# Tracking MaxWeight: Optimal Control for Partially Observable and Controllable Networks

Bai Liu, Qingkai Liang, and Eytan Modiano, *Fellow, IEEE*

***Abstract*— Modern networks are complex and may include components that cannot be fully controlled or observed. Such network models can be characterized by overlay-underlay structures, where the network controller can only observe and operate on overlay nodes, and the underlay nodes are neither observable nor controllable. Classic network control algorithms may fail to work properly if they are only applied to the overlay nodes. To tackle this issue, we propose the Tracking MaxWeight (TMW*) algorithm that does not require direct observations of underlay nodes and only operates on overlay nodes. TMW* maintains virtual queues that track the dynamics of the underlay nodes and makes control decisions based on those virtual queues. We show that TMW* is throughput optimal as long as the network is stabilizable. We further extend our analysis to the setting that the estimates of the underlay state is erroneous and show that as long as the errors scale sub-linearly in time, TMW* preserves throughput optimality.**

***Index Terms*— Network control, resource allocation, routing, queueing theory.**

## I. Introduction

MODERN communication networks are growing rapidly in scale and often the network controller cannot have full access to the entire network. For instance, under the software-defined networking (SDN) paradigm, the controller usually can only control certain key nodes, with the rest of the nodes being uncontrollable or even unobservable. Another example is that due to security or economic concerns, some network modules might have restricted access. Such network characteristics can be captured by an overlay-underlay structure [1], where only the overlay nodes can be observed and controlled, while the underlay nodes appear as unobservable and uncontrollable "black boxes."

We consider an overlay-underlay network where only a subset of nodes can be observed and controlled by the network controller (i.e., overlay nodes). The rest nodes are underlay nodes of which the state may not be directly obtained by the network controller. Moreover, the underlay nodes operate legacy control policies and do not execute commands given by the network controller. Therefore, we propose the Tracking MaxWeight algorithm (TMW*)[1] to stabilize such an overlay-underlay network. To the best of our knowledge, TMW* is the first algorithm to stabilize networks with unobservable and uncontrollable nodes.

Classical network control algorithms such as MaxWeight and BackPressure [2] are capable of stabilizing queue backlogs effectively, yet directly applying them to our overlay-underlay network model might lead to instability. In section VI-B, our simulation results show that applying MaxWeight only to overlay nodes can lead to linear growth in queue backlogs.

The design of overlay control algorithms for overlay-underlay networks has been studied from different perspectives. In [3], the authors model the overlay nodes as routers and the underlay nodes as forwarders, assuming that only routers are controllable. They then propose the Threshold-based Backpressure (BP-T) algorithm that is shown to be throughput optimal when the paths between routers do not overlap. Based on [3], the work of [4] further studies the minimal necessary placement of routers and proposed the Overlay Backpressure (OBP) algorithm for more general network topologies. In [5], the authors construct a counter-example network where OBP fails to stabilize and propose the Optimal Overlay Routing Policy (OORP) algorithm with more general applicability. However, OORP requires the instantaneous knowledge of the underlay queue backlogs, for which approximation methods are applied and strict theoretical performance guarantees cannot be obtained. Model-based reinforcement learning techniques have also been applied to overlay-underlay networks [6], [7], where the network controller estimates the dynamics of the underlay nodes and computes the optimal control policy accordingly.

Most stochastic queueing networks can be modeled as Markov Decision Processes (MDP), and when the observability is constrained, the model becomes Partially Observable Markov Decision Processes (POMDP). POMDP problems have been receiving much attention in machine learning communities since real-world problems often involve limited observability and controllability. However, most of the POMDP algorithms are heuristic and lack theoretical performance guarantees. Among the works with theoretical guarantees, classic methods [8], [9], [10], [11], [12] focus on solving the value iteration problem for POMDPs, yet are only applicable to POMDPs with small state spaces and special structures. From a policy search perspective, [13] introduces

Manuscript received 25 August 2021; revised 5 January 2022, 19 April 2022, and 6 September 2022; accepted 27 November 2022; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor S. Moharir. Date of publication 7 December 2022; date of current version 18 August 2023. This work was supported in part by National Science Foundation (NSF) under Grant CNS-1524317, Grant CNS-1907905, and Grant CNS-1735463; and in part by Office of Naval Research (ONR) under Grant N00014-20-1-2119. Part of the material in this paper was presented at IEEE International Conference on Computer Communications (INFOCOM), 2019 [DOI: 10.1109/INFOCOM.2019.8737528]. *(Corresponding author: Bai Liu.)*

Bai Liu and Eytan Modiano are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: bailiu@mit.edu).

Qingkai Liang is with Celer Network, Mountain View, CA 94043 USA.

Digital Object Identifier 10.1109/TNET.2022.3225752

[1]We use TMW* to distinguish from our earlier version of TMW that required instantaneous observation of uncontrollable nodes.

a gradient-based policy learning algorithm with proof of convergence but can only guarantee local optimality. In communication networks that involve POMDP modeling, [14], [15], [16] study channel cooperation problems and propose algorithms with performance guarantees, yet they focus on channel scheduling problems instead of general network problems.

Another related field is distributed network control, where each node can only access the dynamics of their neighbors. Since only partial information is available, this setting is relevant to partially observable and controllable networks. For distributed routing, [17], [18], [19] use fluid model to characterize the network and develop distributed algorithms to optimize the assigned data transmission rates. In the context of distributed scheduling, many works focus on stabilizing interference constrained networks using randomized or MaxWeight style algorithms and analyzes the corresponding stability regions [20], [21], [22].

As far as we know, the existing algorithms either require instantaneous observations of underlay nodes or can only be applied to constrained settings. In this work, we consider general overlay-underlay networks with unobservable and uncontrollable underlay nodes. A preliminary conference version of our algorithm was presented in [6], and was shown to be throughput optimal when the underlay nodes are fully observable. This journal version extends TMW to settings where underlay nodes are unobservable and their backlogs can only be estimated. In order to distinguish from its earlier conference version henceforth, we will refer to the improved version of TMW as TMW*.

We propose the TMW* algorithm that uses estimates of the underlay queue backlogs instead of direct observations and only needs to operate on overlay nodes. We rigorously prove that TMW* is throughput optimal for general overlay-underlay networks. To the best of our knowledge, TMW* is the first throughput optimal control algorithm for networks with unobservable and uncontrollable nodes. We analyze the performance of TMW* when the estimates are erroneous, and show that as long as the errors grow sub-linearly in time, our algorithm remains throughput optimal. Simulation results on a 15-node overlay-underlay network corroborate the validity of our theoretical guarantees. We also show that when the estimation errors grow linearly (or superlinearly) in time, there exists an overlay-underlay network to which no queue-based throughput optimal stochastic policy exists. Therefore, TMW* is "maximally robust" to estimation errors.

The rest of the paper is organized as follows. We introduce the network model in Section II. We propose TMW* in Section III. In Section IV, we show that TMW* is throughput optimal. We consider estimation errors in Section V and analyze the its impact on stablity. In Section VI, we conduct simulations on three network models. Section VII concludes the paper.

## II. Model

We consider a multi-hop network $\mathcal{G}$ consisting of $N$ nodes and denote the set of nodes by $\mathcal{N}$. We assume that the network topology is known to the controller. The nodes are partitioned into overlay nodes $\mathcal{O}$ that are observable and controllable, and underlay nodes $\mathcal{U}$ that are unobservable and uncontrollable. The network has $K$ classes of traffic and traffic of class $k$ is destined for sink node $d_k$. The set of traffic classes is defined as $\mathcal{K}$. The link capacity between node $i$ and $j$ is $C_{ij}$, which is known to the controller. We assume the time is slotted and denote by $T$ the time horizon.

At the beginning of time slot $t$, a node $i \in \mathcal{N}$ has $Q_{ik}(t)$ buffered packets of class $k$. Node $i$ receives $a_{ik}(t)$ external new packets of class $k$. For each $i \in \mathcal{N}$ and $k \in \mathcal{K}$, we assume $a_{ik}(t)$'s are i.i.d across time and let $\lambda_{ik} = \mathbb{E}[a_{ik}(t)]$.

Overlay nodes are observable and controllable, i.e. at each time slot, the controller can observe their queue backlogs, and make routing and scheduling decisions. For an overlay node $i \in \mathcal{O}$, at most $f_{ijk}(t)$ packets of class $k$ are transmitted to its neighbors $j$. However, since the number of packets available to be transmitted is upper bounded by $Q_{ik}(t) + a_{ik}(t)$, the actual number of packets being transmitted might be less than $f_{ijk}(t)$ and we denote by $\tilde{f}_{ijk}(t) = \min\{f_{ijk}(t), Q_{ik}(t) + a_{ik}(t)\}$.

On the other hand, the underlay nodes are not controllable and their state information (e.g. queue size) can only be estimated sparsely (e.g. sending probing packets, making statistical inference and underlay broadcast). We denote by $\Gamma_{ik}$ the set of time slots when the controller estimates the underlay state information $\hat{Q}_{ik}(t)$ of class $k$ at node $i$. We also assume that the controller can obtain an unbiased estimate of the underlay arrival rates $\lambda_{ik}(t)$ at time $t \in \Gamma_{ik}$ (i.e., $\mathbb{E}[\lambda_{ik}(t)] = \lambda_{ik}$). An example approach is to obtain the underlay arrivals simultaneously with $\hat{Q}_{ik}(t)$ and compute the sample mean of the arrivals.

We denote $\tau_{ik}(t)$ as the time slot when the most recent state estimate of class $k$ at node $i$ is obtained, i.e.,

$$\tau_{ik}(t) = \max_{\tau \in \Gamma_{ik}: \tau \leqslant t} \tau,$$

with which we define $L(t) \triangleq \max_i \big(t - \tau_{ik}(t)\big)$, which denotes the largest delay in underlay observations at time $t$ and assume that

$$\sum_{t=0}^{T-1} \frac{L(t)}{T} = o(T).$$

The condition is not hard to satisfy. If the observations of underlay nodes occur with fixed interval, then it is easy to show that $\sum_{t=0}^{T-1} L(t)/T = \mathcal{O}(1)$. More generally, the condition is met as long as the $k^{th}$ observation interval of underlay nodes grows slower than the order of $k^{\alpha}$ where $\alpha \geqslant 0$.

We assume the legacy control policies of the underlay nodes are queue agnostic (i.e. the actions are independent of the queue backlogs), such as randomized routing and shortest path protocols. For an underlay node $i \in \mathcal{U}$, it transmits at most $\mu_{ijk}(t)$ packets of class $k$ to its neighbors $j$. Since the queue agnostic policies are stationary, $\mathbb{E}[\mu_{ijk}(t)]$ remains constants at different times and we denote that $\mathbb{E}[\mu_{ijk}(t)] = \mu_{ijk}$ (while the actual number of packets transmitted is $\tilde{\mu}_{ijk}(t)$ and may be queue-dependent). Note that our algorithm only operates

TABLE I
ASYMPTOTIC NOTATIONS

| | |
|---|---|
| $f(n)=O\big(g(n)\big)$ | $\lvert f\rvert$ is upper bounded by $g$ asymptotically, i.e., $\limsup_{n\to\infty}\big\lvert f(n)\big\rvert/g(n)<\infty$ |
| $f(n)=o\big(g(n)\big)$ | $\lvert f\rvert$ is dominated by $g$ asymptotically, i.e., $\limsup_{n\to\infty}\big\lvert f(n)\big\rvert/g(n)=0$ |
| $f(n)=\Omega\big(g(n)\big)$ | $f$ is lower bounded by $g$ asymptotically, i.e., $\liminf_{n\to\infty} f(n)/g(n)>0$ |
| $f(n)=\Theta\big(g(n)\big)$ | $f(n)=O\big(g(n)\big)$ and $f(n)=\Omega\big(g(n)\big)$ |

on overlay nodes and does not require control over underlay nodes.

