Spatial-Temporal Routing for Supporting
End to End Hard Deadlines in Multi-hop Networks

by

Weichang Wang

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved April 2018 by the
Graduate Supervisory Committee:

Lei Ying, Chair
Junshan Zhang
Ahmed Ewaisha

ARIZONA STATE UNIVERSITY

May 2018

ABSTRACT

We consider the problem of routing packets with end-to-end hard deadlines in multihop communication networks. This is a challenging problem due to the complex spatial-temporal correlation among flows with different deadlines especially when significant traffic fluctuation exists. To tackle this problem, based on the spatial-temporal routing algorithm that specifies where and when a packet should be routed using concepts of virtual links and virtual routes, we proposed a constrained resource-pooling heuristic into the spatial-temporal routing, which enhances the “work-conserving” capability and improves the delivery ratio. Our extensive simulations show that the policies improve the performance of spatial-temporal routing algorithm and outperform traditional policies such as backpressure and earliest-deadline-first (EDF) for more general traffic flows in multihop communication networks.

ACKNOWLEDGMENTS

I would first like to express my sincere gratitude to my advisor Dr. Lei Ying for his support of my Master study and research, for his valuable guidance during the past year. He is very patient and always there to help me. I am fortunate to have the opportunity to work with him. His rigorous attitude and diligence on study inspire me with my life and study.

Besides my advisor, I would like to express my gratitude to my other thesis committee members: Dr. Junshan Zhang and Dr. Ahmed Ewaisha for their time and valuable comments.

Finally, I would like to thank my lab mates, Mr. Xin Liu during the course of this project, and Mr. Zhen Chen, Mr. Dheeraj Narasimha, Mr. Yiqiu Liu, Mr. Pengfei Jiang, Mr. Fnu Hairi, Mr. Honghao Wei, Ms. Qiao Xue, and Mr. Yan Xiong, for their kind help on both study and life.

TABLE OF CONTENTS

	Page
LIST OF FIGURES.....	iv
CHAPTER	
1 INTRODUCTION.....	1
2 NETWORK MODEL AND INTUITION.....	4
2.1 Network Model.....	4
2.2 Virtual Links/Routes and Spatial-temporal Routing.....	5
3 THROUGHPUT REGION AND OPTIMIZATION FRAMEWORK.....	9
3.1 Throughput Region for Communication Networks with Hard Dead- lines.....	9
3.2 Optimization Framework.....	11
3.2.1 Dual Decomposition.....	12
4 SPATIAL-TEMPORAL ROUTING.....	16
4.1 Virtual Queue Architecture.....	16
4.2 Spatial-temporal Backpressure Policy.....	18
4.2.1 Spatial-temporal Water-filling Policy.....	21
4.3 A Constrained Resource-pooling Heuristic.....	24
5 SIMULATIONS.....	28
5.1 First Scenario: Symmetric Topology.....	28
5.2 Second Scenario: Video Transmission with Abilene Network.....	32
6 CONCLUSIONS.....	34
REFERENCES.....	35
BIOGRAPHICAL SKETCH.....	35

LIST OF FIGURES

Figure		Page
2.1	Toy Example for Flow	5
2.2	Illustration of Virtual Routes and Virtual Links	6
2.3	Toy Example: Case 1	7
2.4	Toy Example: Case 2	8
4.1	Illustration of the Virtual Queue Architecture	17
4.2	Main Workflow in Spatial-temporal Routing	18
4.3	Throughput Performance of ST Routing Policy under a Periodic and General Traffic Pattern	24
4.4	Illustration of a Constrained Resource-pooling Heuristic	25
4.5	Illustration of Downside on an “Intuitive” Heuristic	26
5.1	Symmetric Topology	29
5.2	Periodic Traffic Throughput Performance	30
5.3	Periodic Traffic Average Delay Performance	30
5.4	Gen Traffic Throughput Performance	31
5.5	General Traffic Average Delay Performance	32
5.6	Abilene Topology	32
5.7	Throughput Performance of Different Routing Policies for Abilien Net- work	33

INTRODUCTION

Network applications, such as emergency messages, voice calls and video streaming, demand reliable and predictable transmissions over multi-hop communication networks. Despite remarkable progress on the design of communication networks with maximum throughput and low delays, routing packets with end-to-end hard deadlines in multi-hop communication networks remains to be a challenging problem. When end-to-end hard deadlines are imposed on packets, routing decisions at each hop are coupled with the decisions in the sequential hops. Each link faces the dilemma of allocating its capacity to flows with large backlogs or to flows with short deadlines.

This challenging problem has been studied under several settings in the literature. Li and Eryilmaz (2012) studied communication with per-packet deadline constraints in a multi-hop network and proposed a novel queue structure to dynamically adjust the service discipline to meet end-to-end deadlines. However, Li and Eryilmaz (2012) does not establish any performance guarantee of the proposed resource allocation policies. Hou (2015) proposed a throughput optimal policy for tree networks with a single sink node (i.e., a single destination) and frame-based traffic flows. Mao *et al.* (2014) presented an online algorithm with throughput guarantees for general traffic flows and networks, where the competitive ratio is inversely proportional to the hops of the longest route. More recently, Singh and Kumar (2015) developed a maximal throughput scheduling policy for multi-hop wireless networks in which links do not interfere with each other and the transmit power can be adjusted. A common assumption made in these work is that they all consider static routing and assume each flow is associated with a prefixed route, so routing packets with end-to-end

deadlines remains to be a problem that has not been well studied in the literature.

Liu and Ying (2016) considers dynamic routing and focuses on routing packets with end-to-end hard deadlines in multi-hop communication networks. In this work, no prefixed route is given and a packet is allowed to use any route along which it can reach its destination before the deadline expires. In order to guarantee end-to-end hard deadlines, a routing algorithm not only needs to decide “where” to send the packet (the next hop) but also “when” to send the packet (the time slot the packet should be transmitted). To answer the “when” question, a routing algorithm needs to control the timing of transmitting each packet. To incorporate this temporal perspective, they introduced the concept of *virtual links*, where each virtual link represents the use of a physical link at a specific time slot (within a frame, the concept of frame will be introduced later). In a network consisting of virtual links, a route (called virtual route) for a packet specifies both the spatial path (the sequence of nodes to traverse) to the destination but also the time slot each transmission should occur so that the packet will reach the destination before the deadline expires. These two novel concepts, virtual links and virtual routes, enable this work to characterize the complicated spatial-temporal correlation of packets from different flows and with different deadlines.

Based on Liu and Ying (2016), we propose a new heuristic improved algorithm which maintains the optimality and stability. Based on that, a resource-pooling heuristic improvement method, which using the spirit of “work-conserving”, is developed to dynamically balance real traffic flows among all possible virtual routes. The main contributions of my thesis are summarized below.

- Based on the spatial-temporal routing proposed in Liu and Ying (2016), we propose a resource-pooling heuristic improvement method that can significantly improve its performance when there exists significant traffic fluctuation. The

algorithm enhances the “work-conserving” capability and improves the delivery ratio and delay.

- The improved routing does not require additional information from other nodes compared to the spatial-temporal routing, and the computational complexity is small. Also, it can be proved that the throughput of the improved routing is at least as large as that under the spatial-temporal routing.
- We evaluated the performance of the improved algorithms by numerical simulations. From the simulations, the improved policies significantly outperform backpressure and EDF for frame-based periodic traffic flows and the heuristic supports a higher delivery ratio under general traffic flows.

