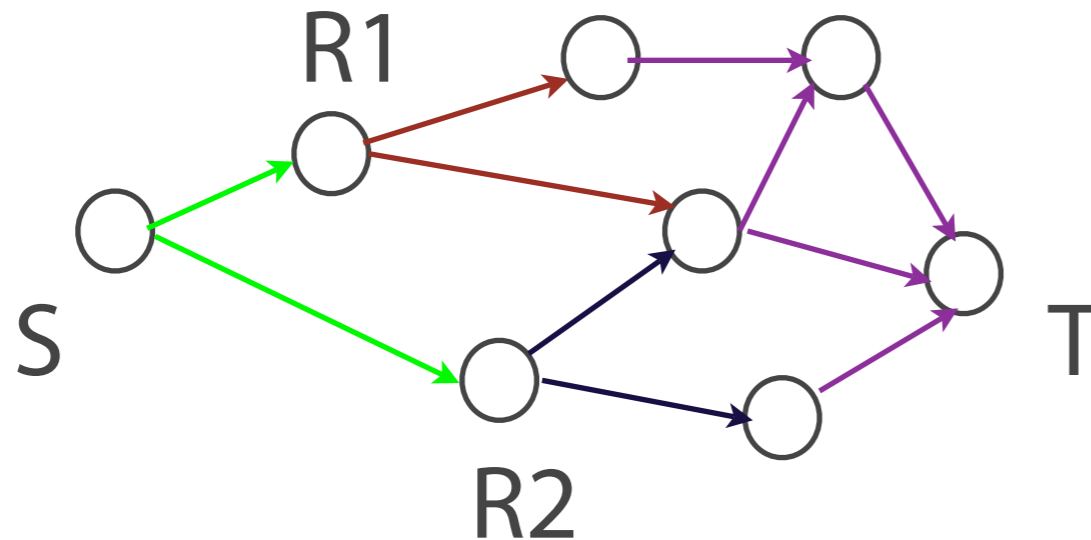


CodeOR: Opportunistic Routing in Wireless Mesh Networks with Segmented Network Coding



Yunfeng Lin, Baochun Li, Ben Liang
Department of Electrical and Computer Engineering
University of Toronto

