Optimal Control for Networks with Unobservable Malicious Nodes

Bai Liu
Massachusetts Institute of Technology
bailiu@mit.edu

Eytan Modiano
Massachusetts Institute of Technology
modiano@mit.edu

With the rapid development of information technology, modern network systems are becoming increasingly complex and are increasingly vulnerable to attacks such as Distributed Denial-of-Service (DDoS) attack. However, existing network control algorithms either require full observability and/or controllability for all nodes [2, 9, 1, 4, 6], or the network dynamics to be time-invariant and stochastic [7, 3, 8, 5]. 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 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, which 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. 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 $\pi = \left\{f_{ijk}(t)\right\}_{i\in\mathcal{A}, j\in\mathcal{N}, k\in\mathcal{K}, 0\leqslant t\leqslant T-1}$. For a malicious node $i \in \mathcal{M}$, the network controller cannot directly observe $Q_{ik}(t)$ or implement control policies. Instead, the network controller only has an estimate $\hat{Q}_{ik}(t)$ of queue backlog $Q_{ik}(t)$ for $t \in \Gamma_i$. In addition to not being observable, malicious nodes are controlled by an adversary. The actions taken by the adversary can be a function of the history up to time $t-1$. Our goal is to determine a policy π 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 $\mathcal{M}$.

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*

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

To characterize the power of the adversary, we use the concept of maliciousness metrics, which place constraints on the sequence of possible network events $\left\{\boldsymbol{a}(t), \boldsymbol{\mu}(t)\right\}_{0\leqslant t\leqslant T-1}$. Existing maliciousness metrics include the W constraint [2] and the V_T constraint [4]. We propose a more relaxed maliciousness metric called the Q_T constraint.

Definition 2. *A network event sequence is Q_T-constrained if under this network event sequence, the following holds*

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

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

Definition 3. *A network is said to have $W/V_T/Q_T$-constrained dynamics if all generated network event sequences are $W/V_T/Q_T$-constrained, respectively.*

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 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. 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. For a malicious node $i \in \mathcal{M}$, its queue backlog may 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. 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.

For each time slot, the network controller observes the queue backlogs of all accessible nodes $i \in \mathcal{A}$. When $t \in \Gamma_i$, the network controller obtains an estimate $\hat{Q}_{ik}(t)$ of the queue backlog $Q_{ik}(t)$ for the malicious node i and updates the estimate of $Y_{ik}(t)$ as $\hat{Y}_{ik}(t) = \hat{Q}_{ik}(t) - X_{ik}(t)$. When $t \notin \Gamma_i$, $\hat{Y}_{ik}(t)$ remains unchanged. The network controller then solves the optimization problem of (1) and applies $\boldsymbol{f}^M(t)$ to the accessible nodes in the real network. The network

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.

$$\boldsymbol{f}^M(t), \boldsymbol{g}^M(t) = \underset{0 \leqslant f_{ijk}, g_{ijk} \leqslant C_{ij}}{\operatorname{argmin}} \sum_{i \in \mathcal{A}, k \in \mathcal{K}} Q_{ik}(t) \left[\sum_{j \in \mathcal{A}} f_{jik} - \sum_{j \in \mathcal{N}} f_{ijk} \right] + \sum_{i \in \mathcal{M}, k \in \mathcal{K}} X_{ik}(t) \left[\sum_{j \in \mathcal{A}} f_{jik} + \sum_{j \in \mathcal{M}} g_{jik} - \sum_{j \in \mathcal{N}} g_{ijk} \right] - \sum_{i \in \mathcal{M}, k \in \mathcal{K}} \max\{\hat{Y}_{ik}(t), 0\} \cdot \left[\sum_{j \in \mathcal{M}} g_{jik} - \min\left\{ \sum_{j \in \mathcal{N}} g_{ijk}, X_{ik}(t) + a_{ik}(t) \right\} \right]. \tag{1}$$

controller uses both $\boldsymbol{f}^M(t)$ and $\boldsymbol{g}^M(t)$ to update $X_{ik}(t)$ for all malicious nodes $i \in \mathcal{M}$.

For a malicious node $i \in \mathcal{M}$, data class $k \in \mathcal{K}$ and $t \in \Gamma_i$, we define the error as $\epsilon_{ik}(t) \triangleq \hat{Q}_{ik}(t) - Q_{ik}(t)$. We define $L(t)$ to be the maximum delay in observations at t, i.e., $\max_{i \in \mathcal{M}, k \in \mathcal{K}} t - \tau_i(t)$, where $\tau_i(t)$ denotes the most recent time when an estimate of node i is made. We show that

Theorem 1. *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 $i \in \mathcal{M}$ and $k \in \mathcal{K}$.*

By Definition 2, 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, for a network with Q_T-constrained dynamics, MWUM is a throughput-optimal algorithm.

We implement MWUM in a 12-node network, as in Figure 1. Node 2, 3, 4 and 6 are unobservable and malicious, while the rest are accessible. All links have the capacity of 5.

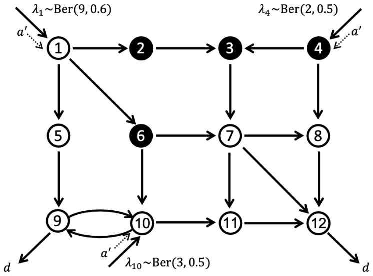
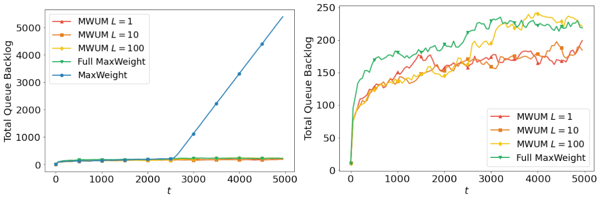


Figure 1: 12-node network model.

At each time slot, 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 simulation results are shown in Figure 2, from which we can see that directly applying the traditional MaxWeight algorithm cannot stabilize the network, while MWUM stabilizes the network.



(a) All policies. (b) Stabilizing policies.

Figure 2: The growth of total queue backlog.

REFERENCES

[1] M. Andrews, B. Awerbuch, A. Fernández, T. Leighton, Z. Liu, and J. Kleinberg. Universal-stability results and performance bounds for greedy contention-resolution protocols. *Journal of the ACM (JACM)*, 48(1):39–69, 2001.

[2] A. Borodin, J. Kleinberg, P. Raghavan, M. Sudan, and D. P. Williamson. Adversarial queueing theory. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 376–385, 1996.

[3] N. M. Jones, G. S. Paschos, B. Shrader, and E. Modiano. An overlay architecture for throughput optimal multipath routing. *IEEE/ACM Transactions on Networking*, 25(5):2615–2628, 2017.

[4] Q. Liang and E. Modiano. Minimizing queue length regret under adversarial network models. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2(1):1–32, 2018.

[5] Q. Liang and E. Modiano. Optimal network control in partially-controllable networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 397–405. IEEE, 2019.

[6] Q. Liang and E. Modiano. Optimal network control with adversarial uncontrollable nodes. In *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 101–110, 2019.

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

[8] A. Rai, R. Singh, and E. Modiano. A distributed algorithm for throughput optimal routing in overlay networks. In *2019 IFIP Networking Conference (IFIP Networking)*, pages 1–9. IEEE, 2019.

[9] P. Tsaparas. *Stability in adversarial queueing theory.* PhD thesis, National Library of Canada= Bibliothèque nationale du Canada.