All nodes receive the packets transmitted by their neighbors. Note that the actual packets received are denoted by $\tilde{f}_{ijk}(t)$ and $\tilde{\mu}_{ijk}(t)$.

Thus, the queue backlogs evolve according to the following,

$$Q_{ik}(t+1)=\begin{cases}\Big[Q_{ik}(t)+a_{ik}(t)-\sum_{j\in\mathcal{N}} f_{ijk}(t)\Big]^+ + \\ \quad \sum_{j\in\mathcal{O}}\tilde{f}_{jik}(t)+\sum_{j\in\mathcal{U}}\tilde{\mu}_{jik}(t), & i\in\mathcal{O}\\ \Big[Q_{ik}(t)+a_{ik}(t)-\sum_{j\in\mathcal{N}} \mu_{ijk}(t)\Big]^+ + \\ \quad \sum_{j\in\mathcal{O}}\tilde{f}_{jik}(t)+\sum_{j\in\mathcal{U}}\tilde{\mu}_{jik}(t), & i\in\mathcal{U},\end{cases}$$

where $[x]^+\triangleq\max\{x,0\}$. We further assume that the system dynamics are bounded, i.e.

$$0\leqslant a_{ik}(t),f_{ijk}(t),\mu_{ijk}(t)\leqslant D,\quad \forall i,j,k,t \tag{1}$$

for some constant $D\geqslant 0$. Moreover, to distinguish the variables under different policies, we use superscripts (e.g., $Q_{ik}^{\pi_A}(t)$ is the queue backlog of node $i$ at time $t$ under policy $\pi_A$).

Our goal is to stabilize the entire network when only overlay dynamics and estimated underlay state information are available, and policies can only be applied to overlay nodes.

### A. Asymptotic Notations

Given two functions $f(n)$ and $g(n)$, their asymptotic relationships are listed in Table I.

### B. Performance Metric

The stability region for an overlay-underlay networks is defined as follows.

*Definition 1: For an overlay-underlay network $\mathcal{G}$, the rate stability region $\Pi$ is the set of $\lambda_{ik}$'s such that there exist a policy $\pi^*$ under which the queues are mean rate stable, i.e.,*

$$\lim_{T\to\infty}\frac{\mathbb{E}\left[\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^*(T)\right]}{T}=0.$$

Mean rate stability implies that as $t\to\infty$, the expected queue backlog grows up to a sublinear factor of $t$ and the arrival rate is no greater than the service rate.

We now define throughput optimality as follows.

*Definition 2: A policy $\pi$ is throughput optimal if for any set of $\lambda_{ik}$'s in $\Pi$, the system is mean rate stable under $\pi$.*

Throughput optimal policies are desirable since they can stabilize the network whenever the network is stabilizable.

For readers' convenience, we summarize the variable notations in Table II.

TABLE II
VARIABLE NOTATIONS

| | |
|---|---|
| $N$ | The number of nodes |
| $C_{ij}$ | The link capacity between node $i$ and $j$ |
| $T$ | The time horizon |
| $\mathcal{N}$ | The set of all nodes |
| $\mathcal{O}$ | The set of overlay nodes |
| $\mathcal{U}$ | The set of underlay nodes |
| $\Gamma_{ik}$ | The set of time slots when an estimation of $Q_{ik}$ was made for node $i\in\mathcal{U}$ |
| $\pi$ | The routing policy for overlay nodes |
| $Q_{ik}^{\pi}(t)$ | Under policy $\pi$, the class $k$ queue backlog of node $i$ at $t$ |
| $\hat{Q}_{ik}^{\pi}(t)$ | Under policy $\pi$, the estimated class $k$ queue backlog of node $i\in\mathcal{U}$ at $t\in\Gamma_{ik}$ |
| $a_{ik}^{\pi}(t)$ | The number of class $k$ external packets arriving at node $i$ at $t$ |
| $f_{ijk}^{\pi}(t)$ | Under policy $\pi$, the planned number of class $k$ packets transmitted from node $i\in\mathcal{O}$ to $j\in\mathcal{N}$ at $t$ |
| $\tilde{f}_{ijk}^{\pi}(t)$ | Under policy $\pi$, the actual number of class $k$ packets transmitted from node $i\in\mathcal{O}$ to $j\in\mathcal{N}$ at $t$ |
| $\mu_{ijk}(t)$ | The planned number of class $k$ packets transmitted from node $i\in\mathcal{U}$ to $j\in\mathcal{N}$ at $t$ |
| $\tilde{\mu}_{ijk}(t)$ | The actual number of class $k$ packets transmitted from node $i\in\mathcal{U}$ to $j\in\mathcal{N}$ at $t$ |
| $g_{ijk}^{\pi}(t)$ | In the imaginary network, under policy $\pi$, the planned number of class $k$ packets transmitted from node $i\in\mathcal{U}$ to $j\in\mathcal{N}$ at $t$ |
| $\tilde{g}_{ijk}^{\pi}(t)$ | In the imaginary network, under policy $\pi$, the actual number of class $k$ packets transmitted from node $i\in\mathcal{U}$ to $j\in\mathcal{N}$ at $t$ |
| $X_{ik}^{\pi}(t)$ | Under policy $\pi$, the virtual class $k$ queue backlog of node $i\in\mathcal{U}$ at $t$ |
| $Y_{ik}^{\pi}(t)$ | $Q_{ik}^{\pi}(t)-X_{ik}^{\pi}(t)$ for $i\in\mathcal{U}$ |
| $\tau_{ik}(t)$ | The most recent time an estimate of $Q_{ik}$ was made for node $i\in\mathcal{U}$ at $t$ |
| $L(t)$ | The maximum delay of estimates at $t$, i.e., $\max_{i\in\mathcal{U}} t-\tau_{ik}(t)$ |

## III. OUR APPROACH

A key challenge in the control of partially observable and controllable networks is that the instantaneous underlay state information cannot be directly observed. If the control actions are only based on overlay information, they may lead to instability. Therefore, the core idea behind our approach is to approximate the underlay queue backlogs and incorporate them into the decision making process.

### A. Overview

Our approach constructs an "imaginary" network that has the same topology and external arrivals as the real network, with the only difference being that the underlay nodes can be instantaneously observed and controlled in the imaginary network. For $i\in\mathcal{O}$, the overlay queue backlogs in the imaginary network are the same as the real network, and we continue to denote them by $Q_{ik}$. For $i\in\mathcal{U}$, the underlay queue backlogs and policies of the imaginary network may differ from the real underlay dynamics. We denote by $X_{ik}$ and $g_{ijk}$ the queue backlog and policy of underlay node $i$ for class $k$ traffic in the imaginary network.

Since the imaginary network is fully observable and controllable, its total queue backlog $\sum_{i\in\mathcal{O},k} Q_{ik}+\sum_{i\in\mathcal{U},k} X_{ik}$

could be stabilized using classical network control algorithms like MaxWeight and BackPressure [2]. Our approach tries to "drive" the real network towards the dynamics of the imaginary network. However, the gap in the underlay queue backlogs between the real system and the imaginary system may grow, making the real system unstable even if the imaginary network has been stabilized.

Therefore, we denote by $Y_{ik} = Q_{ik} - X_{ik}$ the gaps in the underlay queue backlogs between the real system and the imaginary system and aim to stabilize $Y_{ik}$'s as well. Since the total queue backlog of the real system can be expressed as $\sum_{i\in\mathcal{O},k} Q_{ik}+\sum_{i\in\mathcal{U},k} X_{ik}+\sum_{i\in\mathcal{U},k} Y_{ik}$, if we could stabilize $Q_{ik}$'s, $X_{ik}$'s and $Y_{ik}$'s simultaneously, the real system will be stable.

### B. Algorithm

We apply a Lyapunov optimization framework to stabilize $Q_{ik}$, $X_{ik}$ and $Y_{ik}$ simultaneously and name it the Tracking MaxWeight* (TMW*) algorithm. Specifically, we define a Lyapunov function

$$\Phi(t) \triangleq \sum_{i\in\mathcal{O},k} Q_{ik}^2(t) + \sum_{i\in\mathcal{U},k} X_{ik}^2(t) + \sum_{i\in\mathcal{U},k} Y_{ik}^{+2}(t),$$

where $Y_{ik}^+ = \max\{Y_{ik}, 0\}$.

We aim at minimizing the drift $\Delta\Phi(t) = \Phi(t+1) - \Phi(t)$ at each time slot. Directly minimizing $\Delta\Phi(t)$ is hard, and we need to decompose $\Delta\Phi(t)$ into analyzable terms. For simplicity in expression, we make the following definitions of the one-slot changes of $Q_{ik}(t)$'s, $X_{ik}(t)$'s and $Y_{ik}(t)$'s. Note that we use $\delta$ instead of $\Delta$ for $\delta Q_{ik}(t)$ and $\delta X_{ik}(t)$ because they are not the actual one-slot changes but the changes without imposing the work conservation constraints.

$$\begin{cases} \delta Q_{ik}(t) \triangleq & a_{ik}(t) - \sum_{j\in\mathcal{N}} f_{ijk}(t) + \sum_{j\in\mathcal{O}} f_{jik}(t) + \\ & \sum_{j\in\mathcal{U}} \mu_{jik}(t), \quad i \in \mathcal{O} \\ \delta X_{ik}(t) \triangleq & \lambda_{ik}(t) - \sum_{j\in\mathcal{N}} g_{ijk}(t) + \sum_{j\in\mathcal{O}} f_{jik}(t) + \\ & \sum_{j\in\mathcal{U}} g_{jik}(t), \quad i \in \mathcal{U} \\ \Delta Y_{ik}(t) \triangleq & Y_{ik}(t+1) - Y_{ik}(t), \quad i \in \mathcal{U} \end{cases}$$

We first upper bound $Q_{ik}^2(t+1) - Q_{ik}^2(t)$ for $i \in \mathcal{O}$ and $X_{ik}^2(t+1) - X_{ik}^2(t)$ for $i \in \mathcal{U}$ in Lemma 1.

*Lemma 1: For each $k \in \mathcal{K}$ and $t = 0, \cdots, T-1$ and we have*

$$\begin{cases} Q_{ik}^2(t+1) - Q_{ik}^2(t) \leqslant 2\ Q_{ik}(t)\delta Q_{ik}(t) + 6N^2D^2, & i \in \mathcal{O} \\ X_{ik}^2(t+1) - X_{ik}^2(t) \leqslant 2\ X_{ik}(t)\delta X_{ik}(t) + 6N^2D^2, & i \in \mathcal{U} \end{cases}$$

See Appendix A for the proof. We then upper bound $Y_{ik}^{+2}(t+1) - Y_{ik}^{+2}(t)$ for $i \in \mathcal{U}$ in Lemma 2.