Chapter 2

NETWORK MODEL AND INTUITION

2.1 Network Model

We consider a multi-hop communication network denoted by a graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{L}$ is the set of links. The network is assumed to be a time-slotted system, where arrivals occur at the beginning of a time slot and departures occur at the end of a time slot. Denote by (a, b) the link from node a to node b and $C_{(a,b)}$ the capacity of the link (i.e., the number of packets that can be transmitted in one time slot). In this thesis, we assume transmissions at different links do not interfere with each other. So the communication network is either a wired network or a wireless network in which neighboring links operate over orthogonal channels. We further assume the capacity of a link is time invariant. Packets with hard deadlines are injected into the network. We assume a frame structure for traffic flows such that every T consecutive time slots are grouped into a frame. Each incoming packet is associated with a destination and a deadline, where the deadline is the time slot (within a frame) by which the packet should be delivered to its destination otherwise the packet will be dropped. The deadline of a packet expires within the same frame in which the packet arrives. This frame-based traffic structure has been commonly assumed in the literature Hou (2015); Hou *et al.* (2009); Jaramillo and Srikant (2011). Traditionally, in a multi-commodity flow problem, a flow is defined by its source and destination. In this thesis, we define a flow using 4-tuple: its source, destination, the relative arrival time in a frame and the relative expiration time in a frame. In particular, for flow f , let $s(f)$ denote its source, $d(f)$ denote its destination, $t_b(f)$ the

relative arrival time (beginning time) of the packets belonging to flow f and $t_e(f)$ the relative expiration time (end time) of the packets of flow f . Therefore, a flow is not only defined in space (by its source and destination) and also defined over time (by the beginning and end time slots). Denote by $\mathcal{F}$ the collection of flows. Consider a simple example in Fig. 2.1, where the frame size is two. Two flows are defined in the figure, where packets of flow f_1 arrive at node a at the first time slot of a frame and need to be delivered to node c at the end of the frame, and packets of f_2 arrive at node a at the first time slot of a frame and needs to be delivered to node b at the end of the frame. We further assume periodic traffic arrivals such that for each flow, the arrival

Figure 2.1: Illustration of Flows

pattern within each frame remains the same across frames. This models applications such as real-time surveillance systems where remote sensors collect and report data periodically. The periodic traffic model also serves a stepstone to study the general traffic, which we will discuss after introducing the spatial-temporal routing policies for the periodic traffic model.

2.2 Virtual Links/Routes and Spatial-temporal Routing

Traditionally, the throughput region of a communication network is defined based on link capacity, and a set of flows are said to be within the network throughput region if there exists a resource allocation algorithm under which the long-term average throughput is equal to the arrival rate of each flow. So the traditional resource allocation in communication networks concerns the “spatial allocation” (the allocation

of link capacity across flows). For packets with hard deadlines, resource allocation across time becomes critical. In this thesis, we aim at incorporating hard deadline constraints into the characterization of the throughput region. One key step is to introduce two novel ideas: virtual links and virtual routes. The concept of a virtual route expands the traditional route to the temporal domain and is defined to be an $N \times T$ matrix $\mathbf{R}$ such that $R(n, t) = 1$ denotes the route traverses node n at the t th time slot of a frame. A virtual route specifies not only which links to use but when to use them, so can be used to control the end-to-end deadline of transmitting a packet.

Like a virtual route, a virtual link expands the resource of a physical link to a spatial-temporal domain. Denote by $\{(a, b), t\}$ a virtual link, which represents link (a, b) at the t th time slot of a frame. Fig. 2.2 illustrates the concepts of virtual links and virtual routes using the toy example in the previous section. Since the frame size is two, each physical link is represented by two virtual links. For packets of flow f_1 , it has to take the virtual route

$$\{(a, b), 1\} \rightarrow \{(b, c), 2\}$$

to reach destination c by the end of time slot 2. For packets belonging to flow f_2 , both virtual routes $\{(a, b), 1\}$ and $\{(a, b), 2\}$ are feasible routes.

Figure 2.2: Illustration of virtual routes and virtual links. Blue and red lines represent virtual routes for flows f_1 and f_2 , respectively. Black lines represent virtual links.

Now to achieve the full throughput region in the network, a resource allocation

algorithm needs to 1) identify the set of feasible virtual routes for each flow and 2) allocate the load appropriately on the set of routes under the network capacity constraints. Let us assume the capacity of both links are 10 packets/time slot in the toy example. We consider the following two different cases.

- Case 1: Flow f_1 has a periodic arrival of 5 packets at the first time slot of each frame; and flow f_2 has a periodic arrival of 10 packets at the first time slot of each frame. One of the feasible routing solutions is shown in Fig. 2.3, where flow f_2 splits its traffic evenly among the two virtual routes $\{(a, b), 1\}$ and $\{(a, b), 2\}$.
- Case 2: Flow f_1 has a periodic arrival of 10 packets at the first time slot of each frame; and flow f_2 has a periodic arrival of 10 packets at the first time slot of each frame. In this case, the unique routing solution to support the traffic is to have flow f_2 use the unique virtual route $\{(a, b), 2\}$ and flow f_1 use virtual route $\{(a, b), 1\} \rightarrow \{(b, c), 2\}$ as shown in Fig. 2.4.

Figure 2.3: Case 1

The challenge of routing packets with end-to-end hard deadline constraints is that even without introducing virtual routes, the number of possible routes in a network is exponential in the number of links. After representing each physical link with T virtual links, the number of virtual routes becomes even larger. To overcome this difficulty, we exploit the idea in Ying *et al.* (2011), which includes both dynamical

Figure 2.4: Case 2

routing and load balancing in a communication network without requiring per route information.

One of the algorithms proposed in Ying *et al.* (2011) achieves throughput optimality while guaranteeing hop constraints (a constraint on the number of hops a packet is allowed to travel before reaching its destination). A critical component of the algorithm is to maintain per-destination and per-hop queues and to only allow packets to be transmitted to neighboring nodes who can meet the hop constraints. For packets with hard deadlines, with the concept of virtual links and virtual routes, our proposed policies will maintain per-destination, per-deadline and per-time-slot virtual queues, and only allow a packet to be transmitted to virtual links that guarantees the feasibility of delivering the packet before its deadline. Furthermore, our policies are completely distributed. The design of the policies and the performance analysis will be presented in the following sections.

THROUGHPUT REGION AND OPTIMIZATION FRAMEWORK

3.1 Throughput Region for Communication Networks with Hard Deadlines

For traffic flows with hard deadlines, the network throughput region depends on both link capacity and the distributions of traffic arrivals and deadlines. Assuming periodic traffic flows with a frame structure, we next characterize the network throughput region using flow conservation constraints and virtual link capacity constraints based on the necessary conditions for virtual commodities. We define $\mathcal{D}_{\{d,t_e\}} = \{f : d(f) = d, t_e(f) = t_e\}$ to be virtual commodity $\{d, t_e\}$ such that a packet of virtual commodity $\{d, t_e\}$ has to reach the destination d by the end of the t_e th time slot in a frame. The following necessary condition has to be satisfied for each virtual commodity, each node and each time slot in a frame.

$$\begin{aligned} & \sum_{f \in \mathcal{D}_{\{d,t_e\}}} a_f 1_{s(f)=n, t_b(f)=t} + \sum_{a:(a,n) \in \mathcal{L}} \sum_{i=1}^{t-1} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (n,t-1)\}} \\ &= \sum_{b:(n,b) \in \mathcal{L}} \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} u_{\{d,t_e\}}^{\{(n,t) \rightarrow (b,i)\}}, \forall d, n \in \mathcal{N}, t_e, t \in [1, T]. \end{aligned} \quad (3.1)$$

In the condition above, $1_{s(f)=n, t_b(f)=t} = 1$ indicates flow f is injected into node n at the beginning of the t th time slot and a_f is the arrival rate of flow f ; $h_{b \rightarrow d}^{\min}$ denotes the minimum number of hops from node b to node d , so $t_e - h_{b \rightarrow d}^{\min}$ is the maximum number of time slots a packet can be hold at node n before being transmitted to node b so that the packet is still feasible to reach its destination before the deadline expires; $u_{\{d,t_e\}}^{\{(a,i) \rightarrow (n,t)\}}$ denotes the number of packets that arrive at node a at the beginning of the i th time slot and are transmitted to node n at the end of the t th time slot. The

flow conservation constraint (3.1) basically states that the incoming packets of virtual commodity $\{d, t_e\}$ at node n at the beginning of the t th time slot should be sent to node n 's neighbors in the subsequent time slots while guaranteeing the feasibility of delivering them before their deadlines.

