Optimal Control for Networks with Unobservable MaliciousNodes

Optimal Control for Networks with Unobservable Malicious Nodes

BAI LIU and EYTAN MODIANO, Massachusetts Institute of Technology

Classic network optimization theory focuses on network models with stochastic dynamics and all nodes being observable and controllable. However, modern network systems often offer limited access and suffer from adversarial attacks. In this paper, we focus on networks with unobservable malicious nodes, where the network dynamics, such as external arrivals and control actions of malicious nodes can be adversarial. We first extend the existing adversarial network models by introducing a new maliciousness metric that constrains the dynamics of the adversary, and characterize the stability region of a network under adversarial dynamics. We then propose an algorithm that only operates on the accessible nodes and does not require direct observations of the malicious nodes, and show that our algorithm is stabilizing as long as the network dynamics are within the stability region. Finally, we show that our algorithm stabilizes the network even if the estimates of the network state are erroneous, and characterize the necessary and sufficient conditions for networks with unobservable malicious nodes to be stabilizable when subjected to estimation errors.

ACM Reference Format:
Bai Liu and Eytan Modiano. 2021. Optimal Control for Networks with Unobservable Malicious Nodes. 1, 1 (September 2021), 28 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

For decades, control theory for network systems has been a central topic in the field of communication networks. Classic control algorithms like MaxWeight [26] and Drift-plus-Penalty [22] have been studied thoroughly in a variety of contexts. These algorithms usually possess rigorous theoretical performance guarantees when applied to a network in which the controller can observe the network state (e.g., queue backlogs), all nodes cooperatively execute commands given by the controller, and the network dynamics are stochastic and time-invariant (e.g., the external arrivals to the network obey a stationary stochastic process).

With the rapid development of information technology, modern network systems are becoming increasingly complex, which makes the aforementioned framework unrealistic. Moreover, networks are increasingly vulnerable to attacks such as Distributed Denial-of-Service (DDoS) attack. Even worse, some of the nodes may be malicious and attempt to destabilize the network. However, existing network control algorithms either require full observability and/or controllability for all nodes [1–3, 6, 17, 19, 20, 27], or the network dynamics to be time-invariant and stochastic [13, 18, 23, 24]. In this paper, we aim to develop a new algorithm that can stabilize networks with unobservable and uncontrollable nodes under adversarial dynamics (i.e., external arrivals and actions of malicious nodes).

We consider a network where a subset of the nodes are controlled by an adversary that can observe the actions of the network controller and plan its dynamics accordingly to maximize

Authors' address: Bai Liu, email:bailiu@mit.edu; Eytan Modiano, email:modiano@mit.edu, Massachusetts Institute of Technology, Cambridge, MA 02139.

XXXX-XXXX/2021/9-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

disruption to the network. Meanwhile, the network controller may not be able to observe the state of the malicious nodes and can only operate (i.e., control) on the accessible nodes. A concrete example is a Denial-of-Service (DDoS) attack, where the attacker hijacks and takes control of multiple machines in the network by planting Trojans or scanning for security holes [21]. The controlled machines then become malicious nodes, that send a large amount of traffic to other nodes to block the bandwidth under the attacker's commands [10]. Another example of malicious attack is a Structured Query Language (SQL) injection attack which, by injecting and executing malicious commands, can shut down servers in a data center and thus cause congestion [11]. In this work, we propose an algorithm named MWUM (**M**ax**W**eight for Networks with **U**nobservable **M**alicious Nodes) to stabilize networks in such adversarial environments. To the best of our knowledge, MWUM is the first control algorithm to stabilize a network with unobservable malicious nodes.

The major technical challenges addressed in this work are two-fold: 1) unobservable and uncontrollable nodes and 2) adversarial dynamics (i.e., external arrivals and actions of malicious nodes). In the following we briefly discuss prior works pertaining to the above challenges.

Control algorithms for networks with unobservable and/or uncontrollable nodes have been studied in the context of overlay-underlay networks. In an overlay-underlay network, only the overlay nodes can provide instantaneous state information and be controlled, while the underlay nodes are modeled as "black boxes" with limited observability and controllability. The authors of [23] applied a router-forwarder model and proposed the Threshold-based Backpressure (BP-T) algorithm to achieve throughput-optimality. The work in [13] further extended the model in [23] and proposed the Overlay Backpressure (OBP) algorithm. In [24], the authors proposed the Optimal Overlay Routing Policy (OORP) that is applicable to general network topologies. However, OORP requires instantaneous underlay information and lacks strict theoretical performance guarantees. The work in [18] proposed the Tracking-MaxWeight (TMW) and Truncated Upper Confidence Reinforcement Learning (TUCRL) algorithms, which are capable of stabilizing overlay-underlay networks with general topologies, but still require instantaneous observation of underlay queue backlogs. Existing overlay-underlay control algorithms that can be applied to general topologies either lack theoretical performance guarantees or require instantaneous underlay information.

An alternative is to model the network as a Partially Observable Markov Decision Process (POMDP). POMDPs assume the system states to be unobservable and seek to minimize the long-term cost only using indirect information. POMDPs are a popular topic in the machine learning community, yet most works focus on heuristic algorithms and cannot give theoretical performance guarantees. Theoretical studies on POMDPs [5, 8, 9, 14, 25, 29] attempt to solve POMDPs using value iteration or policy search, yet can only be applied to small-scale networks.

There has also been a significant amount of work on control algorithms for networks with adversarial dynamics. A simple version of control problems with adversarial dynamics is the adversarial multi-armed bandit problem. The work in [4] first systematically introduced the concept of adversarial bandits and showed that the achieved reward can be optimal even if the system dynamics are adversarial. A comprehensive analysis for adversarial multi-armed bandit problems and their extensions can be found in [7]. However, multi-armed bandit problems are stateless and cannot capture the queueing dynamics of data networks.

The earliest studies on networks with adversarial dynamics can be traced back to [6], which first proposed the Adversarial Queueing Theory (AQT) framework. Later, the authors of [1] introduced the more general Leaky Bucket (LB) model. Both the AQT and the LB framework only allow the external arrivals to be adversarial while the arrival process is required to satisfy the "W constraint" that restricts the volume of external arrivals during a certain time window.

In [2, 3], the authors considered a single-hop setting of wireless communication between a base station and multiple mobile users. Beyond the adversarial external arrivals, the model also allows

the conditions of communication channels to be adversarial. The results were later extended to multi-hop settings in [20]. However, all these works require full observability of all nodes, and the nodes cannot take adversarial actions.

In [17, 19], the authors extended the network model to general topologies and allowed adversarial actions. They proposed a more relaxed constraint named "V_T constraint", which only requires the peak queue backlog under the optimal policy to be constrained to V_T. The authors showed that as long as V_T is sublinear in the time horizon, the queue backlogs of the network can be stabilized. However, both works [17, 19] require instantaneous information of the underlay dynamics, which may be unrealistic in adversarial network settings.

As far as we know, existing related network control algorithms either only consider stochastic (i.e., non-adversarial) dynamics, or require full observability and can only operate under the relatively restrictive W and V_T constraints. In this paper, we consider networks with both unobservable nodes and adversarial dynamics. Our main contributions are summarized below.

We first propose a new maliciousness metric for the adversary called the Q_T constraint to characterize the adversarial dynamics. The Q_T constraint bounds the queue size at the end of the time horizon (T). We quantitatively analyze the relationship between the Q_T constraint and the existing W constraint [1, 6] and V_T constraint [17, 19], and show that the Q_T constraint is the least restrictive constraint among the three. Thus, the Q_T constraint leads to a more powerful adversary and requires new analysis methods.

Next, we propose MWUM, which uses estimates of the state of the malicious nodes instead of direct observations and only needs to control the accessible nodes. We rigorously show that for networks with Q_T-constrained dynamics, as long as Q_T grows sublinearly in the time horizon T, MWUM can stabilize all queues (including the queues of the malicious nodes). We then use Q_T to characterize the stability region for networks with unobservable malicious nodes and show that MWUM is throughput-optimal. In contrast, existing related network control algorithms either require full observability and/or controllability for all nodes, or the network dynamics to be time-invariant and stochastic. Thus, to the best of our knowledge, MWUM is the first throughput-optimal control algorithm for networks with unobservable malicious nodes.

Furthermore, by applying our new analysis techniques, we strengthen the existing performance guarantees in previous works under the W and V_T constraints from rate stability at the end of the time horizon to sublinear queue backlog during the entire time horizon.

Finally, we consider the impact of estimation errors and show that as long as the estimation errors grow sublinearly in time, MWUM stabilizes the network. We also show that when the estimation errors grow linearly (or superlinearly) in time, there exists a network that is not stabilizable by any state-based algorithm and thus MWUM is "maximally robust" to estimation errors. We further characterize the necessary and sufficient conditions for networks with unobservable malicious nodes to be stabilizable when estimations are erroneous.

The rest of this paper is organized as follows. We introduce the network model and discuss the maliciousness metrics in detail in Section 2. We introduce MWUM in Section 3. In Section 4, we show the stability results under the Q_T constraint, discuss the performance under the W and V_T constraints and characterize the stability region. We consider estimation errors in Section 5, where we characterize the necessary and sufficient conditions for a network to be stabilizable with erroneous estimations. Section 6 presents simulation results and Section 7 concludes the paper.

2 MODEL

We consider a multi-hop network with N nodes and denote the set of nodes by $\mathcal{N}$. The nodes are classified into two types: the set of accessible nodes $\mathcal{A}$ and the set of malicious nodes $\mathcal{M}$. The network has K classes of data and the data of class k is destined for sink d_k. The set of data classes

is denoted by $\mathcal{K}$. The link capacity between node i and j is C_{ij}. We assume that time is slotted and the time horizon is T.

At the beginning of time slot t, a node $i \in \mathcal{N}$ has $Q_{ik}(t)$ buffered packets of class k and receives $a_{ik}(t)$ external packets of class k. Since it is possible for the adversary to inject malicious packets (e.g., DDoS attack), $a_{ik}(t)$ can be non-stochastic and even malicious: the adversary first observes the history, including the past queue backlogs and transmissions, up to time $t-1$, and then decides on $a_{ik}(t)$ for each node.

For an accessible node $i \in \mathcal{A}$, we denote by $f_{ijk}(t)$ the number of packets of class k to be transmitted to a neighbor j as decided by the network controller. Since the packets available to be transmitted cannot exceed $Q_{ik}(t) + a_{ik}(t)$, the actual number of packets transmitted might be less than $f_{ijk}(t)$ and is denoted by $\tilde{f}_{ijk}(t)$. Note that the network controller is only capable of controlling the accessible nodes $\mathcal{A}$. The policy taken by the network controller can be characterized by a set of routing action sequences $\pi = \left\{f_{ijk}(t)\right\}_{i\in\mathcal{A},j\in\mathcal{N},k\in\mathcal{K},0\leqslant t\leqslant T-1}$. We further denote by $\prod$ the set of admissible π's (i.e., the set of $\left\{f_{ijk}(t)\right\}_{i\in\mathcal{A},j\in\mathcal{N},k\in\mathcal{K},0\leqslant t\leqslant T-1}$ with $0 \leqslant \sum_k f_{ijk}(t) \leqslant C_{ij}$).

For a malicious node $i \in \mathcal{M}$, the network controller cannot directly observe $Q_{ik}(t)$ or implement control policies. Note that the word "malicious" does not necessarily mean that the malicious nodes attemp to attack the network. Our setting allows the malicious nodes to be simply uncontrollable. We assume that by applying network inference methods (e.g., probing [12, 16]), the network controller can obtain estimates $\hat{Q}_{ik}(t)$ of queue backlog Q_{ik}, and that such estimates are only available sporadically. We denote by Γ_i the set of time slots when estimates are made for node i. In other words, for a malicious node $i \in \mathcal{M}$, the network controller only has an estimate $\hat{Q}_{ik}(t)$ of queue backlog $Q_{ik}(t)$ for $t \in \Gamma_i$. Note that the estimates do not have to be accurate. We show in Section 5 that as long as the estimation errors grow sublinearly in time, MWUM still stabilizes the network.