*Lemma 2: For each $i \in \mathcal{U}$, $k \in \mathcal{K}$ and $t = 0, \cdots, T-1$, we have*

$$Y_{ik}^{+2}(t+1) - Y_{ik}^{+2}(t) \leqslant 2\hat{Y}_{ik}^{+}(t)\Delta Y_{ik}(t) + \big(8L(t)+6\big)N^2D^2.$$

See Appendix B for the proof. By Lemma 1 and Lemma 2, instead of directly minimizing $\Delta\Phi(t)$, we can minimize

$$\sum_{i\in\mathcal{O},k} Q_{ik}(t)\delta Q_{ik}(t) + \sum_{i\in\mathcal{U},k} X_{ik}(t)\delta X_{ik}(t) + \sum_{i\in\mathcal{U},k} Y_{ik}^{+}(t)\Delta Y_{ik}^{+}(t). \quad (2)$$

However, the controller do not have instantaneous access to $Q_{ik}(t)$ and hence $Y_{ik}^+(t)$ for $i \in \mathcal{U}$. As discussed in Section II, the controller obtain an estimate $\hat{Q}_{ik}(t)$ for node $i \in \mathcal{U}$ at time $t \in \Gamma_{ik}$. Therefore, the controller can use the most recent $\hat{Q}_{ik}(t)$ to estimate $Y_{ik}^+(t)$, i.e.,

$$\hat{Y}_{ik}^{+}(t) \triangleq \big[\hat{Q}_{ik}\big(\tau_{ik}(t)\big) - X_{ik}(t)\big]^+.$$

This optimization can be formulated as (3), shown at the bottom of the next page. The only non-linear component of the problem is $\min\Big\{\sum_{j\in\mathcal{N}} g_{ijk}, X_{ik}(t) + \lambda_{ik}(t)\Big\}$. To tackle the non-linear issue, we can split the problem into two linear programming problems: in the first problem, the component is replaced by $\sum_{j\in\mathcal{N}} g_{ijk}$ with an extra constraint $\sum_{j\in\mathcal{N}} g_{ijk} \leqslant X_{ik}(t) + \lambda_{ik}(t)$. In the second problem, the component is replaced by $X_{ik}(t) + \lambda_{ik}(t)$ with an extra constraint $\sum_{j\in\mathcal{N}} g_{ijk} > X_{ik}(t) + \lambda_{ik}(t)$. The controller solve the two linear programming problems simultaneously and selects the solution with the better result. Therefore, solving (3) is equivalent to solving two linear programming problems. Since numerous efficient algorithms have been developed for linear programming, (3) can be solved efficiently.

We denote the solution to (3) by $\boldsymbol{f}^{\pi_T}(t)$ and $\boldsymbol{g}^{\pi_T}(t)$, where "T" is an abbreviation of the TMW* algorithm. We apply $\boldsymbol{f}^{\pi_T}(t)$ to overlay nodes in the real system. We then use $\boldsymbol{f}^{\pi_T}(t)$, $\boldsymbol{g}^{\pi_T}(t)$ and the estimated underlay arrival rates $\lambda_{ik}(t)$ to update the underlay queue backlogs of the imaginary system according to the update rule

$$X_{ik}(t+1) = \left[X_{ik}(t) + \lambda_{ik}(t) - \sum_{j\in\mathcal{N}} g_{ijk}(t)\right]^+ + \sum_{j\in\mathcal{O}} \tilde{f}_{jik}(t) + \sum_{j\in\mathcal{U}} g_{jik}(t). \quad (4)$$

The complete algorithm is given in Algorithm 1.

**Algorithm 1** The TMW* Algorithm

**Input:** $T$, $Q_{ik}(0)$, $\Gamma_{ik}$
**Initialization:** $X_{ik}(0) \leftarrow Q_{ik}(0)$, $Y_{ik}(0) \leftarrow 0$
**for** $t \leftarrow 0, 1, \cdots, T-1$ **do**
  **for** $k \in \mathcal{K}$ **do**
    Observe $Q_{ik}(t)$ and $a_{ik}(t)$ for $i \in \mathcal{O}$
    **for** $i \in \mathcal{U}$ **do**
      **if** $t \in \Gamma_{ik}$ **then**
        Obtain $\hat{Q}_{ik}(t)$ and $\lambda_{ik}(t)$
      **end if**
      Update $\hat{Y}_{ik}(t)$
    **end for**
  **end for**
  Solve Eqn (3) and obtain $\boldsymbol{f}^{\pi_T}(t), \boldsymbol{g}^{\pi_T}(t)$
  Implement $\boldsymbol{f}^{\pi_T}(t)$ to overlay nodes $\mathcal{O}$ in the real network
  Update $X_{ik}(t+1)$ using Eqn (4)
**end for**
**Output:** Overlay policy $\boldsymbol{f}^{\pi_T}(t)$ for $t = 0, \cdots, T-1$

## IV. Performance Analysis

We aim to design an overlay algorithm that can stabilize the entire network whenever it is stabilizable (i.e. throughput

optimal). We show that our algorithm is throughput optimal as in Theorem 1.

*Theorem 1: TMW* is throughput optimal.*

*Proof:* The outline of the proof is as follows. We first upper bound the queue backlogs by the Lyapunov function $\Phi$ in Lemma 3, so that we only need to analyze the Lyapunov value. In Lemma 4 and 5, we upper bound the drift $\Delta\Phi$. We finally upper bound the sum of drift values in Lemma 4 and 5, which gives us an upper bound for the Lyapunov value and completes the proof.

To show throughput optimality, we consider an arbitrary set of $\lambda_{ik}$'s and $\mu_{ijk}$'s in $\Pi$. In order to analyze the growth of queue backlogs, we first explore the relationship between queue backlogs and the Lyapunov function $\Phi$.

*Lemma 3: For any policy $\pi$, we have*

$$\mathbb{E}\Big[\sum_{i,k} Q_{ik}^{\pi}(T)\Big] \leqslant \sqrt{2KN\mathbb{E}\big[\Phi^{\pi}(T)\big]}$$

See Appendix C for the proof. Lemma 3 indicates that showing $\mathbb{E}\big[\Phi^{\pi}(T)\big] = o(T^2)$ is sufficient for throughput optimality. We use the superscript $\pi_T$ to represent the variables under TMW*. Using Lemma 1 and Lemma 2, we can upper bound $\Delta\Phi^{\pi_T}(t) \triangleq \Phi^{\pi_T}(t+1) - \Phi^{\pi_T}(t)$ as follows,

$$\begin{aligned}&\Delta\Phi^{\pi_T}(t)\\ &\leqslant 2\sum_{i\in\mathcal{O},k} Q_{ik}^{\pi_T}(t)\delta^{\pi_T}Q_{ik}(t) + 2\sum_{i\in\mathcal{U},k} X_{ik}^{\pi_T}(t)\delta^{\pi_T}X_{ik}(t)\\ &\quad + 2\sum_{i\in\mathcal{U},k} \hat{Y}_{ik}^{\pi_T+}(t)\Delta^{\pi_T}Y_{ik}(t) + (8L(t)+18)KN^3D^2.\end{aligned} \tag{5}$$

By Definition 1, there exist a policy $\pi^*$ such that $Q_T^* \triangleq \mathbb{E}\big[\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^*(T)\big] = o(T)$, where we use the superscript $*$ to represent the variables under $\pi^*$. Since TMW* minimizes the first three terms of (5), replacing $\delta^{\pi_T}Q_{ik}(t)$, $\delta^{\pi_T}X_{ik}(t)$ and $\Delta^{\pi_T}Y_{ik}(t)$ with $\delta^*Q_{ik}(t)$, $\delta^*X_{ik}(t)$ and $\Delta^*Y_{ik}(t)$ respectively will not decrease (5), i.e.,

$$\begin{aligned}&\Delta\Phi^{\pi_T}(t)\\ &\leqslant 2\sum_{i\in\mathcal{O},k} Q_{ik}^{\pi_T}(t)\delta^*Q_{ik}(t) + 2\sum_{i\in\mathcal{U},k} X_{ik}^{\pi_T}(t)\delta^*X_{ik}(t)\\ &\quad + 2\sum_{i\in\mathcal{U},k} \hat{Y}_{ik}^{\pi_T+}(t)\Delta^*Y_{ik}(t) + (8L(t)+18)KN^3D^2.\end{aligned} \tag{6}$$

By taking expectation on the sum of (6) from $t=0$ to $t=T-1$, we have

$$\begin{aligned}\mathbb{E}\Big[\Phi^{\pi_T}(T)\Big] &\leqslant 2\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i\in\mathcal{O},k} Q_{ik}^{\pi_T}(t)\delta^*Q_{ik}(t)\right]\\ &\quad + 2\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i\in\mathcal{U},k} X_{ik}^{\pi_T}(t)\delta^*X_{ik}(t)\right]\\ &\quad + 2\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i\in\mathcal{U},k} \hat{Y}_{ik}^{\pi_T+}(t)\Delta^*Y_{ik}(t)\right]\\ &\quad + 2KN^3D^2\left(9T + 4\sum_{t=0}^{T-1}L(t)\right) + \Phi(0).\end{aligned} \tag{7}$$

For the first and second terms in (7), we have the following lemma.

*Lemma 4: For each integer $H > 0$, the following holds*

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i\in\mathcal{O},k} Q_{ik}^{\pi_T}(t)\delta^*Q_{ik}(t) + \sum_{t=0}^{T-1}\sum_{i\in\mathcal{U},k} X_{ik}^{\pi_T}(t)\delta^*X_{ik}(t)\right]\\ &\leqslant \frac{2NDT^2Q_T^*}{H} + 8KN^3D^2HT\end{aligned}$$

See Appendix D for the proof. We next upper bound the third term as follows,

*Lemma 5:*

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i\in\mathcal{U},k} \hat{Y}_{ik}^{\pi_T+}(t)\Delta^*Y_{ik}(t)\right]\\ &\leqslant 2\ KN^3D^2\left(T + 2\sum_{t=0}^{T-1}L(t)\right).\end{aligned}$$

See Appendix E for the proof. Using results in Lemma 4 (with $H = c\sqrt{TQ_T^*/(KN^2D)}$ where $c$ is any positive constant that makes $H$ an integer) and Lemma 5 in (7) and

$$\begin{aligned}\boldsymbol{f}^{\pi_T}(t), \boldsymbol{g}^{\pi_T}(t) = \arg\min_{\boldsymbol{f},\boldsymbol{g}} &\sum_{i\in\mathcal{O},k} Q_{ik}(t)\left[\sum_{j\in\mathcal{O}} f_{jik} + \sum_{j\in\mathcal{U}} g_{jik} - \sum_{j\in\mathcal{N}} f_{ijk}\right] + \sum_{i\in\mathcal{U},k} X_{ik}(t)\left[\sum_{j\in\mathcal{O}} f_{jik} + \sum_{j\in\mathcal{U}} g_{jik} - \sum_{j\in\mathcal{N}} g_{ijk}\right]\\ &- \sum_{i\in\mathcal{U},k} \hat{Y}_{ik}^{+}(t)\left[\sum_{j\in\mathcal{U}} g_{jik} - \min\left\{\sum_{j\in\mathcal{N}} g_{ijk}, X_{ik}(t) + \lambda_{ik}(t)\right\}\right],\\ s.t. \qquad &f_{ijk} \geqslant 0,\ g_{ijk} \geqslant 0,\ \sum_{k\in\mathcal{K}} f_{ijk} \leqslant C_{ij},\ \sum_{k\in\mathcal{K}} g_{ijk} \leqslant C_{ij}.\end{aligned} \tag{3}$$

then applying Lemma 3, we obtain,

$$\begin{aligned}\mathbb{E}\left[\sum_{i\in\mathcal{N}} Q_{ik}^{\pi_T}(T)\right] &\leqslant \left[4KN^3D\left((8c+2/c)\sqrt{KDTQ_T^*}+11KND\right)T \right. \\ &\quad \left. +32K^2N^4D^2\sum_{t=0}^{T-1}L(t)+2KN\Phi(0)\right]^{1/2} \\ &= O\left(T^{3/4}Q_T^{*1/4}+\sqrt{\sum_{t=0}^{T-1}L(t)}\right).\end{aligned}$$

