IEEE Access
The journal for rapid open access publishing

IEEE Xplore
Digital Library
http://ieeexplore.ieee.org/Xplore

Reinforcement Learning-based Adaptive Transmission in Time-Varying Underwater Acoustic Channels

CHAOFENG WANG, ZHAOHUI WANG, *Member, IEEE*, WENSHENG SUN, AND DANIEL R. FUHRMANN, *Fellow, IEEE*

***Abstract*—This work studies adaptive transmission in an underwater acoustic (UWA) point-to-point communication system that operates on an epoch-by-epoch basis for a long term. A fixed amount of information bits periodically arrive at the transmitter data queue, and wait for transmission via a number of packets within each epoch. To trade off energy consumption with transmission latency, the transmitter decides the transmission action at the beginning of each epoch, including to transmit or not, the transmission power and the modulation-and-coding parameters, based on the data queue status and the predicted channel conditions in the current and future epochs. To describe both the fast fading and the large-scale shadowing of UWA channels, the channel within each epoch is characterized by a compound Nakagami-lognormal distribution, and the evolution of the distribution parameters is modeled as an unknown Markov process. Given that the channel can only be observed during active transmissions, we formulate the adaptive transmission problem as a partially observable Markov decision process (POMDP), and develop an online algorithm in a model-based reinforcement learning (RL) framework. The algorithm recursively estimates the channel model parameters, tracks the channel dynamics, and computes the optimal transmission action that minimizes a long-term system cost. Emulated results based on channel measurements from two field experiments demonstrate that the proposed algorithm achieves decent performance relative to a benchmark method that assumes perfect and non-causal channel knowledge.**

***Index Terms*—Adaptive transmission, underwater acoustic channels, energy efficiency, model-based reinforcement learning.**

I. Introduction

UNDERWATER acoustic (UWA) communication is the key technique for wireless information transfer in a wide range of aquatic applications, such as ocean observation, ecosystem health monitoring, and tactical surveillance [1]. Due to the high deployment cost, the lifespan of underwater systems varies from months to years. For instance, underwater monitoring systems, such as scientific data collection systems, could be mounted at the water bottom for months to collect parameters of interest, and large-scale ocean observation systems, such as the NEPTUNE and VENUS ocean observatories [2] and the Ocean Observatory Initiative (OOI) [3], could have projected lifespans of more than 20 years. On the other hand, underwater nodes are often powered by batteries, and battery replacement and recharging are time-consuming and costly. Energy-efficient operation is critical for system longevity.

This work considers a long-term operating underwater system with deterministic data arrivals (e.g., periodic data collection systems), and studies energy-efficient acoustic transmission that adapts the transmission schedule and the transmission parameters, including the transmission power and the modulation-and-coding parameters, to the system state (e.g., the transmitter data queue length) and the current and future predicted channel conditions, with a goal of minimizing a long-term average cost. The UWA channel exhibits both small-scale fast fading and long-term large-scale shadowing. Adapting transmission strategy to the channel dynamics could yield considerable energy saving.

The channel-aware transmission to trade off energy consumption with information delivery latency has been extensively studied in terrestrial radio communications. Particularly for correlated fading channels, most of existing works model the channel as a finite-state Markov chain (FSMC) with known transitional probabilities, and formulate the problem as a Markov Decision Process (MDP) to determine the control variables, such as the transmission schedule, the transmission power, and the modulation-and-coding parameters, based on the channel state and the communication system state (e.g., the data queue length, the incoming traffic rate, and the packet delay constraint). Given that the MDP is generally computationally intractable to solve, special structures of the optimal policy are identified and exploited to find the optimal or near-optimal solution [4]–[7]. However, the channel state transition probability and the traffic statistics could be hard to obtain in practice. Some works propose to solve the MDP *online* using reinforcement learning (RL) [8], where model-free RL methods (e.g., Q-learning, and the actor-critic algorithm) are used to learn from past experiences (namely, how to map "situations" to "actions") without explicit modeling of the channel and/or traffic dynamics [9]–[13]. Recent applications of RL in radio-frequency networks include stochastic power control for energy harvesting systems [14], [15], data scheduling and admission control for backscatter sensor networks [16], and rate and mode adaptation for Wi-Fi/LTE-U coexistence [17].

Compared to radio networks, studies on energy-efficient transmission in UWA networks have been limited. At the physical layer, relevant research includes adapting the transmission power, the frequency band, and the modulation-and-

Department of Electrical and Computer Engineering, Michigan Technological University, 1400 Townsend Drive, Houghton, MI 49931-1295, USA

C. Wang, Z.-H. Wang, and W. Sun are partially supported by NSF grants CNS-1551067 and ECCS-1651135 (CAREER).

Corresponding author: Z.-H. Wang (e-mail: zhaohuiw@mtu.edu).

Digital Object Identifier: 10.1109/ACCESS.2018.2784239

coding parameters to channel dynamics [18]–[21]. At the link layer, assuming a two-state FSMC channel model with known transition probabilities and accounting the non-negligible cost of channel probing, energy-efficient transmission scheduling with partial and discontinuous channel state information (CSI) is studied in [22]. The transmission scheduling is formulated as a dynamic programming problem, and different ways of providing the CSI from the receiver are examined. The above work is extended in [23] when only partial data queue state information is available. In [24], the RL is introduced to optimize the parameters in a slotted Carrier Sensing Multiple Access (slotted CSMA) protocol. Assuming a binary symmetric channel (BSC) with unknown transition probabilities, the model-free RL (Q-learning augmented by virtual experience and state-action aggregation) was introduced in [25] to adapt the link-layer transmission schedule and transmission parameters to the channel dynamics. Q-learning has also been used for designing routing protocols [26] with an aim to balance the workload among network nodes and to prolong the network lifetime.

For long-term operating underwater systems, the UWA channel exhibits both fast fading and large-scale shadowing; see field experiment observations in, e.g., [27]–[29]. Data analysis of different field experiments revealed that the fast fading could follow Rayleigh, Rician, Nakagami-m, or compound-K distributions; see [30] and references therein. Based on field measurements, a lognormal model was suggested for large-scale shadowing [31], [32]. Furthermore, the fading and shadowing statistics could change continuously over time; for instance, channel stationarity over an average of three-minute-long interval [33], nonstationarity and cyclostationarity [30] have been observed in different field experiments.

Existing solutions with the FSMC channel model assumption may not work well for adaptive transmission in long-term operating UWA systems. Specifically, the large channel dynamics require a sufficient number of discrete channel states for an adequate description of the channel behavior. Additionally, the FSMC parameters could change continuously over time. The high-dimensionality of the channel state space and the short-term channel stationarity could prevent model-free RL methods from convergence, which eventually leads to degraded performance.

In this work, we introduce a *continuous* channel model to describe the temporal dynamics of UWA channels, and adopt a model-based RL framework to determine the transmission strategy with the aim of optimizing a long-term system performance measure. Specifically, to better capture the channel variation over a long term, we introduce a compound Nakagami-lognormal distribution to characterize the channel fast fading and the large-scale shadowing, and model the evolution of the distribution parameters as a first-order Markov process. Based on the above channel model, the model-based RL framework is employed for adaptive transmission. The framework has two components: channel model estimation and online planning. Following the maximum likelihood principle and the expectation-maximization concept [34], an algorithm is developed to recursively estimate the channel model parameters and predict the channel state based on newly obtained channel measurements. The online planning is then performed via a Monte Carlo sampling method which finds a near-optimal transmission strategy through constructing an online state-action tree.

The proposed algorithms are validated using data sets collected from two experiments, one held off the coast of Martha's Vineyard, Massachusetts, in 2008, and the other held in the ice-covered Keweenaw Waterway near Michigan Tech, Michigan, in 2014. The experimental results show that: 1) the recursive channel estimation method yields decent performance on tracking the UWA channel dynamics; and 2) the model-based RL algorithm achieves performance close to a genie-aided method that assumes perfect and non-causal channel knowledge.

To the best of our knowledge, this is the first attempt that adopts the model-based RL framework for adaptive transmission in long-term operating UWA systems, where the channel statistical parameters in continuous spaces are explicitly learned from past transmissions.

The rest of the paper is organized as follows. The system model is presented in Section II. The model-based RL algorithm for adaptive transmission is developed in Section III. The Monte Carlo sampling method for online planning is presented in Section IV. A recursive algorithm for channel model estimation and channel tracking is described in Section V. Evaluation of the proposed algorithm is included in Section VI. Conclusions are drawn in Section VII.

Notation: Bold upper case letters and lower case letters are used to denote matrices and column vectors, respectively. $\mathbf{A}^{\mathrm{T}}$ denotes the transpose of matrix $\mathbf{A}$. $[\mathbf{a}]_m$ denotes the mth element of vector $\mathbf{a}$. $|\mathcal{A}|$ denotes the cardinality of set $\mathcal{A}$. ∇_a denotes the derivative w.r.t. a.

II. System Model and Problem Formulation

A. System Description

This work focuses on adaptive transmission in a long-term operating UWA point-to-point data transmission system. The time is divided into epochs as shown in Fig. 1. Each epoch consists of N time slots, and each time slot is used to transmit one data packet. At the end of the epoch, an acknowledgement packet is sent from the receiver through an error-free channel to the transmitter, which includes information of the packets that are successfully delivered and the received signal-to-noise ratio (SNR) of each packet. We further assume that at the transmitter, a fixed amount of information bits are generated at the application layer in each epoch and arrive at the data queue of the transmitter at the beginning of an epoch. The transmission schedule and the transmission parameters will be determined recursively epoch by epoch based on the data queue state and information about the channel state, with an ultimate goal of minimizing a long-term system cost.

For each time epoch, the transmission parameters include the transmission power, the modulation size and the channel coding rate. Note that the acoustic modem in practical systems only maintains a finite number of modulation and coding pairs as well as a finite number of transmission power levels. We consider a finite set of discrete power levels

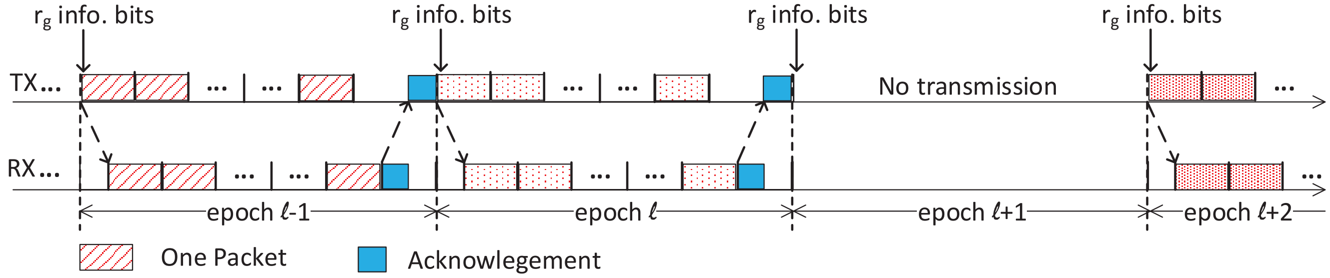


Fig. 1. Epoch structure at the transmitter and the receiver. The transmission parameters, including the transmission power, the modulation size and the channel coding rate, could vary from epoch to epoch.

$\mathcal{P} = \{P_0, P_1, P_2, \cdots\}$, with $P_0 = 0$ for no transmission, a finite set of discrete modulation sizes $\mathcal{M} = \{M_1, M_2, \cdots\}$, and a finite set of channel coding rates $\mathcal{R}_c = \{r_{c,1}, r_{c,2}, \cdots\}$. A combination of the modulation size M_i and the coding rate $r_{c,j}$ yields a data rate of $r_{c,j} \cdot \log_2 M_i$. Stack the triplet of transmission parameters $\{P \in \mathcal{P}, M \in \mathcal{M}, r_c \in \mathcal{R}_c\}$ into a vector, $\mathbf{a} := [P, M, r_c]^T$. Denote $\mathbf{a}(\ell)$ as the transmission parameter vector in the ℓth epoch.

In the next, we will develop an UWA channel model and an evolution model of the transmitter data queue, and then formulate the adaptive transmission as an optimization problem.

B. Underwater Acoustic Channel Model

To model both the fast fading and the large-scale shadowing of UWA channels, the UWA channel within one epoch is statistically characterized via a compound Nakagami-lognormal distribution. Accordingly, the received SNR follows a gamma-lognormal distribution [35]. Denote $\rho := P_{tx}/N_0$ as the transmission signal-power-to-noise ratio in an epoch, and denote x as the corresponding received SNR. The probability density function (PDF) of x can be expressed as

$$f_X(x; m, \mu, \sigma) = \int_0^\infty \frac{x^{m-1} \exp\left(-\frac{mx}{\rho y}\right)}{\Gamma(m)} \times \left(\frac{m}{\rho y}\right)^m \frac{1}{\sqrt{2\pi}\sigma y} \exp\left[-\frac{1}{2\sigma^2}(\ln y - \mu)^2\right] dy, \quad (1)$$

where $\Gamma(\cdot)$ is the gamma function, $m \in [1/2, \infty)$ is the fading parameter in the Nakagami-m fading, and μ and σ are the mean and the standard deviation of the lognormal shadowing, respectively [35]. Therefore, the UWA channel can be statistically parameterized by the triplet $\{m, \mu, \sigma\}$.

Define $\mathbf{s}_{ch} := [m, \mu, \sigma]^T$, and denote $\mathbf{s}_{ch}(\ell)$ as the channel state in the ℓth epoch. We model the long-term channel temporal variation as a first-order Markov process,

$$\mathbf{s}_{ch}(\ell) = \mathbf{A}\mathbf{s}_{ch}(\ell - 1) + \mathbf{w}_{ch}(\ell), \quad (2)$$

where $\mathbf{A}$ is a 3×3 unknown matrix, and $\mathbf{w}_{ch}(\ell)$ is the process noise vector for modeling inaccuracy, and is assumed following a zero-mean Gaussian distribution with an unknown covariance matrix $\mathbf{C}_w$, namely, $\mathbf{w}_{ch}(\ell) \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_w)$.

