & \text{if } \tau < \theta^*, \\ 1, & \text{if } \tau \geq \theta^*. \end{cases}$$

Since the optimal policy f^* is of threshold type, we use the threshold θ to represent a policy when there is no ambiguity.

Remark 2 *Although similar results are available in literatures, either the setup is different [8], [9], [20], or the results are obtained by imposing additional assumptions [22]. Moreover, to our best knowledge, the structure of the Q -factor (monotonicity and submodularity) that is revealed in this work is the first of its kind in the field of sensor scheduling.*

¹A similar result was also reported in [21]. We present it here for completeness and facilitate presentation of the structural learning as we utilized the monotonicity and submodularity of the Q -factor.

B. Constrained Communication

The state space, action space and the transition probability of Problem 2 is the same as those of Problem 1. Nevertheless, two types of one stage cost are involved in the constrained communication problem $c_e(\tau, a) = \text{Tr}(P(\tau))$ and $c_r(\tau, a) = a$. Problem 2 can be formulated as a constrained MDP as

$$\begin{aligned} \inf_{f \in \mathbb{F}} \quad & \limsup_{T \rightarrow \infty} \frac{1}{T+1} \mathbb{E} \left[\sum_{k=0}^T c_e(\tau(k), a(k)) | \tau(0) = 0 \right] \\ \text{s.t.} \quad & \limsup_{T \rightarrow \infty} \frac{1}{T+1} \mathbb{E} \left[\sum_{k=0}^T c_r(\tau(k), a(k)) | \tau(0) = 0 \right] \leq b. \end{aligned}$$

We use the Lagrangian multiplier approach to convert the constrained problem to the following saddle point problem

$$\begin{aligned} \inf_{f \in \mathbb{F}} \sup_{\lambda \geq 0} \limsup_{T \rightarrow \infty} \frac{1}{T+1} \mathbb{E} \left[\sum_{k=0}^T c_e(\tau(k), a(k)) | \tau(0) = 0 \right] \\ + \lambda \left(\limsup_{T \rightarrow \infty} \frac{1}{T+1} \mathbb{E} \left[\sum_{k=0}^T c_r(\tau(k), a(k)) | \tau(0) = 0 \right] - b \right). \end{aligned} \quad (5)$$

As the one-stage cost is bounded below and monotonically increasing, the above problem possesses a solution [23, Theorem 12.8]. If we relax Problem 2 by fixing λ , (5) reduces to Problem 1. Moreover, as the saddle point problem possesses a solution, there exists a λ^* such that the value of (5) with $\lambda = \lambda^*$ is the same as the value of the constrained problem (Remark 1).

The following lemma constitutes a necessary condition for a policy to be optimal.

Lemma 5 *If a scheduling policy $f \in \mathbb{F}$ solves Problem 2, it must satisfy $J_r(f) = b$.*

From [23], we know that as long as the constrained MDP is feasible, the optimal policy randomizes between at most $m + 1$ deterministic policies, where m is the number of constraints. Problem 1 has no constraints, the optimal policy is deterministic. Problem 2 has one constraint, so the optimal policy randomizes between at most two deterministic policies.

Theorem 2 (Constrained communication) *The optimal policy f^* for Problem 2 with known channel condition r_s is of Bernoulli randomized threshold type, i.e., there exist two constants $\theta^* \in \mathbb{S}$ and $0 \leq r_{\theta^*} \leq 1$ such that*

$$f^*(\tau) = \begin{cases} 0, & \text{if } \tau < \theta^*, \\ 0, & \text{with probability } 1 - r_{\theta^*}, \text{ if } \tau = \theta^*, \\ 1, & \text{with probability } r_{\theta^*}, \text{ if } \tau = \theta^*, \\ 1, & \text{if } \tau > \theta^*, \end{cases}$$

where r_{θ^*} and θ^* satisfy

$$\limsup_{T \rightarrow \infty} \frac{1}{T} \mathbb{E} \left[\sum_{k=0}^{T-1} f^*(\tau(k)) \right] = b.$$

We see that the optimal policy for the Problem 2 only depends on the communication budget b and the channel condition r_s . These relations are summarized in the following corollary.

Corollary 1 *The optimal threshold θ^* for Problem 1 and randomization parameter r_{θ^*} are given by*

$$\theta^* = \lfloor \frac{1}{r_s b} - \frac{1}{r_s} \rfloor, \quad r_{\theta^*} = \theta^* + 1 + \frac{b-1}{b r_s},$$

where $\lfloor \cdot \rfloor$ denotes the floor function.

IV. OPTIMAL SCHEDULING POLICY WITH UNCERTAIN CHANNEL CONDITION

The optimal scheduling policy can be obtained if the Q -factor is solved by (4). If the channel condition is not known beforehand, we cannot use classical solution techniques to solve the Bellman optimality equation. We propose two learning-based frameworks, stochastic approximation and parameter learning, to adaptively obtain the optimal policy without knowing the channel statistics a priori.

The stochastic approximation framework yields an iterative method to find a solution of the Bellman optimality equation. The optimal scheduling policy can be directly obtained from the Q -factor. The Bellman optimality equation (4) has a countable infinite state-space and cannot be solved directly. A finite-state approximation is needed. We restrict the largest state to be M , and any states larger than M are treated as M . The optimal action on such states is to transmit the local estimate. As the optimal policy is a threshold-type, the optimal scheduling policy can be captured by solving a finite state approximation as long as M is large enough. In other words, there exists an $M > 0$ such that the optimal policy of any finite state approximation with $|\mathbb{S}| \geq M$ is the same as the optimal policy of the original model. In practice, we have to set a maximal interval between two transmissions for a sensor to avoid the sensor being always idle. The number M can be set as the maximal interval. In the sequel, we denote \mathbb{S}' as the truncated state space.

In parameter learning method, we continuously estimate the channel condition based on the scheduling results and compute the corresponding optimal scheduling policy by taking the estimated channel condition as the actual condition. As we have proven that the optimal policy for Problem 2 can be analytically computed, this method is more suitable for Problem 2.

In the following two sections, we discuss the stochastic approximation method for Problem 1 and 2. The parameter learning method is treated in a third section. Note that since the sensor is aware that whether a transmission succeeds through the feedback acknowledgment from the remote state estimator. The learning algorithm is thus done at the sensor.