We further have the following necessary condition due to link capacity constraint. Note that all traffics $\{(a, i) \rightarrow (n, t)\}$ uses virtual link $\{(a, n), t\}$, so we have the following capacity constraint

$$\sum_{\{d, t_e\} \in \mathcal{D}} \sum_{i=1}^t u_{\{d, t_e\}}^{\{(a, i) \rightarrow (n, t)\}} \leq C_{\{(a, n), t\}}, \quad (a, n) \in \mathcal{L}, \quad t \in [1, T], \quad (3.2)$$

where $C_{\{(a, n), t\}} = C_{(a, n)}$ for any t . Moreover, we define the link capacity region to be $\mathcal{C} = \{\mathbf{u} \mid \mathbf{u} \text{ satisfies (3.2)}\}$, where $\mathbf{u}$ is the vector version of transmission rates.

Now given traffic $\mathbf{A} = \{a_f\}_{f \in \mathcal{F}}$ with deadline constraint $\mathbf{D} = \{t_b(f), t_e(f)\}_{f \in \mathcal{F}}$, we can define the throughput region by relaxing the integer constraint on $\mathbf{u}$ to be

$$\Omega = \{(\mathbf{A}, \mathbf{D}) \mid \text{there exists } \mathbf{u} \text{ that satisfies both (3.1) and (3.2)}\}.$$

Note Ω can be viewed as an outer bound of the “true” throughput region where integer constraints are imposed on $\mathbf{u}$. Then we define a routing policy as follows.

Definition 1. A routing policy is a set of rates allocation for each virtual commodity on each link given the history and current state information.

Further, we give the following definition of and theorem.

Definition 2. The arrival traffic $(\mathbf{A}, \mathbf{D})$ is *supportable* by a routing policy if the packet dropping rate converges to zero as $t \rightarrow \infty$.

Theorem 1. No routing policy can support an arrival traffic $(\mathbf{A}, \mathbf{D}) \notin \Omega$.

Proof. To prove this theorem, we are going to prove that the policy π converges to $\mathbf{u}$ when either flow conservation constraint (3.1) or link capacity constraint (3.2) does

not hold and the long-term average packet dropping rate are zero for all flows does not exist.

Intuitively, the violation of capacity constraint (3.2) on one link will lead to the violation of flow conservation constraint (3.1). If (3.1) does not hold, which means the some incoming packets of virtual commodity $\{d, t_e\}$ at node n at the beginning of the t th time slot are not sent to node n 's neighbors in the subsequent time slots. It contradicts the assumption that under π , the packet dropping rates are zero for all flows.

See Liu and Ying (2016) for detailed proof. $\square$

3.2 Optimization Framework

After defining throughput region Ω , we consider the following optimization problem whose goal is to reduce the average latency while guaranteeing hard deadline constraints

$$\min \sum_{\{d, t_e\} \in \mathcal{D}} \sum_{a=1}^N \sum_{b=1}^N \sum_{t=1}^T \sum_{i=t}^T V \cdot (i - t) U \left(u_{\{d, t_e\}}^{\{(a, t) \rightarrow (b, i)\}} \right) \quad (3.3a)$$

$$\text{s.t. } \mathbf{u} \in \Omega. \quad (3.3b)$$

In the optimization problem, V is the tuning parameter to control the tradeoff between the optimality of the steady-state resource allocation solution and the convergence rate. $(i - t)U(\cdot)$ is a convex utility function of transmission rate $u_{\{d, t_e\}}^{\{(a, t) \rightarrow (b, i)\}}$. The problem above (3.3) is a convex optimization since both objective function and Ω are convex. When $U(\cdot) = 0$, the optimization problem (3.3) is formulated to find a feasible routing policy to support end-to-end deadlines. When $U(\cdot)$ is a strictly increasing and convex function, the optimization problem (3.3) has a unique optimal solution. In our optimization framework, we add a weight $(i - t)$ to the utility on $u_{\{d, t_e\}}^{\{(a, t) \rightarrow (b, i)\}}$ so the policy routes more packets of virtual commodity $\{d, t_e\}$ along $\{t, i\}$

instead of $\{t, i + 1\}$ whenever possible to further reduce end-to-end latency.

3.2.1 Dual Decomposition

To derive a distributed routing policy from the optimization problem (3.3), we keep link constraint (3.2) and define $\lambda_{\{n,t,d,t_e\}}$ to be the Lagrangian multiplier associated with flow conservation constraint (3.1). We obtain the following partial Lagrangian function Boyd and Vandenberghe (2004):

$$\begin{aligned}
& \mathcal{L}(\boldsymbol{\lambda}, \mathbf{u}) \\
&= \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{n=1}^N \sum_{t=1}^T \lambda_{\{n,t,d,t_e\}} \left(\sum_{f \in \mathcal{D}_{\{d,t_e\}}} a_f 1_{s(f)=n, t_b(f)=t} \right. \\
&\quad \left. + \sum_{a:(a,n) \in \mathcal{L}} \sum_{i=1}^{t-1} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (n,t-1)\}} - \sum_{b:(n,b) \in \mathcal{L}} \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} u_{\{d,t_e\}}^{\{(n,t) \rightarrow (b,i)\}} \right) \\
&\quad + \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{a=1}^N \sum_{b=1}^N \sum_{t=1}^T \sum_{i=t}^T V \cdot (i - t) U \left(u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right) \\
&= \underbrace{\sum_{\{d,t_e\} \in \mathcal{D}} \sum_{n=1}^N \sum_{t=1}^T \lambda_{\{n,t,d,t_e\}} \sum_{f \in \mathcal{D}_{\{d,t_e\}}} a_f 1_{s(f)=n, t_b(f)=t}}_X \\
&\quad + \underbrace{\sum_{\{d,t_e\} \in \mathcal{D}} \sum_{a=1}^N \sum_{b=1}^N \sum_{t=1}^T \sum_{i=t}^T V \cdot (i - t) U \left(u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right)}_Y \\
&\quad - \underbrace{\sum_{\{d,t_e\} \in \mathcal{D}} \sum_{n=1}^N \sum_{t=1}^T \lambda_{\{n,t,d,t_e\}} \left(\sum_{b:(n,b) \in \mathcal{L}} \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} u_{\{d,t_e\}}^{\{(n,t) \rightarrow (b,i)\}} \right.} \\
&\quad \left. - \sum_{a:(a,n) \in \mathcal{L}} \sum_{i=1}^{t-1} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (n,t-1)\}} \right)}_Z
\end{aligned}$$