The UWA channel in an epoch can be measured during packet transmissions. We assume that the receiver can measure the received SNR of each packet even if the packet cannot be successfully decoded. The collected received SNR measurements are piggybacked on the acknowledgement packet sent from the receiver to the transmitter at the end of each active epoch. Denote $\{x_{\ell,1}, x_{\ell,2}, \cdots, x_{\ell,N}\}$ as the received SNRs of N packets in the ℓth epoch. Given the knowledge of the transmission SNR $\rho(\ell)$, the channel statistical parameters, $\{m(\ell), \mu(\ell), \sigma(\ell)\}$, can be estimated via the method of moments [34] according to (1).

We denote $\mathbf{z}_{ch}(\ell)$ as the vector stacked by the estimated parameters, $\{\hat{m}(\ell), \hat{\mu}(\ell), \hat{\sigma}(\ell)\}$, and take $\mathbf{z}_{ch}(\ell)$ as the observation vector of $\mathbf{s}_{ch}(\ell)$. Hence,

$$\mathbf{z}_{ch}(\ell) = \mathbf{s}_{ch}(\ell) + \mathbf{v}_{ch}(\ell), \quad (3)$$

where $\mathbf{v}_{ch}(\ell) \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_v)$ is the observation noise with an unknown covariance matrix $\mathbf{C}_v$, and is assumed independent from the process noise $\mathbf{w}_{ch}(\ell)$.

The channel model can then be uniquely represented by the unknown parameter set $\boldsymbol{\Theta} := \{\mathbf{A}, \mathbf{C}_w, \mathbf{C}_v\}$. Due to the water environment dynamics, the parameter set could be slowly time-varying.

Remark 1: For the epochs without active transmissions, a channel probing sequence could be transmitted to collect information about the channel dynamics. Although this work does not consider the probing sequence, the obtained theoretical results can be applied with slight modification to the scenario with channel probing sequences.

C. Evolution of the Data Queue

For each transmission parameter triplet $\mathbf{a} = [P, M, r_c]^T$, the packet error rate (PER) can be determined based on the compound distribution of the received SNR using an information-theoretic approach [36], [37] or an empirical formula estimated by real data [20]. For a channel state $\mathbf{s}_{ch}$ and a transmission parameter vector $\mathbf{a}$, we denote the PER by function $\mathrm{PER}(\mathbf{s}_{ch}, \mathbf{a})$.

At the beginning of epoch ℓ, the data queue length can be recursively represented as

$$q(\ell) = q(\ell - 1) - r(\ell - 1)N_s(\ell - 1) + r_g, \quad (4)$$

where $r(\ell-1)$ is the amount of information bits carried by each packet according to the transmission parameter vector $\mathbf{a}(\ell-1)$, $N_s(\ell - 1)$ is the number of packets that are successfully delivered to the receiver in epoch $(\ell - 1)$, and r_{g} is the amount of information bits from the application layer arriving at the beginning of epoch ℓ. Given $\mathrm{PER}(\mathbf{s}_{\mathrm{ch}}, \mathbf{a})$, the number of packets that can be successfully received follows a binomial distribution $\mathcal{B}(N, 1 - \mathrm{PER}(\mathbf{s}_{\mathrm{ch}}, \mathbf{a}))$, namely,

$$\begin{aligned}&\Pr(N_{\mathrm{s}} = k|\mathbf{s}_{\mathrm{ch}}, \mathbf{a}) = \\ &\quad \binom{N}{k} \left(1 - \mathrm{PER}(\mathbf{s}_{\mathrm{ch}}, \mathbf{a})\right)^{k} \left(\mathrm{PER}(\mathbf{s}_{\mathrm{ch}}, \mathbf{a})\right)^{N-k}. \qquad (5)\end{aligned}$$

Therefore, given the channel state $\mathbf{s}_{\mathrm{ch}}(\ell-1)$ and the transmission parameter vector $\mathbf{a}(\ell-1)$, the probability distribution of $N_{\mathrm{s}}(\ell-1)$, and the transition probability from $q(\ell-1)$ to $q(\ell)$ can be determined.

D. Problem Formulation for Optimal Transmission

We define the *system state* of epoch ℓ as $\mathbf{s}(\ell) := \{\mathbf{s}_{\mathrm{ch}}(\ell), q(\ell)\} \in \mathcal{S}$, $\forall \ell = 0, \cdots, \infty$. The transmission vector in each epoch, $\{\mathbf{a}(\ell) \in \mathcal{A}, \forall \ell\}$, can be determined to minimize the expected total discounted cost,

$$\min_{\{\mathbf{a}(\ell)\in\mathcal{A}\}_{\ell=0}^{\infty}} \mathbb{E}\left\{\sum_{\ell=0}^{\infty} \gamma^{\ell} C(\mathbf{s}(\ell), \mathbf{a}(\ell))\right\}, \qquad (6)$$

where $\gamma \in (0, 1]$ is a discount factor, and the cost function $C(\mathbf{s}, \mathbf{a}) : \mathcal{S} \times \mathcal{A} \to \mathcal{R}$ is application-dependent, and can be defined by the system designer. In this work, we take the cost function as

$$C(\mathbf{s}(\ell), \mathbf{a}(\ell)) = f_{\mathrm{p}}\big(P(\ell)\big) + f_{\mathrm{q}}\big(q(\ell) - r(\ell)N_{\mathrm{s}}(\ell)\big), \forall\ \ell \quad (7)$$

where $f_{\mathrm{p}}(\cdot)$ and $f_{\mathrm{q}}(\cdot)$ are two generic functions that are related to the energy consumption and the queue length, respectively, $(q(\ell) - r(\ell)N_{\mathrm{s}}(\ell))$ is the queue length at the end of epoch ℓ, and the number of successfully delivered packets $N_{\mathrm{s}}(\ell)$ depends on the channel state $\mathbf{s}_{\mathrm{ch}}(\ell)$ and the action $\mathbf{a}(\ell)$. We note that the cost function $C(\mathbf{s}(\ell), \mathbf{a}(\ell))$ is a random variable due to the randomness of the channel state $\mathbf{s}_{\mathrm{ch}}(\ell)$ and the number of successfully delivered packets $N_{\mathrm{s}}(\ell)$.

III. Reinforcement Learning-based Adaptive Transmission

The optimization problem in (6) falls into the category of reinforcement learning (RL) [38], where the transmitter (a.k.a. an agent in RL) interacts with the stochastic and dynamic UWA channel, with a goal of finding an optimal transmission strategy that minimizes the system long-term cost. In this section, we will reformulate the optimization problem in (6) in the model-based RL framework, and provide an overview of the proposed algorithm for online adaptive transmission. For notation convenience, we include the epoch index ℓ as a subscript.

A. Optimality for RL-based Adaptive Transmission

Should the system state be completely observable, the optimal transmission strategy can be determined by solving the Bellman optimality equation (BOE),

$$V^{*}(\mathbf{s}) = \min_{\mathbf{a}\in\mathcal{A}} \left[C(\mathbf{s}, \mathbf{a}) + \gamma \int_{\mathcal{S}} p(\mathbf{s}'|\mathbf{s}, \mathbf{a}) V^{*}(\mathbf{s}') d\mathbf{s}'\right], \qquad (8)$$

where $V^{*}(\mathbf{s})$ is referred to as the optimal value function of state $\mathbf{s}$, and $p(\mathbf{s}'|\mathbf{s}, \mathbf{a})$ is the state transition probability after taking action $\mathbf{a}$. The minimand in (8) consists of two terms: one is the cost of taking action $\mathbf{a}$ at the current state $\mathbf{s}$, and the other is the expected cost in the successor states after taking action $\mathbf{a}$. In the problem under consideration, although the queue state can be completely observed, the UWA channel cannot be directly observed, especially in epochs with no transmissions. The interaction between the transmitter and the underwater channel can be modeled as a partially observable Markov Decision process (POMDP) [38].

We define $b(\mathbf{s}_{\mathrm{ch},\ell})$ as the belief of channel state $\mathbf{s}_{\mathrm{ch},\ell}$, which corresponds to *a priori* PDF of state $\mathbf{s}_{\mathrm{ch},\ell}$, and can be inferred based on past observations $\{\mathbf{z}_{\mathrm{ch},\ell'}; \ell' < \ell\}$. Consider $\mathbf{z}_{\mathrm{ch},\ell} \in \mathcal{Z}, \forall \ell$, with the empty set $\Phi \in \mathcal{Z}$ to represent the scenario without active transmissions, and $q_{\ell} \in \mathcal{Q}, \forall \ell$. To indicate the dependence of the value function on the channel model, we include the model parameter set $\boldsymbol{\Theta}$ in the value function representation. The BOE in (8) can be reformulated as in (9) where q_{ℓ}, $q_{\ell+1}$, $N_{\mathrm{s},\ell}$ and $\mathbf{a}$ are related according to (4). Similar to (8), the minimand in (9) has two terms: the first term is the expected cost in the current epoch based on the current channel belief state and action, and the second term is the expected cost in future epochs. The optimal action in the current epoch is the one that minimizes the total expected cost in the current and future epochs.

We next discuss the probability functions in (9) for two types of actions.

- For the actions leading to packet transmissions, namely, $[\mathbf{a}]_1 \neq 0$ (c.f. Section II), the probability functions in (9) can be determined based on (2), (3) and (4). The channel state belief $b(\mathbf{s}_{\mathrm{ch},\ell+1})$ can be recursively updated as

$$\begin{aligned} b(\mathbf{s}_{\mathrm{ch},\ell+1}) \propto &\int_{\mathcal{S}} f(\mathbf{s}_{\mathrm{ch},\ell+1}|\mathbf{s}_{\mathrm{ch},\ell}) \\ &\times f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{a}) b(\mathbf{s}_{\mathrm{ch},\ell}) d\mathbf{s}_{\mathrm{ch},\ell}. \qquad (10)\end{aligned}$$

- For the action of no transmission, namely, $[\mathbf{a}]_1 = 0$, we have $\mathbf{z}_{\mathrm{ch},\ell} \in \Phi$. The probability function $f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{a})$ is non-informative and is independent of $\mathbf{s}_{\mathrm{ch},\ell}$, hence $\int_{\mathcal{Z}} f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{a}) d\mathbf{z}_{\mathrm{ch},\ell} = 1$. Therefore, the integral w.r.t. $\mathbf{z}_{\mathrm{ch},\ell}$ in the second summand of (9) can be separated from the double integral and be removed. Furthermore, since no transmission is scheduled, $q_{\ell+1}$ can be computed directly based on q_{ℓ} according to (4). The minimand in (9) can be simplified as

$$\begin{aligned} &C(\mathbf{s}_{\mathrm{ch},\ell}, q_{\ell}, \mathbf{a})|_{[\mathbf{a}]_1=0, N_{\mathrm{s},\ell}=0} \\ &\quad + \gamma \int_{\mathcal{S}} b(\mathbf{s}_{\mathrm{ch},\ell}) V^{*}(q_{\ell+1}, b(\mathbf{s}_{\mathrm{ch},\ell+1})) d\mathbf{s}_{\mathrm{ch},\ell}. \qquad (11)\end{aligned}$$

  The channel state belief $b(\mathbf{s}_{\mathrm{ch},\ell+1})$ can be recursively updated as

$$b(\mathbf{s}_{\mathrm{ch},\ell+1}) \propto \int_{\mathcal{S}} f(\mathbf{s}_{\mathrm{ch},\ell+1}|\mathbf{s}_{\mathrm{ch},\ell}) b(\mathbf{s}_{\mathrm{ch},\ell}) d\mathbf{s}_{\mathrm{ch},\ell}. \qquad (12)$$

Given the Gaussian assumption in (2) and (3), the channel state belief in (10) and (12) can be computed through operating over the mean vectors and the covariance matrices of

$$V^*(q_\ell,b(\mathbf{s}_{\mathrm{ch},\ell});\mathbf{\Theta}) = \min_{\mathbf{a}\in\mathcal{A}} \Big[\sum_{k=0}^{N} \int_{\mathcal{S}} C(\mathbf{s}_{\mathrm{ch},\ell}, q_\ell, \mathbf{a}) \Pr(N_{\mathrm{s},\ell}=k|\mathbf{s}_{\mathrm{ch},\ell},\mathbf{a}) b(\mathbf{s}_{\mathrm{ch},\ell}) d\mathbf{s}_{\mathrm{ch},\ell} \\ + \gamma \sum_{k=0}^{N} \int_{\mathcal{Z}} \int_{\mathcal{S}} \Pr(N_{\mathrm{s},\ell}=k|\mathbf{s}_{\mathrm{ch},\ell},\mathbf{a}) f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell},\mathbf{a}) b(\mathbf{s}_{\mathrm{ch},\ell}) V^*(q_{\ell+1}, b(\mathbf{s}_{\mathrm{ch},\ell+1});\mathbf{\Theta}) d\mathbf{s}_{\mathrm{ch},\ell} d\mathbf{z}_{\mathrm{ch},\ell} \Big] \tag{9}$$

relevant random vectors using Kalman filtering [34]. Detailed discussions will be provided in Section V.

B. An Overview of the Proposed Algorithm for Online Adaptive Transmission

Finding the optimal online transmission strategy requires estimation of channel model parameters and online planning at the beginning of each epoch. The model parameter estimation is performed based on channel measurements collected in the past epochs. A recursive estimator is desirable for online implementation, and especially in the presence of temporal variation of UWA channels. Given the model estimation, the optimal transmission strategy can be obtained by solving (9). Due to the mix of continuous and discrete random variables, the optimal solution to the BOE is not straightforward. In Section IV, we will develop a Monte Carlo sampling approach for online approximation of the optimal solution. In Section V, an algorithm will be designed to recursively estimate the unknown model parameter set $\mathbf{\Theta}$ and track the channel state.

At the outset, an overview of the proposed algorithm is in the following. At the beginning of epoch ℓ, the belief state $b(\mathbf{s}_{\mathrm{ch},\ell})$ is computed recursively via (10) or (12), based on the parameter set estimation $\hat{\mathbf{\Theta}}_{\ell-1}$, the belief state $b(\mathbf{s}_{\mathrm{ch},\ell-1})$, and the observation $\mathbf{z}_{\mathrm{ch},\ell-1}$. The queue length q_ℓ can be observed. Based on the current knowledge of the system state and the channel model estimation, the optimal transmission strategy (i.e. action) can be obtained by solving (9). The transmitter applies the obtained transmission strategy. At the end of the epoch, the transmitter collects possible feedback from the receiver. Based on the observation $\mathbf{z}_{\mathrm{ch},\ell}$ and the previous model estimation $\hat{\mathbf{\Theta}}_{\ell-1}$, the transmitter updates the channel model estimation, denoted by $\hat{\mathbf{\Theta}}_\ell$. The belief state $b(\mathbf{s}_{\mathrm{ch},\ell})$, the observation $\mathbf{z}_{\mathrm{ch},\ell}$, and the model estimation $\hat{\mathbf{\Theta}}_\ell$ will be used to compute the belief state $b(\mathbf{s}_{\mathrm{ch},\ell+1})$ in the next epoch. The above process is repeated for each epoch.