In addition to not being observable, malicious nodes are controlled by an adversary. Similar to the aforementioned adversarial external arrivals, the actions taken by the adversary can be a function of the history up to time $t-1$ (i.e., $\{a_{ik}(\tau)\}_{i\in\mathcal{N},k\in\mathcal{K},0\leqslant\tau\leqslant t-1}$, $\{f_{ijk}(\tau)\}_{i\in\mathcal{A},j\in\mathcal{N},k\in\mathcal{K},0\leqslant\tau\leqslant t-1}$). For instance, in DDoS attack, the adversary can hijack a server in the network, and attempt to destroy the stability by sending tremendous amount of requests to the most heavily loaded nodes. We denote by $\mu_{ijk}(t)$ the number of packets of class k to be transmitted to a neighbor j from a malicious node $i \in \mathcal{M}$ and the actual number of packets transmitted by $\tilde{\mu}_{ijk}(t)$.

Our goal is to determine a policy $\pi \in \prod$ that stabilizes the queues for all nodes $\mathcal{N}$ only using sporadic (and possibly erroneous) estimates of the state (queue backlogs) of the malicious nodes $\mathcal{M}$.

Mathematically, the queue backlogs evolve according to the following rule (we use the operator $[x]^+ \triangleq \max\{x, 0\}$)

$$Q_{ik}(t+1) = \begin{cases} \left[Q_{ik}(t) + a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k} f_{ijk}(t)\right]^+ + \sum_{j\in\mathcal{A}} \tilde{f}_{jik}(t) + \sum_{j\in\mathcal{M}} \tilde{\mu}_{jik}(t), & i \in \mathcal{A} \\ \left[Q_{ik}(t) + a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k} \mu_{ijk}(t)\right]^+ + \sum_{j\in\mathcal{A}} \tilde{f}_{jik}(t) + \sum_{j\in\mathcal{M}} \tilde{\mu}_{jik}(t), & i \in \mathcal{M} \end{cases}.$$

We further assume the system dynamics to be bounded, i.e.,

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

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

2.1 Asymptotic Notations

We apply the Bachmann–Landau notations to compare the limiting behavior between functions. Given two functions $f(n)$ and $g(n)$, their asymptotic relationships are listed in Table 1.

Table 1. Asymptotic Notations

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

2.2 Performance Metric

We focus on the rate stability of the queue backlogs for all nodes $\mathcal{N}$, which is defined as follows.

Definition 1. *A network is rate stable if*

$$\lim_{T\to\infty} \frac{\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(T)}{T} = 0.$$

By Definition 1, rate stability implies that when $t \to \infty$, the average arrival rate is no greater than the average service rate. Typically, in order to show rate stability, one needs to upper bound the total queue backlog $\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(T)$ by a sublinear factor of T (i.e., $\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(T) = o(T)$).

2.3 Maliciousness Metrics

To characterize the power of the adversary, we use the concept of maliciousness metrics. In our setting, the external arrivals and actions taken by the malicious nodes are adversarial. Therefore, a meaningful specification of a maliciousness metric places constraints on the sequence of possible network events $\big\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\big\}_{0\leqslant t\leqslant T-1}$.

First proposed in [6], the W constraint is the earliest maliciousness metric used in the study of adversarial network control. The W constraint places restrictions on network events for windows of length W time-slots, as defined below.

Definition 2. *A network event sequence $\big\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\big\}_{0\leqslant t\leqslant T-1}$ is W-constrained if there exists $\pi_W \in \prod$ under which the inequalities*

$$\begin{cases} \sum_{\tau=t}^{t+W-1} \Big(a_{ik}(\tau) - \sum_{j\in\mathcal{N}\cup d_k} \tilde{f}_{ijk}^{W}(\tau) + \sum_{j\in\mathcal{A}} \tilde{f}_{jik}^{W}(\tau) + \sum_{j\in\mathcal{M}} \tilde{\mu}_{jik}(\tau)\Big) \leqslant 0, & \forall i \in \mathcal{A}, k \in \mathcal{K} \\ \sum_{\tau=t}^{t+W-1} \Big(a_{ik}(\tau) - \sum_{j\in\mathcal{N}\cup d_k} \tilde{\mu}_{ijk}(\tau) + \sum_{j\in\mathcal{A}} \tilde{f}_{jik}^{W}(\tau) + \sum_{j\in\mathcal{M}} \tilde{\mu}_{jik}(\tau)\Big) \leqslant 0, & \forall i \in \mathcal{M}, k \in \mathcal{K} \end{cases}$$

are satisfied for $t = 0, W, 2W, \cdots$.

Definition 2 requires the existence of a policy π_W, under which there are at least as many served packets as the arrived packets for each node and each time window of size W. However, the W constraint is relatively restrictive, and to overcome this, the authors in [17] proposed the V_T constraint, defined as follows.

Definition 3. *A network event sequence $\big\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\big\}_{0\leqslant t\leqslant T-1}$ is V_T-constrained if under this network event sequence, the following holds*

$$\min_{\pi\in\prod} \max_{t\leqslant T} \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^{\pi}(t) \leqslant V_T.$$

The minimal value of V_T is defined to be V_T^*.

Definition 3 only requires the existence of a policy that upper bounds the peak queue backlog under the given network event sequence. However, in order to achieve rate stability, we are only concerned with the queue backlog at the end of the time horizon, i.e., $\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(T)$. Therefore, we propose a more relaxed maliciousness metric called the Q_T constraint.

Definition 4. *A network event sequence $\{\boldsymbol{a}(t),\boldsymbol{\mu}(t)\}_{0\leqslant t\leqslant T-1}$ is Q_T-constrained if under this network event sequence, the following holds*

$$\min_{\pi\in\prod}\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^{\pi}(T)\leqslant Q_T.$$

The minimal value of Q_T is defined to be Q_T^*.

Compared with the V_T constraint, the Q_T constraint only requires the queue backlog at the end of the time horizon to be upper bounded, allowing the queue backlog to exceed the bound prior to T. If there does not exist a Q_T constraint sublinear in T, no policy could stabilize the network event sequence. Note that every network event sequence has corresponding V_T and Q_T constraints. However, the scales of V_T and Q_T may vary and reflect different "maliciousness-levels" of the network event sequence.

It is necessary to clarify that in Definition 2, 3 and 4, the network event sequences are predetermined and are not affected by the policies. For instance, suppose we implement a policy π_0, and the system generates a network event sequence S_0. For the given network event sequence S_0, there exists a policy π^* under which the Q_T constraint holds. Of course, if we actually apply π^* to the system, the generated network event sequence S^* can be completely different from S_0. However, with our novel Lyapunov drift analysis technique, **we do not require π^* to be actually implemented to the system**.

We then define the maliciousness metrics for network dynamics as follows.

Definition 5. *A network is said to have W/V_T/Q_T-constrained dynamics if all generated network event sequences $\{\boldsymbol{a}(t),\boldsymbol{\mu}(t)\}_{0\leqslant t\leqslant T-1}$ are W/V_T/Q_T-constrained, respectively.*

The maliciousness metrics discussed above are closely related to each other, described in the following theorem (see Appendix A for the proof).

Theorem 1. *For a given network event sequence $\{\boldsymbol{a}(t),\boldsymbol{\mu}(t)\}_{0\leqslant t\leqslant T-1}$, we have*

$$Q_T^*\leqslant V_T^*\leqslant\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(0)+NKDW.$$

Theorem 1 implies that $W=\Omega(V_T^*)$ and $V_T^*=\Omega(Q_T^*)$. Therefore, the V_T constraint is less restrictive than the W constraint, since $W=o(T)$ guarantees $V_T^*=o(T)$ but not vice versa. Similarly, we have that the Q_T constraint is even less restrictive than the V_T constraint. We use the toy example depicted in Figure 1 to further illustrate the maliciousness metrics. The system consists of an accessible node 1 and link $1\rightarrow d$, with capacity $C_{1d}=2$. During each time slot, node 1 receives one external packet, and then tries to serve $f_{1d}(t)=2$ packets to destination d (note that the actual number of served packets, $\tilde{f}_{1d}(t)$, is smaller than $f_{1d}(t)$ if the queue backlog of node 1 is smaller than 2). The system is attacked and receives another malicious injection of $a_1'(t)$ packets at time t. Different distributions of $a_1'(t)$ result in different maliciousness metrics, as discussed next.

Consider

$$a_1'(t)=\begin{cases}2, & kT/10\leqslant t<kT/10+T/20\\ 0, & kT/10+T/20\leqslant t<(k+1)T/10\end{cases},$$

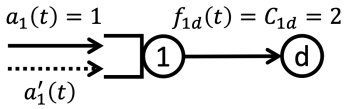


Fig. 1. A toy system to illustrate the relationships among maliciousness metrics.

where $k = 0, 1, \cdots, 9$. It is easy to verify that during each interval $kT/10 \leqslant t < (k+1)T/10$, the total arrived packets equals to the total served packets. Thus, by Definition 2, the network is W-constrained, with $W = T/10$. The peak queue backlog is $Q_1(kT/10 + T/20) = T/20$, which shows that the network is V_T-constrained with $V_T = T/20$. Finally, since all the packets are served by T, the network is Q_T-constrained with $Q_T = 0$.

Consider another malicious injection distribution

$$a_1'(t) = \begin{cases} 2, & t \leqslant T/2 + \sqrt{T}/2 \\ 0, & t > T/2 + \sqrt{T}/2 \end{cases}.$$

Since the malicious injections are not periodic, it is straightforward to verify that the network is not W-constrained. The peak queue backlog is $Q_1(T/2 + \sqrt{T}/2) = T/2 + \sqrt{T}/2$, and the terminal queue backlog is $Q_1(T) = \sqrt{T}$. Therefore, the network has $V_T = T/2 + \sqrt{T}/2 = \Omega(T)$, which dominates $Q_T = \sqrt{T} = o(T)$.

The above examples show that different maliciousness metrics correspond to different adversarial dynamics. The W constraint requires the adversary to be relatively stationary, with attacks sharing similar patterns across different intervals. The V_T constraint does not restrict temporal patterns, but limits the burstiness of the attacks. The Q_T constraint is the most relaxed one, with no requirements on temporal patterns or burstiness, and the adversary can have arbitrary attack patterns. In some cases, $V_T = \Omega(T)$ and $Q_T = o(T)$ coexist, suggesting that our algorithm advances existing works that require V_T to be sublinear in T.

We emphasize that the Q_T constraint is a necessary and sufficient condition for adversarial networks to be stabilizable. Every adversarial network has a Q_T constraint. If $Q_T = \Omega(T)$, then there exists a network event sequence under which the queue backlog at T is at least linear in T. If the adversary is intelligent, it might insist on generating this network event sequence, regardless of our actions. In this sense, no policy can stabilize the network. On the other hand, in Theorem 2, we show that as long as $Q_T = o(T)$, the adversarial network can be stabilized by MWUM. More detailed discussion can be found in Section 4.3.

2.4 Variable Notations

For readers' convenience, we summarize the variable notations in Table 2.

3 OUR APPROACH

The major challenges that need to be addressed are three-fold: 1) the state information of the malicious nodes cannot be observed directly; 2) the external injections and the behaviors of malicious nodes can be adversarial; 3) the malicious nodes cannot be controlled by the network controller. The above limitations render classical algorithms such as MaxWeight [26] unusable, and traditional analytical techniques based on stochastic analysis and stationary assumptions ineffective.

Table 2. Variable Notations

N	The number of queues in the queueing network
C_{ij}	The link capacity between node i and j
d_k	The destination of the data of class k
T	The time horizon
$\mathcal{N}$, $\mathcal{A}$, $\mathcal{M}$	The set of all nodes, accessible nodes, malicious nodes
$\mathcal{K}$	The set of data types
Γ_i	The set of time slots when an estimation of Q_{ik} was made for node $i \in \mathcal{M}$
π	The routing action sequence on accessible nodes, i.e., our policy
$Q_{ik}^{\pi}(t)$	Under policy π, the queue backlog of class k at node $i \in \mathcal{N}$ at t
$\hat{Q}_{ik}^{\pi}(t)$	Under policy π, the estimated queue backlog of class k at node $i \in \mathcal{N}$ at $t \in \Gamma_i$
$a_{ik}^{\pi}(t)$	Under policy π, the number of external packets of class k arriving at node $i \in \mathcal{N}$ at t
$f_{ijk}^{\pi}(t)$, $\tilde{f}_{ijk}^{\pi}(t)$	Under policy π, the planned and actual number of packets of class k transmitted from node $i \in \mathcal{A}$ to $j \in \mathcal{N}$ at t
$\mu_{ijk}^{\pi}(t)$, $\tilde{\mu}_{ijk}^{\pi}(t)$	Under policy π, the planned and actual number of packets of class k transmitted from node $i \in \mathcal{M}$ to $j \in \mathcal{N}$ at t
$g_{ijk}^{\pi}(t)$, $\tilde{g}_{ijk}^{\pi}(t)$	In the imaginary network, under policy π, the planned and actual number of packets of class k transmitted from node $i \in \mathcal{M}$ to $j \in \mathcal{N}$ at t
W, V_T, Q_T	Maliciousness metrics defined in Definition 2, 3, 4, respectively
$X_{ik}^{\pi}(t)$	Under policy π, the virtual queue backlog of class k at node $i \in \mathcal{M}$ at t
$Y_{ik}^{\pi}(t)$	$Q_{ik}^{\pi}(t) - X_{ik}^{\pi}(t)$ for $i \in \mathcal{M}$
$\tau_i(t)$	The most recent time an estimate of node i was made for node $i \in \mathcal{M}$ at t
$L(t)$	The maximum delay of estimates at t, i.e., $\max_{i \in \mathcal{M}, k \in \mathcal{K}} t - \tau_i(t)$

While some of the aforementioned challenges have been addressed in the past in various contexts (e.g., delayed state information), no approach handles the combination of unobservability, uncontrollability, and adversarial dynamics together.

