Reinforcement Learning for Multi-Hop Scheduling and Routing of Real-Time Flows

Aria HasanzadeZonuzy
Dept. of ECE
Texas A&M University
azonuzy@tamu.edu

Dileep Kalathil
Dept. of ECE
Texas A&M University
dileep.kalathil@tamu.edu

Srinivas Shakkottai
Dept. of ECE
Texas A&M University
sshakkot@tamu.edu

***Abstract*—We consider the problem of serving real-time flows over a multi-hop wireless network. Each flow is composed of packets that have strict deadlines, and the goal is to maximize the weighted timely throughput of the system. Consistent with recent developments using mm-wave communications, we assume that the links are directional, but are lossy, and have unknown probabilities of successful packet transmission. An average link utilization budget (similar to a power constraint) constrains the system. We pose the problem in the form of a Constrained Markov Decision Process (CMDP) with an unknown transition kernel. We use a duality approach to decompose the problem into an inner unconstrained MDP with link usage costs, and an outer link-cost update step. For the inner MDP, we develop model-based reinforcement learning algorithms that sample links by sending packets to learn the link statistics. While the first algorithm type samples links at will at the beginning and constructs the model, the second type is an online approach that can only use packets from flows to sample links that they traverse. The approach to the outer problem follows gradient descent. We characterize the sample complexity (number of packets transmitted) to obtain near-optimal policies, to show that a basic online approach has a poorer sample complexity bound, it can be modified to obtain an online algorithm that has excellent empirical performance.**

I. Introduction

The next generation of cellular communication networks that are taking shape under the moniker of 5G are expected to provide ultra-low latency, ultra-high throughput communications to support a variety of real-time applications. They will utilize a large bandwidth from the directional and loss-prone mm wave spectrum. In particular, the low coherence time of these bands implies that channel state must be dynamically determined, and the approach to effective utilization of these channels lies in learning link statistics as they change, and employing those that are most likely to be able to support the offered loads.

Some of the first planned deployments using mm-wave spectrum are using the *Integrated Access and Backhaul* (IAB) architecture [1], [2]. As the name suggests, the user-equipment (UE) is provided access via mm-wave spectrum. However, a fundamental departure with the current cellular architecture is a dense deployment of small cells, with a small number of base stations acting as gateways being equipped with fiber backhaul, some base stations being connected purely over mm-wave backhaul, and a large number of inexpensive analog repeaters adding to the mm-wave backhaul creating many additional links to account for occlusion effects. These wireless backhaul links are directional, and essentially interference free [2]. However, their reliance on mm-wave implies that their statistics are prone to change, which requires dynamic packet routing that accounts for these changes.

The availability of a large, dynamically changing spectrum band with many links, and also the fact that the analog repeaters cannot generate packets of their own implies that the traditional approach of sounding each link using pilot packets is untenable, and an online learning approach to efficiently sample the system is called for. With this motivation in mind, the goal of this work is to address the problem of learning how to maximize throughput, while meeting hard end-to-end deadline guarantees over multihop wireless networks composed of unreliable links. The real-time nature of the flows implies that packets that are not delivered by the deadline are simply dropped. In the IAB context, the model pertains to the problem of routing such deadline constrained packets across the IAB base stations to the one connected to the end-user. Once this happens, the packet is immediately transmitted to the requesting end-user.

The problem of maximizing timely throughput can be posed in the manner of reinforcement learning (RL) over a Constrained Markov Decision Process (CMDP). Here, the state of the system is the tuple of location and remaining lifetime of each packet, and a unit reward is obtained each time that an unexpired packet is delivered successfully to end-user. The available actions are the choices of links that can be used for forwarding the packet at each node, and the randomness of the MDP kernel stems from the randomness of the links. The constraints of this problem are on the number of tranmissions permissible per link at each time, while the fact that the

Research was supported in part by grants NSF CRII:CPS-1850206, ECCS-1839816, NSF-Intel CNS-1719384, ARO W911NF-19-1-0367 and W911NF-19-2-024.

ISBN 978-3-903176-29-4 © 2020 IFIP

probabilities of success or failure at each link is unknown implies the need for a learning approach.

Multiple challenges must be addressed to successfully solve the CMDP problem of deadline constrained flows. First, online reinforcement learning must be employed to estimate the link reliabilities using as few packets as possible. Second, we must ensure that per-packet deadline guarantees are met. Finally, it is untenable to solve a global MDP that requires state information about every packet and node in the system, and a simple distributed implementation of the policy is desired.

Main Results

We build on a framework of a general solution methodology for constrained MDPs using a dual decomposition approach of Altman [3]. Here, the CMDP problem is solved via a two step procedure of (i) maximizing the objective (solving an MDP) under fixed Lagrange multipliers corresponding to the constraints, and (ii) a gradient descent step over the Lagrange multipliers. This approach is tailored to the case of deadline constrained flows in Singh et al. [4]. The main insight of Singh et al. [4] is that the CMDP for timely throughput maximization under an average link utilization constraint (number of tranmissions allowed per time slot) decomposes into a simpler set of per-packet MDPs, thus permitting a distributed solution. However, the work assumes that the transition kernel of the MDP, which depends on the success probabilities of links is known apriori.

Our work is perhaps the first to consider a learning approach towards solving constrained MDPs in the context of optimal wireless scheduling design. The main contribution of our work is to design algorithms that explicitly account for the overhead of learning link reliabilities while computing the optimal packet and link scheduling policy. We follow the general theme of *model-based reinforcement learning,* under which the intent is to efficiently determine the transition kernel of the MDP under study, and explicitly solve it to obtain the optimal policy. This approach is particularly suited to our problem, as it has a well defined structure under which the unknown sources of randomness in the system are parametrized by the success probabilities of the links. Our performance analysis goal is to characterize the so-called *sample complexity* of our algorithms, *i.e.,* we wish to determine the number of packet transmissions needed to ensure that the value of the packet transmission policy differs from that of the optimal policy at most by a parameter ϵ with a high probability.