IV. Monte Carlo Sampling for Online Approximation

The mix of continuous and discrete random variables in the BOE (9) makes it intractable to solve. In this section, we develop a Monte Carlo sampling-based approach [39] to approximate the value function and to find a near-optimal solution. The approach is also known as Monte Carlo planning.

A. Value Function Approximation

The BOE in (9) has a recursive form. Given an estimation of the model parameters $\hat{\mathbf{\Theta}}$, sampling-based methods [40] can be applied to approximate the value function recursively through constructing a state-action tree (see Fig. 2 for an illustration, details provided later). The approximation accuracy increases as the number of samples in the state-action tree increases, which however, incurs higher computational complexity.

In this work, the idea of sparse sampling [39] is applied during the state-action tree construction. To guide the selection of "important" samples, a linear regression (LR) method [41] is introduced to approximate the value function of the system state based on past value function approximations. Specifically, for the system state $\{q, b(\mathbf{s}_{\mathrm{ch}}), \hat{\mathbf{\Theta}}\}$, denote $\mathbf{x}$ as a vector stacked by q and the scalar elements in the mean vector and the covariance matrix of the channel belief state $b(\mathbf{s}_{\mathrm{ch}})$. The value function can be approximated as

$$V(\mathbf{x};\boldsymbol{\phi}) = \phi_0 + \mathbf{x}^{\mathrm{T}}\boldsymbol{\phi}_1, \tag{13}$$

where $\boldsymbol{\phi}^{\mathrm{T}} := [\phi_0, \boldsymbol{\phi}_1^{\mathrm{T}}]$ is the LR coefficient vector[1]. The LR coefficient vector can be updated via the stochastic gradient decent method [41] based on past value function approximations.

The proposed Monte Carlo sampling approach has two steps. The first step is to construct a state-action planning tree, as depicted in Fig. 2. The second step is to approximate the value function recursively based on the state-action tree, as described in Algorithm 1. Details about the two steps are in the following.

1) State-action Tree Construction: Given a root node which represents the current system state, the state-action tree is constructed by sequentially drawing samples of actions and samples of the system states up to a certain planning depth (denoted by D). Specifically,

- Let the current system state described by a triplet $(q, b, \hat{\mathbf{\Theta}})$ be the root state node of the state-action tree, where q is the queue length, and b is the channel belief state;
- For each system state node (including the root node) in the state-action tree, a small number (N_{a}) of actions which yield less approximated expected costs will be selected to expand the tree. To do so, one first enumerates all the actions in the action space. For each enumerated action, a number (N_{o}) of child nodes describing the system states in the next epoch can be obtained through drawing samples of the channel state, the observations of the channel state, and the number of successfully delivered packets; see Algorithm 2. The value of each child system state node can be approximated by the LR

[1] For the elements in $\mathbf{x}$ which have relatively higher orders of magnitude, they can be multiplied by constants to reduce their orders of magnitude. For example, the values of μ and q are multiplied by 0.1 and $1/r_{\mathrm{g}}$, respectively, in Section VI for the LR.

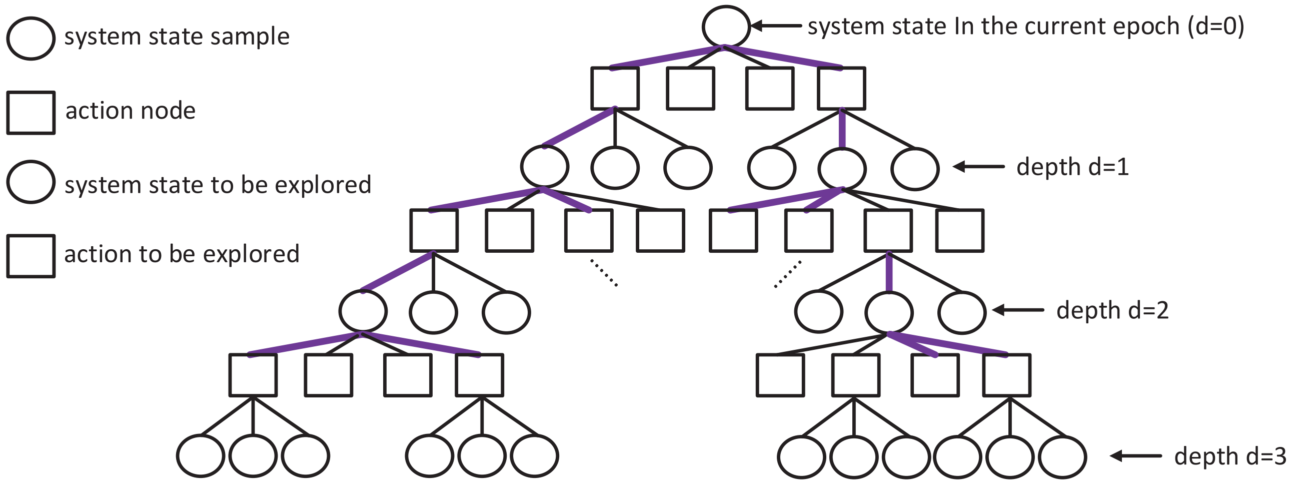


Fig. 2. An illustration of the state-action tree for online planning, with the tree depth $D = 3$. There are 4 actions in the action space $\mathcal{A}$. At depth d, $N_{\rm o} = 3$ system state samples are drawn based on the action and the system state at depth $(d-1)$. $N_{\rm a} = 2$ actions and 1 child system state node are further explored at each depth.

(c.f. Lines 8 to 11 in Algorithm 1). The expected cost induced by each action can be approximated by summing up the expected immediate cost and the averaged value of its child system state nodes.

The expected immediate cost of each action can be computed by drawing a sufficient number of channel samples according to the belief state. The immediate cost corresponding to each channel sample can be obtained based on the packet error rate of the channel sample according to (5) and (7). The average of the immediate costs corresponding to all the channel samples yields the expected immediate cost.

- For each action to be further explored via the tree expansion, for computational efficiency, only one of its child nodes is randomly selected and serves as the system state to be explored in the next epoch.

The above process is repeated until the tree reaches the maximal planning depth, namely, the maximal number of future epochs to be evaluated.

The values of D, $N_{\rm a}$, and $N_{\rm o}$ can be determined to strike a balance between the approximation accuracy and the computational complexity. Benefited from the LR-based value function approximation and the sparse sampling, the structure of the state-action tree can be much simplified compared to the case when all the actions or a large amount of child system state samples are explored to reach similar approximation accuracy.

2) Valuation Function Calculation: The value of the root state node (i.e., the current system state), can be calculated by propagating the values of all the child nodes in the state-action tree to the root node. Specifically,

- The value of a particular system state node at the planning depth d $(d < D)$ is set as the minimal expected cost induced by the selected actions to be explored, and the action with the minimal expected cost is taken as the optimal action. For the system state nodes at the tree leaves, their values are approximated by the LR (c.f. Lines 2 to 4 in Algorithm 1).
- For each action, we follow the concept of the temporal difference (TD) learning [8] to approximate its expected cost (as shown in Line 16 of Algorithm 1), based on its expected immediate cost, the value of its child system state node in the state-action tree, and the approximated cost obtained via the LR method (c.f. Lines 8 to 11 in Algorithm 1). Compared to the method that calculates the expected cost as the summation of the expected immediate cost and the value of the child system state node, the above method can exploit the historical value function approximation results obtained via the LR to achieve higher approximation accuracy.

B. Computational Complexity

Denote K as the total number of channel state samples to calculate the expected immediate cost of each action and $C_{\rm PER}$ as the complexity of calculating the PER. The computational complexity of the expected immediate cost is $C_{\rm cur} = O(KC_{\rm PER})$. The computational complexity to sample the triplet in Algorithm 2 in the worst case, namely, every action indicating packet transmissions, is $C_2 = O(C_{\rm est} + C_{\rm PER})$ where $C_{\rm est}$ is the computational complexity for the channel model estimation. The complexity of Algorithm 1 in the worst case is $C_1 = O(|\mathcal{A}|N_{\rm a}^{D-1}C_{\rm cur}^{D} + |\mathcal{A}|N_{\rm a}^{D-1}N_{\rm o}C_2)$. Hence, the total complexity of the algorithm in the worst case is $O(C_1 + C_{\rm est})$.

V. Recursive Estimation of Unknown Channel Model Parameters

To facilitate online implementation, we will develop a low-complexity recursive algorithm to estimate the parameter set $\boldsymbol{\Theta}$ and the channel state vector based on the sequentially obtained observations $\{\mathbf{z}_{{\rm ch},\ell}\}$. For notation convenience, we denote $\mathbf{z}_{{\rm ch},\ell_1}^{\ell_2} := \{\mathbf{z}_{{\rm ch},\ell_1}, \cdots, \mathbf{z}_{{\rm ch},\ell_2}\}$ and $\mathbf{s}_{{\rm ch},\ell_1}^{\ell_2} := \{\mathbf{s}_{{\rm ch},\ell_1}, \cdots, \mathbf{s}_{{\rm ch},\ell_2}\}$.

At time epoch ℓ, the unknown parameters can be estimated by maximizing the log-likelihood function with respect to

Algorithm 1 Value function approximation: $\mathcal{V}(q, b, \hat{\mathbf{\Theta}}, d, \gamma, \alpha, \beta, \lambda)$

```
Input: Discount factor $\gamma$, temporal difference (TD) learning rate $\alpha$, learning rate in the linear regression (LR) $\beta$, regularization parameter $\lambda$, current planning depth $d$, and system state $(q, b, \hat{\mathbf{\Theta}})$
Set the LR coefficient vector $\boldsymbol{\phi}^{\mathrm{T}} := [\phi_0, \boldsymbol{\phi}_1^{\mathrm{T}}]$ as a global parameter
Output: Approximated value function $V_{\mathrm{opt}}$
Assign an integer value to $D$ $(D > 0)$ and set $V_{\mathrm{opt}}$=$+\infty$
if $d = D$ then
    For system state $(q, b, \hat{\mathbf{\Theta}})$, set $\mathbf{x}$ as a vector consisting of $q$ and all the scalar elements in the mean vector and the covariance matrix of the channel belief state $b$
    return $V(\mathbf{x}; \boldsymbol{\phi}) = \phi_0 + \mathbf{x}^{\mathrm{T}}\boldsymbol{\phi}_1$
for $i = 1$ to $|\mathcal{A}|$ do
    Select an action $\mathbf{a}$ from the action space $\mathcal{A}$ without replacement
    Compute the expected immediate cost $c_i$, and set $v_i = c_i$
    for $j = 1$ to $N_{\mathrm{o}}$ do
        Obtain a state sample $(q', b', \hat{\mathbf{\Theta}}')$ according to action $\mathbf{a}$ based on Algorithm 2
        For $(q', b', \hat{\mathbf{\Theta}}')$, set $\mathbf{x}'_{ij}$ as a vector consisting of $q'$ and all the scalar elements in the mean vector and the covariance matrix of the channel belief state $b'$
        $v_i \leftarrow v_i + \frac{\gamma}{N_{\mathrm{o}}} V(\mathbf{x}'_{ij}; \boldsymbol{\phi})$
Sort elements in $\{v_1, v_2, \cdots, v_{|\mathcal{A}|}\}$ in an increasing order as $\{v_{(1)}, v_{(2)}, \cdots, v_{(|\mathcal{A}|)}\}$
for $i = 1$ to $N_{\mathrm{a}}$ do
    Choose action $\mathbf{a}$ yielding $v_{(i)}$
    Randomly select a state sample $(q', b', \hat{\mathbf{\Theta}}')$ obtained after taking action $\mathbf{a}$
    Perform the TD learning:
    $v_{(i)} \leftarrow v_{(i)} + \alpha(c_{(i)} + \gamma\mathcal{V}(q', b', \hat{\mathbf{\Theta}}', d+1, \gamma, \alpha, \beta, \lambda) - v_{(i)})$
    Update the LR vector:
    $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \beta(V(\mathbf{x}; \boldsymbol{\phi}) - v_{(i)})\nabla_{\boldsymbol{\phi}} V(\mathbf{x}; \boldsymbol{\phi}) - \beta\lambda\boldsymbol{\phi}$
    if $v_{(i)} < V_{\mathrm{opt}}$ then
        $V_{\mathrm{opt}} = v_{(i)}$
        $\mathbf{a}_{\mathrm{opt}} = \mathbf{a}$
return $V_{\mathrm{opt}}$
```

the complete data set $L_\ell(\mathbf{\Theta}) := \ln f(\mathbf{z}^\ell_{\mathrm{ch},0}, \mathbf{s}_{\mathrm{ch},-1}, \mathbf{s}^\ell_{\mathrm{ch},0}|\mathbf{\Theta})$. However, the channel state process $\{\mathbf{s}_{\mathrm{ch},\ell'}\}$ is not observable. Instead, the expectation-maximization (EM) algorithm [41] can be used, which estimates the unknown parameters iteratively through an expectation step and a maximization step. Given a parameter set estimation $\hat{\mathbf{\Theta}}$, in the expectation step, the expectation of the log-likelihood function can be approximated as

$$\mathbb{E}\left[L_\ell(\mathbf{\Theta})|\hat{\mathbf{\Theta}}\right] = \int \left[\ln f(\mathbf{z}^\ell_{\mathrm{ch},0}, \mathbf{s}_{\mathrm{ch},-1}, \mathbf{s}^\ell_{\mathrm{ch},0}|\mathbf{\Theta})\right] \times f\left(\mathbf{s}^\ell_{\mathrm{ch},-1}|\mathbf{z}^\ell_{\mathrm{ch},0}, \hat{\mathbf{\Theta}}\right) d\mathbf{s}^\ell_{\mathrm{ch},-1}. \tag{14}$$

The parameter set estimation can be updated in the maximization step as $\hat{\mathbf{\Theta}}^{\mathrm{new}} = \arg\ \max\ \mathbb{E}\big[L_\ell(\mathbf{\Theta})|\hat{\mathbf{\Theta}}\big]$.