3.1 Overview

To tackle these challenges, we introduce the MWUM algorithm. The core idea behind our approach is to "track" the state of the malicious nodes as well as the adversarial dynamics, and then make decisions based on the tracked information.

We first construct an "imaginary" network that shares the same topology and external arrivals as the real network, except that in the imaginary network, all nodes are fully observable and controllable. We denote by $g_{ijk}(t)$ the number of packets of class k transmitted to neighbor j from a malicious node $i \in \mathcal{M}$ in the imaginary network (also upper bounded by D). For an accessible node $i \in \mathcal{A}$ in the imaginary network, we force its queue backlog Q_{ik} to always be the same as that of the real system, i.e., Q_{ik} is synchronized with the real system instead of being updated using the actions taken in the imaginary network. For a malicious node $i \in \mathcal{M}$, its queue backlog might differ between the two networks, and we denote by Q_{ik} and X_{ik} the queue backlogs of class k at node i in the real network and the imaginary network, respectively.

It is possible to stabilize the queue backlog of the imaginary network $\sum_{i \in \mathcal{A}, k \in \mathcal{K}} Q_{ik} + \sum_{i \in \mathcal{M}, k \in \mathcal{K}} X_{ik}$ by taking proper action in the imaginary network. However, the actual queue size of a malicious node $i \in \mathcal{M}$ might deviate significantly from X_{ik}. Thus, stabilizing the queues of the imaginary

network does not guarantee the stability of the real network. We define the gap between Q_{ik} and X_{ik} by $Y_{ik} \triangleq Q_{ik} - X_{ik}$ and aim at stabilizing Q_{ik} for $i \in \mathcal{A}$, X_{ik} and Y_{ik} for $i \in \mathcal{M}$, together. In other words, we decompose the queue backlog in the real system in the following manner,

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(t) = \sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} Y_{ik}(t), \tag{2}$$

and attempt to stabilizable the three terms on the right side simultaneously.

3.2 Algorithm

We apply the Lyapunov optimization framework to stabilize (2). We first define a Lyapunov function

$$\Phi(t) \triangleq \sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}^2(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}^2(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} Y_{ik}^{+2}(t), \tag{3}$$

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

To control the growth of $\Phi(t)$, we define the Lyapunov drift as $\Delta\Phi(t) \triangleq \Phi(t+1) - \Phi(t)$ and minimize $\Delta\Phi(t)$ at each time slot. We define $\Delta Q_{ik}(t)$, $\Delta X_{ik}(t)$ and $\Delta Y_{ik}^+(t)$ in a similar manner. Minimizing $\Delta\Phi(t)$ can be shown to be equivalent to minimizing

$$\sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(t)\Delta Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(t)\Delta X_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} Y_{ik}^+(t)\Delta Y_{ik}^+(t).$$

However, for a malicious node $i \in \mathcal{M}$, the network controller doe not have instantaneous access to its queue backlog $Q_{ik}(t)$ and thus the value of $Y_{ik}(t)$ is unavailable to the network controller. As discussed in Section 2, the network controller can obtain estimates of Q_{ik} at certain time slots Γ_i. Therefore, the network controller can use the most recently estimated $\hat{Q}_{ik}(t)$ to estimate $Y_{ik}(t)$, i.e.,

$$\hat{Y}_{ik}(t) = \hat{Q}_{ik}(\tau_i(t)) - X_{ik}(t), \tag{4}$$

where $\tau_i(t)$ is the most recent time when an estimation of Q_{ik} was made, i.e., $\tau_i(t) \triangleq \max_{\tau\in\Gamma_i:\tau\leqslant t} \tau$ and the objective of minimization now becomes

$$\sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(t)\Delta Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(t)\Delta X_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} \hat{Y}_{ik}^+(t)\Delta Y_{ik}^+(t). \tag{5}$$

We denote by $\boldsymbol{f}^M(t)$ and $\boldsymbol{g}^M(t)$ the flow assignments that minimize (5), which can be expressed as,

$$\begin{aligned}\boldsymbol{f}^M(t), \boldsymbol{g}^M(t) = \underset{0\leqslant f_{ijk}, g_{ijk}\leqslant C_{ij}}{\operatorname{argmin}} & \sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(t)\left[\sum_{j\in\mathcal{A}} f_{jik} - \sum_{j\in\mathcal{N}\cup d_k} f_{ijk}\right] + \\ & \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(t)\left[\sum_{j\in\mathcal{A}} f_{jik} + \sum_{j\in\mathcal{M}} g_{jik} - \sum_{j\in\mathcal{N}\cup d_k} g_{ijk}\right] - \\ & \sum_{i\in\mathcal{M},k\in\mathcal{K}} \hat{Y}_{ik}^+(t)\left[\sum_{j\in\mathcal{M}} g_{jik} - \min\left\{\sum_{j\in\mathcal{N}\cup d_k} g_{ijk}, X_{ik}(t) + a_{ik}(t)\right\}\right].\end{aligned} \tag{6}$$

For each time slot, the network controller solves (6) and applies $\boldsymbol{f}^M(t)$ to the accessible nodes in the **real network**, meanwhile using both $\boldsymbol{f}^M(t)$ and $\boldsymbol{g}^M(t)$ to update $X_{ik}(t)$ for all malicious nodes $i \in \mathcal{M}$, according to

$$X_{ik}(t+1) = \left[X_{ik}(t) + a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k} g_{ijk}(t)\right]^+ + \sum_{j\in\mathcal{A}} \tilde{f}_{jik}(t) + \sum_{j\in\mathcal{M}} g_{jik}(t), \tag{7}$$

where, for technical reasons, we assume that in the **imaginary network**, malicious nodes can transmit dummy packets when the allotted packets to be transmitted are less than the queue backlog (i.e., $\tilde{g}_{ijk} \equiv g_{ijk}$ for $i \in \mathcal{M}$).

The complete algorithm is given in Algorithm 1.

Algorithm 1 The MWUM algorithm

Input: T, $Q_{ik}(0)$, Γ_i for $i \in \mathcal{M}$
Initialization: $X_{ik}(0) \leftarrow Q_{ik}(0)$ for $i \in \mathcal{N}$, $\hat{Y}_{ik}(0) \leftarrow 0$ for $i \in \mathcal{M}$
for $t \leftarrow 0, 1, \cdots, T-1$ **do**
  Obtain $Q_{ik}(t)$ for $i \in \mathcal{A}$, $X_{ik}(t)$ for $i \in \mathcal{M}$, and $a_{ik}(t)$ for $i \in \mathcal{N}$
  for $i \in \mathcal{M}$ **do**
    if $t \in \Gamma_i$ **then**
      Obtain an estimation $\hat{Q}_{ik}(t)$ for $k \in \mathcal{K}$
    end if
    Update $\hat{Y}_{ik}(t)$ using Eqn (4) for $k \in \mathcal{K}$
  end for
  Solve Eqn (6) and obtain $\boldsymbol{f}^M(t), \boldsymbol{g}^M(t)$
  Implement $\boldsymbol{f}^M(t)$ to accessible nodes $\mathcal{A}$ in the real network
  Update $X_{ik}(t+1)$ using Eqn (7) for $i \in \mathcal{M}$ and $k \in \mathcal{K}$
end for
Output: action sequence for accessible nodes $\boldsymbol{f}^M(t)$ for $t = 0, \cdots, T-1$, i.e., π_M

4 PERFORMANCE ANALYSIS

To analyze the stability of MWUM, we start with the case where the estimates $\hat{Q}_{ik}(t)$ are accurate, i.e., $\hat{Q}_{ik}(t) = Q_{ik}(t)$ for all malicious nodes $i \in \mathcal{M}$ and $t \in \Gamma_i$. We first prove stability under the most challenging setting - the Q_T constraint. We then extend our analysis to include the V_T constraint and the W constraint and obtain results stronger than rate stability. Finally, with the rate stability results, we are able to characterize the stability regions for networks with unobservable malicious nodes.

4.1 Stability for Networks with Q_T-Constrained Dynamics

As mentioned in Section 3.2, $\tau_i(t)$ is the most recent time, prior to time t, an estimate of Q_{ik} was obtained. We define $L(t)$ to be the maximum delay in observations at t, i.e., $\max_{i\in\mathcal{M},k\in\mathcal{K}} t - \tau_i(t)$. Intuitively, for a network with Q_T-constrained dynamics, if $Q_T = o(T)$, then there exists a stabilizing network control policy. Moreover, if $L(t)$ is also sublinear in T, the delay in obtaining queue information should not affect stability significantly [15, 28].

We show that the action sequence on accessible nodes $\boldsymbol{f}^M(t)$ generated by MWUM can achieve rate stability under the aforementioned mild conditions of Q_T and $L(t)$, as stated in Theorem 2.

Theorem 2. *A network with Q_T-constrained dynamics is rate stable under MWUM if $Q_T = o(T)$ and $\sum_{t=0}^{T-1} L(t)/T = o(T)$.*

Proof. The outline of the proof is as follows. We first upper bound the queue backlog at time T with the Lyapunov function Φ in Lemma 1. We then analyze and upper bound the drift $\Delta\Phi$ in Lemma 2 and 3. Finally, we obtain an upper bound of $\Phi(T)$ via Lemma 4 and 5, which shows that the queue backlog at time T is sublinear in T and thus concludes the proof.

Directly analyzing the growth of queue backlogs is difficult, thus we first explore the relationship between queue backlogs and the Lyapunov function Φ (defined in (3)).

Lemma 1. *For any policy, we have*

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(T) \leqslant \sqrt{2NK\Phi(T)}.$$

See Appendix B for the proof. Lemma 1 shows that the total queue backlog grows sublinearly in T if the terminal value of the Lyapunov function is subquadratic in T, i.e., $\Phi(T) = o(T^2)$. We next turn to deriving an upper bound for $\Phi(T)$.

For simplicity of exposition, we make the following definitions of the one-slot changes in $Q_{ik}(t)$'s, $X_{ik}(t)$'s and $Y_{ik}(t)$'s. Note that we use δ instead of Δ for $\delta Q_{ik}(t)$ and $\delta X_{ik}(t)$ because they are not the actual one-slot changes (using $\tilde{f}_{ijk}$, $\tilde{g}_{ijk}$ and $\tilde{\mu}_{ijk}$) but the planned one-slot changes (using f_{ijk}, g_{ijk} and μ_{ijk}).

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

Next we decompose $\Phi(t+1)-\Phi(t)$ into analyzable terms. We first upper bound $Q_{ik}^2(t+1)-Q_{ik}^2(t)$ for $i \in \mathcal{A}$ and $X_{ik}^2(t+1) - X_{ik}^2(t)$ for $i \in \mathcal{M}$ in Lemma 2.

Lemma 2. *For each $t = 0, \cdots, T-1$, we have*

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

See Appendix C for the proof. We then upper bound $Y_{ik}^{+2}(t+1) - Y_{ik}^{+2}(t)$ for $i \in \mathcal{M}$ in Lemma 3.