A. Problem 1 with Stochastic Approximation

At each time step k , an action $a(k)$ is selected for state τ in an ε -greedy pattern as

$$a(k) = \begin{cases} \arg \min_u Q_k(\tau, u), & \text{with probability } 1 - \varepsilon, \\ \text{any action,} & \text{with probability } \varepsilon, \end{cases}$$

where $\varepsilon > 0$ is a randomization parameter² We then observe that the state transits to $\tau(k+1) = \tau'$. The iterative update of the Q -factor is

$$Q_{k+1}(\tau(k), a(k)) = Q_k(\tau(k), a(k)) + \alpha(\nu_k(\tau(k), a(k))) \left[c(\tau(k), a(k)) + \min_{u \in \mathbb{A}} Q_k(\tau(k+1), u) - Q_k(\tau(k), a(k)) - Q_k(\tau_0, a_0) \right], \quad (6)$$

where (τ_0, a_0) is a fixed reference state-action pair, which can be arbitrarily chosen. The step size $\alpha(n)$ satisfies³

$$\sum_{n=0}^{\infty} \alpha(n) = \infty, \quad \sum_{n=0}^{\infty} [\alpha(n)]^2 < \infty,$$

and in (6) this step size depends on $\nu_k(\tau, a) = \sum_{n=0}^k \mathbf{1}[(\tau(n), a(n)) = (\tau, a)]$, which is the number of times that the state-action pair (τ, a) has been visited.

The above scheme is proven to converge [24], but the convergence rate is slow in practice. One reason is that the scheme is asynchronous as only one state-action pair is updated at each time step. We propose two improvements for this scheme by updating as many state-action pairs as possible. We denote them structured learning and synchronous update.

Remark 3 *The asynchronous algorithm does not converge to the actual Q -value under transition probability $\Pr(\tau_+|\tau, a)$ but a perturbed one as follows*

$$\tilde{\Pr}(\tau_+|\tau, a) = (1 - \varepsilon)\Pr(\tau_+|\tau, a) + \frac{\varepsilon}{|\mathbb{A}|} \sum_{u \in \mathbb{A}} \Pr(\tau_+|\tau, a).$$

This scheme is suboptimal. A smaller ε leads to a more accurate learning result but slows down the learning rate. The synchronous scheme, which will be introduced later, however, guarantees that the Q -value converges to its actual value as ε can be set to zero.

Remark 4 *In addition to the randomization parameter ε , the truncation parameter M and the stepsizes α also affects the learning process. A greater M leads to a higher accuracy. As we mentioned in the beginning of this section, the communication rate of a sensor should be above certain values. When M is large enough so that the optimal threshold is below M , the size M has very little effects on the accuracy. In light of transient behavior, big stepsizes lead to severe oscillation while small stepsizes lead to slow convergence rate. In practice, stepsizes of the form $\alpha(k) = \frac{c}{(1+k)^a}$, where c is a constant and $0.5 < a \leq 1$, can be selected to tradeoff between fast convergence rate and small oscillations.*

Structural learning. The first improvement is based on the structural results proven in the previous section. We can infer the unvisited state-action pair by using the monotonicity and the submodularity structure on the Q -factor. From this information, the Q -factor is closer to the solution of the Bellman optimality equation.

²The randomness is necessary because every state-action pairs should be visited with infinite number of times to guarantee convergence.

³Examples of such $\alpha(\cdot) > 0$ include $1/n^p$ with $0.5 < p \leq 1$, $\log(n)/n$ and $1/\lceil n \log(n) \rceil$.

Submodularity of the Q -factor gives

$$Q(\tau, 1) - Q(\tau, 0) - Q(\tau + 1, 1) + Q(\tau + 1, 0) \geq 0, \quad \tau \in \mathbb{S}'. \quad (7)$$

Stack the Q -factor for all state-action pair as a vector

$$\mathbf{Q} = [Q(0, 0), \quad Q(0, 1), \quad \dots, \quad Q(M, 0), \quad Q(M, 1)]^\top$$

We can then write (7) as $\mathbf{T}_s \mathbf{Q} \geq 0$, where

$$\mathbf{T}_s = \begin{bmatrix} -1 & 1 & 1 & -1 & 0 & 0 & \dots \\ 0 & 0 & -1 & 1 & 1 & -1 & \dots \\ & & & \vdots & & & \\ \dots & 0 & 0 & -1 & 1 & 1 & -1 \end{bmatrix}_{M \times 2(M+1)}$$

and the inequality is performed element-wisely. Similarly, we can use the monotonicity constraint $Q(\tau + 1, a) - Q(\tau, a) \geq 0$ for all τ to write $\mathbf{T}_m \mathbf{Q} \geq 0$, where

$$\mathbf{T}_m = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & \dots \\ 0 & -1 & 0 & 1 & 0 & \dots \\ & & \ddots & & \ddots & \\ \dots & 0 & -1 & 0 & 1 & \dots \end{bmatrix}_{2M \times 2(M+1)}$$

The two constraints can be compactly written as $\mathbf{TQ} \geq 0$.

Suppose there is a function $g(\mathbf{Q})$ such that its gradient with respect to $\mathbf{Q}(\cdot, \cdot)$ fulfills

$$\nabla_{\mathbf{Q}} g = \left[c(\tau, a) + \sum \Pr(\tau'|\tau, a) \min_{u \in \mathbb{A}} Q(\tau', u) - Q(\tau, a) - Q(\tau_0, a_0) \right]. \quad (8)$$

This iterative learning scheme is a gradient ascent algorithm for the maximization problem

$$\max_{\mathbf{Q}} g(\mathbf{Q}).$$

In the Q -learning algorithm, the expectation term in (8) is replaced with its noisy sample $\min_{u \in \mathbb{A}} Q_k(\tau(k+1), u)$. We take the noisy sample of $(\tau(k), a(k))$ component of $\nabla_{\mathbf{Q}} g$, i.e., $\nabla_{\mathbf{Q}} g(\tau(k), a(k)) + N_k$, where

$$N(k) = \left[c(\tau(k), a(k)) + \min_{u \in \mathbb{A}} Q(\tau(k+1), u) - Q(\tau(k), a(k)) - Q(\tau_0, a_0) \right] - \nabla_{\mathbf{Q}} g(\tau(k), a(k)).$$