When λ is given, term X is fixed, and we decompose term Z as follows:

$$\begin{aligned}
Z &\stackrel{(a)}{=} \sum_{n=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{t=1}^T \lambda_{\{n,t,d,t_e\}} \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} u_{\{d,t_e\}}^{\{(n,t) \rightarrow (b,i)\}} \\
&\quad - \sum_{a=1}^N \sum_{n=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{t=1}^T \lambda_{\{n,t,d,t_e\}} \sum_{i=1}^{t-1} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (n,t-1)\}} \\
&\stackrel{(b)}{=} \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \left(\sum_{t=1}^T \lambda_{\{a,t,d,t_e\}} \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right. \\
&\quad \left. - \sum_{t=1}^T \lambda_{\{b,t,d,t_e\}} \sum_{i=1}^{t-1} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t-1)\}} \right) \\
&= \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \left(\sum_{t=1}^T \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} \lambda_{\{a,t,d,t_e\}} u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right. \\
&\quad \left. - \sum_{t=1}^T \sum_{i=1}^{t-1} \lambda_{\{b,t,d,t_e\}} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t-1)\}} \right) \\
&\stackrel{(c)}{=} \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \left(\sum_{t=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} \lambda_{\{a,t,d,t_e\}} u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right. \\
&\quad \left. - \sum_{t=2}^{t_e - h_{b \rightarrow d}^{\min} + 1} \sum_{i=1}^{t-1} \lambda_{\{b,t,d,t_e\}} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t-1)\}} \right) \\
&\stackrel{(d)}{=} \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \left(\sum_{i=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{t=1}^i \lambda_{\{a,t,d,t_e\}} u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right. \\
&\quad \left. - \sum_{t=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{i=1}^t \lambda_{\{b,t+1,d,t_e\}} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right) \\
&\stackrel{(e)}{=} \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \left(\sum_{t=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{i=1}^t \lambda_{\{a,i,d,t_e\}} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right. \\
&\quad \left. - \sum_{t=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{i=1}^t \lambda_{\{b,t+1,d,t_e\}} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right) \\
&= \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{t=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{i=1}^t u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \left(\lambda_{\{a,i,d,t_e\}} \right. \\
&\quad \left. - \lambda_{\{b,t+1,d,t_e\}} \right)
\end{aligned} \tag{3.4}$$

where (a) is obtained by interchanging the summations of commodities and links; (b) holds by changing notation n to a and b in the first and second terms respectively; (c) holds by restricting t to the feasible values for both terms; (d) holds by interchanging the summations of t and i in the first term and changing variable $t - 1$ to t in the second term; (e) holds by exchanging notation t and i in the first term;

Similar to steps (a), (c) and (d) above, by restricting t to feasible values and interchanging the summations of t and i in term Y , we obtain

$$\begin{aligned}
Y &= \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{t=1}^T \sum_{i=t}^T V \cdot (i - t) U \left(u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right) \\
&= \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{i=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{t=1}^i V \cdot (i - t) U \left(u_{\{d,t_e\}}^{\{(a,t) \rightarrow (b,i)\}} \right) \\
&= \sum_{a=1}^N \sum_{b=1}^N \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{t=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{i=1}^t V \cdot (t - i) U \left(u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right) \tag{3.5}
\end{aligned}$$

Given λ , we then consider the following problem:

$$\min \mathcal{L}(\lambda, \mathbf{u}) \tag{3.6a}$$

$$\text{s.t. } \mathbf{u} \in \mathcal{C}. \tag{3.6b}$$

After substituting (3.4) and (3.5) into (3.6a), we decouple objective (3.6a) into individual optimization problems for each link (a, b) as follows:

$$\begin{aligned}
\max \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{t=1}^{t_e - h_{b \rightarrow d}^{\min}} \sum_{i=1}^t &\left(\lambda_{\{a,i,d,t_e\}} - \lambda_{\{b,t+1,d,t_e\}} \right) u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \\
&+ V \cdot (i - t) U \left(u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right) \tag{3.7a}
\end{aligned}$$

$$\begin{aligned}
\text{s.t. } \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{i=1}^t &u_{\{d,t_e\}}^{\{(a,i) \rightarrow (n,t)\}} \leq C_{\{(a,n),t\}}, \\
\forall (a, b) \in \mathcal{L}, \quad &t \in [1, T]. \tag{3.7b}
\end{aligned}$$

Further, (3.7) is decoupled into sub-optimization problems at each time slot in a frame

$$\begin{aligned} \max \quad & \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{i=1}^t \left(\lambda_{\{a,i,d,t_e\}} - \lambda_{\{b,t+1,d,t_e\}} \right) u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \\ & + V \cdot (i - t) U \left(u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right) \end{aligned} \quad (3.8a)$$

$$\begin{aligned} \text{s.t.} \quad & \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{i=1}^t u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \leq C_{\{(a,b),t\}}, \\ & \forall (a,b) \in \mathcal{L}, t \in [1, T]. \end{aligned} \quad (3.8b)$$

Now the original problem (3.3) has been decoupled into individual sub-optimization problem (3.8) for each link at each time slot in a frame. Based on the reduced formulation (3.8), we then develop two fully-distributed routing policies for different choices of $U(\cdot)$.

SPATIAL-TEMPORAL ROUTING

Motivated by the connection between Lagrangian dual variables and queue lengths (page 145 to 149) Srikant and Ying (2014), we first introduce the virtual queue architecture.

4.1 Virtual Queue Architecture

Let $d_{\{n,t,d,t_e\}}[k]$ denote virtual queue associated with the Lagrangian dual variable $\lambda_{\{n,d,t_e,t\}}$ which updates as

$$\begin{aligned} d_{\{n,t,d,t_e\}}[k+1] = & \left(d_{\{n,t,d,t_e\}}[k] + \sum_{f \in \mathcal{D}_{\{d,t_e\}}} A_f[k] 1_{s(f)=n, t_b(f)=t} \right. \\ & \left. + \sum_{a:(a,n) \in \mathcal{L}} \sum_{i=1}^{t-1} u_{\{d,t_e\}}^{\{(a,i) \rightarrow (n,t-1)\}}[k] - \sum_{b:(n,b) \in \mathcal{L}} \sum_{i=t}^{t_e - h_{b \rightarrow d}^{\min}} u_{\{d,t_e\}}^{\{(n,t) \rightarrow (b,i)\}}[k] \right)^+. \end{aligned} \quad (4.1)$$