Our first algorithm entitled *Generative Model-Based Learning (GMBL)* follows a procedure under which each link is sampled a given number of times to determine its statistics to a desired level of accuracy, and the resulting (noisy) model of the system is used as an input to the CMDP framework of Altman [3]. Although difficult to actually implement (since all nodes have to generate packets on their own to sample links), this approach sets up both the analytical methodology and a baseline sample complexity bound. The main result here is that the sample complexity is proportional to the number of links in the system, which is consistent with the number of unknown parameters (link success probabilities). Furthermore, the sample complexity is inversely proportional to the square of the desired accuracy ϵ.

Our second algorithm entitled *Constrained Model Based Interval Estimation (Con-MBIE)* is an extension of the MBIE algorithm [5], appropriately modified to the finite horizon [6]. Con-MBIE follows a procedure of targeted routing of packets to learn link parameters of the most attractive links, and solves for an $\epsilon-$optimal policy under the model. Since Con-MBIE learns the model to a desired accuracy for each given value of the Lagrange multipliers, it has to re-learn the model for each change in the Lagrange multipliers, resulting in a higher bound on sample complexity than GMBL. However, empirically it turns out that this tradeoff between the focus on attractive links (much like an online algorithm that focuses on high-value arms in a multi-armed bandit setting), and re-learning the model for each Lagrange multiplier update balance out, and the performance is very close to GMBL.

Finally, we propose a heuristic algorithm, Con-MBIE-RU, which is identical to Con-MBIE, except that it cumulatively uses all data samples gathered to progressively increase the model accuracy as the Lagrange multipliers are updated at each step (i.e., it ”re-uses” samples over updates). It thus retains the targeted learning idea from Con-MBIE, while ensuring high model accuracy by using all samples in the manner of GMBL. We show empirically that Con-MBIE-RU is near-optimal, and significantly outperforms both GMBL and Con-MBIE in terms of sample complexity and timely throughput attained.

Our numerical evaluation is over topologies similar to those proposed for IAB trails [2]. We compare our RL-based algorithms with the optimal solution value assuming that the model (link success probabilities) are known to show how the accuracy improves with increasing sample complexity. We also compare the sample complexities of the variants of our algorithms to compare their efficiencies of link sampling, and show how Con-MBIE-RU uses its samples effectively.

II. Related Work

There has been much work in the past several years on provably throughput optimal scheduling policies, starting with seminal work of Tassiulas et al. [7], and follow up works [8], [9] leading to the so-called backpressure type scheduling policies. Recent work in this space has focused on throughput optimal broadcast under networks with different topologies [10], [11]. With the rise of real-time streaming applications that require hard delay guarantees, a different approach is needed as backpressure cannot provide delay optimality. Work in this space focuses on scheduling such real-time flows, wherein an MDP formulation is avoided due to the emphasis on a single (typically downlink) wireless hop [12], [13].

The design of scheduling algorithms that can support hard deadline constrains in the multi-hop context has been the topic of recent study. For instance, Xiong et al. [14] introduce delay-awareness into the protocol, without, however, enabling hard deadline guarantees. Other work, such as that by Mao et al. [15] provide such guarantees under fixed routing, while that by Li et al. [16] is only able to do so in a heuristic manner without optimality guarantees. The fundamental issue here is the need to solve a global MDP for taking scheduling/routing decisions, and the work of Singh et al. [4] is the first to use an average link utilization constraint to enable a simple and distributed solution. The approach has been further generalized to the broadcast setting by HasanzadeZonuzy et al. [17].

The use of AI methods in communication networks has recently been the subject of much interest, with most work focusing on bandit-style approaches to learning the sources of randomness in the system. For example, Krishnasamy et al. [18] use posterior sampling with some additional learning effort in order to small queuing regret in a system with a single queue and many wireless channel. Combes et al. [19] and Gupta et al. [20] both use a marginal posterior sampling approach in the context of power allocation in the context of a system in which channel statistics are unknown. Talebi et al. [21] also consider a bandit approach to routing over links whose statistics are unknown.

Unlike the above body of work on learning in wireless networks, our problem of delay constrained unicast flows does not admit a bandit-type of solution due to the hard delay constraint that implies that the state of each packet in the system consists of both a location and a time to live. Hence, while the source of randomness in our problem lies in unreliable links (like earlier work), our formulation is very different and takes the form of a constrained MDP that explicitly accounts for state, rather than the bandit formulation considered earlier.

III. Problem Formulation

In this section, we formally describe our model and the constrained MDP (CMDP) formulation for maximizing the weighted timely throughput of the system. The setup is similar to Singh et al. [4], and employs the relaxed transmission constraint and Lagrangian decomposition technique proposed in that work to obtain simple per-packet MDPs that are conducive to the RL approach that will be developed in the next section.

A. System Model

We consider a communication network described by a directed graph $\mathcal{G} = (\mathcal{S}, \mathcal{L})$, where $\mathcal{S}$ is the set of nodes and $\mathcal{L}$ is the set of links. The cardinality of $\mathcal{S}, \mathcal{L}$ are denoted by $S, |\mathcal{L}|$ respectively. Let $\mathcal{L}_j$ be set of the outgoing links from node $j \in \mathcal{S}$. A directed link $l = (j, k)$ indicates that node j can transmit data packet to node k. We use self loops to indicate the decision not to transmit at a node, *i.e.,* $(j, j) \in \mathcal{L}_j$ for all $j \in \mathcal{S}$. We model unreliability of network links by assuming that a transmission over link l is successful with a probability p_l. We also assume that the time is slotted, and one time slot is the time needed to transmit one packet over any link in the network.