Imposing the monotonicity and submodularity constraints on this optimization problem gives

$$\begin{aligned} & \max_{\mathbf{Q}} g(\mathbf{Q}) \\ & \text{s.t. } \mathbf{TQ} \geq 0. \end{aligned}$$

For this problem, we consider the following primal-dual algorithm

$$\begin{aligned} \mathbf{Q}_{k+1}(\tau(k), a(k)) &= \mathbf{Q}_k(\tau(k), a(k)) + \alpha(\nu(\tau(k), a(k))) \\ &\quad \times \left[\nabla_{\mathbf{Q}} g(\tau(k), a(k)) + N_k + [\mathbf{T}^\top \boldsymbol{\mu}_k]^{(\tau(k), a(k))} \right], \quad (9) \\ \boldsymbol{\mu}_{k+1} &= \boldsymbol{\mu}_k - \alpha(k) \mathbf{TQ}_k, \quad (10) \end{aligned}$$

where $[\mathbf{T}^\top \boldsymbol{\mu}_k]^{(\tau, a)}$ corresponds to component (τ, a) of $\mathbf{T}^\top \boldsymbol{\mu}_k$. This algorithm converges to the solution of the Bellman optimality equation as stated in the following theorem.

Theorem 3 *The structured Q-learning (9)-(10) converges to a solution of (4) with probability 1.*

Remark 5 *Standard Q-learning uses the sample average to estimate the Q-factor. One sample is used to update one state-action pair. Our proposed method utilizes the monotonicity and submodularity of the Q-factor. This fully utilizes the samples, and potentially increases the convergence performance.*

Synchronous update. The second improvement is updating synchronously. In most cases, the synchronous update is not applicable for stochastic approximation-based real-time optimal control. In our problem, however, the randomness of the state transition is independent of the state. We can run a parallel virtual model with the actual model. The virtual model keeps track of the Q-factor and the actual model takes actions according to the Q-factor stored in the virtual model. Every time after the actual model transmits, we either observe successful transmission or failure. If the transmission is successful, the Q-factor is updated as

$$Q_{k+1}(\tau, 1) = Q_k(\tau, 1) + \alpha \left(\sum_{n=0}^k a(n) \right) \left[c(\tau, 1) + \min_{u \in \mathbb{A}} Q_k(0, u) - Q_k(\tau, 1) - Q_k(\tau_0, a_0) \right], \quad \tau \in \mathbb{S}'. \quad (11)$$

If the transmission fails, the Q-factor is updated as

$$Q_{k+1}(\tau, 1) = Q_k(\tau, 1) + \alpha \left(\sum_{n=0}^k a(n) \right) \left[c(\tau, 1) + \min_{u \in \mathbb{A}} Q_k(\tau + 1, u) - Q_k(\tau, 1) - Q_k(\tau_0, a_0) \right], \quad \tau \in \mathbb{S}'. \quad (12)$$

For $a = 0$, the Q-factor is updated as

$$Q_{k+1}(\tau, 0) = Q_k(\tau, 0) + \alpha \left(k - \sum_{n=0}^k a(n) \right) \left[c(\tau, 0) + \min_{u \in \mathbb{A}} Q_k(\tau + 1, u) - Q_k(\tau, 0) - Q_k(\tau_0, a_0) \right], \quad \tau \in \mathbb{S}'. \quad (13)$$

To summarize, the update of the Q-factor can be written as

$$Q_{k+1}(\tau, a) = Q_k(\tau, a) + \alpha(i) \left[c(\tau, a) + \min_{u \in \mathbb{A}} Q_k(\tau', u) - Q_k(\tau, a) - Q_k(\tau_0, a_0) \right], \quad \tau \in \mathbb{S}', \quad a \in \mathbb{A}, \quad (14)$$

where the next state τ' can be determined according to whether the transmission succeeds or not and the parameter i in $\alpha(i)$ is

$$i = \begin{cases} \sum_{n=0}^k a(n), & \text{if } a(k) = 1, \\ k - \sum_{n=0}^k a(n), & \text{if } a(k) = 0. \end{cases}$$

With this improvement, the randomness in the action selection is not necessary because every state-action pair is now updated simultaneously (Remark 3).

As the synchronous version is a standard Q-learning algorithm satisfying the assumptions made in [24], its convergence automatically holds.

Remark 6 *The structural learning we introduce above can also be used for the synchronous version as the source of*

noise and the associated limiting ordinary differential equation (ODE) are the same.

Remark 7 *The randomization parameter ε can be set to zero for the synchronous algorithm. Therefore, the Q-factor converges under the synchronous algorithm to the actual value of the model with original probability transition law $\Pr(\tau'|\tau, a)$. From the Bellman optimality equation, we can see that the average cost is a continuous function of the Q-factor. By the continuous mapping theorem [25, Theorem 3.2.4], the average cost also converges to the optimal one.*

B. Problem 2 with Stochastic Approximation

From the structural results for Problem 2, we know that, for each communication budget b , there exists a $\lambda^*(b)$ such that the optimal total cost with communication cost being $\lambda^*(b)$ for Problem 1 equals to the optimal average estimation error under communication budget b plus $\lambda^*(b)b$. We use a gradient-based update for the communication cost to obtain $\lambda^*(b)$ as follows

$$\lambda_{k+1} = \lambda_k + \beta(k) (a(n) - b), \quad (15)$$

where $\beta(k)$ is the step size at time k . From previous analysis, we know that the optimal randomized policy for Problem 2 is also an optimal policy for Problem 1 with communication cost being $\lambda^*(b)$.