Lemma 3. *For each $i \in \mathcal{M}$ 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 D for the proof. Using Lemma 2 and Lemma 3, we can upper bound $\Delta\Phi^M(t) \triangleq \Phi^M(t+1) - \Phi^M(t)$ as follows (the superscript M denotes that the variable is obtained under the action sequence π_M generated by our algoritm MUWM),

$$\begin{aligned} \Delta\Phi^M(t) \leqslant & 2\sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}^M(t)\delta^M Q_{ik}(t) + 2\sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}^M(t)\delta^M X_{ik}(t) + \\ & 2\sum_{i\in\mathcal{M},k\in\mathcal{K}} \hat{Y}_{ik}^{M+}(t_{ik}^*)\Delta^M Y_{ik}(t) + (8L(t)+18)N^3KD^2. \end{aligned} \tag{8}$$

By the definition of Q_T-constrained dynamics, for the network event sequence $\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\}_{0\leqslant t\leqslant T-1}$ generated under the application of MWUM, there exists a policy π^* such that $\sum_{i,k} Q_{ik}^*(T) \leqslant Q_T$. Since MWUM minimizes (5), replacing the actions $\{\boldsymbol{f}^M(t), \boldsymbol{g}^M(t)\}_{0\leqslant t\leqslant T-1}$ with $\{\boldsymbol{f}^*(t), \boldsymbol{\mu}(t)\}_{0\leqslant t\leqslant T-1}$ will not decrease (8), i.e.,

$$\begin{aligned} \Delta\Phi^M(t) \leqslant & 2\sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}^M(t)\delta^* Q_{ik}(t) + 2\sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}^M(t)\delta^* X_{ik}(t) + \\ & 2\sum_{i\in\mathcal{M},k\in\mathcal{K}} \hat{Y}_{ik}^{M+}(t)\Delta^* Y_{ik}(t) + (8L(t)+18)N^3KD^2. \end{aligned} \tag{9}$$

Using (9) and summing up $\Delta\Phi^M(t)$ from $t = 0$ to time $t = T-1$, the value of $\Phi^M(T)$ is upper bounded by

$$\Phi^M(T) \leqslant \Phi(0) + 2\sum_{i\in\mathcal{A},k\in\mathcal{K}}\sum_{t=0}^{T-1} Q_{ik}^M(t)\delta^* Q_{ik}(t) + 2\sum_{i\in\mathcal{M},k\in\mathcal{K}}\sum_{t=0}^{T-1} X_{ik}^M(t)\delta^* X_{ik}(t) + 2\sum_{i\in\mathcal{M},k\in\mathcal{K}}\sum_{t=0}^{T-1}\hat{Y}_{ik}^{M+}(t)\Delta^* Y_{ik}(t) + 18N^3KD^2T + 8N^3KD^2\sum_{t=0}^{T-1}L(t). \tag{10}$$

We next need to upper bound the second, third and fourth term in (10). For the second and third terms we use the following lemma.

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

$$\sum_{i\in\mathcal{A},k\in\mathcal{K}}\sum_{t=0}^{T-1} Q_{ik}^M(t)\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}}\sum_{t=0}^{T-1} X_{ik}^M(t)\delta^* X_{ik}(t) \leqslant \frac{2NKDT^2Q_T}{H} + 8N^3KD^2HT.$$

See Appendix E for the proof. We next upper bound the fourth term as follows,

Lemma 5. *For each $i \in \mathcal{M}$ and $k \in \mathcal{K}$, we have*

$$\sum_{t=0}^{T-1}\hat{Y}_{ik}^{M+}(t)\Delta^* Y_{ik}(t) \leqslant 4N^2D^2\sum_{t=0}^{T-1}L(t) + 2N^2D^2T.$$

See Appendix F for the proof. Now, let $H = c\sqrt{TQ_T/(N^2D)}$ (where c is any positive constant that makes H an integer). Using results in Lemma 4 and Lemma 5 in (10) and then applying Lemma 1, we obtain,

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^M(T) \leqslant \left[4N^3K^2D\left((8c+2/c)\sqrt{DTQ_T} + 11ND\right)T + 32N^4K^2D^2\sum_{t=0}^{T-1}L(t) + 2NK\Phi(0)\right]^{1/2} = O\left(T^{3/4}Q_T^{1/4} + \sqrt{\sum_{t=0}^{T-1}L(t)}\right).$$

When $Q_T = o(T)$ and $\sum_{t=0}^{T-1}L(t) = o(T^2)$, we have $\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^M(T) = o(T)$ and the network is rate stable. □

As discussed before, if the Q_T-constrained dynamics do not satisfy $Q_T = o(T)$, then the adversary can generate a series of $\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\}_{0\leqslant t\leqslant T-1}$ such that no policy can stabilize the network. But as long as $Q_T = o(T)$ and $\sum_{t=0}^{T-1}L(t)/T = o(T)$, MWUM can stabilize the network. Therefore, MWUM is "throughput-optimal" (i.e., can stabilize a network if the network is stabilizable). More detailed discussion is provided in Section 4.3.

4.2 Stability for Networks with V_T/W-Constrained Dynamics

As discussed in Section 2.3, the Q_T constraint is the most relaxed maliciousness metric. If the network further satisfies the V_T constraint or the W constraint, the adversary is less malicious, and it is possible to obtain stronger results than rate stability.

For a network with V_T-constrained dynamics, since the V_T constraint bounds the maliciousness for all time slots, we provide a stronger result below,

Theorem 3. *For a network with V_T-constrained dynamics, if $V_T = o(T)$ and $\sum_{t=0}^{T-1} L(t)/T = o(T)$, we have*

$$\lim_{T\to\infty} \frac{\max_{t\leqslant T} \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^M(t)}{T} = 0,$$

under MWUM.

Proof. Define $T^* \triangleq \arg\max_{t\leqslant T} \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^M(t)$. By replacing T with T^* in the proof of Theorem 2 and changing the definition of π^* to the policy corresponding to Definition 3, we show that

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^M(T^*) = O\left(T^{*3/4}V_T^{1/4} + \sqrt{\sum_{t=0}^{T^*-1} L(t)}\right), \tag{11}$$

which completes the proof. □

Since the W constraint is even more restrictive than the V_T constraint, we can further extend the analysis in Theorem 3 to W-constrained dynamics as follows,

Theorem 4. *For a network with W-constrained dynamics, if $W = o(T)$ and $\sum_{t=0}^{T-1} L(t)/T = o(T)$, we have*

$$\lim_{T\to\infty} \frac{\max_{t\leqslant T} \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^M(t)}{T} = 0,$$

under MWUM.

Proof. The definition of T^* remains identical as the proof of Theorem 3. We replace π^* with π^W (as defined in Definition 2). By Theorem 1, the upper bound (11) still holds after replacing V_T with $\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(0) + NKDW$, which completes the proof. □

Obviously, Theorem 3 and 4 also imply the conditions to achieve rate stability under V_T and W constraints. Moreover, they imply that when the maliciousness metrics are sublinear in T and $\sum_{t=0}^{T-1} L(t)/T = o(T)$, the peak queue backlog along the time horizon is sublinear in time. Theorem 3 and 4 extend previous results [17, 19] which only discussed the rate stability at the end of the time horizon.

4.3 Stability Region

For networks with stochastic dynamics, the stability region is the set of external arrival rates such that there exists a policy under which the sum of arrival rates is no greater than the sum of service rates for each node. If network dynamics (i.e., arrivals, channel conditions) are inside its stability region, then there exists a policy to achieve rate stability. On the other hand, no policy can stabilize the network when the dynamics are outside the stability region.

For networks with malicious nodes, the concept of "rate" is no longer applicable (since the dynamics might be non-stochastic) and the adversarial actions taken by the malicious nodes also need to be considered.

By Definition 4, when $Q_T = \Omega(T)$, the adversary might implement a sequence of $\left\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\right\}_{0\leqslant t\leqslant T-1}$ which cannot be stabilized by any policy. However, as long as $Q_T = o(T)$, we have shown that MWUM could stabilize the network. Therefore, we could use Q_T to characterize the stability region.

Proposition 1. *For a given network, its stability region is the set of $\left\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\right\}_{0\leqslant t\leqslant T-1}$ with $Q_T = o(T)$.*

Since Theorem 2 has shown that when $Q_T = o(T)$ (i.e. inside the stability region), the network is rate stable under MWUM, we have the following corollary.

Corollary 1. *For a network with Q_T-constrained dynamics, MWUM is a throughput-optimal algorithm.*

5 PERFORMANCE WITH ESTIMATION ERRORS

In Section 4, we analyze stability under MWUM when the estimates $\hat{Q}_{ik}(t)$ are accurate. However, in practice, it is common that the estimation is erroneous due to the limits of statistical methods, transmission errors, or even errors injected by the adversary. For a malicious node $i \in \mathcal{M}$, data class $k \in \mathcal{K}$ and $t \in \Gamma_i$, we define the error as $\epsilon_{ik}(t) \triangleq \hat{Q}_{ik}(t) - Q_{ik}(t)$. To distinguish from the estimate $\hat{Y}_{ik}(t)$ without estimation error, we denote by $\tilde{Y}_{ik}(t)$ the corresponding erroneous version of $\hat{Y}_{ik}(t)$ and have $\tilde{Y}_{ik}(t) = \hat{Y}_{ik}(t) + \epsilon_{ik}(\tau_i(t))$.

5.1 Stability

Having obtained $\tilde{Y}^+_{ik}(t)$, in Algorithm 1 $\hat{Y}^+_{ik}(t)$ is replaced by $\tilde{Y}^+_{ik}(t)$ and the goal is to minimize

$$\sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(t)\Delta Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(t)\Delta X_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} \tilde{Y}^+_{ik}(t)\Delta Y^+_{ik}(t). \tag{12}$$

By expanding the Lyapunov optimization analysis in Theorem 2, we show that as long as the scale of $\epsilon_{ik}(t)$ is sublinear in t, rate stability still holds, i.e.,

Theorem 5. *A network with Q_T-constrained dynamics is rate stable under MWUM if $Q_T = o(T)$, $\sum_{t=0}^{T-1} L(t)/T = o(T)$, and $\left|\epsilon_{ik}(t)\right| = o(t)$ for each $i \in \mathcal{M}$, $k \in \mathcal{K}$ and $0 \leqslant t \leqslant T$.*

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

Lemma 6. *For each $i \in \mathcal{M}$ and $t = 0, \cdots, T-1$, we have*

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

Lemma 7. *For each $i \in \mathcal{M}$, we have*

$$\sum_{t=0}^{T-1} \tilde{Y}^{M+}_{ik}(t)\Delta^* Y_{ik}(t) \leqslant 4N^2D^2\sum_{t=0}^{T-1} L(t) + 2N^2D^2T + 2ND\sum_{t=0}^{T-1}\left|\epsilon_{ik}(\tau_i(t))\right|.$$

Proof of Lemma 6 and 7 can be found in Appendix G and H respectively. With Lemma 6 and 7, a similar analysis to the proof of Theorem 2 shows that

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q^M_{ik}(T) \leqslant O\left(T^{3/4}Q_T^{1/4} + \sqrt{\sum_{t=0}^{T-1} L(t)} + \sqrt{\sum_{t=0}^{T-1}\left|\epsilon_{ik}(\tau_i(t))\right|}\right).$$

Since $\left|\epsilon_{ik}(t)\right| = o(t)$,

$$\sum_{t=0}^{T-1}\sum_{i\in\mathcal{M},k\in\mathcal{K}}\left|\epsilon_{ik}(\tau_i(t))\right| \leqslant \sum_{i\in\mathcal{M},k\in\mathcal{K}}\sum_{t=0}^{T-1} o(t) = o(T^2),$$

if $Q_T = o(T)$, $\sum_{t=0}^{T-1} L(t)/T = o(T)$, and $\left|\epsilon_{ik}(t)\right| = o(t)$, then $\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q^M_{ik}(T) = o(T)$, which completes the proof of Theorem 5. □

Theorem 5 shows that as long as the estimation error $\left|\epsilon_{ik}(t)\right|$ is sublinear in t, then the results of Theorem 2 still holds. In other words, as long as estimation errors grow sublinearly in time, the stability of MWUM is not affected. Similar to the performance guarantees provided in Theorem 3 and Theorem 4, the performance of networks with V_T-constrained and W-constrained dynamics under MWUM is as follows.

Theorem 6. *For a network with V_T(or W)-constrained dynamics, if $V_T = o(T)$ (or $W = o(T)$), $\sum_{t=0}^{T-1} L(t)/T = o(T)$, and $\left|\epsilon_{ik}(t)\right| = o(t)$ for each $i \in \mathcal{M}$ and $0 \leqslant t \leqslant T$, we have*

$$\lim_{T\to\infty} \frac{\max_{t\leqslant T} \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^{M}(t)}{T} = 0,$$

under MWUM.