Algorithm 2 Sample the queue state and the belief state in the next epoch

```
Input: Belief state $b$, queue length $q$, action $\mathbf{a}$, and model estimation $\hat{\mathbf{\Theta}}$
Output: Belief state $b'$, and queue length $q'$ in the next epoch, and updated estimated $\hat{\mathbf{\Theta}}'$
if $\mathbf{a}$ indicates transmissions then
    Sample the channel state $\mathbf{s}_{\mathrm{ch}}$ from the belief state $b$
    Sample the observation noise $\mathbf{w}$ from $\mathcal{N}(0, \hat{\mathbf{C}}_{\mathbf{w}})$
    Compute the observation $\mathbf{z} = \mathbf{s}_{\mathrm{ch}} + \mathbf{w}$
    Compute $b'$ via Kalman filtering based on based on $b$ and observation $\mathbf{z}$
    Sample the number of packets that are successfully decoded by the receiver, $N_{\mathrm{s}}$, based on the channel state samples and the action $\mathbf{a}$, according to (5).
    Compute the queue length in the next epoch $q' = q + r_{\mathrm{g}} - rN_{\mathrm{s}}$
else
    Set $q' = q + r_{\mathrm{g}}$
    Compute $b'$ based on $b$ via Kalman filtering without channel observation
Update $\hat{\mathbf{\Theta}}'$ as described in Section V
return $(q', b', \hat{\mathbf{\Theta}})$
```

The algorithm, however, requires processing within each iteration the data in the current and all the past epochs, hence is not amenable to online implementation. We next introduce several approximations, and then develop an EM-type and low-complexity recursive algorithm that estimates the parameter set $\mathbf{\Theta}$ in each epoch iteratively based on the new observation vector and the parameter estimation in the last epoch. We denote $\hat{\mathbf{\Theta}}_{\ell'}$ as the estimation at epoch ℓ'.

A. Approximation for Recursive Operation

Consider that

$$\ln f(\mathbf{z}^\ell_{\mathrm{ch},0}, \mathbf{s}^\ell_{\mathrm{ch},-1}|\mathbf{\Theta}) = \ln f(\mathbf{z}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \mathbf{\Theta}) + \ln f(\mathbf{z}^{\ell-1}_{\mathrm{ch},0}, \mathbf{s}^{\ell-1}_{\mathrm{ch},-1}|\mathbf{\Theta}). \tag{15}$$

The expectation in (14) can be decomposed as

$$\mathbb{E}\big[L_\ell(\mathbf{\Theta})|\hat{\mathbf{\Theta}}\big] = \int [\ln f(\mathbf{s}_{\mathrm{ch},-1}|\mathbf{\Theta})] f(\mathbf{s}_{\mathrm{ch},-1}|\mathbf{z}^\ell_{\mathrm{ch},0}, \hat{\mathbf{\Theta}}) d\mathbf{s}_{\mathrm{ch},-1} + \sum_{\ell'=0}^{\ell} \int [\ln f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{z}_{\mathrm{ch},\ell'}|\mathbf{s}_{\mathrm{ch},\ell'-1}, \mathbf{\Theta})] \times f\left(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}^\ell_{\mathrm{ch},0}, \hat{\mathbf{\Theta}}\right) d\mathbf{s}_{\mathrm{ch},\ell'} d\mathbf{s}_{\mathrm{ch},\ell'-1}. \tag{16}$$

It can be approximated in two steps,

$$
\begin{aligned}
\mathbb{E}\big[L_\ell(\boldsymbol{\Theta})|\hat{\boldsymbol{\Theta}}\big] \approx & \int [\ln f(\mathbf{s}_{\mathrm{ch},-1}|\boldsymbol{\Theta})] f(\mathbf{s}_{\mathrm{ch},-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}) d\mathbf{s}_{\mathrm{ch},-1} \\
& + \sum_{\ell'=0}^{\ell} \int [\ln f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{z}_{\mathrm{ch},\ell'}|\mathbf{s}_{\mathrm{ch},\ell'-1}, \boldsymbol{\Theta})] \\
& \times \underbrace{f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell'}, \hat{\boldsymbol{\Theta}})}_{\approx f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}) \text{ in Eq. (16)}} d\mathbf{s}_{\mathrm{ch},\ell'} d\mathbf{s}_{\mathrm{ch},\ell'-1}, \quad (17a)
\end{aligned}
$$

$$
\begin{aligned}
\approx & \int [\ln f(\mathbf{s}_{\mathrm{ch},-1}|\boldsymbol{\Theta})] f\left(\mathbf{s}_{\mathrm{ch},-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}_{-1}\right) d\mathbf{s}_{\mathrm{ch},-1} \\
& + \sum_{\ell'=0}^{\ell-1} \int [\ln f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{z}_{\mathrm{ch},\ell'}|\mathbf{s}_{\mathrm{ch},\ell'-1}, \boldsymbol{\Theta})] \\
& \times \underbrace{f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell'}, \hat{\boldsymbol{\Theta}}_{\ell'})}_{\approx f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell'}, \hat{\boldsymbol{\Theta}}) \text{ in Eq. (17a)}} d\mathbf{s}_{\mathrm{ch},\ell'} d\mathbf{s}_{\mathrm{ch},\ell'-1} \\
& + \int [\ln f(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \boldsymbol{\Theta})] \\
& \times f(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}) d\mathbf{s}_{\mathrm{ch},\ell} d\mathbf{s}_{\mathrm{ch},\ell-1}, \quad (17b)
\end{aligned}
$$

where the expectation of $\ln f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{z}_{\mathrm{ch},\ell'}|\mathbf{s}_{\mathrm{ch},\ell'-1}, \boldsymbol{\Theta})$ in (17a) is performed with respect to $f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell'}, \hat{\boldsymbol{\Theta}})$ instead of $f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}})$, and in (17b), the expectation of $[\ln f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{z}_{\mathrm{ch},\ell'}|\mathbf{s}_{\mathrm{ch},\ell'-1}, \boldsymbol{\Theta})]$ can be computed at epoch ℓ' based on $f(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell'}, \hat{\boldsymbol{\Theta}}_{\ell'})$. The above approximations enable recursive computation of the summation on the right side of (17b).

One more approximation is made for recursive computation of the PDF $f(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}})$. Note that

$$
\begin{aligned}
f(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}) &= 1/c_0 \times f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell}, \hat{\boldsymbol{\Theta}}) \\
&\times f(\mathbf{s}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \hat{\boldsymbol{\Theta}}) f(\mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell-1}, \hat{\boldsymbol{\Theta}}) \quad (18)
\end{aligned}
$$

where c_0 is a normalization constant. We approximate the joint PDF by

$$
\begin{aligned}
\tilde{f}(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}) &:= 1/c_0' \times f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell}, \hat{\boldsymbol{\Theta}}) \\
&\times f(\mathbf{s}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \hat{\boldsymbol{\Theta}}) \tilde{f}(\mathbf{s}_{\mathrm{ch},\ell-1}|\hat{\boldsymbol{\Theta}}_{\ell-1}), \quad (19)
\end{aligned}
$$

through replacing $f(\mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell-1}, \hat{\boldsymbol{\Theta}})$ by $\tilde{f}(\mathbf{s}_{\mathrm{ch},\ell-1}|\hat{\boldsymbol{\Theta}}_{\ell-1})$ in (18), where$\tilde{f}(\mathbf{s}_{\mathrm{ch},\ell'}|\hat{\boldsymbol{\Theta}}_{\ell'})$ is defined as the marginalization of $\tilde{f}(\mathbf{s}_{\mathrm{ch},\ell'}, \mathbf{s}_{\mathrm{ch},\ell'-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell'}, \hat{\boldsymbol{\Theta}}_{\ell'})$ with respect to $\mathbf{s}_{\mathrm{ch},\ell'}$, and c_0' is a normalization constant.

Finally, based on (17b) and (19), the expectation $\mathbb{E}\big[L_\ell(\boldsymbol{\Theta})|\hat{\boldsymbol{\Theta}}\big]$ is approximated by $Q_\ell(\boldsymbol{\Theta}|\hat{\boldsymbol{\Theta}})$ which is recursively defined as

$$
\begin{aligned}
Q_\ell(\boldsymbol{\Theta}|\hat{\boldsymbol{\Theta}}) = & \gamma_{\mathrm{ch}} Q_{\ell-1}(\boldsymbol{\Theta}|\hat{\boldsymbol{\Theta}}_{\ell-1}) + \\
& \int [\ln f(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \boldsymbol{\Theta})] \\
& \times \tilde{f}(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}) d\mathbf{s}_{\mathrm{ch},\ell} d\mathbf{s}_{\mathrm{ch},\ell-1}, \quad (20)
\end{aligned}
$$

where $\gamma_{\mathrm{ch}} \in (0, 1]$ is a forgetting factor that accounts for the temporal variation of unknown parameters. Based on (20), the expectation and maximization operations in the EM algorithm can be applied for recursive and iterative parameter estimation and channel tracking, as described in the next subsection.

B. Recursive Model and Channel State Estimation

Denote $\hat{\boldsymbol{\Theta}}_\ell^{(i)} = \{\hat{\mathbf{A}}_\ell^{(i)}, \hat{\mathbf{C}}_{\mathrm{w},\ell}^{(i)}, \hat{\mathbf{C}}_{\mathrm{v},\ell}^{(i)}\}$ as the estimation of the unknown parameters in the ith iteration at epoch ℓ. The parameter estimation can be updated via maximizing $Q_\ell(\boldsymbol{\Theta}|\hat{\boldsymbol{\Theta}}_\ell^{(i)})$. Note that $f(\mathbf{z}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \boldsymbol{\Theta}) = f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell}, \boldsymbol{\Theta}) f(\mathbf{s}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \boldsymbol{\Theta})$. Substitute

$$
\begin{aligned}
f(\mathbf{z}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell}, \boldsymbol{\Theta}) &\sim \mathcal{N}(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{C}_{\mathrm{w}}), \\
f(\mathbf{s}_{\mathrm{ch},\ell}|\mathbf{s}_{\mathrm{ch},\ell-1}, \boldsymbol{\Theta}) &\sim \mathcal{N}(\mathbf{A}\mathbf{s}_{\mathrm{ch},\ell-1}, \mathbf{C}_{\mathrm{v}})
\end{aligned}
$$

into the log-likelihood function in (20). Set the partial derivative of $Q_\ell(\boldsymbol{\Theta}|\hat{\boldsymbol{\Theta}}_\ell^{(i)})$ with respect to each unknown parameter to zero. A set of recursive equations can be obtained,

$$
\begin{aligned}
\hat{\mathbf{A}}_\ell^{(i+1)} = & \hat{\mathbf{A}}_{\ell-1} + \\
& \left(\mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell}\mathbf{s}_{\mathrm{ch},\ell-1}^{\mathrm{T}}] - \hat{\mathbf{A}}_{\ell-1}\mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell-1}\mathbf{s}_{\mathrm{ch},\ell-1}^{\mathrm{T}}]\right) \mathbf{M}_{\ell-1}^{-1}, \quad (21a) \\
\hat{\mathbf{C}}_{\mathrm{w},\ell}^{(i+1)} = & \hat{\mathbf{C}}_{\mathrm{w},\ell-1} + \frac{1-\gamma_{\mathrm{ch}}}{1-\gamma_{\mathrm{ch}}^{\ell}} \\
& \times \left\{\mathbb{E}\left[(\mathbf{s}_{\mathrm{ch},\ell} - \hat{\mathbf{A}}_\ell^{(i+1)}\mathbf{s}_{\mathrm{ch},\ell-1})(\mathbf{s}_{\mathrm{ch},\ell} - \hat{\mathbf{A}}_\ell^{(i+1)}\mathbf{s}_{\mathrm{ch},\ell-1})^{\mathrm{T}}\right] \right. \\
& \left. - \hat{\mathbf{C}}_{\mathrm{w},\ell-1}\right\}, \quad (21b) \\
\hat{\mathbf{C}}_{\mathrm{v},\ell}^{(i+1)} = & \hat{\mathbf{C}}_{\mathrm{v},\ell-1} + \frac{1-\gamma_{\mathrm{ch}}}{1-\gamma_{\mathrm{ch}}^{\ell+1}} \\
& \times \left\{\mathbb{E}\left[(\mathbf{z}_{\mathrm{ch},\ell} - \mathbf{s}_{\mathrm{ch},\ell})(\mathbf{z}_{\mathrm{ch},\ell} - \mathbf{s}_{\mathrm{ch},\ell})^{\mathrm{T}}\right] - \hat{\mathbf{C}}_{\mathrm{v},\ell-1}\right\}, \quad (21c)
\end{aligned}
$$

where an auxiliary matrix is defined as

$$
\mathbf{M}_{\ell-1} := \gamma_{\mathrm{ch}}\mathbf{M}_{\ell-2} + \mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell-1}\mathbf{s}_{\mathrm{ch},\ell-1}^{\mathrm{T}}], \quad (22)
$$

and the expectations are performed with respect to $\tilde{f}(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}_\ell^{(i)})$.