We consider a set of finite number of flows F with size $|F|$ indexed by $f \in \{1, \ldots, |F|\}$. Each flow $f \in F$ has a positive bounded weight denoted by β_f. Besides, s_f and d_f indicate the source node and destination node of flow $f \in F$, respectively. Let $A_f(t)$ denotes the set of packets arriving at node s_f at time t that are in flow f. The average arrival rate of flow f is then defined as $\rho_f = \lim_{T\to\infty} \sum_{t=1}^{T} |A_f(t)|/T$. We denote $\rho_{\text{tot}} = \sum_f \rho_f$. Each packet of flow f has a maximum end-to-end delay τ_f associated with it. A packet of flow f that has arrived at s_f at time t needs to be delivered to d_f before time $t + \tau_f$, or else it will be discarded. We assume that $max_f \tau_f = \tau_{\max} < \infty$.

The *timely throughput* for flow f under a scheduling policy π, R_f^π, is the expected value of the number of packets delivered prior to deadline expiry per unit time,

$$R_f^\pi = \liminf_{T\to\infty} \frac{1}{T} \mathbb{E} \sum_{t=1}^{T} x_f^\pi(t), \tag{1}$$

where $x_f^\pi(t)$ is the number of packets of flow f successfully delivered to d_f under policy π at time t.

The *average link utilization* for link l under policy π, denoted by C_l^π is defined as

$$C_l^\pi := \limsup_{T\to\infty} \frac{1}{T} \mathbb{E} \sum_{f\in F} \sum_{t=1}^{T} c_{l,f}^\pi(t), \tag{2}$$

where $c_{l,f}^\pi(t)$ is the number of packet transmissions for flow f on link l under policy π at time t. This is the relaxation proposed in [4]. In practice, such a relaxation might correspond to an average transmit power constraint. It is pointed out in [4] that the gap between this approach and the hard constraint becomes small in the heavy traffic regime. We will also consider such a hard constraint in the numerical simulations.

The optimal scheduling problem is to find a policy π^* that solves the following optimization problem

$$[\textbf{OSP}] \quad \max_{\pi} \sum_{f\in F} \beta_f R_f^\pi, \quad \text{s.t } C_l^\pi \le C_l, \forall l \in \mathcal{L}, \tag{3}$$

where β_f is the weight assigned to flow f.

B. Constrained MDP Formulation

We now formulate OSP using the framework of constrained Markov Decision Processes (CMDP). We first specify the states, actions, rewards and transition kernel of the corresponding MDP.

State. Let $s_{i,f}(t)$ denote the state of the packet i from flow f at time t, defined as the node at which that packet is located at time t. If the packet has been delivered to its destination, or if it has been discarded from the network by time t, then $s_{i,f}(t)$ is defined as the terminal state s_{term}. The state of the network at time t, $s(t)$, is then defined as $s(t) = (s_{i,f}(t), i \in \cup_{\tau=0}^{\tau_f} A_f(t-\tau), f \in F)$.

Action. The scheduling action $a_{i,f}(t)$ for packet i in flow f at time t is defined the link on which that packet is transmitted at time t. Hence, $a_{i,f}(t) \in \mathcal{L}_{s_{i,f}(t)}$. The scheduling action for the network at time t, $a(t)$, is then defined as $a(t) = (a_{i,f}(t), i \in \cup_{\tau=0}^{\tau_f} A_f(t-\tau), f \in F)$. A scheduling policy π maps the state of the system $s(t)$ to the scheduling action $a(t)$, *i.e.,* $a(t) = \pi(s(t))$.

Transition Kernel. We denote the transition kernel of the MDP as $P(k|j,l)$, which is the probability that the $s_{i,f}(t+1) = k$ given that $s_{i,f}(t) = j$ and $a_{i,f}(t) = l$. Clearly,

$$P(k|j,l) = \begin{cases} p_l & \text{if } l = (j,k) \\ 1-p_l & \text{if } j = k \\ 0 & \text{O.W.} \end{cases} \tag{4}$$

We assume that $p_l = 1$ for $l = (j,j)$ for all $j \in \mathcal{S}$. Note that the transition kernel is the same for all packets in all the flows.

Reward. Let $r_f(j)$ denote the reward for a packet in flow f for being in state j. We define

$$r_f(j) = \begin{cases} \beta_f & \text{if } j = d_f \\ 0 & \text{O.W.} \end{cases} \tag{5}$$

The OSP is equivalent to [**CMDP**],

$$\max_{\pi} \quad \lim_{T\to\infty}\frac{1}{T}\ \mathbb{E}\sum_{t=1}^{T}\sum_{f\in F}\sum_{i\in A_f(t)}\sum_{\tau=0}^{\tau_f} r_f(s^{\pi}_{i,f}(t+\tau)) \tag{6}$$

$$\text{s.t.} \quad \lim_{T\to\infty}\frac{1}{T}\ \mathbb{E}\sum_{t=1}^{T}\sum_{f=1}^{F}\sum_{i\in A_f(t)}\sum_{\tau=0}^{\tau_f}\mathbb{I}\{a^{\pi}_{i,f}(t+\tau)=l\} \le C_l, \quad \forall l \in \mathcal{L} \tag{7}$$

where the states and actions are generated according the policy π. The expectation is taken with respect to the arrival process, the transition kernel P and the policy π.

C. Packet-by-Packet Decomposition

We next describe the decomposition approach that reduces the complexity of the problem by turning it into a per-packet MDP, rather than having to consider a global problem that accounts for the states of all packets in the system at each transmission decision.