5.2 Impact of Estimation Errors

Although we have shown that MWUM achieves rate stability as long as the estimation error is sublinear in t, a possible question of interest is: what will happen if the estimation error is larger, i.e., $\epsilon_{ik}(t) = \Omega(t)$?

A quick answer to the question is that estimation errors do not affect the existence of a stabilizing policy. From the definition of Q_T-constrained dynamics, for each network event sequence generated by the adversary, there always exists a policy that ensures the queue backlog at time T is sublinear in T. This policy does not need queue backlog estimates and thus estimation errors technically do not affect the stability region of the network.

However, such a policy requires full knowledge of the adversarial dynamics and is not practical. Instead, the network controller usually can only make decisions based on observable network state (i.e., queue backlogs of accessible nodes). We define this large class of "state-based" algorithms in the following manner and only discuss such algorithms in this section.

Definition 6. *A state-based network control algorithm generates actions solely based on the queue backlogs of the accessible nodes $\mathcal{A}$ (i.e., Q_{ik} for $i \in \mathcal{A}$) and the estimated queue backlogs of the malicious nodes $\mathcal{M}$ (i.e., X_{ik} and $\hat{Q}_{ik}$ for $i \in \mathcal{M}$).*

MWUM is a state-based algorithm by Definition 6. Since state-based algorithms rely on estimates of the malicious nodes, estimation errors could affect stability. We highlight this with an example below.

Theorem 7. *There exists a network with Q_T-constrained dynamics (where $Q_T = o(T)$) and $\epsilon_{ik}(t) = \Omega(t)$ for some $i \in \mathcal{M}$ and $k \in \mathcal{K}$ such that no state-based algorithm can achieve rate stability.*

Proof. Theorem 7 states that although the network is stabilizable by "some" algorithms, no state-based algorithm can achieve rate stability. We construct a 2-node network as shown in Figure 2. Node 1 is an accessible node and can directly transmit packets to the destination d or relay through node 2, while node 2 is unobservable and malicious.

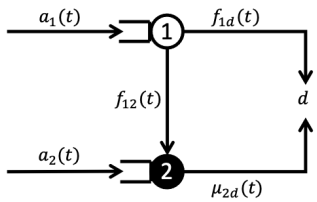


Fig. 2. The example system for Theorem 7

Since we assume the external arrivals $a_1(t)$ and $a_2(t)$ to be finite, the queue backlogs $Q_1(t)$ and $Q_2(t)$ can grow at most linearly in t. Therefore, when the estimation error $\epsilon_2(t)$ grows linearly in t, the error can completely "mask" the actual queue growth of node 2 and make the estimates $\hat{Q}_2(t)$ always zero, enticing the network 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 attack 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 network becomes unstable. A detailed proof of this phenomenon is provided in Appendix I. □

Theorem 7 shows that when the estimation error scales linearly or sup-linearly in t, there does not exist a state-based algorithm that can stabilize all networks. Combining Theorem 5 and 7, we have the following theorem on the necessary and sufficient conditions of achieving rate stability with estimation error.

Theorem 8. *There exists a state-based algorithm that stabilizes all networks with Q_T-constrained dynamics that has $Q_T = o(T)$ and $\sum_{t=0}^{T-1} L(t)/T = o(T)$, if and only if $\left|\epsilon_{ik}(t)\right| = o(t)$ for each $i \in \mathcal{M}$ and $0 \leqslant t \leqslant T$.*

6 NUMERICAL EXPERIMENTS

We conduct several simulations to validate the performance analysis of MWUM. We first examine a simple 3-node case, which has a clear lower bound to illustrate the gap between our algorithm and the optimum. We then study a more complex system of 12 nodes to show the performance of our algorithm in a more complex setting. We finally study the impact of estimation errors.

6.1 3-Node Network

We start from a simple network with 3 nodes, as shown in Figure 3, assuming that there is no estimation error. The network can be analyzed explicitly, and we are able to obtain a lower bound for queue backlog.

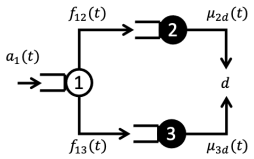


Fig. 3. 3-node network model.

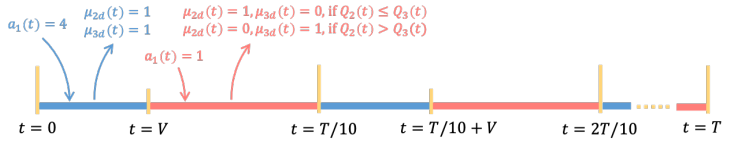
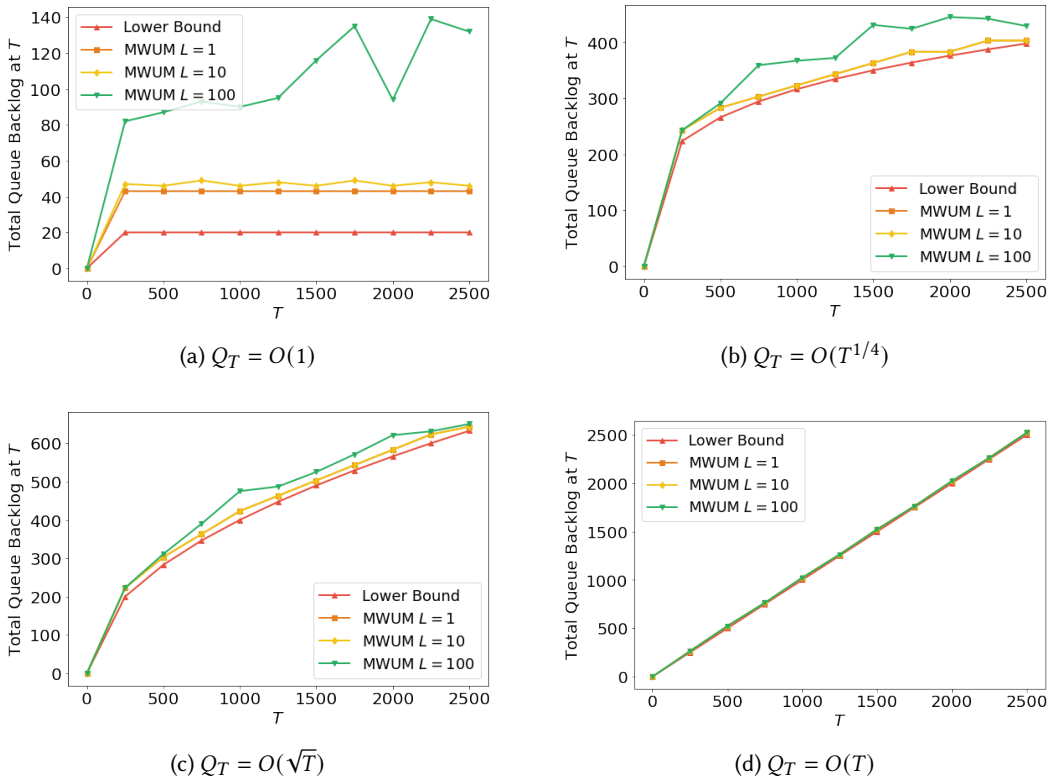


Fig. 4. Dynamics of the 3-node network model.

In the network, node 1 is accessible, while both node 2 and 3 are unobservable and malicious. All links have capacity of 4. The estimates of node 2 and 3 are obtained every L time slots. We use the parameter $V \leqslant T/10$ to describe the network dynamics as follows (also shown in Figure 4). For each $n = 0, 1, \cdots, 9$,

- When $nT/10 \leqslant t < nT/10 + V$, the external arrivals are $a_1(t) = 4$, and both node 2 and 3 transmit one packet to the destination d, i.e., $\mu_{2d}(t) = \mu_{3d}(t) = 1$.
- When $nT/10 + V \leqslant t < (n+1)T/10$, the external arrival is reduced to $a_1(t) = 1$. Moreover, between node 2 and 3, only the node with **smaller** queue continues transmitting at a rate of one packet to destination d, while the other node pauses transmission. The strategy of node 2 and 3 is malicious in the sense that the node with larger queue is likely to remain unserved forever even as it may grow unbounded.

The total number of packets received from external arrivals is $10 \times 4V + 10(T/10 - V) = T + 30V$, while node 2 and 3 can serve at most $10 \times 2V + 10(T/10 - V) = T + 10V$ packets, thus a lower bound for the queue backlog at T is $T + 30V - (T + 10V) = 20V$. The control action of node 1 is to decide



(a) $Q_T = O(1)$

(b) $Q_T = O(T^{1/4})$

(c) $Q_T = O(\sqrt{T})$

(d) $Q_T = O(T)$

Fig. 5. Simulation results for the 3-node network.

how to route packets at node 1 by the choice of f_{12} and f_{13}. Given any network event sequence $\{\mu_{2d}(t), \mu_{3d}(t)\}_{0 \leqslant t \leqslant T-1}$, the policy that sets $f_{12}(t) = \mu_{2d}(t+1)$ and $f_{13}(t) = \mu_{3d}(t+1)$ guarantees $\sum_{i \in \mathcal{N}, k \in \mathcal{K}} Q_{ik}(T)$ to achieve the lower bound of $20V$. Therefore, the network has Q_T-constrained dynamics with $Q_T = 20V$.

We conduct simulations with different scalings of Q_T (i.e., V) in the time horizon T. For each Q_T, we obtain the total queue backlog at T for different T's and draw the curve illustrating how $\sum_{i \in \mathcal{N}, k \in \mathcal{K}} Q_{ik}(T)$ grows with different scalings of Q_T under the MWUM algorithm. We also compare the performance of MWUM under different values of estimation interval L to analyze the influence of estimation frequency. Note that the upper bound of $\sum_{i \in \mathcal{N}, k \in \mathcal{K}} Q_{ik}(T)$ given in the proof of Theorem 2 is much larger than the actual performance and is omitted from the plots. The simulation results are shown in Figure 5.

From Figure 5, we can see that $\sum_{i \in \mathcal{N}, k \in \mathcal{K}} Q_{ik}(T)$ has the same order as the lower bound. As Q_T increases, the absolute gap between MWUM and the lower bound remains bounded, while the relative gap diminishes. This shows that MWUM is order-optimal for the 3-node network example. Also, the comparison among different values of L shows that if the estimates are obtained too sparsely, the performance of MWUM downgrades, which is consistent with the upper bound on $\sum_{i \in \mathcal{N}, k \in \mathcal{K}} Q_{ik}(T)$ in the proof to Theorem 2. Note that $\sum_{i \in \mathcal{N}, k \in \mathcal{K}} Q_{ik}(T)$ might not grow monotonically with T, and a possible reason is due to the choice of L. Having fresh information of malicious nodes can greatly affect $\sum_{i \in \mathcal{N}, k \in \mathcal{K}} Q_{ik}(T)$. Sometimes larger L divides T and helps to obtain fresher information of malicious nodes, thus improves the performance.

6.2 12-Node Network

We now implement MWUM in a more complex network to illustrate its practicality. The network, as in Figure 6, contains 3 external arrival sources, 2 destinations and 12 nodes. Among them, node 2, 3, 4 and 6 are unobservable and malicious, while the rest are accessible. All links, including link $9 \rightarrow d$ and $10 \rightarrow d$, have the capacity of 5.

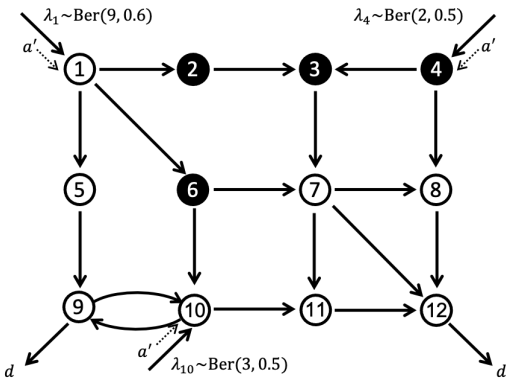


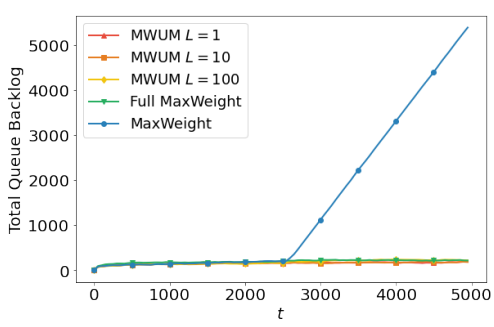
Fig. 6. 12-node network model.