Since $Q_T^*=o(T)$ by Definition 1, when $\sum_{t=0}^{T-1}L(t)/T=o(T)$, the expected queue backlog at $T$ is upper bounded by a sublinear factor of $T$. □

## V. Performance With Estimation Errors

In Section IV, we showed that TMW* is a throughput optimal network control algorithm when the estimates $\hat{Q}_{ik}(t)$ are accurate. However, in practice, it is hard to obtain accurate estimates because statistical methods have fundamental performance limits, and transmission errors can pollute the collected data. For an underlay node $i\in\mathcal{U}$ and $t\in\Gamma_{ik}$, we define the error as $\epsilon_{ik}(t)\triangleq\hat{Q}_{ik}(t)-Q_{ik}(t)$ and the erroneous estimates of $Y_{ik}(t)$ as $\tilde{Y}_{ik}(t)=\hat{Y}_{ik}(t)+\epsilon_{ik}(\tau_{ik}(t))$.

### A. Performance of TMW*

The variable $\hat{Y}_{ik}^+(t)$ in Algorithm 1 is now replaced by $\tilde{Y}_{ik}^+(t)$ and the goal becomes to minimize

$$\begin{aligned}\Delta\Phi(t)=&\sum_{i\in\mathcal{O},k}Q_{ik}(t)\Delta Q_{ik}(t)+\sum_{i\in\mathcal{U},k}X_{ik}(t)\Delta X_{ik}(t)\\&+\sum_{i\in\mathcal{U},k}\tilde{Y}_{ik}^+(t)\Delta Y_{ik}^+(t).\end{aligned}\tag{8}$$

In Theorem 2, we show that as long as the scale of $\epsilon_{ik}(t)$ is sublinear in $t$, TMW* is still a throughput optimal algorithm.

*Theorem 2: TMW* is a throughput optimal network control policy if $|\epsilon_{ik}(t)|=o(t)$ for each $i\in\mathcal{U}$.*

*Proof:* The analysis is nearly identical to the proof of Theorem 1, with the only difference in upper bounding $Y_i^{+2}(t+1)-Y_i^{+2}(t)$ and $\sum_{t=0}^{T-1}\tilde{Y}_i^{\pi_T+}(t)\Delta^*Y_{ik}(t)$, as given by Lemma 6 and 7 below (see Appendix F and G for the proofs).

*Lemma 6:* For each $i\in\mathcal{U}$, $k\in\mathcal{K}$ and $t=0,\cdots,T-1$, we have

$$\begin{aligned}&Y_{ik}^{+2}(t+1)-Y_{ik}^{+2}(t)\\&\quad\leqslant 2\tilde{Y}_{ik}^+(t)\Delta Y_{ik}(t)+\big(8L(t)+6\big)N^2D^2\\&\qquad+4ND\left|\epsilon_{ik}(\tau_{ik}(t))\right|.\end{aligned}$$

*Lemma 7:*

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i\in\mathcal{U},k}\hat{Y}_{ik}^{\pi_T+}(t)\Delta^*Y_{ik}(t)\right]\\&\quad\leqslant 2\ KN^3D^2\left(T+2\sum_{t=0}^{T-1}L(t)\right)\\&\qquad+2KN^2\sum_{t=0}^{T-1}\sum_{i\in\mathcal{U}}\left|\epsilon_{ik}(\tau_{ik}(t))\right|.\end{aligned}$$

With Lemma 6 and 7, by applying a similar analysis to the proof of Theorem 1, we have that

$$\begin{aligned}&\mathbb{E}\left[\sum_{i,k}Q_{ik}^{\pi}(T)\right]\\&=O\left(T^{3/4}Q_T^{*1/4}+\sqrt{\sum_{t=0}^{T-1}L(t)}+\sum_{t=0}^{T-1}\sum_{i\in\mathcal{U}}\left|\epsilon_{ik}(\tau_{ik}(t))\right|\right)\end{aligned}\tag{9}$$

When $|\epsilon_{ik}(t)|=o(t)$, we have

$$\sum_{t=0}^{T-1}\sum_{i\in\mathcal{U}}\left|\epsilon_{ik}(\tau_{ik}(t))\right|\leqslant\sum_{i\in\mathcal{U}}\sum_{t=0}^{T-1}o(t)=o(T^2).$$

Therefore, if $Q_T^*=o(T)$, $\sum_{t=0}^{T-1}L(t)/T=o(T)$ and $|\epsilon_{ik}(t)|=o(t)$, we have $\sum_{i\in\mathcal{N}}Q_{ik}^{\pi_T}(T)=o(T)$, which completes the proof of Theorem 2. □

### B. Impact of Estimation Errors

By Theorem 2, as long as the estimation error grows sublinearly in $t$, TMW* is still throughput optimal. However, when the estimation error grows linearly or even faster in $t$, whether the network is stabilizable becomes a question of interest.

By Definition 1, for a given set of external data arrival rates inside the stability region, there always exists a randomized policy that reaches rate stability. The randomized policy is independent of network state and is immune to estimation errors. However, to obtain a stabilizing randomized policy usually requires the knowledge of network dynamics (i.e., arrival rates), which is unrealistic in practice. Moreover, a given randomized policy may only support a subset of the stability region, and may not be throughput optimal. Practical control algorithms like MaxWeight and BackPressure [2] usually only utilize queue information and are throughput optimal. Therefore, we focus on the "queue-based throughput optimal policies" defined as follows

*Definition 3: A queue-based throughput optimal policy generates actions solely based on the current queue backlogs of the overlay nodes $\mathcal{O}$ and the current estimated queue backlogs of the underlay nodes $\mathcal{U}$, and stabilizes the entire network whenever the arrival rates are inside the stability region.*

MaxWeight and BackPressure are examples of queue-based throughput optimal policies in fully observable and controllable networks. The actions taken are only decided by the queue backlogs,are independent of the arrival rates, and could stabilize the system whenever inside the stability region.

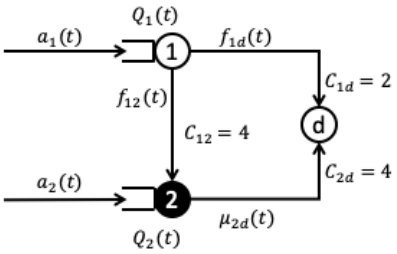



Fig. 1. Constructed example system for Theorem 3.

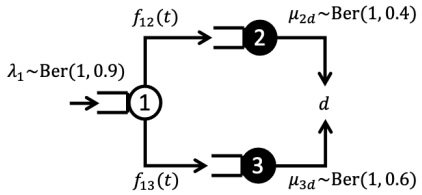



Fig. 2. The 3-node network for simulation.

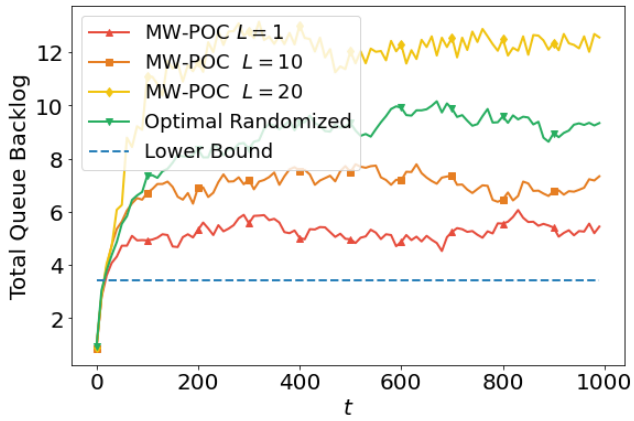



Fig. 3. Average queue backlog for the 3-node network.

In contrast, randomized policies may fail to stabilize the system if the arrival rates change (while still inside the stability region).

The following theorem shows that when noise is superlinear in $t$, there is no queue-based throughput optimal policy for all network topologies and dynamics.

*Theorem 3: There exists a network with $\epsilon_{ik}(t) = \Omega(t)$ for some $i \in \mathcal{U}$ such that no queue-based throughput optimal policy exists.*

*Proof:* We construct a 3-node network with a single class of data as in Figure 1. Node 1 can directly transmit data packets to destination $d$ or relay through node 2, while node 2 is neither observable nor controllable.

The idea behind the proof is that for any arrival rates, the queue backlogs can grow at most linearly in $t$. Therefore, when the estimation error grows linearly in $t$, the error can completely "mask" the actual queue growth of node 2 and makes $\hat{Q}_2(t) \equiv 0$, causing the controller to transmit packets from node 1 to node 2. However, the external arrival rate to node 2 might be very close to $C_{2d}$ and the total arrival rate of node 2 may exceed $C_{2d}$ even if $f_{12}$ is small. The queue backlog at node 2 then grows linearly in the time horizon and the entire network becomes unstable. The detailed proof is given in Appendix H. □

Theorem 3 shows that when the estimation errors scale linearly or sup-linearly in $t$, there does not exist a universal queue-based throughput optimal policy for all partially observable and controllable networks. On the other hand, Theorem 2 shows that TMW* is throughput optimal as long as the estimation errors grow sublinearly in $t$. Therefore, TMW* is optimally robust to estimation errors.

## VI. Numerical Experiments

We conduct simulations on serveral network models to validate the performance analysis of TMW*. We start from a simple 3-node network, which has an explicit lower bound and can be used to evaluate the gap between TMW* and optimum. We then implement TMW* on a 15-node network to examine the performance of TMW* in a more compex network model. We finally consider the 15-node network model with different scales of estimation errors.

### A. 3-Node Network

We first consider a simple 3-node network, as shown in Figure 2.

In the network, only node 1 is an overlay node, while nodes 2 and 3 are underlay nodes. We assume all links have a capacity of 1. The network only has a single class of traffic to be transmitted from node 1 to destination $d$. During each time slot, node 1 receives a new packet with probability 0.9 and receives no packets otherwise (i.e., Bernoulli process $\text{Ber}(1, 0.9)$). Node 1 then transmit packets to node 2 and node 3 according to the applied policy. Node 2 and 3 are uncontrollable and apply randomized policies. Node 2 serves one packet with probability 0.4, and serves no packet otherwise (i.e., $\text{Ber}(1, 0.4)$). Similarly, the action taken by node 3 is a Bernoulli process $\text{Ber}(1, 0.6)$.

We first derive a lower bound of the expected queue backlog. We consider a dominant network with the same topology and dynamics except that the service process at node 2 is changed to $\mu_{2d}(t) \sim \text{Ber}(1, 0.6)$. The dominant network has smaller expected queue backlog and becomes an $M/M/2$ queueing system. By applying the analytical techniques for $M/M/c$ queueing systems in [23], the expected queue backlog of the dominant network is $24/7$, which serves as a lower bound for the 3-node network.