The Lagrange Dual is a usual approach towards the solution of a CMDP [3]. The Lagrangian can be written as,

$$L(\pi,\lambda) = \sum_{l\in\mathcal{L}}\lambda_l C_l + \lim_{T\to\infty}\frac{1}{T}\ \mathbb{E}\sum_{t=1}^{T}\sum_{f\in F}\sum_{i\in A_f(t)}\sum_{\tau=0}^{\tau_f} (r_f(s^{\pi}_{i,f}(t+\tau)) - \sum_{l}\lambda_l\mathbb{I}\{a^{\pi}_{i,f}(t+\tau)=l\}), \tag{8}$$

considering equivalent formulation of **OSP**, presented by (6) and (7). Noting that the rewards and transition probabilities are the same for each packet i in a given flow f, we define

$$V^{\pi}_f(\lambda) = \mathbb{E}[\sum_{\tau=0}^{\tau_f}(r_f(s^{\pi}_{i,f}(t+\tau)) - \sum_{l}\lambda_l\mathbb{I}\{a^{\pi}_{i,f}(t+\tau)=l\}) \,|i,f,s^{\pi}_{i,f}(t)=s_f], \tag{9}$$

where $\mathbb{E}$ is the expectation w.r.t. to the underlying transition kernel under the policy π. Then the Lagrangian (8) can be written as

$$\begin{aligned} L(\pi,\lambda) &= \sum_{l\in\mathcal{L}}\lambda_l C_l + \sum_{f\in F}\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\sum_{i\in A_f(t)} V^{\pi}_f(\lambda) \\ &= \sum_{l\in\mathcal{L}}\lambda_l C_l + \sum_{f\in F}\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}|A_f(t)|V^{\pi}_f(\lambda) \\ &= \sum_{l\in\mathcal{L}}\lambda_l C_l + \sum_{f\in F}\rho_f V^{\pi}_f(\lambda). \end{aligned} \tag{10}$$

The dual function $D(\lambda)$ and 'dual policy' $\pi(\lambda)$, and the optimal dual variable are defined as

$$D(\lambda) = \max_{\pi}\ L(\pi,\lambda),\ \pi(\lambda) = \arg\max_{\pi}\ L(\pi,\lambda), \quad \lambda^* = \arg\min_{\lambda\ge 0}\ D(\lambda) \tag{11}$$

Since there is no duality gap [3], the optimal policy π^* for the **[CMDP]** is the same as $\pi(\lambda^*)$.

Note that given a λ, $V^{\pi}_f(\lambda)$ for a given flow f does not depend on other flows. Hence, rather than finding an optimal joint policy $\pi(\lambda)$ for all flows, we can instead find an optimal policy $\pi_f(\lambda)$ for each flow separately. More precisely,

$$\begin{aligned} D(\lambda) = \max_{\pi} L(\pi,\lambda) &= \sum_{l\in\mathcal{L}}\lambda_l C_l + \max_{\pi}\sum_{f\in F}\rho_f V^{\pi}_f(\lambda) \\ &= \sum_{l\in\mathcal{L}}\lambda_l C_l + \sum_{f\in F}\rho_f \max_{\pi_f} V^{\pi_f}_f(\lambda) \\ &= \sum_{l\in\mathcal{L}}\lambda_l C_l + \sum_{f\in F}\rho_f V^{*}_f(\lambda), \end{aligned}$$

where,

$$V^{*}_f(\lambda) = \max_{\pi_f} V^{\pi_f}_f(\lambda),\ \text{and},\ \pi_f(\lambda) = \arg\max_{\pi_f} V^{\pi_f}_f(\lambda) \tag{12}$$

Now, $\pi_f(\lambda)$ and $V^{*}_f(\lambda)$ can be computed by standard finite horizon dynamic programming if we know the transition kernel P (equivalently, the link probabilities p_l).

However, as discussed in the Introduction, p_ls are unknown a priori. We thus propose a reinforcement learning approach for learning p_ls and at the same time solving for the optimal policy.

IV. Reinforcement Learning Solutions

In this section, we propose two model-based Reinforcement Learning (RL) algorithms for solving the CMDP corresponding to timely throughput maximization. Both algorithms operate in a loop consisting of two steps. First, they solve the per-packet MDP under the current model to obtain sub-optimal solution of $V_f^*(\lambda)$ with high probability. Next, the Lagrange multipliers, λ are updated according to estimated model. The difference between the two algorithms is in how they sample the system in order to construct the model and update it, using an offline or online approach. We will show how both algorithms would result in an $\epsilon-$optimal policy with high probability. We also characterize the *sample complexity*, which in this case is the number of packets that are transmitted before the model is learned with a sufficiently high accuracy to ensure $\epsilon-$optimality. We do not present any proofs due to space constraints.

A. Generative Model-Based Learning

According to the GMBL algorithm, we use a traditional channel sounding approach, and simply send n packets over each link for estimating the reliability of each link p_l. For each link $l=(j,k)$, the transmission of a packet is 'successful' if the packet transmitted from node j in the link l reaches node j in one time slot. We define the empirical link reliability, $\hat{p}_l$, as the ratio of the successful transmission to the total number of transmission. Given $\hat{p}_l$, we can define the approximate transmission kernel $\hat{P}$ as (4) by replacing p_l with $\hat{p}_l$. It is straight forward to see that $\hat{p}_l$ is an unbiased estimator of p_l and $\hat{P}$ is an unbiased estimator of P. The expectation w.r.t. to this approximate transition kernel $\hat{P}$ is denoted by $\hat{\mathbb{E}}[\cdot]$.

We now consider a different constrained MDP that is identical to the CMDP defined in Section III except that its transition kernel is $\hat{P}$ instead of P. The expectation w.r.t. $\hat{P}$ is denoted by $\hat{\mathbb{E}}[\cdot]$. We define the quantities $\hat{V}_f^{\pi}(\lambda)$ in the same way as in (9) but by replacing $\mathbb{E}$ by $\hat{\mathbb{E}}$. The quantities $\hat{L}(\pi,\lambda), \hat{D}(\lambda)$ can also now be defined in a similar way as in (10) and (11) by replacing $V_f^{\pi}(\lambda)$ with $\hat{V}_f^{\pi}(\lambda)$. The optimal dual variable $\hat{\lambda}^*$ is defined as $\hat{\lambda}^* = \arg\min_{\lambda} \hat{D}(\lambda)$. We also define

$$\hat{\pi}_f(\lambda) = \arg\max_{\pi_f} \hat{V}_f^{\pi_f}(\lambda), \quad \hat{V}_f^*(\lambda) = \hat{V}_f^{\hat{\pi}_f(\lambda)}(\lambda). \tag{13}$$

Note that $\hat{\pi}_f(\lambda)$ and $\hat{V}_f^*(\lambda)$ can be computed by standard finite horizon dynamic programming [22], and we omit the details.