Combing (14)-(15)⁴, the iterative learning algorithm for the Problem 2 is

$$Q_{k+1}(\tau, a) = Q_k(\tau, a) + \alpha \left[\sum_{n=0}^k a(n) \right] \left[c_{\lambda_k}(\tau, a) + \min_{u \in \mathbb{A}} Q_k(\tau', u) - Q_k(\tau, a) - Q_k(\tau_0, a_0) \right], \quad \tau \in \mathbb{S}', \quad a \in \mathbb{A},$$

$$\lambda_{k+1} = \lambda_k + \beta(k) (a(k) - b),$$

where the subscript λ_k in $c_{\lambda_k}(\cdot, \cdot)$ is used to emphasize the dependence of the one stage cost on the communication cost. The step sizes $\alpha(\cdot)$ and $\beta(\cdot)$ satisfy

$$\sum_n \alpha(n) = \sum_n \beta(n) = \infty, \quad \sum_n (\alpha(n))^2 + (\beta(n))^2 < \infty, \quad (16)$$

$$\text{and } \lim_{n \rightarrow \infty} \frac{\beta(n)}{\alpha(n)} = 0. \quad (17)$$

The last requirement imposes that that the communication cost λ is updated in a slower time scale. This is called a quasi-static condition because the updates of λ seem “static” when Q is updating. By using either the standard asynchronous Q-learning or its improved version discussed before, for every “static” cost λ , the vector Q converges to the corresponding solution of the Bellman optimality equation (4). Consequently, the scheduling policy will also converge to the optimal one. If the algorithm over the slower time scale also converges, the two-time scale algorithm converges. This result is stated in the following theorem.

⁴Such combination is also applicable for the original asynchronous version and the structural learning. Convergence analysis of these are the same.

Theorem 4 *The two-time scale Q-learning (14)-(15) converges with probability 1. The asymptotic communication cost $\lambda_\infty = \lambda^*$ and the Q-factor are the solutions to the Bellman optimality equation*

$$Q(\tau, a) = c(\tau, a) + \sum_{\tau_+} \min_{a \in \mathcal{A}} Q(\tau_+, a) \Pr(\tau_+ | \tau, a) - \mathcal{J}^*,$$

with $c(\tau, a) = \text{Tr}(P(\tau)) + \lambda^* a$. The optimal policy $f(\lambda^*)$ satisfies $J_r(f(\lambda^*)) = b$.

C. Problem 1 and Problem 2 with Parameter Learning

The stochastic approximation method iteratively updates the Q-factor. In the sensor scheduling problem, only the transmission success probability is unknown. If we can sample the channel condition infinitely many times, the empirical success probability converges to the actual success probability almost surely by the strong law of large numbers. Based on this observation, we develop the direct learning schemes for Problems 1 and 2, respectively. Different from previous sections, we discuss Problem 2 first.

1) *Problem 2*: Thanks to Theorem 2, the optimal policy in this case only depends on the channel condition r_s and the communication budget b . Once we know the channel condition r_s , the optimal threshold θ^* and switching probability r_{θ^*} can be analytically computed as shown in Corollary 1. We propose the following learning method. Let $N_s(k)$ and $N_f(k)$ denote the number of successful transmission and failed transmission at time k . The maximum likelihood estimate of r_s is

$$\hat{r}_s(k) = \frac{N_s(k)}{N_s(k) + N_f(k)}. \quad (18)$$

We use \hat{r}_s instead of r_s to determine the corresponding optimal scheduling policy as

$$\theta^*(\hat{r}_s(k)) = \lfloor \frac{1}{\hat{r}_s(k)b} - \frac{1}{\hat{r}_s(k)} \rfloor, \quad (19)$$

$$r_{\theta^*}(\hat{r}_s(k)) = \theta^*(\hat{r}_s(k)) + 1 + \frac{b-1}{b\hat{r}_s(k)}. \quad (20)$$

If $\hat{r}_s = 0$, the corresponding threshold is defined to be infinity. This can be avoided through proper initialization. In the initialization phase, we keep transmitting until $N_s(k) = 1$. After that, we use the randomized threshold policy $(\theta^*(\hat{r}_s(k)), r_{\theta^*}(\hat{r}_s(k)))$ to determine the scheduling policy while learning r_s .

This scheme separates the parameter estimation and the optimal control problem. Its convergence is immediate.

Theorem 5 *The schedule policy, which uses (18)-(20) converges almost surely to the optimal policy. Moreover, also the average estimation error converges to the optimal average estimation error almost surely.*

2) *Problem 1*: In this case, the estimation of r_s and its initialization remains the same as in Problem 2. As shown before, there is no analytic expression for the optimal policy in Problem 1. For every given channel condition estimate $\hat{r}_s(k)$, we need to solve the Bellman optimality equation. As the initialization guarantees that $\hat{r}_s(k)$ will not be zero, the corresponding policy is a finite-threshold policy, which ensures

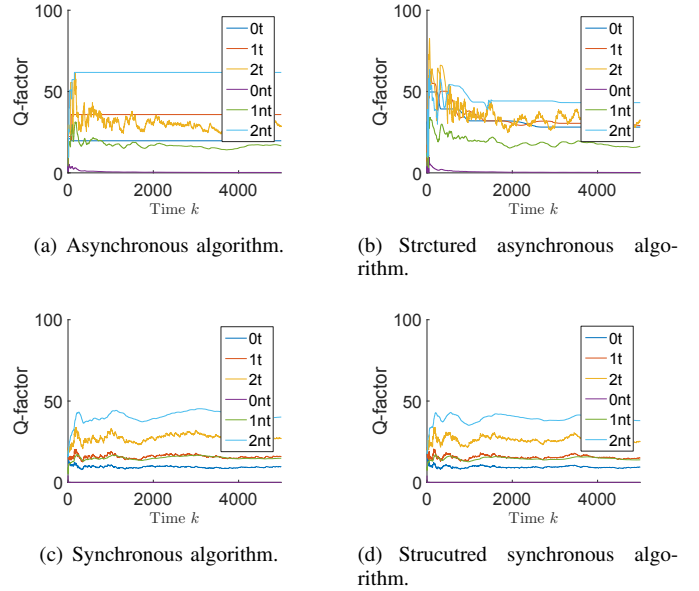


Fig. 3: Q-factor in the learning process for Problem 1.

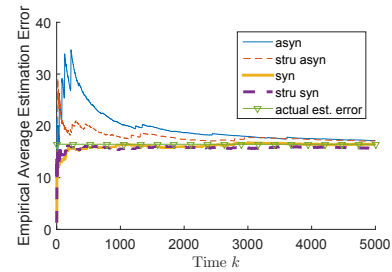


Fig. 4: Average estimation error in Problem 1.

that the trial does not stop. However, the computation overhead is large as the Bellman optimality equation needs to be solved at each time step. We present a numerical example to illustrate the computational issue in this scenario.