The expectations in (21) and (22) can be computed via performing marginalization of $\tilde{f}(\mathbf{s}_{\mathrm{ch},\ell}, \mathbf{s}_{\mathrm{ch},\ell-1}|\mathbf{z}_{\mathrm{ch},0}^{\ell}, \hat{\boldsymbol{\Theta}}_\ell^{(i)})$ (c.f. (19)). For convenience, denote $\tilde{f}(\mathbf{s}_{\mathrm{ch},\ell-1}|\hat{\boldsymbol{\Theta}}_{\ell-1}) \sim \mathcal{N}(\boldsymbol{\mu}_{\ell-1}, \mathbf{C}_{\ell-1})$. It can be shown that [41]

$$
\begin{aligned}
\mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell}|\hat{\boldsymbol{\Theta}}_\ell^{(i)}] = \boldsymbol{\mu}_\ell^{(i)} = \hat{\mathbf{A}}_\ell^{(i)}\boldsymbol{\mu}_{\ell-1} + & \\
\mathbf{K}_\ell^{(i)}(\mathbf{z}_{\mathrm{ch}} - \hat{\mathbf{A}}_\ell^{(i)}\boldsymbol{\mu}_{\ell-1}) & \quad (23a) \\
\mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell-1}|\hat{\boldsymbol{\Theta}}_\ell^{(i)}] = \breve{\boldsymbol{\mu}}_{\ell-1}^{(i)} = \boldsymbol{\mu}_{\ell-1} + & \\
\mathbf{J}_{\ell-1}^{(i)}(\boldsymbol{\mu}_\ell^{(i)} - \hat{\mathbf{A}}_\ell^{(i)}\boldsymbol{\mu}_{\ell-1}) & \quad (23b) \\
\mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell}\mathbf{s}_{\mathrm{ch},\ell}^{\mathrm{T}}|\hat{\boldsymbol{\Theta}}_\ell^{(i)}] = \mathbf{C}_\ell^{(i)} + \boldsymbol{\mu}_\ell^{(i)}\boldsymbol{\mu}_\ell^{(i),\mathrm{T}} & \quad (23c) \\
\mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell-1}\mathbf{s}_{\mathrm{ch},\ell-1}^{\mathrm{T}}|\hat{\boldsymbol{\Theta}}_\ell^{(i)}] = \breve{\mathbf{C}}_{\ell-1}^{(i)} + \breve{\boldsymbol{\mu}}_{\ell-1}^{(i)}\breve{\boldsymbol{\mu}}_{\ell-1}^{(i),\mathrm{T}} & \quad (23d) \\
\mathbb{E}[\mathbf{s}_{\mathrm{ch},\ell}\mathbf{s}_{\mathrm{ch},\ell-1}^{\mathrm{T}}|\hat{\boldsymbol{\Theta}}_\ell^{(i)}] = \mathbf{C}_\ell^{(i)}\mathbf{J}_{\ell-1}^{(i),\mathrm{T}} + \boldsymbol{\mu}_\ell^{(i)}\breve{\boldsymbol{\mu}}_{\ell-1}^{(i),\mathrm{T}} & \quad (23e)
\end{aligned}
$$

where $\mathbf{K}_\ell^{(i)} = \mathbf{P}_\ell^{(i)}\big(\hat{\mathbf{C}}_{\mathrm{v}}^{(i)} + \mathbf{P}_\ell^{(i)}\big)^{-1}$ with $\mathbf{P}_\ell^{(i)} = \hat{\mathbf{A}}_\ell^{(i)}\mathbf{C}_{\ell-1}\hat{\mathbf{A}}_\ell^{(i),\mathrm{T}} + \hat{\mathbf{C}}_{\mathrm{w}}^{(i)}$, $\mathbf{J}_{\ell-1}^{(i)} = \mathbf{C}_{\ell-1}\hat{\mathbf{A}}_\ell^{(i),\mathrm{T}}\big(\mathbf{P}_\ell^{(i)}\big)^{-1}$, $\mathbf{C}_\ell^{(i)} = (\mathbf{I} - \mathbf{K}_\ell^{(i)})\mathbf{P}_\ell^{(i)}$, and $\breve{\mathbf{C}}_{\ell-1}^{(i)} = \mathbf{C}_{\ell-1} + \mathbf{J}_{\ell-1}^{(i)}(\mathbf{C}_\ell^{(i)} - \mathbf{P}_\ell^{(i)})\mathbf{J}_{\ell-1}^{(i),\mathrm{T}}$.

In summary, when $\mathbf{z}_{\mathrm{ch},\ell}$ is available at the end of epoch ℓ, the iterative model parameter estimation can be initialized as $\hat{\boldsymbol{\Theta}}_\ell^{(0)} = \hat{\boldsymbol{\Theta}}_{\ell-1}$. The expectation and maximization operations are performed iteratively based on (23) and (21). Consider that the operation terminates after a pre-determined number of iterations, denoted by N_{iter}. We set $\hat{\boldsymbol{\Theta}}_\ell = \hat{\boldsymbol{\Theta}}_\ell^{(N_{\text{iter}})}$ and $\tilde{f}(\mathbf{s}_{\text{ch},\ell}|\hat{\boldsymbol{\Theta}}_\ell) \simeq \mathcal{N}(\boldsymbol{\mu}_\ell, \mathbf{C}_\ell)$ with $\boldsymbol{\mu}_\ell = \boldsymbol{\mu}_\ell^{(N_{\text{iter}})}$ and $\mathbf{C}_\ell = \mathbf{C}_\ell^{(N_{\text{iter}})}$, which will be used for the operation in the next epoch. If no transmission is scheduled in epoch ℓ, namely, $\mathbf{z}_{\text{ch},\ell}$ is an empty set, no model parameter estimation is needed. One can set $\hat{\boldsymbol{\Theta}}_\ell = \hat{\boldsymbol{\Theta}}_{\ell-1}$, $\boldsymbol{\mu}_\ell = \hat{\mathbf{A}}_{\ell-1}\boldsymbol{\mu}_{\ell-1}$ and $\mathbf{C}_\ell = \hat{\mathbf{A}}_{\ell-1}\mathbf{C}_{\ell-1}\hat{\mathbf{A}}_{\ell-1}^{\text{T}} + \hat{\mathbf{C}}_{\text{w},\ell-1}$. In both cases, the *a posteriori* PDF $\tilde{f}(\mathbf{s}_{\text{ch},\ell}|\hat{\boldsymbol{\Theta}}_\ell)$ and the conditional PDF $\tilde{f}(\mathbf{s}_{\text{ch},\ell+1}|\mathbf{s}_{\text{ch},\ell}, \hat{\boldsymbol{\Theta}}_\ell)$ can be used to compute the belief state $b(\mathbf{s}_{\text{ch},\ell+1})$ according to (10) or (12).

Remark 2: The proposed algorithm does not guarantee that the Nakagami-fading parameter $m \geq 1/2$. In the Monte Carlo sampling method for online approximation, we only draw samples of m which are greater than $1/2$ based on the channel belief state.

VI. Algorithm Evaluation

The proposed algorithm is evaluated using data sets collected from two experiments: one is the Surface Processes and Acoustic Communications Experiment (SPACE08), and the other is an experiment conducted in the Keweenaw Waterway near Michigan Tech in Nov. 2014 (KW-NOV14).

A. Experiment Description

The SPACE08 experiment was conducted near the coast of Martha's Vineyard, MA, from Oct. 14 to Nov. 1, 2008. We consider the data collected by a receiver which is 200 meters away from the transmitter, from Julian date 287 to Julian date 302. Due to the appearance of severe weather conditions during the experiment, some of the data files were damaged hence are excluded for algorithm evaluation. A waveform of 10 seconds was transmitted every two hours from the source to the receiver, leading to 12 transmissions per day. The waveform consists of 60 signaling blocks within the frequency band $[8,\ 18]$ kHz, and each block has 672 symbols. In this work, we take each transmission as one epoch and take each signaling block as one packet. There are 117 epochs in total. The channel distribution parameters μ, σ and m within each epoch are estimated via the method of moments [34] based on the received SNR samples obtained within that epoch. The evolution of the distribution parameters is shown in Fig. 3.

The KW-NOV14 experiment was held in the Keweenaw Waterway adjacent to Michigan Tech from Nov. 22 to Nov. 28, 2014 when the water surface was covered by a thin layer of ice. The distance between the transmitter and the receiver is 312 m. A waveform of about 9 seconds was transmitted every 15 minutes. The waveform consists of 20 signaling blocks within the frequency band $[14,\ 20]$ kHz, and each block has 672 symbols. Similar to SPACE08, we take each transmission as one epoch and take each signaling block as one packet. A total of 117 epochs are used for algorithm evaluation. Artificial Gaussian noise is added to the received signal in KW-NOV14 such that the two experiments have similar average channel losses over all the epochs. Evolution of the KW-NOV14 channel distribution parameters is shown in Fig. 3.

TABLE I
Transmission Modes.

Mode Index	Coding rate	Modulation	TSNR
1	N/A	N/A	0
2	1/2	BPSK	76 dB
3	1/2	BPSK	79 dB
4	1/2	QPSK	79 dB
5	1/2	BPSK	82 dB
6	1/2	QPSK	82 dB
7	3/4	QPSK	85 dB
8	3/4	QPSK	88 dB

Comparing the channels in the two experiments, one can see that the channel in SPACE08 varies faster than that in KW-NOV14 due to a larger time interval between two consecutive transmissions. Especially about KW-NOV14, the mean of the channel lognormal shadowing per epoch (μ) is quite stable from epoch 30 to 75, and the values of σ on the order of 10^{-3} reveals very slow variation.

B. Emulation Setup and Performance Metric

We consider 8 transmission modes as listed in Table I. Mode 1 refers to no transmission. There are five non-zero discrete transmission power levels according to the listed transmission SNRs (TSNRs) (P_{tx}/N_0). We set the ambient noise level using an empirical formula N_0 [dB] $= 55 + 10\log_{10}(\text{bandwidth})$ *re* $1\mu\text{Pa}^2$ [42], which leads to 94.9 dB for SPACE08 and 92.8 dB for KW-NOV14. For a given transmission mode and a channel parameter triplet $\{\mu, \sigma, m\}$, the PER is computed using an information-theoretic method [36, Eq. (4)].

We define the cost function as

$$C(\mathbf{s}_\ell, \mathbf{a}_\ell) = \log_2(1 + P_\ell/P_{\max}) + \big(q_\ell - r_\ell N_{\text{s},\ell}\big)/r_{\max}, \quad (24)$$

where P_ℓ is the transmission power in the ℓth epoch in Watts, $P_{\max}$ is the maximal transmission power in Watts, and $r_{\max}$ is the maximal amount of information bits that can be carried during one epoch. According to Table I, $r_{\max}$ can be computed based on the mode with the highest data rate, namely, Mode 8, as $r_{\max} = 672 \times \frac{3}{4} \times \log_2 4 \times N_{\text{pa}}$, where 672 is the number of symbols per packet, and N_{pa} is the total number of packets within one epoch.

The average observed cost is used as the performance metric,

$$\bar{C} = \frac{1}{N_{\text{epoch}}} \sum_{\ell=1}^{N_{\text{epoch}}} C(\mathbf{s}_\ell, \mathbf{a}_\ell), \quad (25)$$

where N_{epoch} is the total number of epochs in the algorithm evaluation.

To establish a performance upper bound, we consider a genie-aided transmission scheme with non-causal and perfect knowledge. It assumes that at the beginning of each epoch, the transmitter knows the number of successfully delivered packets corresponding to each transmission action in the current and all the future epochs. With the above knowledge, the system state only consists of the queue state. The optimal action selection can be formulated as a dynamic programming (DP) problem,

$$V^*_{\text{Genie}}(q_\ell) = \min_{\mathbf{a} \in \mathcal{A}} \left[C(q_\ell, \mathbf{a}) + \gamma V^*_{\text{Genie}}(q_{\ell+1}) \right], \quad (26)$$

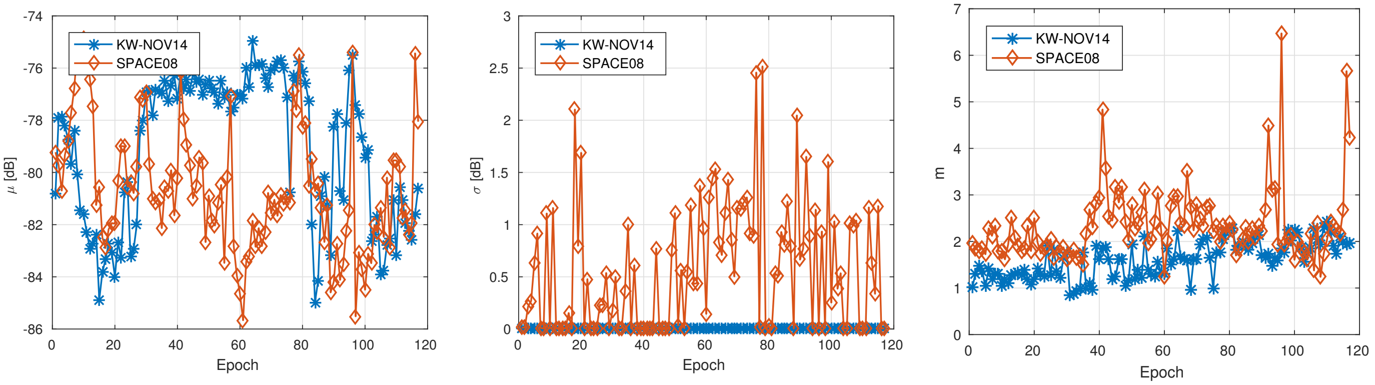


Fig. 3. Estimated parameters $\{\mu, \sigma, m\}$ in two experiments. In KW-NOV14, the estimated σ's are on the order of 10^{-3}.

TABLE II
AVERAGE PERFORMANCE USING THE SPACE08 DATA SET.

	Scheme 1	Scheme 2	Scheme 3	Scheme 4	Scheme 5
Average Queue Length [kilobits]	10.9	13.1	254.8	949.6	7.1
Average Transmission Power [dB]	76.2	81.1	81.3	76.0	84.4
Average Cost	0.47	0.58	4.48	15.79	0.65

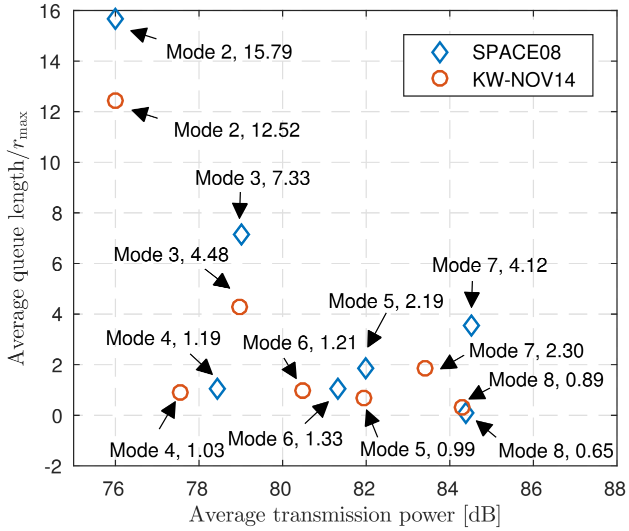


Fig. 4. The performance of fixed-mode transmissions. The number next to each mode is the average cost calculated based on the cost function in (24).

where $C(q_\ell, \mathbf{a})$ is defined as in (24), and $q_{\ell+1}$ and q_ℓ are related as in (4) with perfect knowledge of $N_{\mathrm{s},\ell}$ for a given $\mathbf{a}$. The optimization problem (26) is essentially a deterministic DP problem. However, the DP solver cannot be applied to (26) directly due to the curse of dimensionality [8] induced by the large total number of epochs and a large queue state space. To obtain a near-optimal solution, we modify Algorithm 2 to approximate the value function in (26). Specifically, to approximate the expected cost induced by one action (c.f. Lines 8 to 11 in Algorithm 1), the process of drawing system state samples is replaced by using the true system state directly. Correspondingly, the TD learning is performed based on the true system state instead of a system state sample in the next epoch (c.f. Line 16 in Algorithm 1).