We also define the following quantities for describing the GMBL algorithm succinctly.

$$\hat{C}_{l,f}^{\pi_f} = \hat{\mathbb{E}}[\sum_{\tau=0}^{\tau_f} \mathbb{1}\{a_{i,f}^{\pi_f}(t+\tau) = l\} | i, f, s_{i,f}^{\pi_f}(t) = s_f], \tag{14}$$

$$\hat{R}_f^{\pi_f} = \mathbb{E}[\sum_{\tau=0}^{\tau_f} r_f(s_{i,f}^{\pi_f}(t+\tau) | i, f, s_{i,f}^{\pi_f}(t) = s_f], \tag{15}$$

$$\hat{C}_l^{\pi} = \sum_f \rho_f \hat{C}_{l,f}^{\pi_f}, \quad \hat{R}^{\pi} = \sum_f \rho_f \hat{R}_f^{\pi_f}, \text{ for } \pi = (\pi_f)_{f\in\mathcal{F}}, \tag{16}$$

Here π is the joint policy given by the collection of each individual policy π_f. Note that from (9), (14), (15), we can write

$$\hat{V}_f^{\pi_f}(\lambda) = \hat{R}_f^{\pi_f} - \sum_l \lambda_l \hat{C}_{l,f}^{\pi_f}, \tag{17}$$

$$\hat{L}(\pi,\lambda) = \hat{R}^{\pi} + \sum_l \lambda_l (C_l - \hat{C}_l^{\pi}). \tag{18}$$

The GMBL algorithm is summarized in Algorithm 1.

Algorithm 1 Generative Model-Based Learning (GMBL)

```
Input: accuracy $\epsilon, \delta$. Initialize $\lambda_l(0) = 0, \forall l \in \mathcal{L}$
Send $n = n(\epsilon,\delta)$ packets in each link $l \in \mathcal{L}$
Estimate the link probability $\hat{p}_l$ by transmitting $n$ packets across all links uniformly
Construct the approximate transition kernel $\hat{P}$
for $m$ from 1 to $M$ do
    For each flow $f$, compute $\pi_f(m) = \hat{\pi}_f(\lambda(m))$ according to (13). Define $\pi(m) = (\pi_f(m))_{f\in\mathcal{F}}$
    Compute $\hat{C}_l^{\pi(m)}$ according to (14) and (16)
    Compute $\lambda_l(m+1)$ for each link $l$ as
        $\lambda_l(m+1) = \Pi_{\Lambda}(\lambda_l(m) - \alpha(C_l - \hat{C}_l^{\pi(m)}))$[1]
Compute $\hat{\lambda}(M) = \frac{1}{M}\sum_{m=1}^{M} \lambda(m)$.
Compute $\pi_f(M+1) = \hat{\pi}_f(\hat{\lambda}(M))$
Output: $\hat{\pi} = (\pi_f(M+1))_{f\in F}, \quad \hat{\lambda} = \hat{\lambda}(M)$
```

We next present the sample complexity of GMBL.

Theorem 1. *GMBL algorithm with*

$$n(\epsilon,\delta) \geq \frac{18(\rho_{tot}\beta_{\max}\tau_{\max})^2}{\epsilon^2} \log \frac{6|\mathcal{L}||F|}{\delta} \tag{19}$$

and parameters

$$M = \frac{36|\mathcal{L}|(\tau_{\max}\rho_{tot} + C_{\max})^2 \lambda_{\max}^2}{\epsilon^2}, \tag{20}$$

$$\alpha = \frac{\epsilon}{3|\mathcal{L}|(\tau_{\max}\rho_{tot} + C_{\max})^2}, \tag{21}$$

where $\lambda_{\max} = \frac{\rho_{tot}\beta_{\max}}{C_{\min}}$ *and* $C_{\min} = \min_l C_l$, *achieves a* $\hat{\lambda}$ *and* $\hat{\pi}$ *such that*

$$\mathbb{P}\left(|L(\hat{\pi},\hat{\lambda}) - L(\pi^*,\lambda^*)| \leq \epsilon\right) \geq (1-\delta).$$

[1] Π_{Λ} is projection to set $\Lambda = [0, 2\lambda_{\max}]$, where $C_{\min} = \min_l C_l$.

Algorithm 2 MBIE (Model Based Interval Estimation) for flow f

```
Input: accuracy $\epsilon$, $\delta$
$n(j,l) = n'(k,l,j) = 0;\ \tilde{V}_{\tau_f+1} := 0 \quad \forall j,k \in \mathcal{S}, l \in \mathcal{L}_s$
$m = \frac{4S\tau_f^4}{\epsilon^2} + \frac{\tau_f^4}{\epsilon^2}\ln\left(\frac{S|\mathcal{L}|H}{\epsilon\delta}\right)$
$lc = \ln(2S-2) + \ln\left(\frac{2S|\mathcal{L}|m}{\delta}\right)$
while Model Stops Updating do
    for $t = \tau_f$ to 1 do
        for $j \in \mathcal{S}$ do
            for $l \in \mathcal{L}_j$ do
                $\tilde{p}(l|j,l) = \frac{n'(l,l,j)}{n(j,l)} + \sqrt{\frac{2lc}{n(j,a)}}$
                $\tilde{p}(j|j,l) = 1 - \tilde{p}(l|j,l)$
                $Q(l) = r_f(j,l) + \sum_{k \in \mathcal{L}_j} \tilde{p}(k|j,l)\tilde{V}_{t+1}(k)$
            $\pi_f(j,t) = \arg\max_l Q(l), \quad \tilde{V}_t(j) = Q(\pi(j,t))$
    $j_0 = s_f$
    for $t = 1$ to $\tau_f$ do
        $l_t = \pi_f(j_t,t), \quad j_{t+1} \sim P(j_t,l_t)$
        $n(j_t,l_t)++, \quad n'(j_{t+1},l_t,j_t)++$
Output: $\pi_f$
```

B. Con-MBIE

As mentioned in the Introduction, many IAB nodes are analog repeaters that cannot generate packets for channel sounding. Thus, we require an online approach that utilizes packets to learn the model as they are routed through the network.