Note that $d_{\{n,t,d,t_e\}}[k]$ measures the congestion level of transmitting the commodity $\{d, t_e\}$ at node n at the t th time slot during frame k . We remark that k is the frame index, so the virtual queues are updated once every frame (instead of every time slot). If $d_{\{n,t,d,t_e\}}[k]$ is large, packets belonging to $\{d, t_e\}$ are less likely to get through if being routed to node n at the t th time slot, so should be routed to a different time slot t in the frame. As convention, we assume $d_{\{n, \cdot, n, \cdot\}}[k] = 0$ for $\forall n \in \mathcal{N}$ since packets will be moved to the upper layer after arriving at their destinations. We still use a toy example to illustrate the virtual queue architecture in Fig. 4.1, where packets of commodities $\{c, 2\}$ (blue) and $\{b, 2\}$ (red) are injected into the common source a at the beginning of the first time slot, so node a maintains virtual queue $d_{\{a,1,b,2\}}$ for $\{b, 2\}$ and $d_{\{a,1,c,2\}}$ for $\{c, 2\}$. The virtual queues are counters and do

not hold real packets. Node b maintains $d_{\{b,2,b,2\}}$ and $d_{\{b,3,b,2\}}$ for $\{b,2\}$ and $d_{\{b,2,c,2\}}$ for $\{c,2\}$, since for $\{b,2\}$, packets can be routed along either $d_{\{a,1,b,2\}} \rightarrow d_{\{b,2,b,2\}}$ or $d_{\{a,1,b,2\}} \rightarrow d_{\{b,3,b,2\}}$; and for $\{c,2\}$, $d_{\{a,1,c,2\}} \rightarrow d_{\{b,2,c,2\}}$ is the only feasible route in its first hop. Node c maintains $d_{\{c,3,c,2\}}$ for commodity $\{c,2\}$. As defined, $d_{\{b,2,b,2\}}$, $d_{\{b,3,b,2\}}$ and $d_{\{c,3,c,2\}}$ are always 0. Note in (4.1), we take a slightly perturbed version of the real periodic arrival $A_f[k] = (1 - \theta)a_f + \theta A_f^\theta[k]$ as the virtual arrival into virtual queues, where θ is a small positive number and $A_f^\theta[k]$ is a poisson random variable with unit mean. Then virtual queues $d_{\{n,t,d,t_e\}}[k], \forall n, t, d, t_e$ compose of a Markov chain. The reason to add this “randomness” is to guarantee the existence and uniqueness of the stationary distribution of the virtual queues.

We analyze the complexity of virtual queues as follows: For t outside the range $[\min\{h_{n \rightarrow d}^{\min}, t_e\}, \max\{h_{n \rightarrow d}^{\min}, t_e\}]$, $d_{\{n,t,d,t_e\}} = 0$. So for those values, there is no need to maintain virtual queues. Therefore, the maximum number of virtual queues maintained at node n for the destination d is $O((T - h_{n \rightarrow d}^{\min} + 1)^2)$, which is much smaller than the number of feasible routes in the network.

Figure 4.1: Illustration of the Virtual Queue Architecture

Based on the virtual queue architecture, we propose two spatial-temporal routing policies in the following subsections. Before jumping into details of spatial-temporal routing policies, we illustrate the main workflow in Fig. 4.2. Two types of queues ex-

ist in spatial-temporal routing policies, real queues (real packets) and virtual queues (counters). A_f^{real} and $A_f^{virtual}$ are arrivals into real queues and virtual queues, respectively, where $A_f^{virtual}$ is a slightly perturbed version if A_f^{real} is the periodic arrival; otherwise $A_f^{virtual} = A_f^{real}$. Spatial-temporal routing returns virtual service rates $\bar{\mathbf{u}}$ and real service rates are obtained by rounding operation $\hat{\mathbf{u}} = \lfloor \bar{\mathbf{u}} \rfloor$.

Figure 4.2: Main Workflow in Spatial-temporal Routing

4.2 Spatial-temporal Backpressure Policy

Assuming $U(\cdot) = 0$ in (3.8), we propose the following spatial-temporal backpressure policy in Policy 1.

Denote $Q_{\{d,t_e\}}^t[k]$ to be the number of real packets of $\{d, t_e\}$ at t th time slot in frame k . Spatial-temporal backpressure $w_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t+1)\}}[k]$ is the difference between the values of virtual queues of (a, i) and $(b, t+1)$ for commodity $\{d, t_e\}$, which extends the traditional backpressure to the spatial-temporal domain. In policy 1, virtual link $\{(a, b), t\}$ increases the capacity allocation to the tuple $\{d, t_e, i\}^*$ with the maximum backpressure $w_{\{d,t_e\}}^{\{(a,i^*) \rightarrow (b,t+1)\}}$ (breaking ties arbitrarily). If $Q_{\{d,t_e\}}^t[k] < \hat{u}_{\{d,t_e\}}^{\{(a,b),t\}}[k]$, then all real packets are transmitted. Again, we consider the example in Fig. 4.1. Suppose $a_{f_1} = a_{f_2} = 10$, $d_{\{a,1,c,2\}}[k] = 30$, $d_{\{b,2,c,2\}}[k] = 10$ and $d_{\{a,1,b,2\}}[k] = 10$, then

Policy 1 Spatial-Temporal Backpressure Policy

- 1: For virtual link $\{(a, b), t\}$ at the beginning of the k th frame, calculate spatial-temporal backpressure

$$w_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t+1)\}}[k] = \left(d_{\{a, i, d, t_e\}}[k] - d_{\{b, t+1, d, t_e\}}[k] \right)^+$$

- 2: Compute virtual transmission rate at the beginning of the k th frame

$$u_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t)\}}[k] \in \arg \max_{\mathbf{u} \in \mathcal{C}} \sum_{\{d, t_e\} \in \mathcal{D}} \sum_i w_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t+1)\}}[k] \cdot u_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t)\}}$$

- 3: Compute aggregated transmission rate for $\{d, t_e\}$ on virtual link $\{(a, b), t\}$ by

$$\bar{u}_{\{d, t_e\}}^{\{(a, b), t\}}[k] = \sum_{i=1}^t \bar{u}_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t)\}}[k],$$

where $\bar{u}_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t)\}}[k]$ is the time average of the virtual transmission rate up to the k th frame, i.e.,

$$\bar{u}_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t)\}}[k] = \frac{1}{k} \sum_{i=1}^k u_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t)\}}[i].$$

- 4: Transmit packets of virtual commodity $\{d, t_e\}$ at node a with $\hat{u}_{\{d, t_e\}}^{\{(a, b), t\}}[k] = \left\lfloor \bar{u}_{\{d, t_e\}}^{\{(a, b), t\}}[k] \right\rfloor$ to node b at the t th time slot of the k th frame.
 - 5: Update virtual queues at the end of the k th frame as (4.1).
-

$w_{\{c,2\}}^{\{(a,1) \rightarrow (b,2)\}}[k] = 20$, $w_{\{b,2\}}^{\{(a,1) \rightarrow (b,2)\}}[k] = 10$ and $w_{\{b,2\}}^{\{(a,1) \rightarrow (b,3)\}}[k] = 10$. For ease of exposition, we assume a non-perturbated version of arrival $A_{f_1}[k] = a_{f_1}$ and $A_{f_2}[k] = a_{f_2}$. During the first time slot at the k th frame, as $w_{\{c,2\}}^{\{(a,1) \rightarrow (b,2)\}}[k] > w_{\{b,2\}}^{\{(a,1) \rightarrow (b,2)\}}[k]$, virtual link $\{(a,b),1\}$ will increase its capacity allocation to the commodity $\{c,2\}$. Similarly in the second time slot, virtual link $\{(a,b),2\}$ and $\{(b,c),2\}$ will increase its capacity allocation to $\{b,2\}$ and $\{c,2\}$, respectively. Then at the beginning of the $k+1$ th frame, the values of virtual queues are updated to be $d_{\{a,1,c,2\}}[k+1] = 30$, $d_{\{b,2,c,2\}}[k+1] = 10$ and $d_{\{a,1,b,2\}}[k+1] = 10$, which remain unchanged in the k th frame and real transmission rates $\hat{\mathbf{u}}$ converges to a feasible solution for $a_{f_1} = a_{f_2} = 10$ as shown in Fig. 2.4.

In Policy 1, steps 1, 2 and 5 are related to the virtual queue system and generate the virtual transmission rate; steps 3 and 4 transfer the virtual transmission rate into the real transmission rate by “average”, “aggregate” and “rounding” operations. Note “average” in step 3 is to avoid oscillation and guarantee the real transmission rate converges to a unique solution. Denote η to be the ratio between the link capacity and the packet size. We are ready to present the following theorem.