At each time slot, node 1 receives 9 packets with probability 0.6 and receives no packet otherwise (i.e., Bernoulli process Ber$(9, 0.6)$). Similarly, the external arrival process for node 4 and 10 are Ber$(2, 0.5)$ and Ber$(3, 0.5)$, respectively. Moreover, an adversary injects at each time slot $a' = 2$ packets into the network through node 1, 4 or 10. In an attempt to destabilize the network, the adversary chooses to inject the a' packets into the node with the **largest** queue. Similarly, node 4 and 6 apply the "join the longest queue" (JLQ) policy that transmits 5 packets to the neighboring node with the larger queue size and transmits nothing to the other neighboring node. JLQ, in contrast to the stabilizing "join the shortest queue" (JSQ) policy, is adversarial since the node with the larger queue is more heavily loaded and hence, easier to destabilize. Node 3 simply transmits 5 packets to node 7 at each time slot. Node 2 transmits 5 packets to node 3 for the first $T/2$ time slots, but starting at $T/2$, it only transmits 1 packet to node 3.

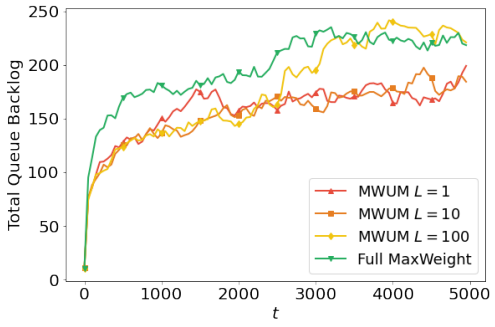
The network is challenging since the expected number of external arrivals at each time slot is $9 \times 0.6 + 2 \times 0.5 + 3 \times 0.5 + a' = 9.9$ (packets), while the total service rate is $C_{9d} + C_{12,d} = 10$ (packets). The network is heavily loaded and can easily become unstable without proper control decisions. Moreover, starting at $T/2$, the service rate of node 2 drops sharply, which requires the algorithm to sense the change in time and alter the policy accordingly.

We conduct the simulation for 5000 time slots and compare the performance under different policies: 1) directly applying MaxWeight to accessible nodes (MaxWeight), 2) assuming all nodes are accessible, and applying MaxWeight to all nodes (Full MaxWeight), and 3) MWUM under different estimation intervals (assuming accurate estimates). The simulation results are shown in Figure 7.

From Figure 7, we can see that directly applying the traditional MaxWeight algorithm cannot stabilize the network. Because of the sudden change of μ_{23} at $T/2$, node 2 can only serve 1 packet during the second half of the time horizon. However, the traditional MaxWeight algorithm cannot observe it and may continue transmitting more than 1 packet to node 2, leading to linear growth in the queue size. Both the full MaxWeight algorithm and MWUM stabilize the network, yet surprisingly, MWUM achieves a smaller queue backlog. This is due to the fact that the MaxWeight algorithm minimizes the drift rather than the queue backlog and can only guarantee stability rather



(a) Results of all policies.



(b) Results of stabilizing policies.

Fig. 7. The growth of total queue backlog.

than minimal queue backlog. In addition, when $L = 100$, MWUM has significantly downgraded performance after $T/2$. The reason is that since the estimation is heavily delayed, it takes a much longer time to notice the abnormal growth of the queue at node 2.

We further study whether MWUM successfully tracks the queues of malicious nodes with $L = 10$, as shown in Figure 8. From the figure, for all malicious nodes, the gaps between X_{ik} and Q_{ik} are bounded and small. This shows that MWUM tracks the real queue backlogs of malicious nodes well.

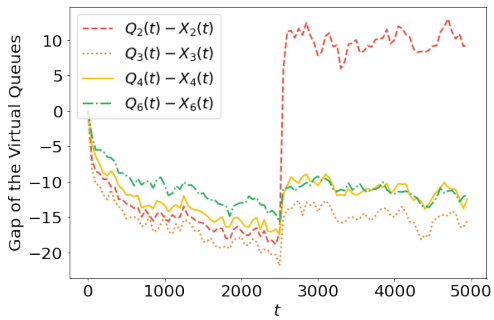


Fig. 8. The gaps between the "imagined" queues X_{ik} and actual queues Q_{ik} for malicious nodes ($L = 10$).

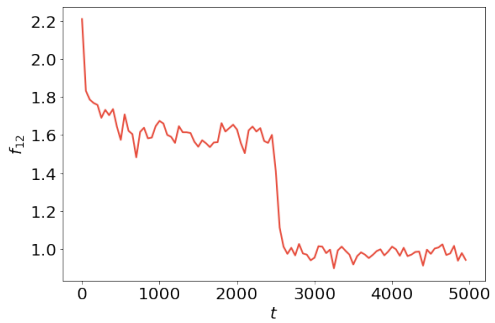


Fig. 9. The evolution of $f_{12}(t)$ ($L = 10$).

Finally, we trace $f_{12}(t)$ with $L = 10$ to see how MWUM responds to the sudden drop of μ_{12} at $T/2$, as shown in Figure 9. From the figure, for the first $T/2$ time slots, since node 2 can serve at the rate of 5 (packets), MWUM transmits more than 1 packet from node 1 to 2. After $T/2$, MWUM learns the change in μ_{12} and reduce the transmission rate to 1 without exceeding the service capacity of node 2.

6.3 Network with Estimation Errors

We continue using the 12-node network model designed in 6.2, but assuming the existence of estimation errors when applying MWUM. We take the estimation interval $L = 10$, i.e., the network controller obtains an estimate for the malicious nodes 2, 3, 4, and 6 every 10 time slots. In simulation, for a given error scale ε, the estimation error of node i at time t is uniformly distributed between $-\varepsilon$

and ε, i.e. $\epsilon_{ik}(t) = \text{Unif}(-\varepsilon, \varepsilon)$. The network controller only obtains erroneous estimates $\hat{Q}_{ik}(t) = Q_{ik}(t) + \epsilon_{ik}(t)$. We conduct simulations under different values of error scale ε: 1) no error, 2) constant error, 3) error grows in $t^{1/4}$, 4) error grows in $\sqrt{t}$, and 5) error grows linearly in t. The results are shown in Figure 10.

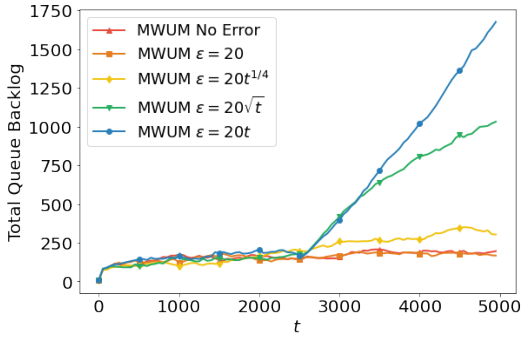


Fig. 10. The growth of total queue backlog with estimation errors.

From the figure, as the error scale grows, the total queue backlog becomes larger. When the error has a constant bound, the impact is minor. For errors that grow sublinearly in t, the total queue backlog is larger, but still grows sublinearly in the time horizon and thus stabilizes. For errors that grow linearly in t, MWUM fails to stabilize. The simulation results validate Theorem 8.

7 CONCLUSIONS

In this paper, we focus on networks with unobservable and uncontrollable nodes, under adversarial dynamics (i.e. external arrivals and actions of malicious nodes). We first propose a new maliciousness metric named Q_T constraint to characterize the adversarial network model and make a comprehensive comparison to the W and V_T constraints from previous works. We then propose the MWUM algorithm that only needs to be operated on accessible nodes, and show that MWUM achieves rate stability when $Q_T = o(T)$. We also strengthen the existing stability results under the W and V_T constraints using our analysis framework. We further characterize the stability region for adversarial network systems and show that MWUM is a throughput-optimal network control algorithm. Moreover, we discuss the case when estimates are erroneous and show that MWUM can still stabilize the network, as long as the errors grow sublinearly in time. We finally provide the necessary and sufficient conditions for networks to be stabilizable under estimation errors.

A possible direction for future work is to develop explicit estimation methods for unobservable malicious nodes and analyze their estimation error bounds. Moreover, we focus on stabilizing the queue backlogs in this paper, yet going beyond stability to reach optimality for general networks largely remains an open problem. Therefore, another possible problem of interest is how to minimize the queue backlog of general networks under various settings, e.g., cooperative environment, adversarial environment, arbitrary environment. The recent emergence of machine learning techniques may provide new tools in this direction.

ACKNOWLEDGMENTS

This work was funded by NSF grants CNS-1524317, NSF CNS-1907905 and by Office of Naval Research (ONR) grant award N00014-20-1-2119.

REFERENCES

[1] Matthew Andrews, Baruch Awerbuch, Antonio Fernández, Tom Leighton, Zhiyong Liu, and Jon Kleinberg. 2001. Universal-stability results and performance bounds for greedy contention-resolution protocols. *Journal of the ACM (JACM)* 48, 1 (2001), 39–69.

[2] Matthew Andrews and Lisa Zhang. 2004. Scheduling over nonstationary wireless channels with finite rate sets. In *IEEE INFOCOM 2004*, Vol. 3. IEEE, 1694–1704.

[3] Matthew Andrews and Lisa Zhang. 2005. Scheduling over a time-varying user-dependent channel with applications to high-speed wireless data. *Journal of the ACM (JACM)* 52, 5 (2005), 809–834.

[4] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. 1995. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*. IEEE, 322–331.

[5] Jonathan Baxter and Peter L Bartlett. 2001. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research* 15 (2001), 319–350.

[6] Allan Borodin, Jon Kleinberg, Prabhakar Raghavan, Madhu Sudan, and David P Williamson. 1996. Adversarial queueing theory. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 376–385.

[7] Sébastien Bubeck and Nicolo Cesa-Bianchi. 2012. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721* (2012).

[8] Anthony R Cassandra, Michael L Littman, and Nevin Lianwen Zhang. 2013. Incremental pruning: A simple, fast, exact method for partially observable Markov decision processes. *arXiv preprint arXiv:1302.1525* (2013).

[9] Hsien-Te Cheng. 1988. *Algorithms for partially observable Markov decision processes*. Ph.D. Dissertation. University of British Columbia.

[10] Christos Douligeris and Aikaterini Mitrokotsa. 2004. DDoS attacks and defense mechanisms: classification and state-of-the-art. *Computer Networks* 44, 5 (2004), 643–666.

[11] William G Halfond, Jeremy Viegas, Alessandro Orso, et al. 2006. A classification of SQL-injection attacks and countermeasures. In *Proceedings of the IEEE international symposium on secure software engineering*, Vol. 1. IEEE, 13–15.

[12] Junghee Han, David Watson, and Farnam Jahanian. 2005. Topology aware overlay networks. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, Vol. 4. IEEE, 2554–2565.

[13] Nathaniel M Jones, Georgios S Paschos, Brooke Shrader, and Eytan Modiano. 2017. An overlay architecture for throughput optimal multipath routing. *IEEE/ACM Transactions on Networking* 25, 5 (2017), 2615–2628.

[14] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence* 101, 1-2 (1998), 99–134.

[15] Joy Kuri and Anurag Kumar. 1995. Optimal control of arrivals to queues with delayed queue length information. *IEEE Trans. Automat. Control* 40, 8 (1995), 1444–1450.

[16] Zhi Li and P Mohapatra. 2006. QRON: QoS-aware routing in overlay networks. *IEEE Journal on Selected Areas in Communications* 22, 1 (2006), 29–40.

[17] Qingkai Liang and Eytan Modiano. 2018. Minimizing Queue Length Regret Under Adversarial Network Models. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 2, 1 (2018), 1–32.

[18] Qingkai Liang and Eytan Modiano. 2019. Optimal network control in partially-controllable networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 397–405.

[19] Qingkai Liang and Eytan Modiano. 2019. Optimal Network Control with Adversarial Uncontrollable Nodes. In *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 101–110.

[20] Sungsu Lim, Kyomin Jung, and Matthew Andrews. 2013. Stability of the max-weight protocol in adversarial wireless networks. *IEEE/ACM Transactions on Networking* 22, 6 (2013), 1859–1872.

[21] Jelena Mirkovic and Peter Reiher. 2004. A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review* 34, 2 (2004), 39–53.

[22] Michael J Neely, Eytan Modiano, and Chih-Ping Li. 2008. Fairness and optimal stochastic control for heterogeneous networks. *IEEE/ACM Transactions On Networking* 16, 2 (2008), 396–409.

[23] Georgios S Paschos and Eytan Modiano. 2014. Throughput optimal routing in overlay networks. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 401–408.