C. General Results

We set the data arrival rate $r_{\mathrm{g}} = 20$ kilobits per epoch for SPACE08 and $r_{\mathrm{g}} = 6$ kilobits per epoch for KW-NOV14. For an epoch with a small queue length, if the number of encoded data packets according to a chosen transmission mode is less than the number of time slots within that epoch (see Fig. 1), the remaining time slots will be used to transmit dummy packets at a very low power level (with TSNR = 70 dB) for the purpose of channel probing. The average packet transmission power will be used to calculate the cost defined in (24).

To shed light on the tradeoff between energy consumption and information delivery latency, Fig. 4 depicts the performance of fixed-mode transmissions in both experiments. According to the cost function defined in (24), Mode 8 achieves the least average cost in both experiments.

We compare in details the performance of five schemes for the transmission action selection.

- *Scheme 1*: The genie-aided transmission scheme;
- *Scheme 2*: The proposed online algorithm;
- *Scheme 3*: Randomly select a transmission action from the action space in each epoch;
- *Scheme 4*: Select the action with the least transmission power and rate, namely, Mode 2, in all epochs;
- *Scheme 5*: Select the action with the highest transmission power and rate, namely, Mode 8, in all epochs.

In the proposed algorithm, the number of the child system nodes for each action N_{o}, the number of the actions to be explored N_{a}, and the planning depth D are set to be 3, 3, and 5, respectively. We set the discount factor $\gamma = 0.8$ in both the genie-aided scheme and the proposed algorithm. The unknown channel model parameters in the first epoch $\hat{\mathbf{\Theta}}_0$ are initialized as $\hat{\mathbf{A}}_0 = \hat{\mathbf{C}}_{\mathrm{w},0} = \hat{\mathbf{C}}_{\mathrm{v},0} = \mathrm{diag}([1, 1, 1])$. We set the forgetting factor $\gamma_{\mathrm{ch}} = 0.8$ and the number of iterations $N_{\mathrm{iter}} = 20$. The learning rates for the TD learning and the LR, i.e., α and β, are set to be 0.01 and 0.01, respectively. The regularization parameter λ is set to be 1. The initial values of

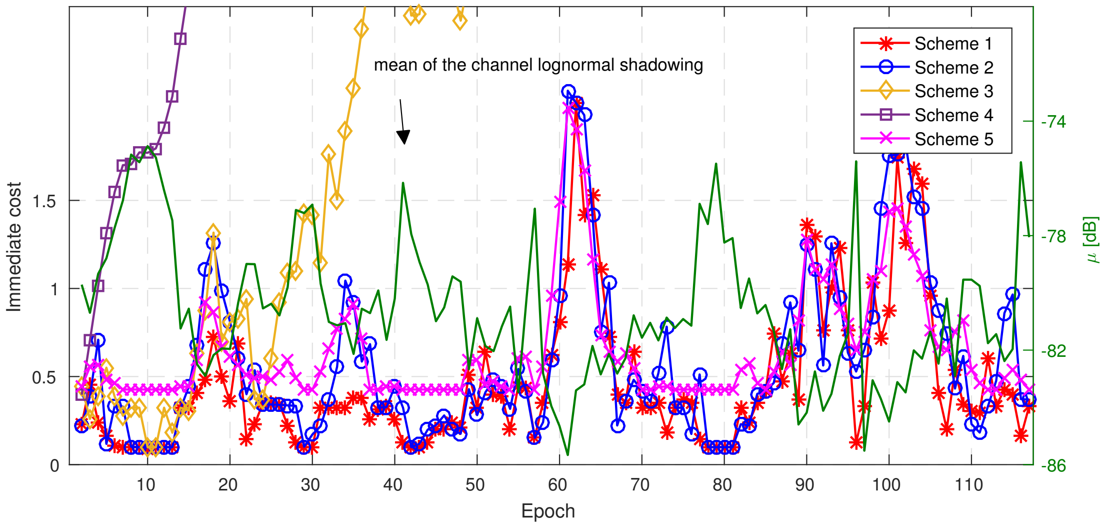


Fig. 5. SPACE08: The mean of the channel lognormal shadowing and immediate collected costs by different schemes.

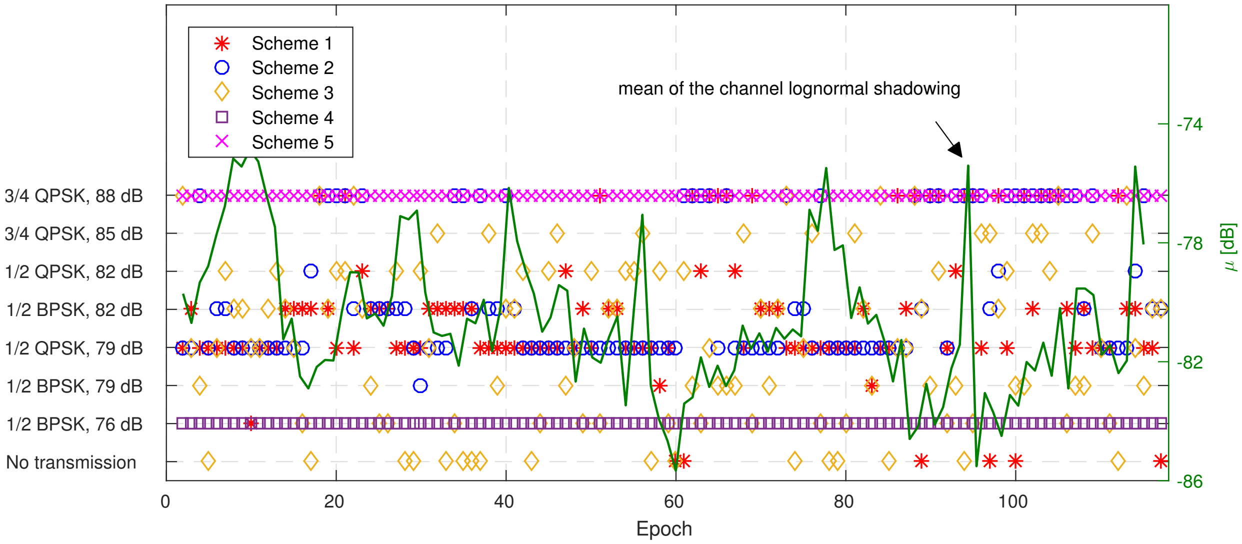


Fig. 6. SPACE08: The mean of the channel lognormal shadowing and selected actions in different schemes.

all the elements in ϕ are set as 0. The number of channel state samples to calculate the expected immediate cost is set to be 100.

1) SPACE08: The performance of different schemes is shown in Table II. It can be seen that the proposed algorithm has the least performance gap with the genie-aid method. Schemes 3 and 4 suffer from very large average queue lengths. Compared to the proposed algorithm, Scheme 5 has a smaller average queue length but requires more average transmission power.

The immediate costs per epoch of different schemes are shown in Fig. 5. One can see that the immediate cost of the proposed algorithm is close to that of the genie-aided method. With the immediate costs fluctuating with the mean of the channel lognormal shadowing, the proposed algorithm and the genie-aided method are able to maintain low costs when the average channel loss is small (i.e., when μ is large). When the average channel loss is large, the proposed algorithm still can maintain relatively low immediate costs. The immediate costs of Schemes 3 and 4 increase drastically due to the random selection of transmission actions in Scheme 3 and the adoption of the least transmission power and data rate in Scheme 4. Scheme 5 has larger immediate costs than the proposed algorithm and the genie-aided method in most epochs, due to its adoption of the largest transmission power.

The actions selected by different schemes are shown in Fig. 6. The proposed algorithm and the genie-aided method prefer in most epochs the transmission action with a moderate transmission power level and a moderate data rate, i.e., 1/2 QPSK and 79 dB. In the epochs with large channel losses, the proposed algorithm opts for the transmission actions with larger transmission power levels to suppress the increase of the data queue length.

The channel state vector estimation and the normalized root mean squared error (NRMSE) of the estimation are depicted in Fig. 7. The results reveal that the proposed channel model

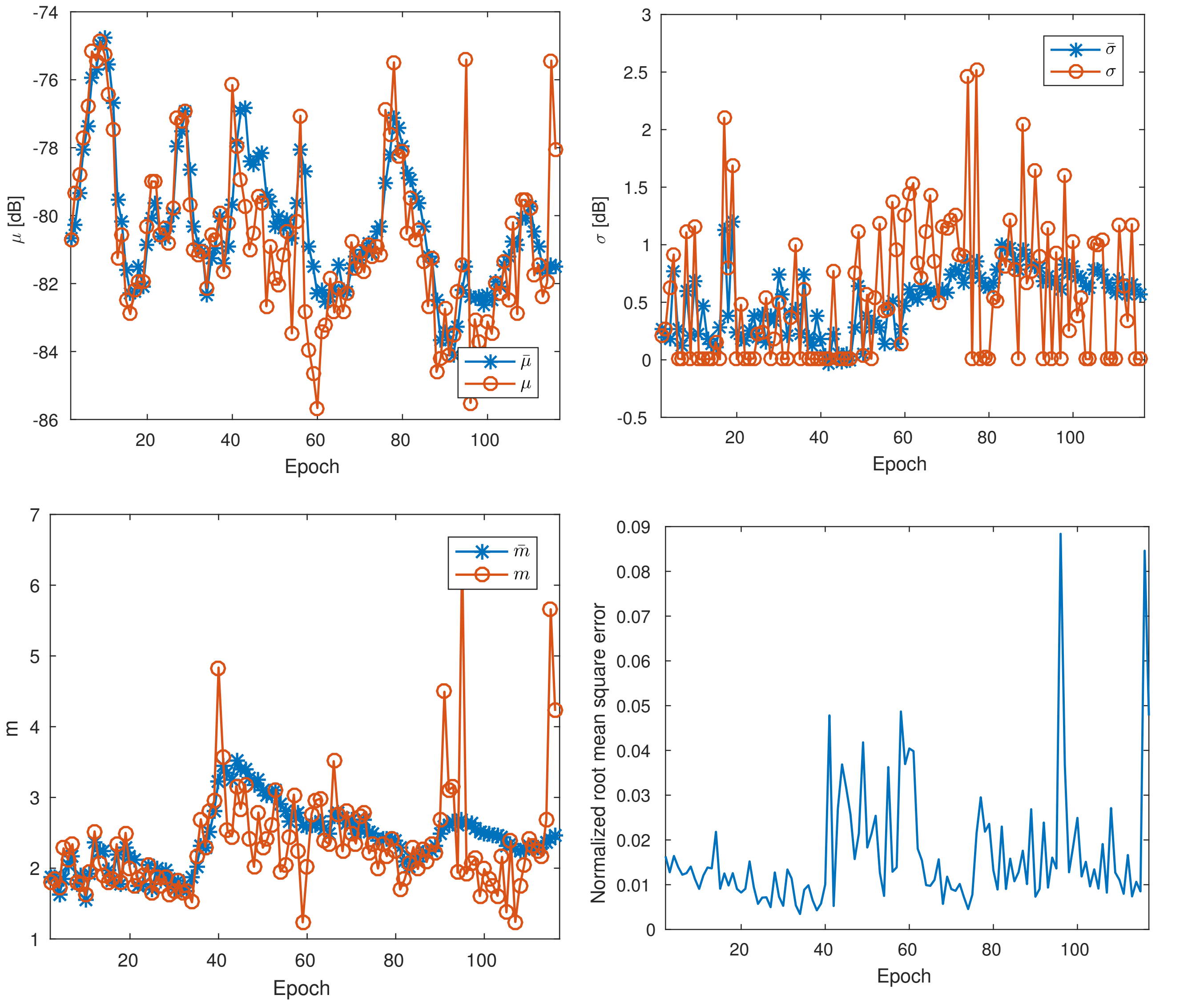


Fig. 7. SPACE08: Comparison between the mean $(\bar{\mu}, \bar{\sigma}, \bar{m})$ of the estimated channel belief state and the true channel state (μ, σ, m), and the NRMSE.

can capture the channel dynamics reasonably well, and the NRMSE less than 0.1 in each epoch shows the superior performance of the proposed recursive estimation algorithm.

2) KW-NOV14: The performance of different schemes is shown in Table III. It can be seen that the proposed algorithm has the least performance gap with the genie-aid method. Schemes 3 and 4 suffer from large average queue lengths. Although Scheme 5 has a small average queue length, it requires the most average transmission power among all schemes.

The immediate costs and actions of different schemes are shown in Figs. 8 and 9, respectively. The immediate cost of Scheme 4 grows drastically due to its adoption of the least transmission power and data rate. Schemes 3 and 5 have larger immediate costs than the proposed algorithm and the genie-aided method in most epochs. The immediate cost of the proposed algorithm is close to that of the genie-aided method. A large performance gap between the proposed algorithm and the genie-aided method can be observed during epochs 8 to 32. Due to large channel dynamics and large channel losses in those epochs, the immediate cost of the proposed algorithm grows greater than that of the genie-aided method which can adapt the transmission mode more precisely. A little lag around epoch 30 can be observed between the changes of the immediate costs of those two schemes. During epochs 32 to 85, the immediate costs obtained by the proposed algorithm and the genie-aided method are almost identical, as the transmitter in the proposed algorithm has learned adequate channel knowledge. Both schemes prefer the transmission action with a moderate transmission power level and a moderate data rate, i.e., 79 dB and 1/2 QPSK.

The channel state vector estimation and the NRMSE of the estimation are depicted in Fig. 10. Similar to the case in SPACE08, the difference between the mean values of the channel belief state and the true channel states is small and the NRMSE is less than 0.1 in every epoch. The results validate the effectiveness of the proposed recursive estimation algorithm.