We can also derive the optimal randomized policy for the 3-node network. Suppose node 1 attempts to transmit one packet to node 2 with probability $p$, and to node 3 otherwise, then the arrival rate to node 2 and node 3 are $0.9p$ and $0.9 - 0.9p$, respectively. Using analytical techniques for $M/M/1$ queueing systems, we can express the expected queue backlog using $p$ and further obtain the optimal choice of $p$ is $p^* = (2 - \sqrt{2/3})/3$.

We conduct simulation on the 3-node network using TMW* with different estimation intervals $L$ and the optimal randomized policy with $p^*$. The results are shown in Figure 3.

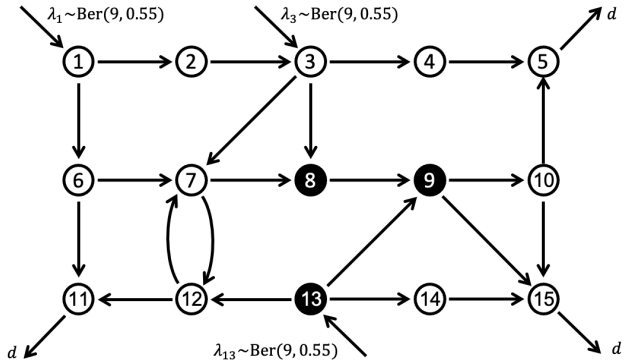



Fig. 4. The 15-node network for simulation.

From Figure 3, we can see that when the estimation interval is small, TMW* significantly outperforms the optimal randomized policy. A possible reason is that the randomized policy fails to consider the queue backlog information and may transmit to the node with larger queue. Moreover, the gap between TMW* and the lower bound is relatively small, which shows that TMW* is close to the optimal policy in this case.

### *B. 15-Node Network*

We next study a 15-node partially observable queueing network as in Figure 4. The system consists of 12 overlay nodes and 3 underlay nodes. At the beginning of each time slot, external packets arrive at nodes 1, 3 and 13 at random with rates of $\lambda_1$, $\lambda_3$ and $\lambda_{13}$ respectively. Each node then decides to which neighbors to relay the buffered packets. The destination $d$ can be reached via nodes 5, 11 and 15. We aim to stabilize the entire network by implementing policies only on overlay nodes.

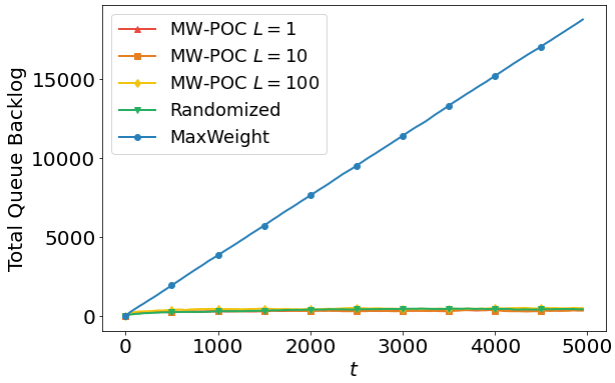
For conciseness in illustration, we let all link capacities be 5 (including the links $5 \to d$, $11 \to d$ and $15 \to d$). We let the underlay transmission rates for nodes 8, 9 and 13 be random and uniform between 0 and 5 packets on each outgoing link, i.e.

$$\mu_{8\to 9}(t), \mu_{9\to 15}(t), \mu_{13\to 9}(t), \mu_{13\to 12}(t), \mu_{13\to 14}(t) \sim \text{Unif}\{0, \cdots, 5\},$$

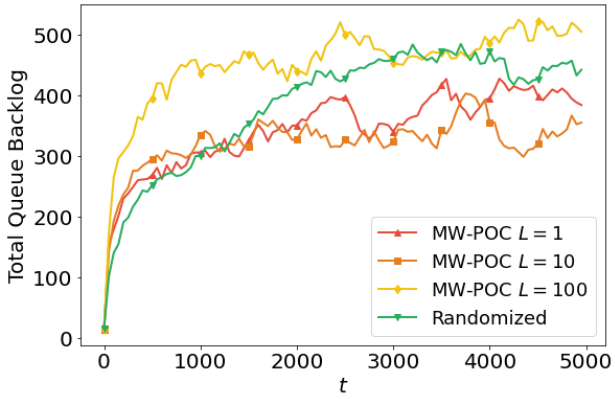
and the updates from underlay nodes have a fixed interval of $L$ time slots. We also set the external arrivals for node $\lambda_1$, $\lambda_3$ and $\lambda_{13}$ be i.i.d. Bernoulli with 9 packet arrivals with probability 0.55 and no arrivals with probability 0.45.

It can be easily shown that a stabilizing randomized overlay policy is to fix the transmission rates on route $1 \to 6 \to 11 \to d$, $3 \to 4 \to 5 \to d$, $12 \to 7 \to 8$, $10 \to 15 \to d$ and $14 \to 15 \to d$ to 5, while keeping other overlay link rates to zero. In the simulation, we compared the evolution of the total queue backlog under the above randomized policy, and TMW* with different update intervals $L$. We also directly applied the traditional MaxWeight algorithm to the overlay nodes as a baseline method. The results are shown in Figure 5.

From Figure 5a, we can see that under the traditional MaxWeight algorithm, the average queue backlog grows linearly in time. Therefore, traditional MaxWeight might not




(a) Comparison of all policies.




(b) Comparison of stabilizing policies.

Fig. 5. Average queue backlog for the 15-node network.

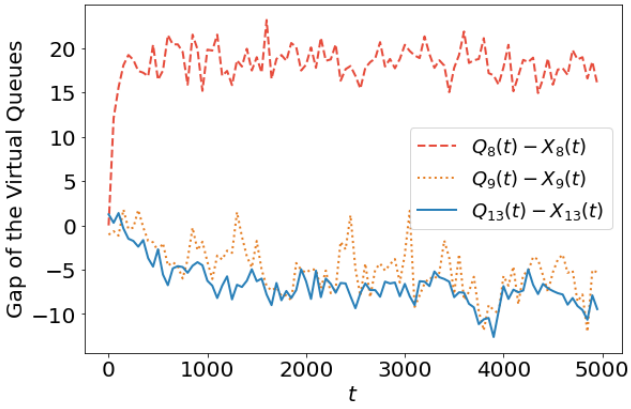



Fig. 6. The gaps between the virtual queues and actual queues for underlay nodes ($L = 10$).

be capable of stabilizing the network. We then focus on the performance of stabilizing policies in Figure 5b, which shows that while all of the shown values of $L$ can stabilize the system, smaller $L$'s leads to smaller average queue backlogs. This phenomenon matches intuition, as smaller $L$'s give fresher information about the underlay nodes.

We then plot the gaps between virtual queues $X_{ik}$'s and actual queue $Q_{ik}$'s for $i \in \mathcal{U}$ under $L = 10$. As can be seen from Figure 6, the gaps between $X_{ik}$'s and $Q_{ik}$'s are bounded by constants, which indicates that our estimates $X_{ik}$'s

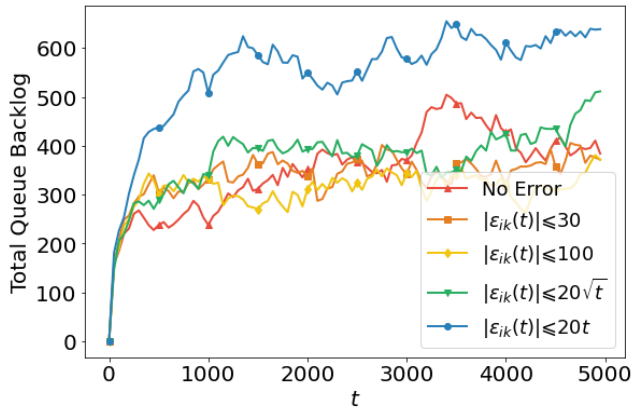



Fig. 7. Average queue backlogs in noisy environments.

for underlay queues are effective (otherwise the gap would grow without bound). Note that all the gaps have deviations from zero and a possible reason is that, in our simulation, the initial buffers are empty, and it takes some time to reach the stationary distribution.

### C. 15-Node Network With Estimation Errors

We continue using the 15-node network model in Section VI-B. Theorem 2 indicates that when the estimation error $|\epsilon_{ik}(t)| = o(t)$, our algorithm can stabilize the queues. We conducted simulations for different noise settings: when $|\epsilon_{ik}(t)| \leqslant 30$, $|\epsilon_{ik}(t)| \leqslant 100$, $|\epsilon_{ik}(t)| \leqslant 20\sqrt{t}$, $|\epsilon_{ik}(t)| \leqslant 20\ t$ and when there is no estimation error. The estimation error imposed is sampled uniformly inside the error scale region.

As can be seen from Figure 7, as the estimation error scale increases, the average queue backlogs grow larger, yet the system is still stable even when the error grows at the rate of $t$. Note that this result does not contradict Theorem 3, which only gives a specific example of a system that cannot be stabilized when the noise grows linearly in the time horizon.

## VII. Conclusion

In this paper, we focus on overlay-underlay networks in which the underlay nodes are unobservable and uncontrollable. We propose the TMW* algorithm that only requires sparse estimates of the underlay queue state and only needs to be implemented on the overlay nodes. We rigorously show that TMW* is throughput optimal. We then analyze the performance of TMW* when the estimation is erroneous and show that as long as the error scales sublinearly in time, TMW* still remains throughput optimal. We further explore the theoretical limit of queue-based control algorithm and show that TMW* is optimally robust to estimation errors. Simulations on multiple overlay-underlay networks validate our performance analysis.

For future works, a potential direction is to apply machine learning techniques to further optimize the control algorithms for overlay-underlay networks. Another possible direction is to develop inference methods for underlay queue backlogs and analyze the error bounds.

## Appendix A
## Proof of Lemma 1

We first upper bound $Q_{ik}^2(t+1)-Q_{ik}^2(t)$ for $i \in \mathcal{O}$. Writing down the update rule for $Q_{ik}^2(t)$, we have that

$$\begin{aligned} Q_{ik}(t+1) &= \left[Q_{ik}(t) + a_{ik}(t) - \sum_{j\in\mathcal{N}} f_{ijk}(t)\right]^+ \\ &\quad + \sum_{j\in\mathcal{O}} \tilde{f}_{jik}(t) + \sum_{j\in\mathcal{U}} \tilde{\mu}_{jik}(t) \\ &\leqslant \left[Q_{ik}(t) + a_{ik}(t) - \sum_{j\in\mathcal{N}} f_{ijk}(t)\right]^+ \\ &\quad + \sum_{j\in\mathcal{O}} f_{jik}(t) + \sum_{j\in\mathcal{U}} \mu_{jik}(t). \end{aligned}$$

It is easy to show that for $x, y, z \geqslant 0$, the inequality

$$\left([x-y]^+ + z\right)^2 \leqslant x^2 + y^2 + z^2 + 2x(z-y)$$

holds. By replacing $x$ with $Q_{ik}(t) + a_{ik}(t)$, $y$ with $\sum_{j\in\mathcal{N}} f_{ijk}(t)$ and $z$ with $\sum_{j\in\mathcal{O}} f_{jik}(t) + \sum_{j\in\mathcal{U}} \mu_{jik}(t)$, we upper bound $Q_{ik}^2(t+1)$ as

$$\begin{aligned} Q_{ik}^2(t+1) &\leqslant Q_{ik}^2(t) + \left(\sum_{j\in\mathcal{N}} f_{ijk}(t)\right)^2 \\ &\quad + \left(\sum_{j\in\mathcal{O}} f_{jik}(t) + \sum_{j\in\mathcal{U}} \mu_{jik}(t)\right)^2 \\ &\quad + 2\ a_{ik}(t)\delta Q_{ik}(t) + 2\ Q_{ik}(t)\delta Q_{ik}(t) \\ &\leqslant Q_{ik}^2(t) + 2\ Q_{ik}(t)\delta Q_{ik}(t) + 6N^2D^2, \quad (10) \end{aligned}$$

where the last inequality holds by utilizing (1).