Remark 8 *To summarize this section, the parameter learning method is suitable for Problem 2. Meanwhile, the stochastic approximation causes less computation overhead than the parameter learning method for Problem 1.*

V. NUMERICAL EXAMPLE

In this section, we illustrate the convergence of our algorithms with a specific example. We consider the following system:

$$x(k+1) = \begin{bmatrix} 1.2 & 1 \\ 0 & 0.8 \end{bmatrix} x(k) + w(k), \quad y(k) = x(k) + v(k),$$

where

$$\mathbb{E}[w(k)w(k)^\top] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbb{E}[v(k)v(k)^\top] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The successful transmission rate is $r_s = 0.7$.

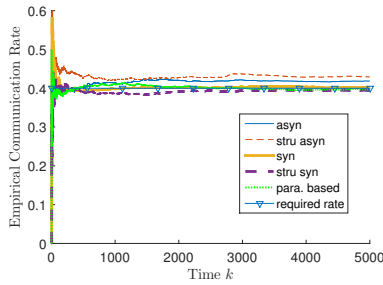


Fig. 5: Communication rate in Problem 2.

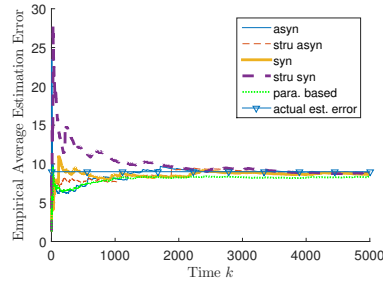


Fig. 6: Average estimation error in Problem 2.

We first consider the Problem 1. We set the communication cost to be $\lambda = 20$ per transmission. We compare four algorithms: the original asynchronous algorithm (6), the structure-based asynchronous algorithm in (9)-(10), the synchronous algorithm (14) and the structure-based synchronous algorithm (a combination of the structural learning and the synchronous algorithm). The learning processes of all algorithms converge as shown in Fig. 3. The label nt (nnt) stands for transmit (not transmit) when $\tau = n$. We can see that the Q -factor in the original asynchronous algorithm does not satisfy the monotonicity condition. By comparing (a) and (b), we can see that the structure-based learning ensures monotonicity and submodularity of the Q -factor. The average cost, which is the empirical sum of the time average of the estimation error and the average communication cost, is shown in Fig. 4. The computation detail of the empirical estimation error and the empirical communication rate at time k is available in online version [27]. For comparison, we also provide the true value of cost. We can see that all four algorithms converge to the true value. As the structure-based learning imposes the monotonicity and the submodularity of the Q -factor, the average estimation error in the structure-based asynchronous version converges to the true value faster than the basic asynchronous version. Moreover, the synchronous algorithms have much faster convergence rate than the asynchronous ones as expected.

We then consider the Problem 2. We set the desired communication rate to be $b = 0.4$. In addition to the four algorithms compared for Problem 1, we include the parameter-based learning algorithm in (18)-(20). We show the results of the communication rate and the average cost in Figs. 5 and 6. The empirical value of the communication rate and the average estimation error are computed in the same way as before. It can be seen that the four stochastic approximation-

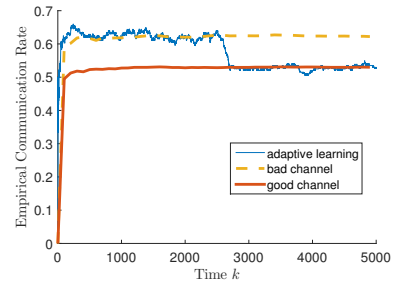


Fig. 7: The learning method is adaptive to time-varying channel condition.

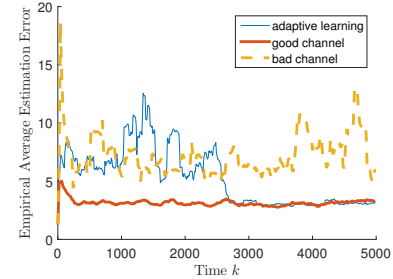


Fig. 8: Average estimation error of three scheduling policies.

based algorithms have comparable performances in terms of the communication rate. Moreover, their empirical average estimation errors are comparable to that of the direct parameter learning method.

We next illustrate the effectiveness of the learning method for a time-varying channel. We consider Problem 1 and set the communication cost to be $\lambda = 10$ per transmission. The channel condition is initially good with a successful transmission rate being $r_s = 0.9$. At iteration time step $k = 2500$, the successful transmission rate decreases to $r_s = 0.6$. We compare the transient performance of the synchronous structured learning method with the performance under constantly “good” or “bad” conditions. Figs. 7 and 8 show that the learning method is adaptive to time-varying channel conditions as the empirical communication rate and the empirical average estimation error converge to the optimal value. The computation of the empirical values of the communication rate and the average estimation error is computed using a sliding window (details available in online version [27]). The solid blue lines are the adaptive learning method, while the solid red line and the dotted orange line are under “good” and “bad” channel conditions, respectively. Note that as the entropy of a Bernoulli random variable with mean being 0.6 is greater than that with mean being 0.9, the empirical average estimation error under bad channel has a greater fluctuation.

We mentioned in the introduction that adaptive control methods can be computationally intense. We show how the Q -learning-based methods outperform direct parameter learning. In particular, we consider the remote estimation of the same dynamic process as previous examples for Problem 1. We simultaneously run the synchronous Q -learning with the parameter learning algorithm. In the parameter learning algorithm, we first estimate r_s based on the history of transmission

success and failures, and then calculate the optimal policy of the corresponding MDP using the relative value iteration. As the relative value iteration fails to converge within finite time, we forcefully stop the algorithm within 1, 5 and 50 iterations. The time-averaged costs of each algorithm is presented in Fig. 9. The label “MDP-x” stands for the parameter learning method with x iterations at each time step. If only one

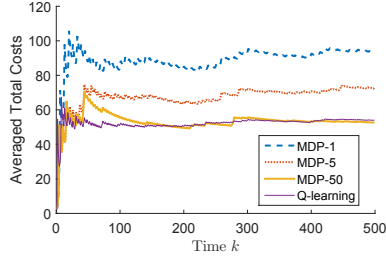


Fig. 9: Performance comparison between the Q -learning and the parameter learning. The label “MDP-x” stands for the parameter learning method with x allowable iterations at each time step.

iteration is allowed for the MDP algorithm at each time step, the performance of the parameter learning is much worse than the Q -learning. If the number of iterations increases, the performance improves. The performance of the parameter learning is close to the Q -learning for 50 iterations at each time step. The computation overhead of the Q -learning is equivalent to one iteration of the relative value iteration for MDP. In this particular example, it costs approximately 50 times more computation resources for the parameter learning method to reach the same performance as that of the Q -learning.