Theorem 2. Given any traffic with end-to-end deadline constraints such that $((1 + \varepsilon + \delta)\mathbf{A}, \mathbf{D}) \in \Omega$ for some $\varepsilon > 0$ and $\delta = O\left(\frac{1}{\eta}\right)$, $\hat{\mathbf{u}}[k]$ generated by Policy 1 satisfies both conditions (1) and (2) for the given traffic. Policy 1 is near optimal as η becomes sufficiently large.

Proof. Define a Lyapunov function:

$$\mathbf{L}[k] = \frac{1}{2} \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{n=1}^N \sum_{t=1}^T (d_{\{n,t,d,t_e\}}[k])^2$$

such that for a given virtual queue $\mathbf{d}[k]$ at k th frame, Lyapunov drift is

$$\begin{aligned}\delta[k] &= E[\mathbf{L}[k+1] - \mathbf{L}[k] | \mathbf{d}[k] = \mathbf{d}] \\ &\leq B_0 - \varepsilon \sum_{f \in \mathcal{F}} \sum_{n=1}^N \sum_{t=1}^T a_f d_{\{n,t,d,t_e\}}\end{aligned}$$

where B_0 is upper bound defined by capacity C , arrival a_f and network topology. With Foster theorem in Srikant and Ying (2014), we can prove that $\mathbf{d}[k]$ is positive recurrent under Policy 1 for any arrival strictly in Ω . Then, with Renewal Reward Theorem in Gallager (1996), we can get that $\mathbf{u}^*$ is a feasible point due to the fact $(\mathbf{d}[k])$ is positive recurrent.

See Liu and Ying (2016) for detailed proof. $\square$

Spatial-temporal backpressure policy will result in a feasible routing solution whenever possible. The policy exploits all feasible virtual routes. As a result, packets might traverse routes with unnecessary large end-to-end delay. To further improve delay performance, we propose a water-filling policy to reduce end-to-end latency while guaranteeing throughput performance.

4.2.1 Spatial-temporal Water-filling Policy

To solve (3.8), we define v to be the Lagrange dual variable of (3.8b) and derive its Lagrange function

$$\begin{aligned}L(\mathbf{u}, v) &= C_{\{(a,b),t\}} \cdot v + \sum_{\{d,t_e\} \in \mathcal{D}} \sum_{i=1}^t u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \left(\lambda_{\{a,i,d,t_e\}} \right. \\ &\quad \left. - \lambda_{\{b,t+1,d,t_e\}} - v \right) + V \cdot (i-t) U \left(u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right)\end{aligned}$$

According to the KKT conditions Boyd and Vandenberghe (2004), the optimal

solution of the problem satisfies

$$\lambda_{\{a,i,d,t_e\}} - \lambda_{\{b,t+1,d,t_e\}} - v + V(i-t)U' \left(u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} \right) = 0,$$

$$\sum_{\{d,t_e\} \in \mathcal{D}} \sum_{i=1}^t u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}} = C_{\{(a,b),t\}},$$

from which, we derive spatial-temporal water-filling routing in Policy 2.

Policy 2 Spatial-Temporal Water-filling Policy

- 1: For virtual link $\{(a,b),t\}$ at the beginning of the k th frame, calculate virtual transmission rates

$$u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}}[k] = U'^{-1} \left(\frac{(w_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t+1)\}}[k] - v)^+}{V \cdot (t-i)} \right) \quad (4.2)$$

where v satisfies

$$\sum_{\{d,t_e\} \in \mathcal{D}} \sum_{i=1}^t u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}}[k] = C_{\{(a,b),t\}}$$

- 2: Compute aggregated transmission rate for $\{d,t_e\}$ on virtual link $\{(a,b),t\}$ by

$$\bar{u}_{\{d,t_e\}}^{\{(a,b),t\}}[k] = \sum_{i=1}^t \bar{u}_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}}[k]$$

where $\bar{u}_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}}[k]$ is time average of virtual transmission rate up to the k th frame as

$$\bar{u}_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}}[k] = \frac{1}{k} \sum_{i=1}^k u_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t)\}}[i]$$

- 3: Transmit packets of virtual commodity $\{d,t_e\}$ at node a with $\hat{u}_{\{d,t_e\}}^{\{(a,b),t\}}[k] = \left\lceil \bar{u}_{\{d,t_e\}}^{\{(a,b),t\}}[k] \right\rceil$ to node b at the t th time slot of the k th frame.
 - 4: Update virtual queues at the end of the k th frame as (4.1).
-

As suggested in (4.2), virtual transmission rates are allocated in a “water-filling” way, v is regarded to “water level”. This policy scales backpressure $w_{\{d,t_e\}}^{\{(a,i) \rightarrow (b,t+1)\}}$

with inverse of holding time $t - i$ on link (a, b) . Since $U'^{-1}(\cdot)$ is a non-decreasing function, link (a, b) is likely to increase capacity allocation to $\bar{u}_{\{d, t_e\}}^{\{(a, i) \rightarrow (b, t)\}}$ rather than $\bar{u}_{\{d, t_e\}}^{\{(a, j) \rightarrow (b, t)\}}$, $j < i$, which reduces holding time for $\{d, t_e\}$ on link (a, b) , and thus decreases the end-to-end latency for $\{d, t_e\}$. Again, we prove the optimality of spatial-temporal water-filling policy in the following theorem.

Theorem 3. Given any traffic with end-to-end deadline constraints such that $((1 + \varepsilon + \delta)\mathbf{A}, \mathbf{D}) \in \Omega$ for some $\varepsilon > 0$ and $\delta = O\left(\frac{1}{\eta}\right)$, $\hat{\mathbf{u}}[k]$ generated by Policy 2 satisfies both conditions (1) and (2) for the given traffic. Policy 2 achieves the trade-off $[O(1/V), O(V)]$ between the gap to the network utility and average virtual queue length.

Proof. See Liu and Ying (2016) for detailed proof. $\square$

We analyze the complexity of spatial-temporal routing policy as follows: Denote T_d to be the length of the maximum end-to-end deadline. For node n at time slot t , only virtual commodities $\{d, t_e\}$, where t_e belongs to $[t - T_d, t + T_d]$, are “activated” and required to be considered at time slot t . The number of “active” virtual commodities is $O(T_d)$ at time slot t , and the computation complexity of node n at time slot t in spatial-temporal routing policy is $O(VT_d^2)$.

Theorem 2 and Theorem 3 prove 1) spatial-temporal backpressure policy and spatial-temporal water-filling policy can support any arrival within outer throughput region when η is sufficiently large, and imply 2) both policies achieve high delivery ratio when the traffic fluctuation is low. We verify these two arguments with simulation in Fig. 4.3a) and 4.3b), respectively, where the detailed parameters can be found in the section of Simulation. To tackle the general traffic, we propose a constrained resource-pooling heuristic to improve the original spatial-temporal routing policies in the next section.

(a) A periodic traffic pattern

(b) A general traffic pattern

Figure 4.3: Throughput performance of spatial-temporal routing policy under a periodic and general traffic pattern. (a) verifies spatial-temporal routing is “near optimal” in a periodic traffic; (b) verifies spatial-temporal routing performs well when the traffic fluctuation is low in a general arrival. (b) ToDo

4.3 A Constrained Resource-pooling Heuristic

In this section, we enhance spatial-temporal policies by a constrained resource-pooling heuristic (CRPH) imposed on real packets transmission, i.e. step 4 in Policy 1 and step 3 in Policy 2.