Opportunistic Routing with Network Coding



Opportunistic routing

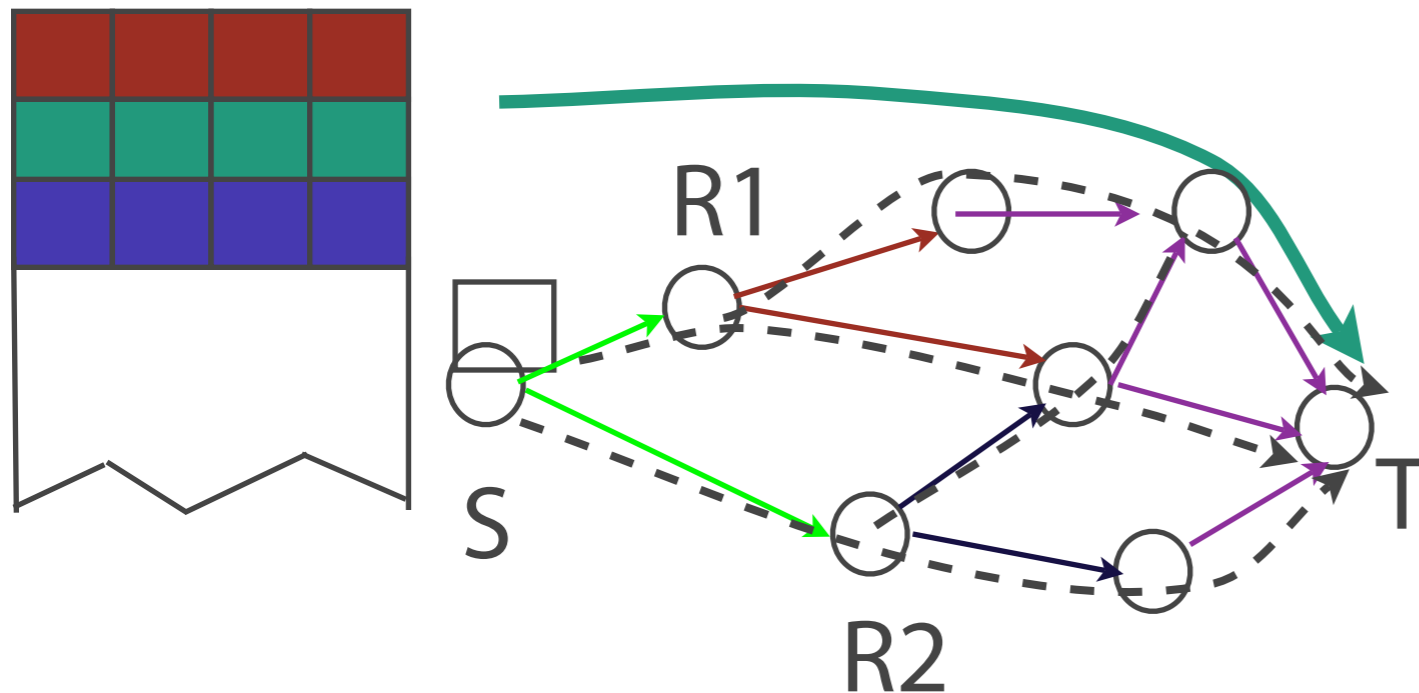
Utilizes wireless broadcast medium

Multi-path routing

Network coding simplifies protocol design

Coded packets are equivalent—no complicated packet scheduling

Network Coding Based Opportunistic Routing



Encoding and decoding complexity

Coding cannot be operated on all packets

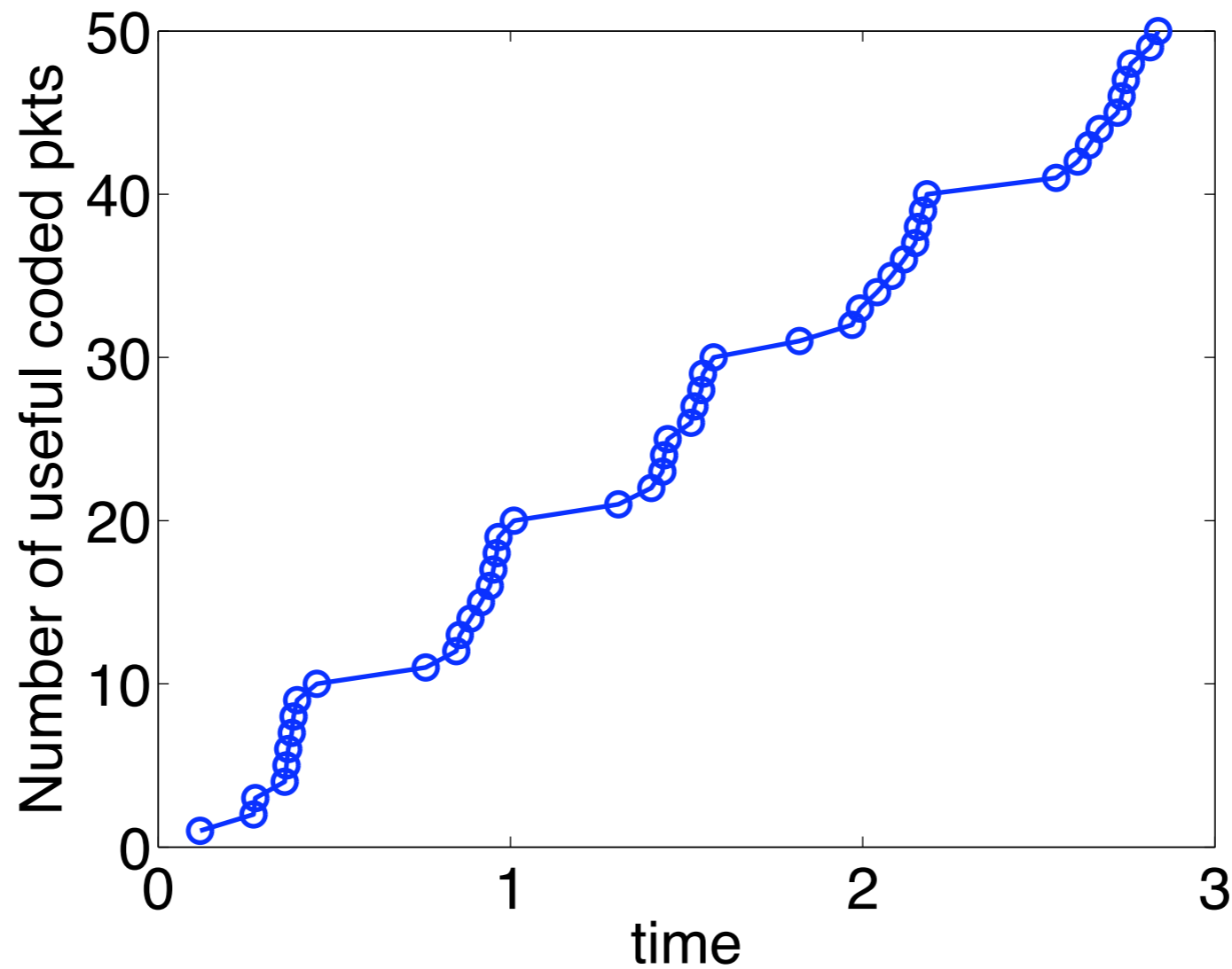
Data stream is divided to multiple segments