We start with a simple model-based RL scheme called Model-Based Interval Estimation (MBIE) [5], and modify it to our constrained MDP setting[2]. The finite horizon MBIE algorithm, described in Algorithm 2 proceeds in *episodes*, where each episode corresponds to the time interval between the generation and expiry of a packet. The model $\tilde{P}$ is updated at the end of each such episode, and a new (optimistic) policy π_f for each flow f is generated. The algorithm proceeds over episodes until we obtain a PAC guarantee on the accuracy of the policy. Note that unlike GMBL, sampling is online, and "targeted" towards routes (and hence links) that are likely to yield high reward, much like the difference between sampling of all arms of a multi-armed bandit vs. using an online learning algorithm.

Under Constrained MBIE, or Con-MBIE we have two time scales of updates. The algorithm consists of an alternate MBIE [5] step (consisting of multiple episodes), and a stochastic subgradient step at which Lagrange multipliers are updated (called an *epoch*). Hence, we start with an initial $\lambda(0)$ and compute optimistic policies π_f for each flow f with respect to $\lambda(0)$ by means of MBIE (Algorithm 2). The MBIE step results in π_f for each flow f (which are likely sub-optimal during the initial epochs), and an optimistic transition kernel $\tilde{P}$ according to the samples from links.

We denote expectation w.r.t $\tilde{P}$ by $\tilde{\mathbb{E}}$. After the MBIE procedure, we apply stochastic subgradient method and calculate new Lagrange multipliers, disregard the samples from MBIE step and repeat the identical process for the new Lagrange multipliers. At each epoch, we define the quantities $\tilde{V}_f^{\pi}(\lambda)$ similar to equation (9) by replacing $\mathbb{E}$ with $\tilde{\mathbb{E}}$. Then, quantities $\tilde{L}(\pi,\lambda)$ and $\tilde{D}(\lambda)$ are defined similar to (10) and (11) by replacing $V_f^{\pi}(\lambda)$ with $\tilde{V}_f^{\pi}(\lambda)$. We also define

$$\tilde{\pi}_f(\lambda) = \arg\max_{\pi_f} \tilde{V}_f^{\pi_f}(\lambda), \quad \tilde{V}_f^*(\lambda) = \tilde{V}_f^{\tilde{\pi}_f(\lambda)}(\lambda). \tag{22}$$

Note that every time the MBIE step is conducted with failure probability δ' for accuracy of ϵ' for a flow f and given λ, some samples are collected such that

$$\mathbb{P}(|\tilde{V}_f^*(\lambda) - V_f^*(\lambda)| \le \epsilon') \ge 1 - \delta'.$$

In fact, there are countably many sets of other samples that would yield the same result, and any execution of MBIE picks one set of samples among all possible set of samples. Therefore, all the quantities of $\tilde{V}_f^{\pi_f}(\lambda)$, $\tilde{L}(\pi,\lambda)$ and $\tilde{D}(\lambda)$ would be random variables.

Call each set of samples an observation O, and denote expectation w.r.t observations by $\mathbb{E}_O$. Since we do not have any information about distribution on O, we can treat $\mathbb{E}_O$ of any random variable as another random variable. Denote $\mathbb{E}_O[\tilde{V}_f^{\pi}(\lambda)]$ by $\bar{V}_f^{\pi}(\lambda)$ for each flow f, and replace $V_f^{\pi}(\lambda)$ with $\bar{V}_f^{\pi}(\lambda)$ in equations (10) and (11) to obtain $\bar{L}(\pi,\lambda)$ and $\bar{D}(\lambda)$. It is obvious that $\bar{D}(\lambda) = \mathbb{E}_O[\tilde{D}(\lambda)]$ and is a convex function.

Now, instead of minimizing $D(\lambda)$, we minimize $\bar{D}(\lambda)$. Then, the optimal dual variable $\bar{\lambda}^*$ is defined as $\bar{\lambda}^* = \arg\min_\lambda \bar{D}(\lambda)$. We also define the following quantities required for describing the Con-MBIE algorithm.

$$\tilde{C}_{l,f}^{\pi_f} = \tilde{\mathbb{E}}[\sum_{\tau=0}^{\tau_f} \mathbb{1}\{a_{i,f}^{\pi_f}(t+\tau) = l\}|i,f,s_{i,f}^{\pi_f}(t) = s_f], \tag{23}$$

$$\tilde{R}_f^{\pi_f} = \mathbb{E}[\sum_{\tau=0}^{\tau_f} r_f(s_{i,f}^{\pi_f}(t+\tau)|i,f,s_{i,f}^{\pi_f}(t) = s_f], \tag{24}$$

$$\tilde{C}_l^{\pi} = \sum_f \rho_f \tilde{C}_{l,f}^{\pi_f}, \quad \tilde{R}^{\pi} = \sum_f \rho_f \tilde{R}_f^{\pi_f}, \text{ for } \pi = (\pi_f)_{f\in\mathcal{F}}, \tag{25}$$

Here, π is the joint policy given by the collection of each individual policy π_f. Note that from (9), (23), (24), we can write

$$\tilde{V}_f^{\pi_f}(\lambda) = \tilde{R}_f^{\pi_f} - \sum_l \lambda_l \tilde{C}_{l,f}^{\pi_f}, \tag{26}$$

$$\tilde{L}(\pi,\lambda) = \tilde{R}^{\pi} + \sum_l \lambda_l (C_l - \tilde{C}_l^{\pi}). \tag{27}$$

Con-MBIE is described by Algorithm 3 and the result below presents the sample complexity of Con-MBIE.

[2] Note that MBIE is an RL algorithm designed originally for infinite-horizon MDPs with a discount factor γ. However, it can be utilized as a finite-horizon algorithm while the analysis still holds with minor modifications.