The motivation for integrating a heuristic on real packets transmission is to guarantee spatial-temporal policies to be “work-conserving” in the general traffic by pooling the link capacity among virtual commodities with the same destination. We explain how the heuristic works in the following scenario: at the t th time slot in frame k , two virtual commodities $\{d, t_e\}$ and $\{d, t_e + 1\}$ with the same destination d share the virtual link $\{(a, b), t\}$ and their corresponding number of packets are m and n . Spatial-temporal policies are able to tell us the service rates $\hat{u}_{\{d, t_e\}}^{\{(a, b), t\}}[k]$ and $\hat{u}_{\{d, t_e + 1\}}^{\{(a, b), t\}}[k]$. Suppose $m < \hat{u}_{\{d, t_e\}}^{\{(a, b), t\}}[k]$ and $n > \hat{u}_{\{d, t_e + 1\}}^{\{(a, b), t\}}[k]$, the proposed heuris-

tic suggests to shift $\min \left(n - \hat{u}_{\{d, t_e+1\}}^{\{(a,b),t\}}[k], \hat{u}_{\{d, t_e\}}^{\{(a,b),t\}}[k] - m \right)$ packets from $\{d, t_e + 1\}$ to $\{d, t_e\}$ and mark them belonging to $\{d, t_e\}$. Note the shifted packets can be delivered successfully since $\hat{u}_{\{d, t_e\}}^{\{(a,b),t\}}[k]$ are the guaranteed service rate obtained from spatial-temporal routing.

Further, we illustrate CRPH with a toy example. Consider a physical link (a, b) with capacity=10 and the frame size has 3 time slots in Fig. 4.4. Two virtual commodities $\{b, 2\}$ and $\{b, 3\}$ share a virtual links $\{(a, b), 2\}$. The arrivals for both virtual

Figure 4.4: Illustration of a constrained resource-pooling heuristic

commodities are random integer variables from 14 to 16, equally likely. Spatial-temporal routing will return the feasible solution: $\hat{u}_{\{b,2\}}^{\{(a,b),1\}} = 10$, $\hat{u}_{\{b,2\}}^{\{(a,b),2\}} = 5$, $\hat{u}_{\{b,3\}}^{\{(a,b),2\}} = 5$, and $\hat{u}_{\{b,3\}}^{\{(a,b),3\}} = 10$. Consider a traffic pattern with $A_{\{b,2\}} = 14$ and $A_{\{b,3\}} = 16$ in Fig. 4.4. If following spatial-temporal routing $\hat{\mathbf{u}}$, one packet of $\{b, 3\}$ would be lost. However, by integrating CRPH into $\hat{\mathbf{u}}$, one packet of $\{b, 3\}$ is shifted to $\{b, 2\}$ at time slot 2 and it will be delivered successfully. As we can see, the heuristic smooths the stochastic arrival across time horizon to enhance “work-conserving” capability.

Note CRPH shares link resource among the virtual commodities with the same destination, one may consider an “intuitive” heuristic, where link resources pooling among the virtual commodities with distinct destinations. However, we will see this “intuitive” heuristic might degrade the performance. Consider three nodes network

with three links (a, b) , (b, c) and (a, c) in Fig. 4.5. Link capacity are fixed all the time to be $C_{a,b} = 10$, $C_{b,c} = 5$ and $C_{a,c} = 5$, respectively. The frame size is composed of two time slots and there exists two virtual flows $(b, 1)$ and $(c, 2)$ arriving at the beginning of the frame with mean $E[A_{(b,1)}] = 15$ and $E[A_{(c,2)}] = 5$. By spatial temporal routing, we have $\hat{u}_{\{b,1\}}^{\{(a,b),1\}} = 5$, $\hat{u}_{\{c,2\}}^{\{(a,b),1\}} = 5$, $\hat{u}_{\{c,2\}}^{\{(b,c),2\}} = 5$, $\hat{u}_{\{c,2\}}^{\{(a,c),1\}} = 5$, and $\hat{u}_{\{c,2\}}^{\{(a,c),2\}} = 5$. One virtual route is marked with red color for $(b, 1)$ and three virtual routes are marked with blue color for $(c, 2)$ as shown in Fig. 4.5. Suppose in frame k , we have a traffic pattern $A_{(b,1)}[k] = 2$ and $A_{(c,2)}[k] = 16$. If following the “intuitive” heuristic, the virtual link $\{(a, b), 1\}$ will transmit 8 packets of $\{c, 2\}$ and 2 packets of $\{b, 1\}$, resulting in 3 packets of $\{c, 2\}$ lost due to the bottle link $C_{\{b,c\}} = 5$. In fact, by allocating 5 packets of $\{c, 2\}$ according to spatial-temporal routing, we have only 1 packet of $\{c, 2\}$ lost in this traffic pattern.

Figure 4.5: Illustration of downside on an “intuitive” heuristic.

Then we summarized spatial-temporal policies with CRPH in Policy 3.

In step 2 in CRPH, $l_{\{d,t_e\}}^{\{(a,b),t\}}$ is the surplus resource of $\{d, t_e\}$ and $\nu_{\{d,t_e+j\}}^{\{(a,b),t\}}$ is the number of packets of $\{d, t_e + j\}$ to fill up. It is worth to mention that CRPH improves the robustness to traffic fluctuation without affecting the convergence of the original

Policy 3 ST Policy with resource-pooling heuristic

- 1: Follow step 1, 2 and 3 in STBP or step 1 and 2 STWF to obtain $\{\hat{u}\}$.
- 2: Perform the constrained resource-pooling heuristic for $\{d, t_e\}$ on $\{(a, b), t\}$

$$\begin{aligned} r_{\{d, t_e\}}^{\{(a, b), t\}} &= \min \left\{ \hat{u}_{\{d, t_e\}}^{\{(a, b), t\}}[k], Q_{\{d, t_e\}}^t[k] \right\}, \\ l_{\{d, t_e\}}^{\{(a, b), t\}} &= \left(\hat{u}_{\{d, t_e\}}^{\{(a, b), t\}}[k] - Q_{\{d, t_e\}}^t[k] \right)^+, \\ \nu_{\{d, t_e+j\}}^{\{(a, b), t\}} &= \left(Q_{\{d, t_e+j\}}^t[k] - \hat{u}_{\{d, t_e+j\}}^{\{(a, b), t\}}[k] \right)^+, j \geq 1, \end{aligned}$$

and allocate real packet transmission rates to be

$$r_{\{d, t_e\}}^{\{(a, b), t\}}[k] + \min \left\{ l_{\{d, t_e\}}^{\{(a, b), t\}}[k], \sum_{j \geq 1} \nu_{\{d, t_e+j\}}^{\{(a, b), t\}}[k] \right\}$$

- 3: Update virtual queues at the end of the k th frame as in ST policy.
-

spatial-temporal policies.

Chapter 5

SIMULATIONS

In this section, we evaluate the performance of the proposed policies via simulations in two scenarios. The first scenario is dedicated to justify the theoretical results in the chapter 4 and to verify the effectiveness of CRPH proposed in the section 4.3; and the second scenario is to demonstrate spatial-temporal routing policies with CRPH is practical for video transmission in Abilene network Knight *et al.* (2011).