Transmit next segment after receiving the ACK of the current segment

A Shortcoming

When the ACK is in propagation — current segment is decoded

No new useful data is transmitted



Is It a Serious Problem?

Segment size cannot be too large

Encoding and decoding complexity

Large segment size implies long decoding delay

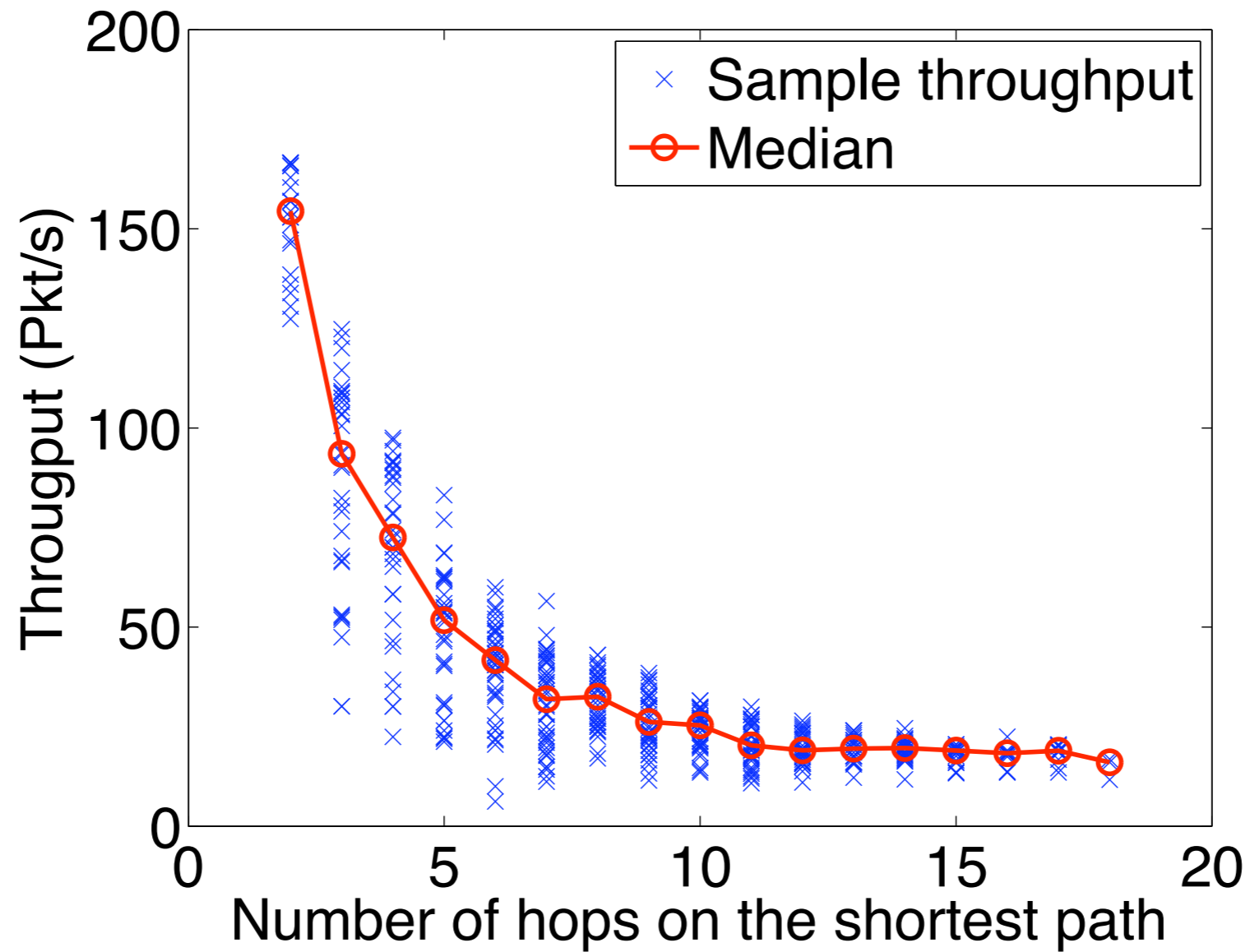
A few dozens in practice