D. Performance of the Proposed Algorithm with Different System Setups

The performance of the proposed algorithm is examined in different system setups, including different data arrival rates from the application layer, different numbers of child system state samples in online approximation, different numbers of actions to be explored, and different depths of the state-action tree, in the Monte Carlo planning.

To quantify the performance of the proposed algorithm in different setups, we take the performance of the genie-aided scheme as a benchmark, and evaluate the *normalized difference* which is defined as $\big(\bar{C} - \bar{C}_{\rm Genie}\big)/\bar{C}_{\rm Genie}$, where $\bar{C}$ is the average cost defined in (25). For comparison purpose, $\bar{C}_{\rm Genie}$ is obtained based on $N_{\rm o} = 3$, $N_{\rm a} = 3$, and $D = 5$.

1) Performance with different data arrival rates: The data arrival rate will impact the performance of the proposed algorithm. As the data arrival rate increases, both the proposed algorithm and the genie-aided method prefer the transmission modes with high data rates to suppress the increase of the data queue length. Without precise channel knowledge, there are high chances that the proposed algorithm could schedule high-data-rate transmissions in epochs with bad channel conditions. Consequently, the proposed algorithm suffers an increased performance gap with the genie-aided method that determines the transmission actions based on non-causal and perfect knowledge. Fig. 11 shows the normalized performance difference of the proposed algorithm w.r.t. the genie-aided method with different data arrival rates. It can be seen that as the data arrival rate increases from a small value to a moderately large value, the normalized difference increases. However, with further increase of the data arrival rate, the normalized difference starts decreasing. This is caused by the large value of the average cost $\bar{C}_{\rm Genie}$ that increases monotonically with the data arrival rate.

2) Performance with different numbers of child system state samples and actions to be explored in online approximation: The normalized performance difference of the proposed algorithm w.r.t. the genie-aided method with different numbers of child system state samples and different numbers of actions to be explored in online approximation are shown in Fig. 12(a) and Fig. 12(b), respectively. The performance improvement is minor with the increase of the numbers of child system state samples and actions to be explored. This indicates that with a small number of child system state samples and a small number of actions to be explored, the proposed algorithm can achieve good online approximation performance with a low computational complexity.

3) Performance with different depths of Monte Carlo planning: The depth of Monte Carlo planning is a key factor in the tradeoff between the approximation accuracy and the computational complexity; see Section IV-B. Fig. 12(c) shows the normalized performance difference of the proposed algorithm w.r.t. the genie-aided method with different planning depths. It can be seen that considerable performance improvement is achieved when the depth of planning is increased from 1 to 2 in SPACE08 and from 1 to 3 in KW-NOV14. Further increase of the planning depth in both experiments leads to slight performance improvement, which, however, is accompanied with exponentially increased computational cost. The results demonstrate that the proposed algorithm achieves decent performance with a small depth of planning since it stores and exploits the historical knowledge of the value function via the TD learning and the LR when evaluating the future expected costs.

VII. CONCLUSIONS

This work focused on an UWA point-to-point transmission system which operates on an epoch-by-epoch basis over a long term, and developed an adaptive transmission algorithm which exploits the UWA channel dynamics to trade off energy consumption with information delivery latency. To describe both the short-term fading and the large-scale shadowing of UWA channels, the Nakagami-lognormal distribution was adopted for channel characterization. To account for the channel variation across epochs, the evolution of the channel distribution parameters was modeled as a Markov process with unknown parameters. Given that the channel can only be observed during active transmissions, we formulated the adaptive transmission problem as a POMDP to strike an optimal tradeoff between learning the channel dynamics via active transmissions and exploiting the learned channel knowledge for transmission efficiency. An algorithm in the model-based RL framework was developed, which recursively estimates the channel model parameters and computes the optimal transmission strategy that minimizes a long-term system cost. Thorough algorithm evaluation was performed using channel measurements from two field experiments. The emulated results revealed that the proposed algorithm achieves decent performance relative to a benchmark method that assumes perfect and non-causal channel knowledge.

REFERENCES

[1] I. F. Akyildiz, D. Pompili, and T. Melodia, “Underwater acoustic sensor networks: Research chanllenges,” *Ad Hoc Networks*, vol. 3, pp. 257–279, 2005.

[2] “Ocean Networks Canada,” http://www.oceannetworks.ca/.

[3] “The Ocean Observatory Initiative (OOI),” http://oceanobservatories.org/.

[4] D. V. Djonin and V. Krishnamurthy, “MIMO transmission control in fading channels - A constrained Markov decision process formulation with monotone randomized policies,” *IEEE Trans. Signal Processing*, vol. 55, no. 10, pp. 5069–5083, Oct. 2007.

[5] M. H. Ngo and V. Krishnamurthy, “Optimality of threshold policies for transmission scheduling in correlated fading channels,” *IEEE Trans. Commun.*, vol. 57, no. 8, pp. 2474–2483, Aug. 2009.

[6] R. Srivastava and C. E. Koksal, “Energy optimal transmission scheduling in wireless sensor networks,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 5, pp. 1550–1560, May 2010.

[7] N. Ding, P. Sadeghi, and R. A. Kennedy, “On monotonicity of the optimal transmission policy in cross-layer adaptive m-QAM modulation,” *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3771–3785, Sep. 2016.

[8] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. Cambridge:Cambridge Univ. Press, 2011.

[9] C. Pandana and K. Liu, “Near-optimal reinforcement learning framework for energy-aware sensor communications,” *IEEE J. Select. Areas Commun.*, vol. 23, no. 4, pp. 788–797, Apr. 2005.

[10] D. V. Djonin and V. Krishnamurthy, “Q-learning algorithms for constrained Markov decision processes with randomized monotone policies: Application to MIMO transmission control,” *IEEE Trans. Signal Processing*, vol. 55, no. 5, pp. 2170–2181, May 2007.

[11] N. Salodkar, A. Bhorkar, A. Karandikar, and V. S. Borkar, “An on-line learning algorithm for energy efficient delay constrained scheduling over a fading channel,” *IEEE J. Select. Areas Commun.*, vol. 26, no. 4, pp. 732–742, May 2008.

TABLE III
AVERAGE PERFORMANCE USING THE KW-NOV14 DATA SET.

	Scheme 1	Scheme 2	Scheme 3	Scheme 4	Scheme 5
Average Queue Length [kilobits]	4.9	11.4	31.3	25.1	7.0
Average Transmission Power [dB]	74.4	75.6	80.8	76.0	84.3
Average Cost	0.40	0.76	1.81	12.52	0.89

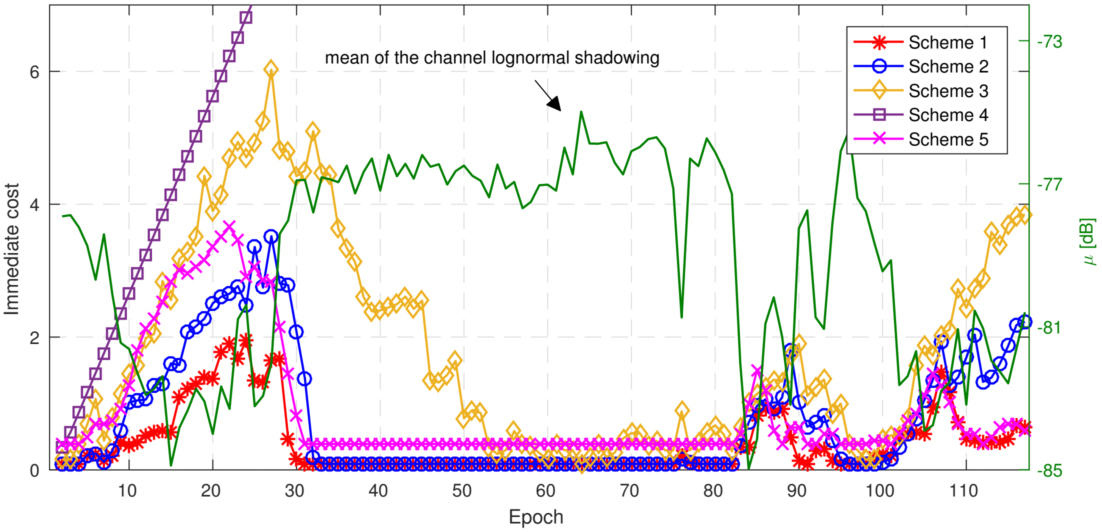


Fig. 8. KW-NOV14: The mean of the channel lognormal shadowing and immediate collected costs by different schemes.

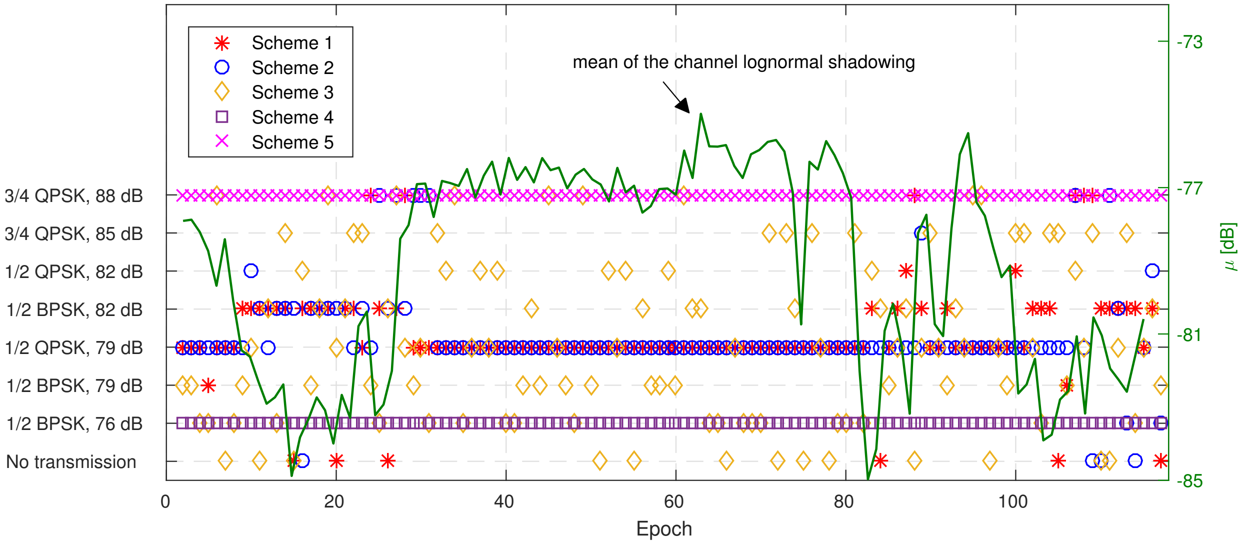


Fig. 9. KW-NOV14: The mean of the channel lognormal shadowing and selected actions in different schemes.

[12] M. H. Ngo and V. Krishnamurthy, "Monotonicity of constrained optimal transmission policies in correlated fading channels with ARQ," *IEEE Trans. Signal Processing*, vol. 58, no. 1, pp. 438–451, Jan. 2010.

[13] N. Mastronarde and M. van der Schaar, "Joint physical-layer and system-level power management for delay-sensitive wireless communications," *IEEE Trans. Mobile Computing*, vol. 12, no. 4, pp. 694–709, Apr. 2013.

[14] L. Lei, Y. Kuang, X. S. Shen, K. Yang, J. Qiao, and Z. Zhong, "Optimal reliability in energy harvesting industrial wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 8, pp. 5399–5413, Aug. 2016.

[15] I. Ahmed, K. T. Phan, and T. Le-Ngoc, "Optimal stochastic power control for energy harvesting systems with delay constraints," *IEEE J. Select. Areas Commun.*, vol. 34, no. 12, pp. 3512–3527, Dec. 2016.

[16] D. T. Hoang, D. Niyato, P. Wang, D. I. Kim, and L. B. Le, "Optimal data scheduling and admission control for backscatter sensor networks," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2062–2077, May 2017.

[17] M. Hirzallah, W. Afifi, and M. Krunz, "Full-duplex-based rate/mode adaptation strategies for Wi-Fi/LTE-U coexistence: A POMDP approach," *IEEE J. Select. Areas Commun.*, vol. 35, no. 1, pp. 20–29, Jan. 2017.

[18] A. Radosevic, R. Ahmed, T. M. Duman, J. G. Proakis, and M. Stojanovic, "Adaptive OFDM modulation for underwater acoustic communications: Design considerations and experimental results," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 2, pp. 357–370, Apr. 2014.

[19] Y. M. Aval, S. K. Wilson, and M. Stojanovic, "On the achievable rate of a class of acoustic channels and practical power allocation strategies for OFDM systems," *IEEE Journal of Oceanic Engineering*, vol. 40, no. 4, pp. 785–795, Oct. 2015.