[24] Anurag Rai, Rahul Singh, and Eytan Modiano. 2019. A distributed algorithm for throughput optimal routing in overlay networks. In *2019 IFIP Networking Conference (IFIP Networking)*. IEEE, 1–9.

[25] Edward J Sondik. 1978. The optimal control of partially observable Markov processes over the infinite horizon: Discounted costs. *Operations research* 26, 2 (1978), 282–304.

[26] Leandros Tassiulas and Anthony Ephremides. 1990. 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, 2130–2132.

[27] Panayiotis Tsaparas. [n.d.]. *Stability in adversarial queueing theory*. Ph.D. Dissertation. National Library of Canada= Bibliothèque nationale du Canada.

[28] Lei Ying and Sanjay Shakkottai. 2011. On throughput optimality with delayed network-state information. *IEEE Transactions on Information Theory* 57, 8 (2011), 5116–5132.
[29] Nevin L Zhang and Wenju Liu. 1996. *Planning in stochastic domains: Problem characteristics and approximation.* Technical Report. Technical Report HKUST-CS96-31, Hong Kong University of Science and Technology.

A PROOF OF THEOREM 1

For any given network, we define the policies that achieve V_T^* and Q_T^* in Definition 3 and 4 as π_V and π_Q, respectively. Since π_V may not minimize $\sum_{i,k} Q_{ik}(T)$, we have $Q_T^* \leqslant \sum_{i,k} Q_{ik}^V(T)$. By Definition 3, $V_T^* \geqslant \sum_{i,k} Q_{ik}^V(T)$ and we have

$$Q_T^* \leqslant V_T^*. \tag{13}$$

For any given network with W constraint and time horizon T, we define

$$t^* \triangleq \arg\max_{t \leqslant T} \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^W(t).$$

Since π_V minimized the peak queue backlog, we have

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^W(t^*) \geqslant V_T^*. \tag{14}$$

We define $M \triangleq t^* \bmod W$ and there exists an integer K such that $t^* = KW + M$. We then upper bound the total queue backlog at t^* as

$$\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^W(t^*) \leqslant \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}^W(KW-1) + NDW \leqslant \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(0) + NKDW, \tag{15}$$

where the first inequality comes from the fact that the total queue backlog grows at most NKD packets during each time slot and $M \leqslant W$, and the second inequality holds by Definition 2.

Combining (13), (14) and (15) completes the proof.

B PROOF OF LEMMA 1

From the definitions of X_{ik} and Y_{ik}, we can decompose the total queue backlog at T as

$$\begin{aligned}\sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(T) &= \sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} Y_{ik}(T)\\ &\leqslant \sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} Y_{ik}^+(T).\end{aligned}$$

Then by applying Cauchy–Schwarz inequality, we have

$$\begin{aligned}&\sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} Y_{ik}^+(T)\\ \leqslant&\sqrt{NK + |\mathcal{M}|K} \cdot \sqrt{\sum_{i\in\mathcal{A},k\in\mathcal{K}} Q_{ik}^2(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} X_{ik}^2(T) + \sum_{i\in\mathcal{M},k\in\mathcal{K}} Y_{ik}^{+2}(T)}\\ \leqslant&\sqrt{2NK} \cdot \sqrt{\Phi(T)},\end{aligned}$$

which completes the proof.

C PROOF OF LEMMA 2

We first upper bound $Q_{ik}^2(t+1) - Q_{ik}^2(t)$ for $i \in \mathcal{A}$. We first have that

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

It is easy to show that the inequality

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

holds for $x, y, z \geqslant 0$. By replacing x with $Q_{ik}(t) + a_{ik}(t)$, y with $\sum_{j \in \mathcal{N} \cup d_k} f_{ijk}(t)$ and z with $\sum_{j \in \mathcal{A}} f_{jik}(t) + \sum_{j \in \mathcal{M}} \mu_{jik}(t)$, we upper bound $Q_{ik}^2(t+1)$ as

$$\begin{aligned} Q_{ik}^2(t+1) \leqslant& Q_{ik}^2(t) + \left(\sum_{j \in \mathcal{N} \cup d_k} f_{ijk}(t) \right)^2 + \left(\sum_{j \in \mathcal{A}} f_{jik}(t) + \sum_{j \in \mathcal{M}} \mu_{jik}(t) \right)^2 + 2a_{ik}(t)\delta Q_{ik}(t) + 2Q_{ik}(t)\delta Q_{ik}(t) \\ \leqslant& Q_{ik}^2(t) + 2Q_{ik}(t)\delta Q_{ik}(t) + 6N^2D^2, \end{aligned} \tag{16}$$

where the last inequality comes from the setting that $0 \leqslant a_{ik}(t), f_{ijk}(t), \mu_{ijk}(t) \leqslant D$.

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

$$\begin{aligned} X_{ik}(t+1) =& \left[X_{ik}(t) + \hat{a}_{ik}(t) - \sum_{j \in \mathcal{N} \cup d_k} g_{ijk}(t) \right]^+ + \sum_{j \in \mathcal{A}} \tilde{f}_{jik}(t) + \sum_{j \in \mathcal{M}} g_{jik}(t) \\ \leqslant& \left[X_{ik}(t) + \hat{a}_{ik}(t) - \sum_{j \in \mathcal{N} \cup d_k} g_{ijk}(t) \right]^+ + \sum_{j \in \mathcal{A}} f_{jik}(t) + \sum_{j \in \mathcal{M}} g_{jik}(t), \end{aligned}$$

by applying similar techniques as (16), we have

$$X_{ik}^2(t+1) \leqslant X_{ik}^2(t) + 2X_{ik}(t)\delta X_{ik}(t) + 6N^2D^2. \tag{17}$$

Equation (16) and (17) complete the proof.

D PROOF OF LEMMA 3

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 $i \in \mathcal{M}$ and $t = 0, \cdots, T-1$, 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

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

Since we assume the value of μ_{ijk}'s and g_{ijk}'s is bounded between 0 and D, we have

$$-2ND \leqslant \Delta Y_{ik}(t) \leqslant 2ND. \tag{18}$$

With (18) at hand, we first have

$$\begin{aligned}
\Delta Y^+_{ik}(t) =& \max\{Y_{ik}(t+1), 0\} - Y^+_{ik}(t) = \max\{Y_{ik}(t+1) - Y^+_{ik}(t), -Y^+_{ik}(t)\}\\
\leqslant& \max\{Y_{ik}(t+1) - Y_{ik}(t), -Y^+_{ik}(t)\} = \max\{\Delta Y_{ik}(t), -Y^+_{ik}(t)\} \leqslant 2ND.
\end{aligned} \tag{19}$$

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

$$\begin{aligned}
\Delta Y^+_{ik}(t) =& Y^+_{ik}(t+1) - \max\{Y_{ik}(t), 0\} = \min\{Y^+_{ik}(t+1) - Y_{ik}(t), Y^+_{ik}(t+1)\}\\
\geqslant& \min\{Y_{ik}(t+1) - Y_{ik}(t), Y^+_{ik}(t+1)\} = \min\{\Delta Y_{ik}(t), Y^+_{ik}(t+1)\} \geqslant -2ND.
\end{aligned} \tag{20}$$

Combining (18), (19) and (20) completes the proof. □

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

$$Y^{+2}_{ik}(t+1) - Y^{+2}_{ik}(t) = \big(Y^+_{ik}(t) + \Delta Y^+_{ik}(t)\big)^2 - Y^{2+}_{ik}(t) = 2Y^+_{ik}(t)\Delta Y^+_{ik}(t) + \big(\Delta Y^+_{ik}(t)\big)^2, \tag{21}$$

we only need to upper bound $Y^+_{ik}(t)\Delta Y^+_{ik}(t)$, as follows

$$\begin{aligned}
Y^+_{ik}(t)\Delta Y^+_{ik}(t) \leqslant& Y^+_{ik}(t)\cdot\max\{\Delta Y_{ik}(t), -Y^+_{ik}(t)\}\\
=& Y^+_{ik}(t)\Delta Y_{ik}(t) + \max\{0, -Y^{+2}_{ik}(t) - Y^+_{ik}(t)\Delta Y_{ik}(t)\}\\
\leqslant& Y^+_{ik}(t)\Delta Y_{ik}(t) + \max\{0, -Y^{+2}_{ik}(t) + 2NDY^+_{ik}(t)\}\\
=& Y^+_{ik}(t)\Delta Y_{ik}(t) + \max\{0, -(Y^{+2}_{ik}(t) - ND)^2 + N^2D^2\}\\
\leqslant& Y^+_{ik}(t)\Delta Y_{ik}(t) + N^2D^2,
\end{aligned} \tag{22}$$

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 (22) into (21) and utilizing Lemma 8, we have that

$$\begin{aligned}
Y^{+2}_{ik}(t+1) - Y^{+2}_{ik}(t) \leqslant& 2Y^+_{ik}(t)\Delta Y_{ik}(t) + \big(\Delta Y^+_{ik}(t)\big)^2 + 2N^2D^2\\
\leqslant& 2Y^+_{ik}(t)\Delta Y_{ik}(t) + 6N^2D^2\\
\leqslant& 2\hat{Y}^+_{ik}(t)\Delta Y_{ik}(t) + 2\big(t - \tau_i(t)\big)\cdot 2ND\cdot 2ND + 6N^2D^2\\
\leqslant& 2\hat{Y}^+_{ik}(t)\Delta Y_{ik}(t) + \big(8L(t) + 6\big)N^2D^2,
\end{aligned} \tag{23}$$

which completes the proof.

E PROOF OF LEMMA 4

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

$$\begin{aligned}
&\sum_{t=0}^{T-1} Q_{ik}^{M}(t)\delta^* Q_{ik}(t) \\
&= \sum_{k=0}^{K-1}\left[Q_{ik}^{M}(kH)\sum_{t=kH}^{(k+1)H-1}\delta^* Q_{ik}(t) + \sum_{t=kH}^{(k+1)H-1}\left(Q_{ik}^{M}(t) - Q_{ik}^{M}(kH)\right)\cdot\delta^* Q_{ik}(t)\right] + \sum_{t=KH}^{T-1} Q_{ik}^{M}(t)\delta^* Q_{ik}(t) \\
&\leqslant \sum_{k=0}^{K-1}\left[2NDT\sum_{t=kH}^{(k+1)H-1}\delta^* Q_{ik}(t) + \sum_{t=kH}^{(k+1)H-1} 2NDH\cdot 2ND\right] + M\cdot 2NDT\cdot 2ND \\
&\leqslant 2KNDT\sum_{t=0}^{T-1}\delta^* Q_{ik}(t) + 8N^2D^2HT \leqslant \frac{2NDT^2}{H}\sum_{t=0}^{T-1}\delta^* Q_{ik}(t) + 8N^2D^2HT,
\end{aligned} \tag{24}$$

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

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

$$\sum_{t=0}^{T-1} X_{ik}^{M}(t)\delta^* X_{ik}(t) \leqslant \frac{2NDT^2}{H}\sum_{t=0}^{T-1}\delta^* X_{ik}(t) + 8N^2D^2HT. \tag{25}$$

We then proceed to analyze $\sum_{i\in\mathcal{A},k\in\mathcal{K}}\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}}\delta^* X_{ik}(t)$ as follows.

$$\begin{aligned}
\sum_{i\in\mathcal{A},k\in\mathcal{K}}\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}}\delta^* X_{ik}(t) &= \sum_{i\in\mathcal{A},k\in\mathcal{K}}\left(a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k} f_{ijk}^*(t) + \sum_{j\in\mathcal{A}} f_{jik}^*(t) + \sum_{j\in\mathcal{M}}\mu_{jik}(t)\right) + \\
&\quad \sum_{i\in\mathcal{M},k\in\mathcal{K}}\left(a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k}\mu_{ijk}(t) + \sum_{j\in\mathcal{A}} f_{jik}^*(t) + \sum_{j\in\mathcal{M}}\mu_{jik}(t)\right) \\
&= \sum_{i\in\mathcal{N},k\in\mathcal{K}} a_{ik}(t) - \sum_{i\in\mathcal{A},k\in\mathcal{K}} f_{id_kk}^*(t) - \sum_{i\in\mathcal{M},k\in\mathcal{K}}\mu_{id_kk}(t),
\end{aligned}$$

with which we have that

$$\begin{aligned}
&\sum_{t=0}^{T-1}\left(\sum_{i\in\mathcal{A},k\in\mathcal{K}}\delta^* Q_{ik}(t) + \sum_{i\in\mathcal{M},k\in\mathcal{K}}\delta^* X_{ik}(t)\right) \\
&\leqslant \sum_{t=0}^{T-1}\left(\sum_{i\in\mathcal{N},k\in\mathcal{K}} a_{ik}(t) - \sum_{i\in\mathcal{A},k\in\mathcal{K}} f_{id_kk}^*(t) - \sum_{i\in\mathcal{M},k\in\mathcal{K}}\mu_{id_kk}(t)\right).
\end{aligned} \tag{26}$$