Algorithm 3 Constrained MBIE (Con-MBIE)

```
Input: accuracy $\epsilon, \delta$. Initialize $\lambda_l(0) = 0, \forall l \in \mathcal{L}$
for $m$ from $1$ to $M$ do
    for $f \in F$ do
        $\pi_f(m) = MBIE(\frac{\delta}{4|F|}, \frac{\epsilon}{5\rho_{\text{tot}}})$
    $\pi(m) = (\pi_f(m))_{f \in F}$
    Compute $\tilde{C}_l^{\pi(m)}$ according to (23) and (25)
    Compute $\lambda_l(m+1)$ for each link $l$ as
        $$\lambda_l(m+1) = \Pi_\Lambda(\lambda_l(m) - \alpha(C_l - \tilde{C}_l^{\pi(m)}))$$
    Reset samples
Compute $\bar{\lambda}(M) = \frac{1}{M}\sum_{m=1}^{M} \lambda(m)$
for $f \in F$ do
    $\pi_f(M+1) = \tilde{\pi}_f(\bar{\lambda}(M)) = MBIE(\frac{\delta}{4|F|}, \frac{\epsilon}{5\rho_{\text{tot}}})$
Output: $\tilde{\pi} = (\pi_f(M+1))_{f \in F}, \quad \bar{\lambda} = \bar{\lambda}(M)$
```

Theorem 2. *Con-MBIE with parameters*

$$M = \frac{100|\mathcal{L}|(\tau_{\max}\rho_{tot} + C_{\max})^2\lambda_{\max}^2}{\epsilon^2}, \tag{28}$$

$$\alpha = \frac{\epsilon}{5|\mathcal{L}|(\tau_{\max}\rho_{tot} + C_{\max})^2}, \tag{29}$$

and

$$n_C(\epsilon, \delta) = O(M|F|\frac{S|\mathcal{L}|\tau_{\max}^6}{\epsilon^2}(S + \ln(\frac{S|\mathcal{L}|\tau_{\max}}{\epsilon\delta}))\ln\frac{1}{\delta}\ln\frac{\tau_{\max}}{\epsilon}), \tag{30}$$

achieves $\tilde{\pi}$ and $\bar{\lambda}$ such that

$$\mathbb{P}\Big(|L(\tilde{\pi}, \bar{\lambda}) - L(\pi^*, \lambda^*)| \leq \epsilon\Big) \geq 1 - \delta.$$

Theorem 3. *[5] Let π_i be the policy of MBIE algorithm 2 in the i^{th} episode on any finite horizon MDP with N states, A actions and horizon H. Then with probability $1 - \delta'$ for all $\epsilon' > 0$ jointly the number of episodes i where $V^* - V^{\pi_i} > \epsilon'$ is at most*

$$O\Big(\frac{NAH^6}{\epsilon'^2}(N + \ln(\frac{NAH}{\epsilon'\delta'}))\ln\frac{1}{\delta'}\ln\frac{H}{\epsilon'}\Big).$$

C. Con-MBIE-RU

The above results suggest that Con-MBIE has a poorer sample complexity to GMBL, due to the fact that it resamples the system at each update of the Lagrange multipliers. Hence, we introduce a heuristic learning algorithm Con-MBIE-RU. This algorithm is identical to Con-MBIE, except for reusing samples obtained in previous epochs. In section V we show that it outperforms both GMBL and Con-MBIE algorithms empirically. This intuition is based on two facts. First, Con-MBIE-RU is an online learning algorithm based on MBIE, which trade offs exploration and exploitation in a structured manner. Hence, within each epoch, it is likely to offer superior performance to a GMBL-like approach (even if its sample complexity bound is the same). Second, it recognizes that the link statistics are unchanged between epochs, and hence aggregates all samples to create an increasingly more accurate model with the passage of epochs. Hence, it is likely to require fewer episodes per epoch than MBIE, particularly in the later epochs.

V. Simulation Results

In this section, we present simulation results to compare the performance of the GMBL, Con-MBIE and Con-MBIE-RU algorithms with respect to the optimal policy in the context of attaining high weighted timely throughput in an IAB network. We need to implement the basic MBIE algorithm described in algorithm 2 for both Con-MBIE and Con-MBIE-RU, and we follow the same steps as [5].

We develop a simulation scenario motivated by the IAB use case, with the green nodes representing gateways, and the white nodes representing fully wireless nodes in Figures 1. Since gateway nodes communicate with zero latency over an optic fiber medium, they can be merged into a single node. Hence, in Figure 1 nodes 1 and 11 can jointly be represented as a single node $(1, 11)$.

For each link l, p_l is uniformly randomly chosen from $[0.5, 1.0]$, while C_l is chosen from $[1, 5]$. We have two unicast flows, with Flow 1 between nodes $(1, 11) \to 6$, and Flow 2 is between nodes $10 \to 6$. Packet arrivals to the system follow a Poisson number of arrivals to each source node in each time slot, and the injection rate is varied based on the experiment of interest (indicated below). Flow 1 has a weight of 3, and Flow 2 has a weight of 4.

The performance metrics of interest are the error in the value engendered by the policy that is the outcome of the algorithm and the optimal reward. We define the error as

$$\left|\sum_l \lambda_l^M C_l + \sum_f V_f^{\pi_f(\lambda_M)}(\lambda_M) - D(\lambda^*)\right|, \tag{31}$$

where λ_M and $\pi_f(\lambda_M)$ are the Lagrange multipliers and policy that result from the execution of a particular learning algorithm. The error depends on the number of sub-gradient updates, M, which we empirically set as 100 for good error performance. Further, optimal reward is simply the sum of the first two terms in (31).

We set a packet budget for learning the model, and identify the error for each of our candidate algorithms. Figures 2 depicts the relation between the error and transmission budget empirically. The graph shows that increasing the transmit budget reduces error for all the algorithms. However, GMBL outperforms Con-MBIE which is consistent with Theorems 1 and 2. Finally, Con-MBIE-RU significantly outperforms both algorithms by benefiting from the targeted routing of the online approach, while building the model from all samples. Figure 3 is also consistent with the fact that Con-MBIE-RU outperforms both GMBL and Con-MBIE.