We then upper bound $X_{ik}^2(t+1) - X_{ik}^2(t)$ for $i \in \mathcal{U}$. With

$$\begin{aligned} X_{ik}(t+1) &= \left[X_{ik}(t) + \lambda_{ik}(t) - \sum_{j\in\mathcal{N}} g_{ijk}(t)\right]^+ \\ &\quad + \sum_{j\in\mathcal{O}} \tilde{f}_{jik}(t) + \sum_{j\in\mathcal{U}} g_{jik}(t) \\ &\leqslant \left[X_{ik}(t) + \lambda_{ik}(t) - \sum_{j\in\mathcal{N}} g_{ijk}(t)\right]^+ \\ &\quad + \sum_{j\in\mathcal{O}} f_{jik}(t) + \sum_{j\in\mathcal{U}} g_{jik}(t), \end{aligned}$$

by applying similar techniques as (10), we have

$$X_{ik}^2(t+1) \leqslant X_{ik}^2(t) + 2\ X_{ik}(t)\delta X_{ik}(t) + 6N^2D^2. \quad (11)$$

## Appendix B
## Proof of Lemma 2

To avoid confusion, we define that $\Delta Y_{ik}^+(t) \triangleq Y_{ik}^+(t+1) - Y_{ik}^+(t)$. Both $\Delta Y_{ik}(t)$ and $\Delta Y_{ik}^+(t)$ are bounded as the following lemma.

*Lemma 8: For each $i \in \mathcal{U}$, $t = 0, \cdots, T-1$ and $k \in \mathcal{K}$, we have*

$$-2ND \leqslant \Delta Y_{ik}(t), \Delta Y_{ik}^+(t) \leqslant 2ND,$$

*Proof:* Here we fix an $i$ and a $t$ arbitrarily. We first discuss the range of $\Delta Y_{ik}(t)$. From the definition of $\Delta Y_{ik}(t)$, we have

$$\begin{aligned}
&\Delta Y_{ik}(t)\\
&\quad= Q_{ik}(t+1) - Q_{ik}(t) - \big(X_{ik}(t+1) - X_{ik}(t)\big)\\
&\quad= a_{ik}(t) - \sum_{j\in\mathcal{N}} \tilde{\mu}_{ijk}(t) + \sum_{j\in\mathcal{O}} \tilde{f}_{jik}(t) + \sum_{j\in\mathcal{U}} \tilde{\mu}_{jik}(t)\\
&\qquad- \lambda_{ik}(t) + \sum_{j\in\mathcal{N}} \tilde{g}_{ijk}(t) - \sum_{j\in\mathcal{O}} \tilde{f}_{jik}(t) - \sum_{j\in\mathcal{U}} g_{jik}(t)\\
&\quad= a_{ik}(t) - \lambda_{ik}(t) + \sum_{j\in\mathcal{U}} \tilde{\mu}_{jik}(t) - \sum_{j\in\mathcal{N}} \tilde{\mu}_{ijk}(t)\\
&\qquad+ \sum_{j\in\mathcal{N}} \tilde{g}_{ijk}(t) - \sum_{j\in\mathcal{U}} g_{jik}(t).
\end{aligned}$$

By applying (1), we have

$$-2ND \leqslant \Delta Y_{ik}(t) \leqslant 2ND. \tag{12}$$

With (12) at hand, we first have

$$\begin{aligned}
\Delta Y^+_{ik}(t) &= \max\{Y_{ik}(t+1), 0\} - Y^+_{ik}(t)\\
&= \max\{Y_{ik}(t+1) - Y^+_{ik}(t), -Y^+_{ik}(t)\}\\
&\leqslant \max\{Y_{ik}(t+1) - Y_{ik}(t), -Y^+_{ik}(t)\}\\
&= \max\{\Delta Y_{ik}(t), -Y^+_{ik}(t)\} \leqslant 2ND.
\end{aligned} \tag{13}$$

For the lower bound $Y^+_{ik}(t)$, we have

$$\begin{aligned}
\Delta Y^+_{ik}(t) &= Y^+_{ik}(t+1) - \max\{Y_{ik}(t), 0\}\\
&= \min\{Y^+_{ik}(t+1) - Y_{ik}(t), Y^+_{ik}(t+1)\}\\
&\geqslant \min\{Y_{ik}(t+1) - Y_{ik}(t), Y^+_{ik}(t+1)\}\\
&= \min\{\Delta Y_{ik}(t), Y^+_{ik}(t+1)\} \geqslant -2ND.
\end{aligned} \tag{14}$$

Combining (12), (13) and (14) completes the proof. □

Since $Y^{+2}_{ik}(t+1) - Y^{+2}_{ik}(t)$ can be decomposed as

$$\begin{aligned}
&Y^{+2}_{ik}(t+1) - Y^{+2}_{ik}(t)\\
&\quad= \big(Y^+_{ik}(t) + \Delta Y^+_{ik}(t)\big)^2 - Y^{2+}_{ik}(t)\\
&\quad= 2\, Y^+_{ik}(t)\Delta Y^+_{ik}(t) + \big(\Delta Y^+_{ik}(t)\big)^2,
\end{aligned} \tag{15}$$

upper bounding $Y^+_{ik}(t)\Delta Y^+_{ik}(t)$ suffices and we have that

$$\begin{aligned}
&Y^+_{ik}(t)\Delta Y^+_{ik}(t)\\
&\leqslant Y^+_{ik}(t)\cdot\max\{\Delta Y_{ik}(t), -Y^+_{ik}(t)\}\\
&= Y^+_{ik}(t)\Delta Y_{ik}(t) + \max\{0, -Y^{+2}_{ik}(t) - Y^+_{ik}(t)\Delta Y_{ik}(t)\}\\
&\leqslant Y^+_{ik}(t)\Delta Y_{ik}(t) + \max\{0, -Y^{+2}_{ik}(t) + 2NDY^+_{ik}(t)\}\\
&= Y^+_{ik}(t)\Delta Y_{ik}(t) + \max\{0, -(Y^+_{ik}(t) - ND)^2 + N^2D^2\}\\
&\leqslant Y^+_{ik}(t)\Delta Y_{ik}(t) + N^2D^2,
\end{aligned} \tag{16}$$

where the first inequality comes from the fact that $Y^+_{ik}(t) \geqslant 0$ and $\Delta Y^+_{ik}(t) \leqslant \max\{\Delta Y_{ik}(t), -Y^+_{ik}(t)\}$. The second inequality holds because $Y^+_{ik}(t) \geqslant 0$ and $\Delta Y_{ik}(t) \geqslant -2ND$.

By inserting (16) into (15) and utilizing Lemma 8, we have that

$$\begin{aligned}
&Y^{+2}_{ik}(t+1) - Y^{+2}_{ik}(t)\\
&\leqslant 2\, Y^+_{ik}(t)\Delta Y_{ik}(t) + \big(\Delta Y^+_{ik}(t)\big)^2 + 2N^2D^2\\
&\leqslant 2\, Y^+_{ik}(t)\Delta Y_{ik}(t) + 6N^2D^2\\
&\leqslant 2\hat{Y}^+_{ik}(t)\Delta Y_{ik}(t) + 2(t - \tau_{ik}(t))\cdot 2ND\cdot 2ND + 6N^2D^2\\
&\leqslant 2\hat{Y}^+_{ik}(t)\Delta Y_{ik}(t) + \big(8L(t) + 6\big)N^2D^2,
\end{aligned} \tag{17}$$

which completes the proof.

## Appendix C
## Proof of Lemma 3

We first have

$$\begin{aligned}
&\sum_{i,k} Q_{ik}(T)\\
&\quad= \sum_{i\in\mathcal{O},k} Q_{ik}(T) + \sum_{i\in\mathcal{U},k} X_{ik}(T) + \sum_{i\in\mathcal{U},k} Y_{ik}(T)\\
&\quad\leqslant \sum_{i\in\mathcal{O},k} Q_{ik}(T) + \sum_{i\in\mathcal{U},k} X_{ik}(T) + \sum_{i\in\mathcal{U},k} Y^+_{ik}(T)\\
&\quad\leqslant \sqrt{KN + K\,|\mathcal{U}|}\\
&\qquad\cdot\sqrt{\sum_{i\in\mathcal{O},k} Q^2_{ik}(T) + \sum_{i\in\mathcal{U},k} X^2_{ik}(T) + \sum_{i\in\mathcal{U},k} Y^{+2}_{ik}(T)}\\
&\quad\leqslant \sqrt{2KN\Phi(T)},
\end{aligned} \tag{18}$$

where the second inequality utilizes Cauchy–Schwarz inequality.

By taking expectation on both sides of (18) and then applying Jensen's inequality, we have

$$\begin{aligned}
\mathbb{E}\left[\sum_{i,k} Q_{ik}(T)\right] &\leqslant \sqrt{2KN}\cdot\mathbb{E}\big[\sqrt{\Phi(T)}\big]\\
&\leqslant \sqrt{2KN\mathbb{E}\big[\Phi(T)\big]},
\end{aligned}$$

which completes the proof.

## Appendix D
## Proof of Lemma 4

We define $M \triangleq T \bmod H$ and there exists an integer $J$ such that $T = JH + M$. Then, we have the following decomposition for $i \in \mathcal{O}$ and $k \in \mathcal{K}$,

$$\begin{aligned}
&\sum_{t=0}^{T-1} Q^{\pi_T}_{ik}(t)\delta^* Q_{ik}(t)\\
&\quad= \sum_{j=0}^{J-1}\Bigg[Q^{\pi_T}_{ik}(jH)\sum_{t=jH}^{(j+1)H-1}\delta^* Q_{ik}(t)\\
&\qquad+ \sum_{t=jH}^{(j+1)H-1}\big(Q^{\pi_T}_{ik}(t) - Q^{\pi_T}_{ik}(jH)\big)\cdot\delta^* Q_{ik}(t)\Bigg]\\
&\qquad+ \sum_{t=JH}^{T-1} Q^{\pi_T}_{ik}(t)\delta^* Q_{ik}(t)
\end{aligned}$$

$$\leqslant \sum_{j=0}^{J-1}\Bigg[2NDT\sum_{t=jH}^{(j+1)H-1}\delta^* Q_{ik}(t) + \sum_{t=jH}^{(j+1)H-1} 2NDH\cdot 2ND\Bigg] + M\cdot 2NDT\cdot 2ND$$
$$\leqslant 2JNDT\sum_{t=0}^{T-1}\delta^* Q_{ik}(t) + 8N^2D^2HT$$
$$\leqslant \frac{2NDT^2}{H}\sum_{t=0}^{T-1}\delta^* Q_{ik}(t) + 8N^2D^2HT, \tag{19}$$

where inequalities hold by using (1), and the fact that $M \leqslant H$ and $J \leqslant T/H$.