Network scales up

Long time wasted in transmitting ACKs

Throughput degrades

Throughput Degrades



Contributions

Transmit multiple data segments simultaneously

Estimate the right number of segments for any data flow

Use small packet size to increase throughput

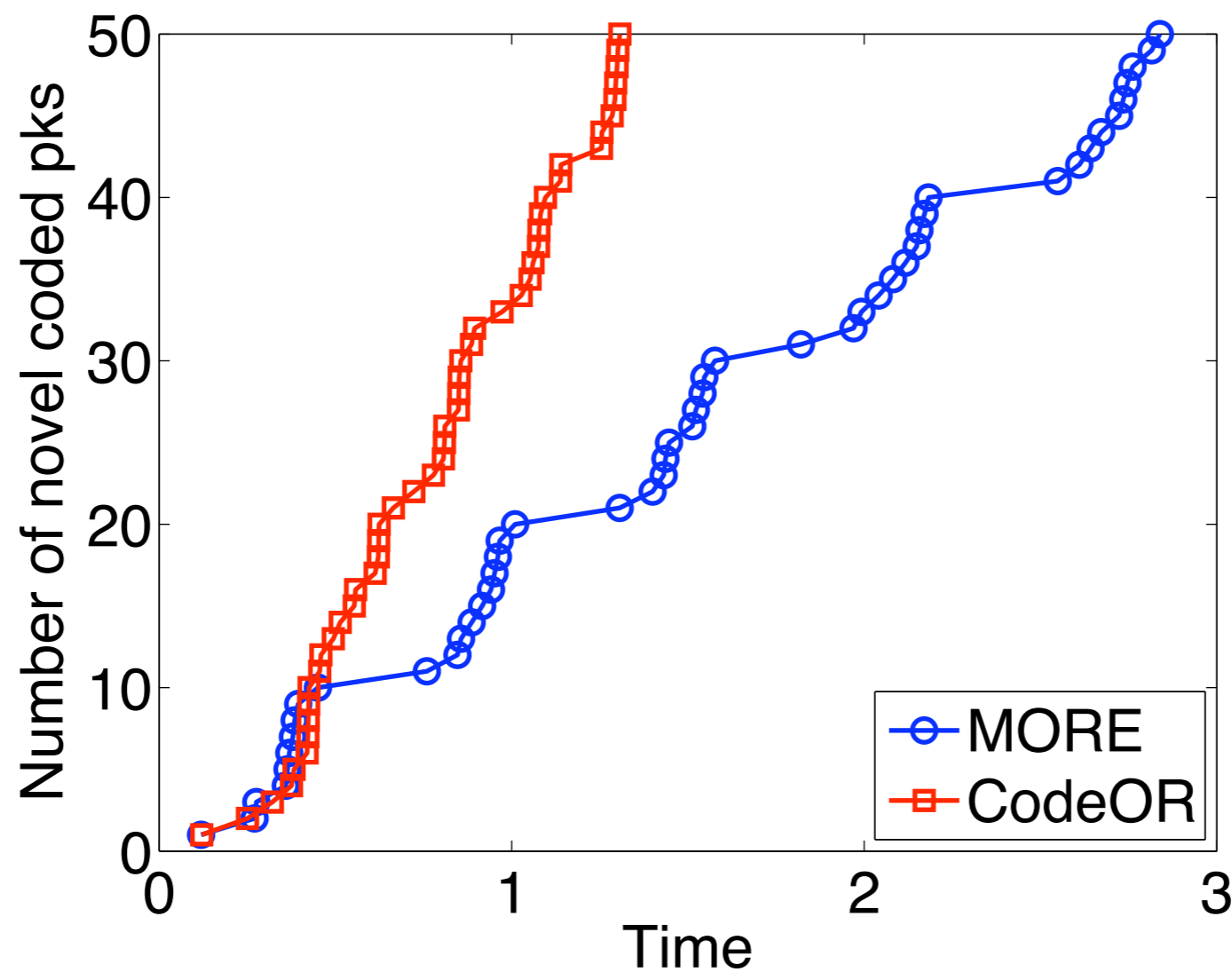
Higher throughput in large networks

Why Multiple Segments in Transmitting?