5.1 First Scenario: Symmetric Topology

We consider a network topology shown in Fig. 5.1, where all links have the same capacity of $C = 10^3$ packets/time slot and we have three flows in the network. Flow 1 is from node 1 to 6, flow 2 is from node 2 to 7, and flow 3 is from node 3 to 5. We call our spatial-temporal backpressure policy “STBP Policy” and spatial-temporal waterfilling policy “STWF Policy” in which we choose the disutility function $U(x) = x^{1.1}$ and trade-off factor $V = 10^4$. We choose $x^{1.1}$ so that the optimization problem has a unique solution, and it is close to x so that the object is to minimize the average latency in addition to guarantee end-to-end deadlines. We compared them with the backpressure and EDF policies. In our simulations, EDF adopts randomized routing to route each packet to a randomized chosen neighbor, and backpressure adopts random packet scheduling for the selected commodity.

Periodic traffics: We assume frame size $T = 6$. Packets of flow 1 and flow 2 arrive at the beginning of the frame and expire at the end of the frame. Packets of flow 3 arrive at time slot 2 and expire at time slot 4 of a frame. Packet arrivals per frame increase from (2000, 1000, 0) to (4000, 3000, 2000) with the step size (200, 200, 200).

Figure 5.1: Symmetric Topology

We observe in Fig. 5.2 that two proposed spatial-temporal policies achieve the same throughput and outperform both backpressure and EDF at all incoming rates. EDF has a higher throughput than backpressure at the low traffic regime and backpressure outperforms EDF at the high traffic regime. It can also be easily verified that (4000, 3000, 2000) is at the boundary of the network capacity given the periodic traffic pattern. The results in Fig. 5.2 verifies ST policies can support (4000,3000,2000) as we proved in Theorem 2 and Theorem 3. As for the average delay performance of delivery packets, we only compare two proposed ST policies because of their significant advantage of throughput performance against EDF and backpressure. Fig. 5.3 shows STWF policy dominates STBP policy on the average delay performance and demonstrates STWF policy can further reduce the average delay by routing packets on these virtual routes with small delay.

More General traffic: We consider a traffic pattern such that each of three flows has packet arrivals at every time slot. Each packet has a deadline of 6 time slots. The number of packet arrivals for each flow at each time slot is a random variable that takes integer values from $10 - i$ to $10 + i$, equally likely. We varied i from 0 to 10. A larger value of i implies larger variance of the incoming traffic. We impose virtual

Figure 5.2: Throughput performance of different routing policies for a periodic traffic pattern

Figure 5.3: Average delay performance of ST routing policies for a periodic traffic pattern

Figure 5.4: Throughput performance of different routing policies for a more general traffic pattern ToDo

frame structure on this general traffic pattern and evaluated the ST policies as the frame size increases, $T = 50$ and 100 . From Fig. 5.4, we can see that the delivery ratio of both ST policies increases as frame T increases and STWF policy (dash line) has better throughput performance than STBP policy (solid line) especially for small $T = 50$. Furthermore, ST policies outperform both backpressure and EDF for almost all values of i even when T is small. The delivery ratio increases as the variance of the incoming traffic decreases. This is intuitive since it is more difficult to guarantee end-to-end deadlines when the randomness in the traffic increases. In Fig. 5.5, we evaluate the average delay performance. Increasingly, we can see that the average delay under STWF decreases as T increases while it increases under STBP. The reason we believe is that STBP is designed for guaranteeing end-to-end deadlines while STWF further attempts to minimize end-to-end latency.

Figure 5.5: Average delay performance of ST routing policies for a more general traffic pattern ToDo

Figure 5.6: Abilene Topology

5.2 Second Scenario: Video Transmission with Abilene Network

In this scenario, we consider a video transmission on a practical network, Abilene network in Fig. 5.6, where three video streaming flows exist, flow 1 is from node 1 to node 8, flow 2 is from node 6 to node 5 and flow 3 is from node 7 to node 11. We configure packet size with 1250 bytes and Ethernet interface with capacity 10G bits/s. Assume the length of a time slot is 10ms, then link capacity in terms of packets

Figure 5.7: Throughput performance of different routing policies for Abilien Network

per time slot is 10^4 packets/time slot. According to the requirements of interactive Video, one-way latency should be less than 150ms, so we choose the latency at 100ms (10 time slots). The number of packets arrival for each video streaming is a random variable that takes integer values from $10^3 \cdot (10 - i)$ to $10^3 \cdot (10 + i)$, equally likely, where i is varied from 0 to 10. The throughput performance is shown in Fig. 5.7. We can see both STBP and STBP-FS outperform EDF and Backpressure significantly for small i . As i increases (“wild” traffic), STBP degrades. However, STBP-FS is robust to traffic fluctuation. Specifically, for $i = 10$, STBP-FS outperform EDF by a large margin, 79.8% v.s. 70.2%.

Chapter 6

CONCLUSIONS

In this thesis, we developed the resource-pooling heuristic based on spatial-temporal routing for supporting end-to-end hard deadlines in communication networks. We first introduced the virtual queue architecture and developed two spatial-temporal routing policies: spatial-temporal backpressure and spatial-temporal water-filling. We then developed the resource-pooling heuristic. Our policies can support any periodic traffic within the network throughput region, and for the general traffic patterns, numerical simulations verified that the proposed policies outperform existing routing policies.

REFERENCES

- Boyd, S. and L. Vandenberghe, *Convex Optimization* (Cambridge University Press, New York, NY, 2004).
- Gallager, R. G., *Discrete Stochastic Processes*, The Springer International Series in Engineering and Computer Science (Springer US, New York, NY, 1996).
- Hou, I.-H., “Packet scheduling for real-time surveillance in multihop wireless sensor networks with lossy channels”, *IEEE Trans. Wireless Commun.* **14**, 2, 1071–1079 (2015).
- Hou, I.-H., V. Borkar and P. R. Kumar, “A theory of QoS for wireless”, in “Proc. IEEE Int. Conf. Computer Communications (INFOCOM)”, pp. 486–494 (Rio de Janeiro, Brazil, 2009).
- Jaramillo, J. J. and R. Srikant, “Optimal scheduling for fair resource allocation in ad hoc networks with elastic and inelastic traffic”, *IEEE/ACM Trans. Netw.* **19**, 1125–1136 (2011).
- Knight, S., H. Nguyen, N. Falkner, R. Bowden and M. Roughan, “The internet topology zoo”, *Selected Areas in Communications, IEEE Journal on* **29**, 9, 1765–1775 (2011).
- Li, R. and A. Eryilmaz, “Scheduling for end-to-end deadline-constrained traffic with reliability requirements in multihop networks”, *IEEE/ACM Trans. Netw.* **20**, 5, 1649–1662 (2012).
- Liu, X. and L. Ying, “Spatial-temporal routing for supporting end-to-end deadlines in multi-hop networks”, Tech. rep., Arizona State University (2016).
- Mao, Z., C. E. Koksal and N. B. Shroff, “Optimal online scheduling with arbitrary hard deadlines in multihop communication networks”, *IEEE/ACM Trans. Netw.* **PP**, 99 (2014).
- Singh, R. and P. R. Kumar, “Decentralized throughput maximizing policies for deadline-constrained wireless networks”, in “2015 54th IEEE Conference on Decision and Control (CDC)”, pp. 3759–3766 (2015).
- Srikant, R. and L. Ying, *Communication Networks: An Optimization, Control and Stochastic Networks Perspective* (Cambridge University Press, 2014).
- Ying, L., S. Shakkottai, A. Reddy and S. Liu, “On combining shortest-path and back-pressure routing over multihop wireless networks”, *IEEE/ACM Trans. Netw.* **19**, 3, 841–854 (2011).