Similarly, we show that for $i \in \mathcal{U}$,

$$\sum_{t=0}^{T-1} X_{ik}^{\pi_T}(t)\delta^* X_{ik}(t) \leqslant \frac{2NDT^2}{H}\sum_{t=0}^{T-1}\delta^* X_{ik}(t) + 8N^2D^2HT. \tag{20}$$

We then proceed to analyze $\sum_{i\in\mathcal{O},k}\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{U},k}\delta^* X_{ik}(t)$. Define the set of destinations (sinks) for the traffic of class $k$ as $\mathcal{D}_k$, we then have that

$$\sum_{i\in\mathcal{O},k}\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{U},k}\delta^* X_{ik}(t)$$
$$= \sum_{i\in\mathcal{O},k}\Bigg(a_{ik}(t) - \sum_{j\in\mathcal{N}} f_{ijk}^*(t) + \sum_{j\in\mathcal{O}} f_{jik}^*(t) + \sum_{j\in\mathcal{U}}\mu_{jik}(t)\Bigg) + \sum_{i\in\mathcal{U},k}\Bigg(\lambda_{ik}(t) - \sum_{j\in\mathcal{N}}\mu_{ijk}(t) + \sum_{j\in\mathcal{O}} f_{jik}^*(t) + \sum_{j\in\mathcal{U}}\mu_{jik}(t)\Bigg)$$
$$= \sum_{i\in\mathcal{O},k} a_{ik}(t) + \sum_{i\in\mathcal{U},k}\lambda_{ik}(t) - \sum_{i\in\mathcal{O},k} f_{id_kk}^*(t) - \sum_{i\in\mathcal{U},k}\mu_{id_kk}(t),$$

with which we have that

$$\sum_{t=0}^{T-1}\Bigg(\sum_{i\in\mathcal{O},k}\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{U},k}\delta^* X_{ik}(t)\Bigg) = \sum_{t=0}^{T-1}\Bigg(\sum_{i\in\mathcal{O},k} a_{ik}(t) + \sum_{i\in\mathcal{U},k}\lambda_{ik}(t) - \sum_{i\in\mathcal{O},k} f_{id_kk}^*(t) - \sum_{i\in\mathcal{U},k}\mu_{id_kk}(t)\Bigg). \tag{21}$$

On the other hand, we have that

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^*(T) = \sum_{t=0}^{T-1}\Bigg(\sum_{i\in\mathcal{N},k} a_{ik}(t) - \sum_{i\in\mathcal{O},k}\tilde{f}_{id_kk}^*(t) - \sum_{i\in\mathcal{U},k}\tilde{\mu}_{id_kk}(t)\Bigg) + \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(0). \tag{22}$$

Combining (21) and (22), and using the fact that $Q_{ik}(0) \geqslant 0$, $\tilde{f}_{ijk}^*(t) \leqslant f_{ijk}^*(t)$, $\tilde{\mu}_{ijk}^*(t) \leqslant \mu_{ijk}^*(t)$ hold for each $i, j, k, t$, we have

$$\mathbb{E}\Bigg[\sum_{t=0}^{T-1}\Bigg(\sum_{i\in\mathcal{O},k}\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{U},k}\delta^* X_{ik}(t)\Bigg)\Bigg] \leqslant Q_T^*. \tag{23}$$

Summing up (19) and (20) over all nodes and traffic classes, and then plugging in (23) complete the proof.

## Appendix E
## Proof of Lemma 5

We first discuss the case when $Q_{ik}(t) < ND$. Since $Y_{ik}^+(t)$ is non-negative and $X_{ik}(t) \geqslant 0$, we have $0 \leqslant Y_{ik}^+(t) < ND$, which gives us that

$$\mathbb{E}\Big[\hat{Y}_{ik}^{\pi_T+}(t)\Delta^* Y_{ik}(t) \mid Q_{ik}(t) < ND\Big] \leqslant \Big(Y_{ik}^+(t) + (t - \tau_{ik}(t))\cdot 2ND\Big)\cdot 2ND \leqslant \big(4L(t) + 2\big)\cdot N^2D^2, \tag{24}$$

where the first inequality utilizes Lemma 8.

When $Q_{ik}(t) \geqslant ND$, $Q_{ik}(t) + a_{ik}(t) - \sum_{j\in\mathcal{N}}\mu_{ijk}(t) \geqslant 0$. Therefore, $\tilde{\mu}_{ijk}(t) = \mu_{ijk}(t)$ and $\Delta^* Y_{ik}(t)$ can be upper bounded as

$$\Delta^* Y_{ik}(t) = Q_{ik}^*(t+1) - Q_{ik}^*(t) - (X_{ik}^*(t+1) - X_{ik}^*(t)) = a_{ik}(t) - \lambda_{ik}(t) - \sum_{j\in\mathcal{N}}\mu_{ijk}(t) + \sum_{j\in\mathcal{U}}\tilde{\mu}_{ijk}(t) + \sum_{j\in\mathcal{N}}\tilde{\mu}_{ijk}(t) - \sum_{j\in\mathcal{U}}\mu_{ijk}(t)$$
$$\leqslant a_{ik}(t) - \lambda_{ik}(t) - \sum_{j\in\mathcal{N}}\mu_{ijk}(t) + \sum_{j\in\mathcal{U}}\mu_{ijk}(t) + \sum_{j\in\mathcal{N}}\mu_{ijk}(t) - \sum_{j\in\mathcal{U}}\mu_{ijk}(t) = a_{ik}(t) - \lambda_{ik}(t). \tag{25}$$

Moreover, since $\hat{Y}_{ik}^{\pi_T+}(t)$ depends on arrivals and actions up to time $t-1$, but is independent of the arrivals and actions at time $t$, we have

$$\mathbb{E}\Big[\hat{Y}_{ik}^{\pi_T+}(t)\Delta^* Y_{ik}(t) \mid Q_{ik}(t) \geqslant ND\Big] \leqslant \mathbb{E}\Big[\hat{Y}_{ik}^{\pi_T+}(t)\cdot\big(a_{ik}(t) - \lambda_{ik}(t)\big) \mid Q_{ik}(t) \geqslant ND\Big] = \mathbb{E}\Big[\hat{Y}_{ik}^{\pi_T+}(t) \mid Q_{ik}(t) \geqslant ND\Big]\cdot\Big(\mathbb{E}\big[a_{ik}(t)\big] - \lambda_{ik}\Big) = 0. \tag{26}$$

Combining (24) and (26), we have

$$\mathbb{E}\Big[\hat{Y}_{ik}^{\pi_T+}(t)\Delta^* Y_{ik}(t)\Big] \leqslant \big(4L(t) + 2\big)\cdot N^2D^2,$$

which completes the proof.

## APPENDIX F
## PROOF OF LEMMA 6

We have the following upper bound

$$\begin{aligned}
&Y_{ik}^{+2}(t+1) - Y_{ik}^{+2}(t) \\
&\quad\leqslant 2\hat{Y}_{ik}^{+}(t)\Delta Y_{ik}(t) + (8L(t)+6)\,N^2D^2 \\
&\quad= 2\tilde{Y}_{ik}^{+}(t)\Delta Y_{ik}(t) + (8L(t)+6)\,N^2D^2 \\
&\qquad+ 2\left(\hat{Y}_{ik}^{+}(t) - \tilde{Y}_{ik}^{+}(t)\right)\Delta Y_{ik}(t) \\
&\quad\leqslant 2\tilde{Y}_{ik}^{+}(t)\Delta Y_{ik}(t) + (8L(t)+6)\,N^2D^2 \\
&\qquad+ 4ND\left|\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}^{+}(t)\right|, \qquad (27)
\end{aligned}$$

where the first inequality comes from (17) and the last inequality utilizes (1).

To analyze $\left|\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}^{+}(t)\right|$, we first have

$$\begin{aligned}
\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}^{+}(t) &= \max\left\{\tilde{Y}_{ik}(t), 0\right\} - \hat{Y}_{ik}^{+}(t) \\
&= \max\left\{\tilde{Y}_{ik}(t) - \hat{Y}_{ik}^{+}(t), -\hat{Y}_{ik}^{+}(t)\right\} \\
&\leqslant \max\left\{\tilde{Y}_{ik}(t) - \hat{Y}_{ik}(t), -\hat{Y}_{ik}^{+}(t)\right\} \\
&\leqslant \max\left\{\epsilon_{ik}(\tau_{ik}(t)), 0\right\}.
\end{aligned}$$

On the other direction, we have a lower bound as follows

$$\begin{aligned}
\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}^{+}(t) &= \tilde{Y}_{ik}^{+}(t) - \max\left\{\hat{Y}_{ik}(t), 0\right\} \\
&= \min\left\{\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}(t), \tilde{Y}_{ik}^{+}(t)\right\} \\
&\geqslant \min\left\{\tilde{Y}_{ik}(t) - \hat{Y}_{ik}(t), \tilde{Y}_{ik}^{+}(t)\right\} \\
&\geqslant \min\left\{\epsilon_{ik}(\tau_{ik}(t)), 0\right\}.
\end{aligned}$$

Therefore, we have an upper bound $\left|\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}^{+}(t)\right| \leqslant |\epsilon_{ik}(\tau_{ik}(t))|$. By inserting it into (27), we complete the proof.

## APPENDIX G
## PROOF OF LEMMA 7

We first discuss the case when $Q_{ik}(t) < ND$. Since now $0 \leqslant Y_{ik}^{+}(t) < ND$ and $\left|\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}^{+}(t)\right| \leqslant |\epsilon_{ik}(t)|$ (as shown in Lemma 6), we have

$$\begin{aligned}
&\mathbb{E}\left[\tilde{Y}_{ik}^{\pi_T+}(t)\Delta^* Y_{ik}(t) \mid Q_{ik}(t) < ND\right] \\
&\quad\leqslant \left(Y_{ik}^{+}(t) + (t - \tau_{ik}(t))\cdot 2ND + \epsilon_{ik}(\tau_{ik}(t))\right)\cdot 2ND \\
&\quad\leqslant \left(4L(t)+2\right)\cdot N^2D^2 + 2ND\,|\epsilon_{ik}(\tau_{ik}(t))|\,. \qquad (28)
\end{aligned}$$

When $Q_{ik}(t) \geqslant ND$, the analysis remains identical as the proof of Lemma 5 and we have

$$\mathbb{E}\left[\tilde{Y}_{ik}^{\pi_T+}(t)\Delta^* Y_{ik}(t) \mid Q_{ik}(t) \geqslant ND\right] \leqslant 0. \qquad (29)$$

By combining (28) and (29), we complete the proof.

## APPENDIX H
## PROOF OF THEOREM 3

Node 2 is set to have a fixed underlay policy $\pi_u^* : \mu_{2d} = C_{2d}$. We reinforce to assume that we can observe $Q_2(t)$ for each time slot, with the observation defined as $\hat{Q}_2(t)$. The observation noise $\epsilon_2(t) = 8t$ and we have $\hat{Q}_2(t) = [Q_2(t) - \epsilon_2(t)]^+$.