VI. CONCLUSION

We considered scheduling for remote state estimation under costly communication and constrained communication, respectively. By using dynamic programming, we established two frameworks to tackle the problems when the channel condition is known. We utilized these results to develop revised algorithms to improve the convergence of the standard asynchronous stochastic approximation algorithm. In addition, as the randomness of the state transition was observed to be independent of the state, we developed a simple synchronous algorithm for the costly communication problem. Although the stochastic approximation method can be used for the constrained communication problem, the parameter learning method possesses faster convergence speed and is easier to implement. For future work, the framework can be extended to a general channel such as a Markovian channel and scheduling multiple sensors.

APPENDIX

A. Proof of Lemma 1 and 2

The proof of Lemma 1 relies on the vanishing discount approach [26, Theorem 5.5.4]. Details are omitted and available in online version [27].

Similar to the existence of an optimal stationary policy, the proof of Lemma 2 relies on a discounted cost setup for the same problem. For a constant $0 < \gamma < 1$, we want to minimize the discounted total cost $\sum_{k=0}^{\infty} \gamma^k \mathbb{E}[c(\tau(k), a(k))]$. The optimal policy satisfies the Bellman optimality equation for the discounted cost problem

$$V_{\gamma}(\tau) = \min_{a \in \mathbb{A}} \left[c(\tau, a) + \gamma \sum_{\tau_+} V_{\gamma}(\tau_+) \Pr(\tau_+ | \tau, a) \right].$$

Note that the right hand side of the discounted Bellman optimality equation is a mapping of $V_{\gamma}(\tau), \tau \in \mathbb{S}$. Define such mapping as the Bellman operator on $V_{\gamma}(\tau), \tau \in \mathbb{S}$ as

$$\mathcal{T}(V_{\gamma}) = \min_{a \in \mathbb{A}} \left[c(\tau, a) + \gamma \sum_{\tau_+} V_{\gamma}(\tau_+) \Pr(\tau_+ | \tau, a) \right].$$

The discounted setup is considered here because the Bellman operator \mathcal{T}_{γ} for the discounted cost problem is a contraction mapping w.r.t. to a norm (Details available in [27]). Since there is a unique fixed point for a contraction mapping iteration, which enables us to use an induction-based method to prove the monotonicity of the discounted value function. Moreover, as the six condition in Lemma 1 hold, we have $V(\tau) = \lim_{\gamma \uparrow 1} V_{\gamma}(\tau)$. The details are omitted due to space limitation and are available in the online version [27].

B. Proof of Lemmas 3 and 4

1) *Proof of Lemma 3:* The monotonicity of the Q -factor holds because

$$\begin{aligned} & Q(\tau, a) - Q(\tau', a) \\ & \geq \sum_{\tau_+} \min_{a \in \mathbb{A}} Q(\tau_+, a) \Pr(\tau_+ | \tau, a) - \sum_{\tau'_+} \min_{a \in \mathbb{A}} Q(\tau'_+, a) \Pr(\tau'_+ | \tau', a) \\ & = \sum_{\tau_+} V(\tau_+) \Pr(\tau_+ | \tau, a) - \sum_{\tau'_+} V(\tau'_+) \Pr(\tau'_+ | \tau', a) \geq 0. \end{aligned}$$

This completes the proof.

2) *Proof of Lemma 4:* Since $a, a' \in \mathbb{A} = \{0, 1\}$, let $a = 1$ and $a' = 0$. We can compute that

$$\begin{aligned} & Q(\tau, 1) - Q(\tau, 0) - Q(\tau', 1) + Q(\tau', 0) \\ & = r_s \min_{a \in \mathbb{A}} Q(0, a) + (1 - r_s) \min_{a \in \mathbb{A}} Q(\tau + 1, a) - \min_{a \in \mathbb{A}} Q(\tau + 1, a) \\ & \quad - r_s \min_{a \in \mathbb{A}} Q(0, a) - (1 - r_s) \min_{a \in \mathbb{A}} Q(\tau' + 1, a) + \min_{a \in \mathbb{A}} Q(\tau' + 1, a) \\ & = r_s \left[\min_{a \in \mathbb{A}} Q(\tau' + 1, a) - \min_{a \in \mathbb{A}} Q(\tau + 1, a) \right] \\ & = r_s (V(\tau' + 1) - V(\tau + 1)) \leq 0, \end{aligned}$$

which completes the proof.

C. Proof of Theorem 1

This argument is equivalent to that, if $Q(\tau, 1) \leq Q(\tau, 0)$, then $Q(\tau', 1) \leq Q(\tau', 0)$ for $\tau \leq \tau'$. Since $V(\tau + 1) \leq V(\tau' + 1)$, we obtain

$$\begin{aligned} Q(\tau', 1) - Q(\tau', 0) & = \lambda + r_s V(0) - r_s V(\tau' + 1) \\ & \leq \lambda + r_s V(0) - r_s V(\tau + 1) \\ & = Q(\tau, 1) - Q(\tau, 0) \leq 0. \end{aligned}$$

This completes the proof.

D. Proof of Lemma 5

As the problem is feasible in the sense that there exists $f \in \mathbb{F}$ such that $J_r(f) < b$. The optimal solution (f^*, λ^*) to the saddle point problem should satisfy

$$\lambda^*(J_r(f^*) - b) = 0.$$

If $\lambda^* = 0$, the optimal scheduling policy is always to transmit, which violates the constraint $J_r(f^*) \leq b$. Therefore, $\lambda^* \neq 0$, and $J_r(f^*) = b$ accordingly.

E. Proof of Theorem 2

The proof relies on concavity and continuity of $J(\theta, \lambda) := J_e(\theta) + \lambda J_r(\theta)$ w.r.t. λ along with a sufficient optimality condition of constrained optimization [28, Theorem 1, Sec 8.4]. Details are available online.

F. Proof of Corollary 1 and Theorem 3

Corollary 1 follows from straightforward computation. Details are omitted and available in online version [27]. The proof of Theorem 3 relies on stochastic approximation in [24], [29]. Details are available in online version.