[20] L. Wan, H. Zhou, X. Xu, Y. Huang, S. Zhou, Z. Shi, and J.-H. Cui, "Adaptive modulation and coding for underwater acoustic OFDM," *IEEE*

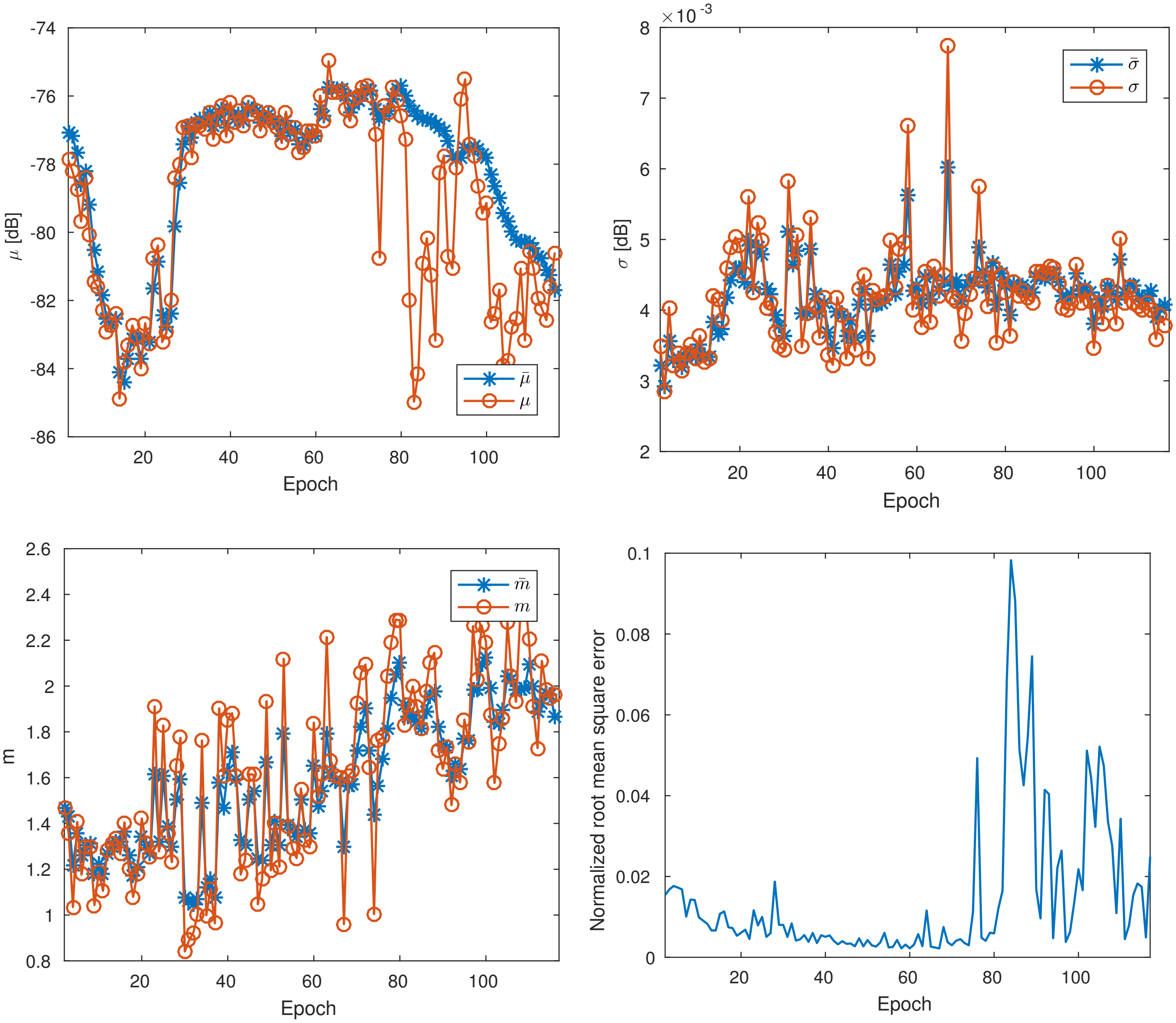


Fig. 10. KW-NOV14: Comparison between the mean ($\bar{\mu}, \bar{\sigma}, \bar{m}$) of the estimated channel belief state and the true channel state (μ, σ, m), and the NRMSE.

J. Ocean. Eng., vol. 40, no. 2, pp. 327–336, Apr. 2015.

[21] E. Demirors, J. Shi, R. Guida, and T. Melodia, "SEANet G2: Toward a high-data-rate software-defined underwater acoustic networking platform," in *Proc. of the ACM Intl. Conf. on Underwater Networks (WUWNet)*, Oct. 2016.

[22] B. Tomasi and J. C. Preisig, "Energy-efficient transmission strategies for delay constrained traffic with limited feedback," *IEEE Trans. Wireless Commun.*, vol. 3, no. 14, pp. 1369–1379, Mar. 2015.

[23] ——, "Efficient heuristic scheduling with partial queue and channel state information," in *Proc. of the ACM Intl. Conf. on Underwater Networks (WUWNet)*, Rome, Italy, Sep. 2014.

[24] L. Jin and D. D. Huang, "A slotted CSMA based reinforcement learning approach for extending the lifetime of underwater acoustic wireless sensor networks," *Computer Communications*, vol. 36, no. 9, pp. 1094–1099, 2013.

[25] V. D. Valerio, C. Petrioli, L. Pescosolido, and M. V. D. Shaar, "A reinforcement learning-based data-link protocol for underwater acoustic communications," in *Proc. of the ACM Intl. Conf. on Underwater Networks (WUWNet)*, Washington DC, Oct. 2015.

[26] T. Hu and Y. Fei, "QELAR: A machine-learning-based adaptive routing protocol for energy-efficient and lifetime-extended underwater sensor networks," *IEEE Trans. Mobile Computing*, vol. 9, no. 6, pp. 796–809, Jun. 2010.

[27] N. M. Carbone and W. S. Hodgkiss, "Effects of tidally driven temperature fluctuations on shallow-water acoustic communications at 18 kHz," *IEEE J. Ocean. Eng.*, vol. 25, no. 1, pp. 84–94, Jan. 2000.

[28] M. Badiey, Y. Mu, J. A. Simmen, and S. E. Forsythe, "Signal variability in shallow-water sound channels," *IEEE J. Ocean. Eng.*, vol. 25, no. 4, pp. 492–500, Oct. 2000.

[29] A. Song, M. Badiey, H. C. Song, W. S. Hodgkiss, and M. B. Porter, "Impact of ocean variability on coherent underwater acoustic communications during the Kauai experiment (KauaiEx)," *J. Acoust. Soc. Am.*, vol. 123, no. 2, pp. 856–865, Feb. 2008.

[30] P. van Walree, "Propagation effects in underwater acoustic communication channels," in *Proc. of the Workshop on Underwater Communications: Channel Modelling & Validation*, Italy, Sep. 2012.

[31] P. Qarabaqi and M. Stojanovic, "Statistical characterization and computationally efficient modeling of a class of underwater acoustic communication channels," *IEEE J. Ocean. Eng.*, vol. 38, no. 4, pp. 701–717, Oct. 2013.

[32] J. Llor and M. P. Malumbres, "Statistical modeling of large-scale signal path loss in underwater acoustic networks," *Sensors*, vol. 13, no. 2, pp. 2279–2294, 2013.

[33] B. Tomasi, P. Casari, L. Badia, and M. Zorzi, "A study of incremental redundancy hybrid ARQ over Markov channel models derived from experimental data," in *Proc. of the ACM Intl. Workshop on Underwater Networks (WUWNet)*, Woods Hole, MA, Sep. 2010.

[34] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. New Jersey: Prentice Hall, 1993, vol. 2.

[35] S. Atapattu, C. Tellambura, and H. Jiang, "A mixture Gamma distri-

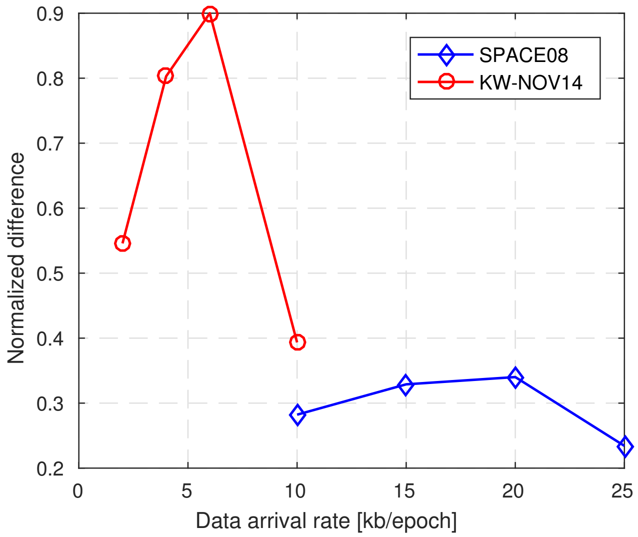


Fig. 11. Normalized difference with respect to the genie-aided method with different data arrival rates with $N_{\rm o} = 3$, $N_{\rm a} = 3$, and $D = 5$.

bution to model the SNR of wireless channels," *IEEE Trans. Wireless Commun.*, vol. 10, no. 12, pp. 4193–4203, Dec. 2011.

[36] B. Lu, G. Yue, and X. Wang, "Performance analysis and design optimization of LDPC-coded MIMO OFDM systems," *IEEE Trans. Signal Processing*, vol. 52, no. 2, pp. 348–361, Feb. 2004.

[37] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. 28, no. 1, pp. 55–67, Jan. 1982.

[38] M. Ghavamzadeh, S. Mannor, J. Pineau, and A. Tamar, "Bayesian reinforcement learning: A survey," *Found. Trends. Mach. Learn.*, vol. 8, no. 5–6, pp. 359–483, 2015.

[39] M. Kearns, Y. Mansour, and A. Ng, "A sparse sampling algorithm for near-optimal planning in large Markov decision processes," *Machine Learning*, vol. 49, no. 2, pp. 193–208, Nov. 2002.

[40] A. Guez, D. Silver, and P. Dayan, "Scalable and efficient Bayes-adaptive reinforcement learning based on Monte-Carlo tree search," *Journal of Artificial Intelligence Research*, vol. 48, pp. 841–883, Nov. 2013.

[41] C. M. Bishop, *Pattern Recognition and Machine Learning*, 6th ed. Springer-Verlag New York, 2006.

[42] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 84–89, Jan. 2009.

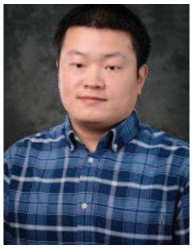
Chaofeng Wang received the B.S. degree in 2010 and M.Sc. degree in 2012, from Southwest Jiaotong University (SWJTU), Chengdu, China, in applied mathematics and vehicle application engineering, respectively. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering at Michigan Technological University (MTU), Houghton MI, USA.

His research interests lie in the areas of signal processing, communications, and machine learning, with recent focus on statistical learning for underwater acoustic communications.

Zhaohui Wang (S'10 - M'13) received her B.S. degree in 2006 from Beijing University of Chemical Technology (BUCT), an M.S. degree in 2009 from the Institute of Acoustics, Chinese Academy of Sciences (IACAS), Beijing, China, and a Ph.D. degree in 2013 from the University of Connecticut (UCONN), Storrs, all in electrical engineering.

Dr. Wang has been with the Department of Electrical and Computer Engineering at Michigan Technological University (Michigan Tech), Houghton, as an assistant professor since 2013. Her research interests lie in the areas of wireless communications, networking, and statistical signal processing, with recent focus on signal processing techniques for wireless communications and networking in underwater acoustic environments.

Dr. Wang served as technical reviewers for many premier journals and conferences. She was recognized as an Outstanding Reviewer by the IEEE Journal of Oceanic Engineering in 2012, 2013-2014, 2015 and 2016 respectively. In 2013, she was honored with the Women of Innovation Award by the Connecticut Technology Council. Dr. Wang is a recipient of the NSF CAREER award in 2017.



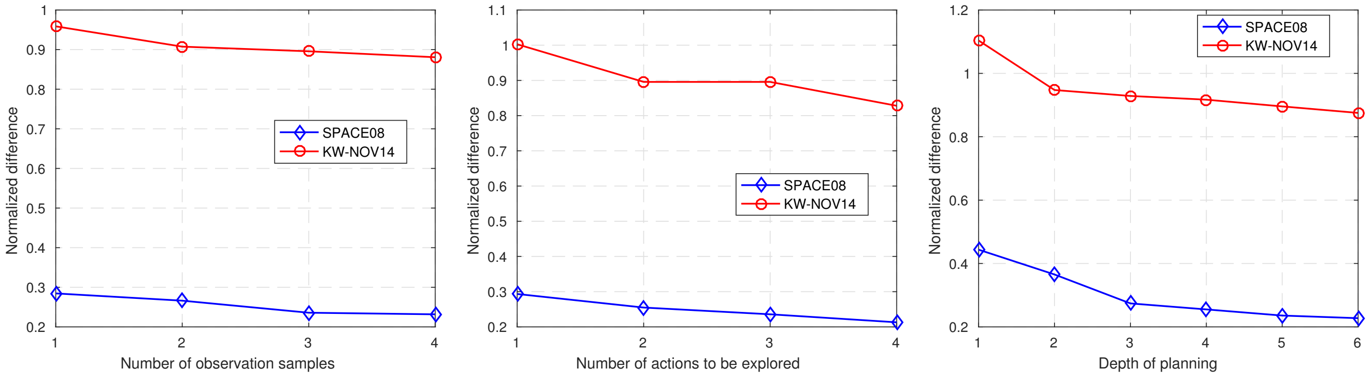
Wensheng Sun received the B.S. degree in 2010, from the Yanshan University (YSU), and the M.Sc. degree in 2012, from the Harbin Institute of Technology (HIT), Harbin, China, both in electrical engineering. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering at the Michigan Technological University (MTU), Houghton, USA.

His research interests lie in the areas of communications, detection and estimation, with the recent focus on online learning and inversion for underwater acoustic communications.

Daniel R. Fuhrmann (S'78-M'84-SM'95-F'10) received the B.S.E.E. degree (cum laude) from Washington University, St. Louis, MO, USA, in 1979 and the M.A., M.S.E., and the Ph.D. degrees from Princeton University, Princeton, NJ, in 1982 and 1984, respectively.

From 1984 to 2008, he was with the Department of Electrical Engineering, now the Department of Electrical and Systems Engineering, Washington University. In the 2000-2001 academic year, he was a Fulbright Scholar visiting the Universidad Nacional de La Plata, Buenos Aires, Argentina. He is currently the Dave House Professor and Chair of the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI. His research interests lie in various areas of statistical signal and image processing, including sensor array signal processing, radar systems, and adaptive sensing.

Dr. Fuhrmann is a former Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He was the Technical Program Chairman for the 1998 IEEE Signal Processing Workshop on Statistical Signal and Array Processing, and General Chairman of the 2003 IEEE Workshop on Statistic-al Signal Processing. He has been an ASEE Summer Faculty Research Fellow with the Naval Underwater Systems Center, New London, CT, a consultant to MIT Lincoln Laboratory, Lexington, MA, and an ASEE Summer Faculty Fellow with the Air Force Research Laboratory, Dayton, OH.



(a) With $N_\mathrm{a} = 3$ and $D = 5$

(b) With $N_\mathrm{o} = 3$ and $D = 5$

(c) With $N_\mathrm{o} = 3$ and $N_\mathrm{a} = 3$

Fig. 12. Normalized difference with respect to the genie-aided method with different Monte Carlo planning parameters. $r_\mathrm{g} = 20$ kb/epoch in SPACE08, and $r_\mathrm{g} = 6$ kb/epoch in KW-NOV14.