We first characterize the stability region. It is easy to show that the stability region of the system is

$$\Pi_0 = \{(\lambda_1, \lambda_2) : \lambda_1 + \lambda_2 < 6, \lambda_2 < 4\}.$$

We fix an arbitrary throughput optimal stochastic policy $\pi_c^0 : (Q_1, \hat{Q}_2) \to (f_{1d}, f_{12})$. To simplify the expression, we define the action taken under $(Q_1, \hat{Q}_2)$ to be $f_{1d}(Q_1, \hat{Q}_2)$ and $f_{12}(Q_1, \hat{Q}_2)$. We now analyze three different cases.

*Case 1:* Let

$$a_1(t) = \begin{cases} 4, & w.p.\ 3/4 \\ 0, & w.p.\ 1/4, \end{cases} \quad a_2(t) \equiv 0.$$

It is easy to verify that we are inside the stability region. Since $\lambda_1 - C_{1d} = 1$, we must have

$$\mu_{12} = \lim_{T\to\infty} \frac{\mathbb{E}_{\pi_c^0}\left[\sum_{t=1}^{\pi_T} \tilde{f}_{12}\left(Q_1(t), \hat{Q}_2(t)\right)\right]}{T} \geqslant 1.$$

Since the every time slot there are at most 4 external packets into the system, we have $\hat{Q}_2(t) \equiv 0$.

Denote $p(Q_1, Q_2)$ as the stationary probability of $(Q_1, Q_2)$ and $\mathcal{Q} \triangleq \{Q_1 : p^{\pi_c^0}(Q_1, 0) > 0\}$, we then have

$$\mu_{12} = \sum_{Q_1\in\mathcal{Q}} p(Q_1, 0)\cdot \mathbb{E}_{\pi_c^0}\left[f_{12}(Q_1, 0)\right] \geqslant 1. \qquad (30)$$

*Case 2:* Let

$$a_1(t) = \begin{cases} 4, & w.p.\ 1/2 \\ 0, & w.p.\ 1/2, \end{cases} \quad a_2(t) \equiv 0.$$

It is easy to verify that we are still inside the stability region and we still have $\hat{Q}_2(t) \equiv 0$.

Denote $p'(Q_1, Q_2)$ as the stationary probability in this case. Since the set of possible values of $a_1(t)$, $f_{12}(t)$ and $f_{1d}(t)$ are the same as case 1, and $\mathcal{Q}$ denotes the set of reachable $Q_1$'s, we also have $p'(Q_1, 0) > 0$ for $Q_1 \in \mathcal{Q}$. Also, (30) ensures that there exists $Q_1^*$ such that $\mathbb{E}_{\pi_c^0}\left[f_{12}(Q_1, 0)\right] > 0$, we thus have

$$\begin{aligned}
\mu'_{12} &= \sum_{Q_1\in\mathcal{Q}} p'(Q_1, 0)\cdot \mathbb{E}_{\pi_c^0}\left[f_{12}(Q_1, 0)\right] \\
&\geqslant\ p'(Q_1^*, 0)\cdot \mathbb{E}_{\pi_c^0}\left[f_{12}(Q_1^*, 0)\right] > 0.
\end{aligned}$$

*Case 3:* We define the value of $\mu'_{1\to 2}$ as $\delta$ and let

$$a_1(t) = \begin{cases} 4, & w.p.\ 1/2 \\ 0, & w.p.\ 1/2, \end{cases}$$

$$a_2(t) = \begin{cases} 4, & w.p.\ 1 - \delta/8 \\ 0, & w.p.\ \delta/8. \end{cases}$$

It is easy to verify that we are still inside the stability region.

Since the every time slot there are at most 8 external packets into the system, we still have $\hat{Q}_2(t) \equiv 0$. Also consider the $a_1$ has the same pattern as in case 2, for us (the overlay controller), the system now "looks" exactly the same as case 2. We therefore have that $\mu''_{12}$, the rate of packets transmitted from node 1 to node 2, equals to $\mu'_{12}$.

Now, the total input rate to node 2 amounts to $\lambda_2 + \mu''_{12} = 4 - \delta/2 + \delta > C_{2d}$ and $Q_2$ is instable.

Case 3 violates the definition of throughput optimal stochastic policy. Since $\pi_c^0$ is selected arbitrarily, we complete the proof.

## REFERENCES


[1] R. K. Sitaraman, M. Kasbekar, W. Lichtenstein, and M. Jain, "Overlay networks: An Akamai perspective," *Adv. Content Del., Streaming, Cloud Services*, vol. 51, no. 4, pp. 305–328, 2014.

[2] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[3] G. S. Paschos and E. Modiano, "Throughput optimal routing in overlay networks," in *Proc. 52nd Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2014, pp. 401–408.

[4] N. M. Jones, G. S. Paschos, B. Shrader, and E. Modiano, "An overlay architecture for throughput optimal multipath routing," *IEEE/ACM Trans. Netw.*, vol. 25, no. 5, pp. 2615–2628, Aug. 2017.

[5] A. Rai, R. Singh, and E. Modiano, "A distributed algorithm for throughput optimal routing in overlay networks," in *Proc. IFIP Netw. Conf. (IFIP Netw.)*, May 2019, pp. 1–9.

[6] Q. Liang and E. Modiano, "Optimal network control in partially-controllable networks," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2019, pp. 397–405.

[7] B. Liu, Q. Xie, and E. Modiano, "Reinforcement learning for optimal control of queueing systems," in *Proc. 57th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2019, pp. 663–670.

[8] E. J. Sondik, "The optimal control of partially observable Markov processes over the infinite horizon: Discounted costs," *Ope. Res.*, vol. 26, no. 2, pp. 282–304, 1978.

[9] H.-T. Cheng, "Algorithms for partially observable Markov decision processes," Ph.D. dissertation, Dept. Commerce Bus. Admin., Univ. British Columbia, Vancouver, BC, Canada, 1988.

[10] N. L. Zhang and W. Liu, "Planning in stochastic domains: Problem characteristics and approximation," Hong Kong Univ. Sci. Technol., Hong Kong, Tech. Rep. HKUST-CS96-31, 1996.

[11] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artif. Intell.*, vol. 101, nos. 1–2, pp. 99–134, 1998.

[12] A. R. Cassandra, M. L. Littman, and N. L. Zhang, "Incremental pruning: A simple, fast, exact method for partially observable Markov decision processes," 2013, *arXiv:1302.1525*.

[13] J. Baxter and P. L. Bartlett, "Infinite-horizon policy-gradient estimation," *J. Artif. Intell. Res.*, vol. 15, pp. 319–350, Jul./Dec. 2001.

[14] R. Urgaonkar and M. J. Neely, "Opportunistic cooperation in cognitive femtocell networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 607–616, Apr. 2012.

[15] S. Kompella, G. Nguyen, C. Kam, J. E. Wieselthier, and A. Ephremides, "Cooperation in cognitive underlay networks: Stable throughput tradeoffs," *IEEE/ACM Trans. Netw.*, vol. 22, no. 6, pp. 1756–1768, Dec. 2014.

[16] T. Stahlbuhk, B. Shrader, and E. Modiano, "Throughput maximization in uncooperative spectrum sharing networks," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2016, pp. 1242–1246.

[17] R. Gallager, "A minimum delay routing algorithm using distributed computation," *IEEE Trans. Commun.*, vol. COM-25, no. 1, pp. 73–85, Jan. 1977.

[18] D. Bertsekas, E. Gafni, and R. Gallager, "Second derivative algorithms for minimum delay distributed routing in networks," *IEEE Trans. Commun.*, vol. COM-32, no. 8, pp. 911–919, Aug. 1984.

[19] J. N. Tsitsiklis and D. P. Bertsekas, "Distributed asynchronous optimal routing in data networks," *IEEE Trans. Autom. Control*, vol. AC-31, no. 4, pp. 325–332, Apr. 1986.

[20] E. Modiano, D. Shah, and G. Zussman, "Maximizing throughput in wireless networks via gossiping," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 34, no. 1, pp. 27–38, 2006.

[21] C. Joo and N. B. Shroff, "Performance of random access scheduling schemes in multi-hop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 17, no. 5, pp. 1481–1493, Oct. 2009.

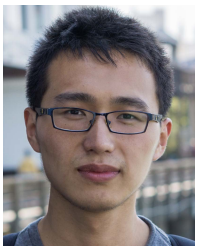
[22] X. Lin and S. B. Rasool, "Constant-time distributed scheduling policies for ad hoc wireless networks," *IEEE Trans. Autom. Control*, vol. 54, no. 2, pp. 231–242, Feb. 2009.

[23] M. Barbeau and E. Kranakis, *Principles of Ad Hoc Networking*. Hoboken, NJ, USA: Wiley, 2007.

[24] C. Hedrick et al., *Routing Information Protocol*, document TR RFC 1058, Rutgers Univ., Piscataway, NJ, USA, 1988.

[25] C. Hopps et al., *Analysis of an Equal-Cost Multi-Path Algorithm*, document TR RFC 2992, Nov. 2000.

[26] J. Moy, *OSPF Version 2*, document RFC2178, 1998.

**Bai Liu** received the B.E. degree (Hons.) from Tsinghua University, Beijing, China, in 2017, and the M.S. degree from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2019, where he is currently pursuing the Ph.D. degree with the Laboratory for Information and Decision Systems. His research interests include learning and control problems in networked systems, with application of reinforcement learning, stochastic optimization, and inference methods.

**Qingkai Liang** received the B.E. degree (Hons.) in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2013, and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2015 and 2018, respectively. He is currently the Co-Founder at Celer Network. His research focuses on various learning and control problems that arise in networked systems, especially on online learning algorithms in adversarial networks, which have been successfully applied in Raytheon BBN Technologies and Bell Laboratories.

**Eytan Modiano** (Fellow, IEEE) received the B.S. degree in electrical engineering and computer science from the University of Connecticut, Storrs, in 1986, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, MD, in 1989 and 1992, respectively.

He is currently the Richard C. Maclaurin Professor with the Department of Aeronautics and Astronautics and the Laboratory for Information and Decision Systems (LIDS), MIT. Prior to joining as the Faculty Member at MIT in 1999, he was a Naval Research Laboratory Fellow from 1987 to 1992, a National Research Council Post-Doctoral Fellow from 1992 to 1993, and a member of the Technical Staff at the MIT Lincoln Laboratory from 1993 to 1999. His research interests include modeling, analysis, and design of communication networks and protocols. In 2020, he received the Infocom Achievement Award for contributions to the analysis and design of cross-layer resource allocation algorithms for wireless, optical, and satellite networks. He is the co-recipient of the Infocom 2018 Best Paper Award, the MobiHoc 2018 Best Paper Award, the MobiHoc 2016 Best Paper Award, the Wiopt 2013 Best Paper Award, and the Sigmetrics 2006 Best Paper Award. He was the Editor-in-Chief for IEEE/ACM TRANSACTIONS ON NETWORKING (2017–2020) and served as an Associate Editor for IEEE TRANSACTIONS ON INFORMATION THEORY and IEEE/ACM TRANSACTIONS ON NETWORKING. He was the Technical Program Co-Chair for IEEE Wiopt 2006, IEEE Infocom 2007, ACM MobiHoc 2007, and DRCN 2015; and the General Co-Chair of Wiopt 2021. He had served on the IEEE Fellow Committee in 2014 and 2015. He is an Associate Fellow of the AIAA.