G. Proof of Theorem 4

The theorem can be proven by showing that the two-time scale iteration converges to the solution of the saddle point problem in (5). This is equivalent to $\lambda_\infty \in \arg \max_\lambda J_e(f^*) + \lambda J_r(f^*)$ and $f^* \in \arg \min_f J_e(f^*) + \lambda^* J_r(f^*)$, where f^* is the policy induced by $\mathbf{Q}_\infty(\cdot, \cdot)$.

Similar to the stochastic approximation in one time scale, the two-time scale approach converges to the constrained communication problem if the two types of conditions in [30, Theorem 3.4] holds. The first type relates to the noise and the second type relates to the stability of the limit ODE. Based on analysis in Problem 1, the remaining task is to check the asymptotic stability of the ODE in the slower time scale.

A major difficulty lies in that the time average limit of the right hand side of (15) is not an ODE but a differential inclusion as

$$\dot{\lambda} \in J_r(\lambda) - b$$

as $J_r(\lambda)$ is discontinuous at countably many λ . Nevertheless, according to [31, Lemma 4.3], the limit ODE can be characterized by the following ODE instead

$$\dot{\lambda}(t) = \frac{\partial}{\partial \lambda} J^*(\lambda(t)),$$

where $J^*(\lambda(t)) = \inf_f J_e(f) + \lambda(t) J_r(f)$. The \inf_f can be achieved as \mathbf{Q} in the faster time scale converges according to previous analysis. The trajectory of $\lambda(t)$ is thus the solution to the following integral equation

$$\lambda(t) = \lambda(0) + \int_0^t J^*(\lambda(s)) ds.$$

This interpretation conquers the discontinuity problem as the set of discontinuity has a zero measure. By the chain rule, the trajectory of the total cost $J^*(t)$ satisfies

$$\dot{J}^*(t) = \frac{\partial}{\partial \lambda} J^*(\lambda) \cdot \dot{\lambda}(t) = \left| \frac{\partial}{\partial \lambda} J^*(\lambda) \right|^2 > 0,$$

for almost all t except when $\lambda(t) \in \arg \max J^*(\cdot)$. This proves that $\lambda(t)$ converges to $\arg \max J^*(\cdot)$, i.e.,

$$J_e(f^*) + \lambda_\infty (J_r(f^*) - b) = J_e(f^*) + \lambda^* (J_r(f^*) - b),$$

where λ^* is the saddle point solution to (5) and

$$f^*(\tau) \in \arg \min_a \left\{ c(\tau, a) + \sum_{\tau'} \min_{u \in \mathbb{A}} \mathbf{Q}(\tau', u) \Pr(\tau' | \tau, a) - \mathcal{J}^* \right\}.$$

This completes the proof.

H. Proof of Theorem 5

The proof relies on the continuous mapping theorem [25, Theorem 3.2.4]. Details are available in online version.

REFERENCES

- [1] D. I. Shuman, A. Nayyar, A. Mahajan, Y. Goykhman, K. Li, M. Liu, D. Teneketzis, M. Moghaddam, and D. Entekhabi, "Measurement scheduling for soil moisture sensing: From physical models to optimal control," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1918–1933, 2010.
- [2] L. Wang and Y. Xiao, "A survey of energy-efficient scheduling mechanisms in sensor networks," *Mobile Networks and Applications*, vol. 11, no. 5, pp. 723–740, 2006.
- [3] O. C. Imer and T. Basar, "Optimal estimation with limited measurements," in *Proc. 44th IEEE Conf. Decision and Control and the European Control Conf.* IEEE, 2005, pp. 1029–1034.
- [4] G. M. Lipsa and N. C. Martins, "Remote state estimation with communication costs for first-order lti systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2013–2025, 2011.
- [5] J. Wu, Q.-S. Jia, K. H. Johansson, and L. Shi, "Event-based sensor data scheduling: Trade-off between communication rate and estimation quality," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 1041–1046, 2013.
- [6] S. Trimpe and R. D'Andrea, "Event-based state estimation with variance-based triggering," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3266–3281, 2014.
- [7] M. Nourian, A. S. Leong, S. Dey, and D. E. Quevedo, "An optimal transmission strategy for kalman filtering over packet dropping links with imperfect acknowledgements," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 3, pp. 259–271, 2014.
- [8] J. Chakravorty and A. Mahajan, "Fundamental limits of remote estimation of autoregressive Markov processes under communication constraints," *IEEE Transactions on Automatic Control*, vol. 62, no. 3, pp. 1109–1124, 2017.
- [9] X. Ren, J. Wu, K. H. Johansson, G. Shi, and L. Shi, "Infinite horizon optimal transmission power control for remote state estimation over fading channels," *IEEE Transactions on Automatic Control*, vol. 63, no. 1, pp. 85–100, 2018.
- [10] X. Gao, E. Akyol, and T. Başar, "Optimal communication scheduling and remote estimation over an additive noise channel," *Automatica*, vol. 88, pp. 57–69, 2018.
- [11] V. K. Lau and Y.-K. R. Kwok, *Channel-adaptive Technologies and Cross-layer Designs for Wireless Systems with Multiple Antennas: Theory and Applications*. John Wiley & Sons, 2006.
- [12] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2005.
- [13] C. J. C. H. Watkins, "Learning From Delayed Rewards," Ph.D. dissertation, King's College, 1989.
- [14] D. P. Bertsekas and J. N. Tsitsiklis, "Neuro-dynamic programming: an overview," in *34th IEEE Conference on Decision and Control*, vol. 1. IEEE, 1995, pp. 560–564.
- [15] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[16] P. R. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification, and Adaptive Control*. SIAM, 1986, vol. 75.

[17] O. Hernández-Lerma, *Adaptive Markov Control Processes*. Springer Science & Business Media, 1989, vol. 79.

[18] E. Altman and A. Schwartz, “Adaptive control of constrained Markov chains,” *IEEE Transactions on Automatic Control*, vol. 36, no. 4, pp. 454–462, 1991.

[19] V. S. Borkar, “Convex analytic methods in Markov decision processes,” in *Handbook of Markov Decision Processes*. Springer, 2002, pp. 347–375.