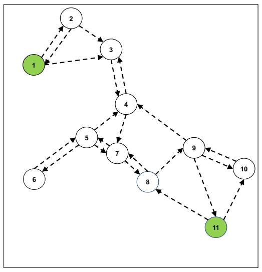

Fig. 1. S1: 11−node network

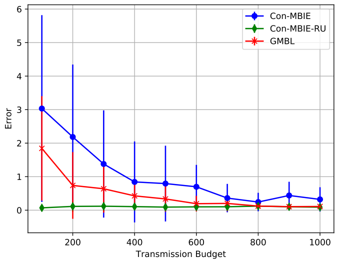

Fig. 2. S1: Error vs. Transmit Budget

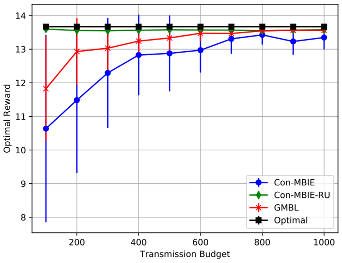

Fig. 3. S1: Optimal Reward vs. Transmit Budget

VI. Conclusion

In this paper, we considered the problem of maximizing the throughput of unicast flows with strict per-packet deadlines over a multi-hop wireless network, motivated by 5G IAB mm-wave networks. The problem formulation took the form on a constrained MDP, and, assuming that the link statistics are known, can be solved using a dual-decomposition approach. We proposed a model-based RL approach, and developed two types of algorithms, based on offline channel sounding, and online learning. We showed that although the basic online approach, Con-MBIE performs targeted sampling of links, suffers from having to start with a fresh model for each Lagrange multiplier update, and so has higher sample complexity than GMBL. An online learning approach that eliminates this wekness, Con-MBIE-RU has excellent empirical performance, and our future goal is to characterize its performance analytically.

References

[1] M. N. Islam, S. Subramanian, and A. Sampath, "Integrated access backhaul in millimeter wave networks," in *Wireless Communications and Networking Conference (WCNC), 2017 IEEE*. IEEE, 2017, pp. 1–6.

[2] M. N. Islam, N. Abedini, G. Hampel, S. Subramanian, and J. Li, "Investigation of performance in integrated access and backhaul networks," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2018, pp. 597–602.

[3] E. Altman, *Constrained Markov decision processes*. CRC Press, 1999, vol. 7.

[4] R. Singh and P. Kumar, "Throughput optimal decentralized scheduling of multihop networks with end-to-end deadline constraints: Unreliable links," *IEEE Transactions on Automatic Control*, vol. 64, no. 1, pp. 127–142, 2019.

[5] A. L. Strehl and M. L. Littman, "An analysis of model-based interval estimation for markov decision processes," *Journal of Computer and System Sciences*, vol. 74, no. 8, pp. 1309–1331, 2008.

[6] C. Dann, T. Lattimore, and E. Brunskill, "Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 5713–5723.

[7] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," in *29th IEEE Conference on Decision and Control*. IEEE, 1990, pp. 2130–2132.

[8] X. Lin and N. B. Shroff, "Joint rate control and scheduling in multihop wireless networks," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 2. IEEE, 2004, pp. 1484–1489.

[9] A. Eryilmaz and R. Srikant, "Joint congestion control, routing, and mac for stability and fairness in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1514–1524, 2006.

[10] A. Sinha, G. Paschos, and E. Modiano, "Throughput-optimal multi-hop broadcast algorithms," *IEEE/ACM Transactions on Networking*, 2017.

[11] A. Sinha and E. Modiano, "Optimal control for generalized network-flow problems," *IEEE/ACM Transactions on Networking (TON)*, vol. 26, no. 1, pp. 506–519, 2018.

[12] I.-H. Hou, V. Borkar, and P. R. Kumar, "A theory of QoS for wireless," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Rio de Janeiro, Brazil, April 2009.

[13] R. Li, A. Eryilmaz, and B. Li, "Throughput-optimal wireless scheduling with regulated inter-service times," in *2013 Proceedings IEEE INFOCOM*. IEEE, 2013, pp. 2616–2624.

[14] H. Xiong, R. Li, A. Eryilmaz, and E. Ekici, "Delay-aware cross-layer design for network utility maximization in multi-hop networks," *Selected Areas in Communications, IEEE Journal on*, vol. 29, pp. 951 – 959, 2011.

[15] Z. Mao, C. E. Koksal, and N. B. Shroff, "Online packet scheduling with hard deadlines in multihop communication networks," in *Proc. of IEEE INFOCOM*, 2013, pp. 2463 – 2471.

[16] R. Li and A. Eryilmaz, "Scheduling end-to-end deadline-constrained traffic with reliability requirements in multi-hop networks," *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 5, pp. 1649 – 1662, Otc 2012.

[17] A. HasanzadeZonuzy, I.-H. Hou, and S. Shakkottai, "Broadcasting real-time flows in integrated backhaul and access 5G networks," in *WiOpt 2019*, 2019.

[18] S. Krishnasamy, R. Sen, R. Johari, and S. Shakkottai, "Regret of queueing bandits," in *Advances in Neural Information Processing Systems*, 2016, pp. 1669–1677.

[19] R. Combes, A. Proutiere, D. Yun, J. Ok, and Y. Yi, "Optimal rate sampling in 802.11 systems," in *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*. IEEE, 2014, pp. 2760–2767.

[20] H. Gupta, A. Eryilmaz, and R. Srikant, "Low-complexity, low-regret link rate selection in rapidly-varying wireless channels," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 540–548.

[21] M. S. Talebi, Z. Zou, R. Combes, A. Proutiere, and M. Johansson, "Stochastic online shortest path routing: The value of feedback," *IEEE Transactions on Automatic Control*, vol. 63, no. 4, pp. 915–930, 2017.

[22] D. P. Bertsekas, D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas, *Dynamic programming and optimal control*. Athena scientific Belmont, MA, 1995, vol. 1, no. 2.