On the other hand, by the definition of Q_T we have that

$$\begin{aligned}
Q_T &\geqslant \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(0) + \sum_{t=0}^{T-1}\sum_{i\in\mathcal{A},k\in\mathcal{K}}\left(a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k}\tilde{f}_{ijk}^*(t) + \sum_{j\in\mathcal{A}}\tilde{f}_{jik}^*(t) + \sum_{j\in\mathcal{M}}\tilde{\mu}_{jik}(t)\right) + \\
&\quad \sum_{t=0}^{T-1}\sum_{i\in\mathcal{M},k\in\mathcal{K}}\left(a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k}\tilde{\mu}_{ijk}(t) + \sum_{j\in\mathcal{A}}\tilde{f}_{jik}^*(t) + \sum_{j\in\mathcal{M}}\tilde{\mu}_{jik}(t)\right) \\
&= \sum_{i\in\mathcal{N},k\in\mathcal{K}} Q_{ik}(0) + \sum_{t=0}^{T-1}\left(\sum_{i\in\mathcal{N},k\in\mathcal{K}} a_{ik}(t) - \sum_{i\in\mathcal{A},k\in\mathcal{K}}\tilde{f}_{id_kk}^*(t) - \sum_{i\in\mathcal{M},k\in\mathcal{K}}\tilde{\mu}_{id_kk}(t)\right).
\end{aligned} \tag{27}$$

Combining (26) and (27), and use the fact that $Q_{ik}(0) \geqslant 0$, $\tilde{f}^*_{ijk}(t) \leqslant f^*_{ijk}(t)$ and $\tilde{\mu}^*_{ijk}(t) \leqslant \mu^*_{ijk}(t)$ hold for each i, j, t, the comparison between shows that

$$\sum_{t=0}^{T-1}\left(\sum_{i\in\mathcal{A},k\in\mathcal{K}}\delta^* Q_{ik}(t)+\sum_{i\in\mathcal{M},k\in\mathcal{K}}\delta^* X_{ik}(t)\right)\leqslant Q_T. \tag{28}$$

By summing up (24) and (25) over all nodes and all data types, and plugging in (28), the proof is completed.

F PROOF OF LEMMA 5

For $\hat{Y}^{M+}_{ik}(t)\Delta^* Y_{ik}(t)$, we first discuss the case when $Q^M_{ik}(t) < ND$. Since $X_{ik}(t) \geqslant 0$ and $Y_{ik}(t) = Q_{ik}(t) - X_{ik}(t) \leqslant Q_{ik}(t)$, we have $0 \leqslant Y^{M+}_{ik}(t) < ND$, which gives us that

$$\begin{aligned}\hat{Y}^{M+}_{ik}(t)\Delta^* Y_{ik}(t) \leqslant&\Big(Y^{M+}_{ik}(t)+\big(t-\tau_i(t)\big)\cdot 2ND\Big)\cdot 2ND\\ \leqslant&\big(ND+L(t)\cdot 2ND\big)\cdot 2ND=\big(4L(t)+2\big)\cdot N^2D^2,\end{aligned} \tag{29}$$

where the first inequality utilizes Lemma 8.

When $Q^M_{ik}(t) \geqslant ND$, we have $Q^M_{ik}(t) + a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k}\mu_{ijk}(t) \geqslant 0$ and thus $\tilde{\mu}_{ijk}(t) = \mu_{ijk}(t)$. To distinguish between the $\tilde{\mu}_{ijk}(t)$ in the real and imaginary network, we use $\tilde{\mu}'_{ijk}(t)$ to denote the actual transmitted packets in the imaginary network. Then $\Delta^* Y_{ik}(t)$ can be upper bounded as

$$\begin{aligned}&\Delta^* Y_{ik}(t) = \delta^* Q_{ik}(t) - \delta^* X_{ik}(t)\\ =&a_{ik}(t) - \sum_{j\in\mathcal{N}\cup d_k}\tilde{\mu}_{ijk}(t) + \sum_{j\in\mathcal{A}}\tilde{f}^*_{jik}(t) + \sum_{j\in\mathcal{M}}\tilde{\mu}_{jik}(t) -\\ &a_{ik}(t) + \sum_{j\in\mathcal{N}\cup d_k}\tilde{\mu}'_{ijk}(t) - \sum_{j\in\mathcal{A}}\tilde{f}^*_{jik}(t) - \sum_{j\in\mathcal{M}}\mu_{jik}(t)\\ =&-\sum_{j\in\mathcal{N}\cup d_k}\mu_{ijk}(t)+\sum_{j\in\mathcal{M}}\tilde{\mu}_{jik}(t)+\sum_{j\in\mathcal{N}\cup d_k}\tilde{\mu}'_{ijk}(t)-\sum_{j\in\mathcal{M}}\mu_{jik}(t)\\ \leqslant&-\sum_{j\in\mathcal{N}\cup d_k}\mu_{ijk}(t)+\sum_{j\in\mathcal{M}}\mu_{jik}(t)+\sum_{j\in\mathcal{N}\cup d_k}\mu_{ijk}(t)-\sum_{j\in\mathcal{M}}\mu_{jik}(t)=0,\end{aligned}$$

with which we have that

$$\hat{Y}^{M+}_{ik}(t)\Delta^* Y_{ik}(t) \leqslant 0. \tag{30}$$

Combining (29) and (30), we have

$$\sum_{t=0}^{T-1}\hat{Y}^{M+}_{ik}(t)\Delta^* Y_{ik}(t) \leqslant 4N^2D^2\sum_{t=0}^{T-1}L(t) + 2N^2D^2T,$$

which completes the proof.

G PROOF OF LEMMA 6

For $Y^{+2}_{ik}(t+1) - Y^{+2}_{ik}(t)$, we have the following upper bound

$$\begin{aligned}&Y^{+2}_{ik}(t+1) - Y^{+2}_{ik}(t)\\ \leqslant&2\hat{Y}^{+}_{ik}(t)\Delta Y_{ik}(t) + \big(8L(t)+6\big)N^2D^2\\ =&2\tilde{Y}^{+}_{ik}(t)\Delta Y_{ik}(t) + \big(8L(t)+6\big)N^2D^2 + 2\big(\tilde{Y}^{+}_{ik}(t) - \hat{Y}^{+}_{ik}(t)\big)\Delta Y_{ik}(t)\\ \leqslant&2\hat{Y}^{+}_{ik}(t^*_{ik})\Delta Y_{ik}(t) + \big(8L(t)+6\big)N^2D^2 + 4ND\Big|\tilde{Y}^{+}_{ik}(t) - \hat{Y}^{+}_{ik}(t)\Big|,\end{aligned} \tag{31}$$

where the first inequality holds because of Lemma 3 and the last inequality utilizes Lemma 8.

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

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

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

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

Therefore, we have the upper bound $\left|\tilde{Y}_{ik}^{+}(t) - \hat{Y}_{ik}^{+}(t)\right| \leqslant \left|\epsilon_{ik}\big(\tau_i(t)\big)\right|$. By inserting it into (31), we complete the proof.

H PROOF OF LEMMA 7

For $\tilde{Y}_{ik}^{M+}(t)\Delta^{*}Y_{ik}(t)$, we first discuss the case when $Q_{ik}^{M}(t) < ND$. Similar to Lemma 5, we still have $0 \leqslant Y_{ik}^{M+}(t) < ND$, which gives us that

$$\begin{aligned}\tilde{Y}_{ik}^{M+}(t)\Delta^{*}Y_{ik}(t) =& \left(\hat{Y}_{ik}^{M+}(t) + \epsilon_{ik}\big(\tau_i(t)\big)\right) \cdot \Delta^{*}Y_{ik}(t) \\ \leqslant& \left(Y_{ik}^{M+}(t) + \epsilon_{ik}\big(\tau_i(t)\big) + \big(t - \tau_i(t)\big) \cdot 2ND\right) \cdot 2ND \\ \leqslant& \big(4L(t) + 2\big) \cdot N^2D^2 + 2ND\left|\epsilon_{ik}\big(\tau_i(t)\big)\right|, \end{aligned} \tag{32}$$

where the first inequality utilizes Lemma 8.

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

$$\tilde{Y}_{ik}^{M+}(t)\Delta^{*}Y_{ik}(t) \leqslant 0. \tag{33}$$

By combining (32) and (33), we have

$$\sum_{t=0}^{T-1}\tilde{Y}_{ik}^{M+}(t)\Delta^{*}Y_{ik}(t) \leqslant 4N^2D^2\sum_{t=0}^{T-1}L(t) + 2N^2D^2T + 2ND\sum_{t=0}^{T-1}\left|\epsilon_{ik}\big(\tau_i(t)\big)\right|,$$

which completes the proof.

I PROOF OF THEOREM 7

We assume each link in the system has a capacity of 1. The policy taken by node 2 is $\mu_{2d}(t) \equiv 1$. We reinforce to assume that we estimate $Q_2(t)$ at each time slot, with the estimation defined to be $\tilde{Q}_2(t)$. The estimation error $\epsilon_2(t) = 2t$ and gives us $\tilde{Q}_2(t) = [Q_2(t) - \epsilon_2(t)]^{+}$.

We assume that there exists a state-based policy $\pi_a : (Q_1, \tilde{Q}_2) \to (f_{1d}^{a}, f_{12}^{a})$ that could stablize any arrival process inside the stability region.

Case 1: Let

$$a_1(t) \equiv 2, \qquad a_2(t) \equiv 0.$$

It is easy to verify that the dynamics are within the stability region by taking $f_{1d}^{*}(t) = f_{12}^{*}(t) \equiv 1$. Since π_a could stabilize the system and $C_{12} = C_{1d} = 1$, we have that under π_a,

$$\lim_{T\to\infty}\frac{\sum_{t=0}^{T}\tilde{f}_{12}(Q_1(t), 0)}{T} = 1.$$

By the definition of limit, we have that there exists a finite constant T_0 such for each $T \geqslant T_0$, we have

$$\frac{\sum_{t=0}^{T} \tilde{f}_{12}(Q_1(t), 0)}{T} \geqslant \frac{1}{2},$$

or equivalently (to help the writing for case 2), for each $T \geqslant 2T_0$, we have

$$\frac{\sum_{t=0}^{T/2-1} \tilde{f}_{12}(Q_1(t), 0)}{T/2} \geqslant \frac{1}{2}. \tag{34}$$

Case 2: Let

$$a_1(t) = \begin{cases} 2, & t = 0, \cdots, T/2-1 \\ 0, & t = T/2, \cdots, T-1 \end{cases}, \qquad a_2(t) \equiv 0.$$

The traffic load is lighter than case 1, which naturally implies that the arrival process is inside the stability region.

During the first $T/2$ time slots, the arrival process of case 2 is identical as case 1 and thus (34) also holds for case 2. Therefore, we have that under π_a, for each $T \geqslant 2T_0$,

$$\frac{\sum_{t=0}^{T-1} \tilde{f}_{12}(Q_1(t), 0)}{T} \geqslant \frac{\sum_{t=0}^{T/2-1} \tilde{f}_{12}(Q_1(t), 0)}{T} \geqslant \frac{1}{4}. \tag{35}$$

Case 3: Let

$$a_1(t) = \begin{cases} 2, & t = 0, \cdots, T/2-1 \\ 0, & t = T/2, \cdots, T-1 \end{cases}, \qquad a_2(t) \equiv 1.$$

It is easy to verify that the dynamics are still within the stability region by taking $f_{1d}^*(t) \equiv 1$ and $f_{12}^*(t) \equiv 0$.

Since for every time slot, there are at most 2 external packets into the system, we have $\hat{Q}_2(t) \equiv 0$. Moreover, a_1 has the same pattern as case 2. Thus, for the network controller, the system "looks" exactly the same as case 2 and (35) holds for case 3. Therefore, for each $T \geqslant 2T_0$, the average input rate to node 2 amounts to at least $5/4$, which exceeds C_{2d} and leads Q_2 to instability.