[20] V. Krishnamurthy and D. V. Jonin, “Structured threshold policies for dynamic sensor scheduling—a partially observed Markov decision process approach,” *IEEE Transactions on Signal Processing*, vol. 55, no. 10, pp. 4938–4957, 2007.

[21] A. S. Leong, S. Dey, and D. E. Quevedo, “On the optimality of threshold policies in event triggered estimation with packet drops,” in *2015 European Control Conference*. IEEE, 2015, pp. 927–933.

[22] Z. Ren, P. Cheng, J. Chen, L. Shi, and H. Zhang, “Dynamic sensor transmission power scheduling for remote state estimation,” *Automatica*, vol. 50, no. 4, pp. 1235–1242, 2014.

[23] E. Altman, *Constrained Markov Decision Processes*. CRC Press, 1999.

[24] J. Abounadi, D. Bertsekas, and V. S. Borkar, “Learning algorithms for Markov decision processes with average cost,” *SIAM Journal on Control and Optimization*, vol. 40, no. 3, pp. 681–698, 2001.

[25] R. Durrett, *Probability: Theory and Examples*. Cambridge University Press, 2010.

[26] O. Hernández-Lerma and J. B. Lasserre, *Discrete-time Markov control processes: basic optimality criteria*. Springer Science & Business Media, 1995, vol. 30.

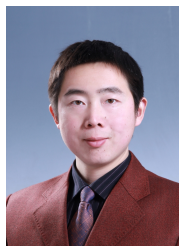
[27] S. Wu, X. Ren, Q.-S. Jia, K. H. Johansson, and L. Shi, “Learning optimal scheduling policy for remote state estimation under uncertain channel condition,” *arXiv preprint arXiv:1810.09820*, 2018.

[28] D. G. Luenberger, *Optimization by Vector Space Methods*. John Wiley & Sons, 1997.

[29] V. S. Borkar and S. P. Meyn, “The O.D.E. method for convergence of stochastic approximation and reinforcement learning,” *SIAM Journal on Control and Optimization*, vol. 38, no. 2, pp. 447–469, 2000.

[30] S. Bhatnagar, H. Prasad, and L. Prashanth, *Stochastic Recursive Algorithms for Optimization: Simultaneous Perturbation Methods*. Springer, 2012, vol. 434.

[31] V. S. Borkar, “An actor-critic algorithm for constrained Markov decision processes,” *Systems & control letters*, vol. 54, no. 3, pp. 207–213, 2005.

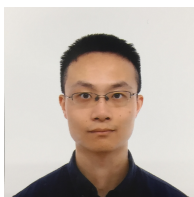


Qing-Shan Jia (S’02-M’06-SM’11) received the B.S. and Ph.D. degrees in automation and control science and engineering from Tsinghua University, Beijing, China, in 2002 and 2006, respectively. He was a visiting scholar in Harvard, HKUST, and MIT in 2006, 2010, and 2013, respectively. He is currently a tenured associate professor with the Center for Intelligent and Networked Systems, Department of Automation, Beijing National Research Center for Information Science and Technology, Tsinghua University. His current research interests include theories and applications of discrete event dynamic systems, simulation-based performance evaluation, and optimization of cyber physical energy systems.

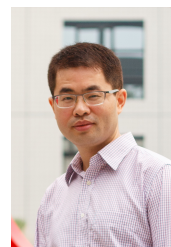
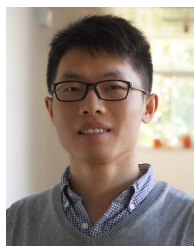


Karl Henrik Johansson is Professor at the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology. He received MSc and PhD degrees from Lund University. He has held visiting positions at UC Berkeley, Caltech, NTU, HKUST Institute of Advanced Studies, and NTNU. His research interests are in networked control systems, cyber-physical systems, and applications in transportation, energy, and automation. He has served on the IEEE Control Systems Society Board of Governors, the IFAC Executive Board, and the European Control Association Council. He has received several best paper awards and other distinctions from IEEE and ACM. He has been awarded Distinguished Professor with the Swedish Research Council and Wallenberg Scholar with the Knut and Alice Wallenberg Foundation. He has received the Future Research Leader Award from the Swedish Foundation for Strategic Research and the triennial Young Author Prize from IFAC. He is Fellow of the IEEE and the Royal Swedish Academy of Engineering Sciences, and he is IEEE Distinguished Lecturer.

Shuang Wu is a senior researcher at Noah’s Ark Lab Huawei. He received the B.Eng. (Hons) degree in the Department of Mechanical Engineering from Zhejiang University in 2015 and the Ph.D. degree in the Department of Electronic and Computer Engineering from Hong Kong University of Science and Technology in 2019. His research interests include learning-based methods in control systems, networked control systems, distributed state estimation and wireless sensor networks.



Xiaoqiang Ren received the B.E. degree in the Department of Control Science and Engineering from Zhejiang University, Hangzhou, China, in 2012 and the Ph.D. degree in the Department of Electronic and Computer Engineering from Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2016. He is currently a professor at the School of Mechatronic Engineering and Automation, Shanghai University, China. Prior to this, he was a postdoctoral researcher in the Hong Kong University of Science and Technology in 2016, Nanyang Technological University from 2016 to 2018, and KTH Royal Institute of Technology from 2018 to 2019. His research interests include security of cyber-physical systems, sequential decision, and networked estimation and control.



Ling Shi received the B.S. degree in electrical and electronic engineering from Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2002 and the Ph.D. degree in Control and Dynamical Systems from California Institute of Technology, Pasadena, CA, USA, in 2008. He is currently an associate professor at the Department of Electronic and Computer Engineering, and the associate director of the Robotics Institute, both at the Hong Kong University of Science and Technology. His research interests include cyber-physical systems security, networked control systems, sensor scheduling, event-based state estimation, and exoskeleton robots. He is a senior member of IEEE. He served as an editorial board member for The European Control Conference 2013-2016. He was a subject editor for International Journal of Robust and Nonlinear Control (2015-2017). He has been serving as an associate editor for IEEE Transactions on Control of Network Systems from July 2016, and an associate editor for IEEE Control Systems Letters from Feb 2017. He also served as an associate editor for a special issue on Secure Control of Cyber Physical Systems in the IEEE Transactions on Control of Network Systems in 2015-2017. He served as the General Chair of the 23rd International Symposium on Mathematical Theory of Networks and Systems (MTNS 2018).