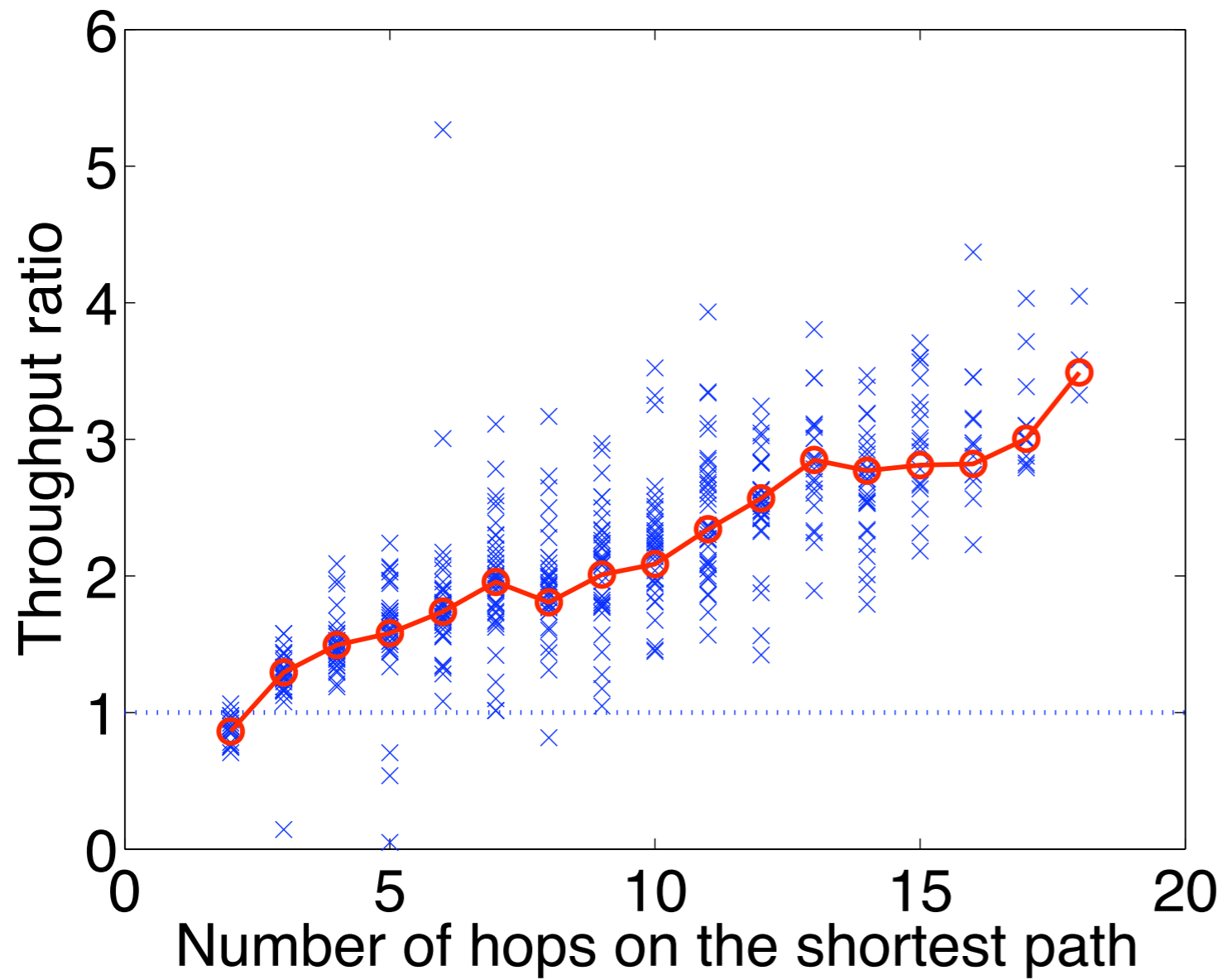
Benefits

Transmitting new segments during ACK propagation

Network is fully utilized at any time



Throughput Increases



5 hops — 1.5 times
18 hops — 3.5 times

Two Implementation Challenges

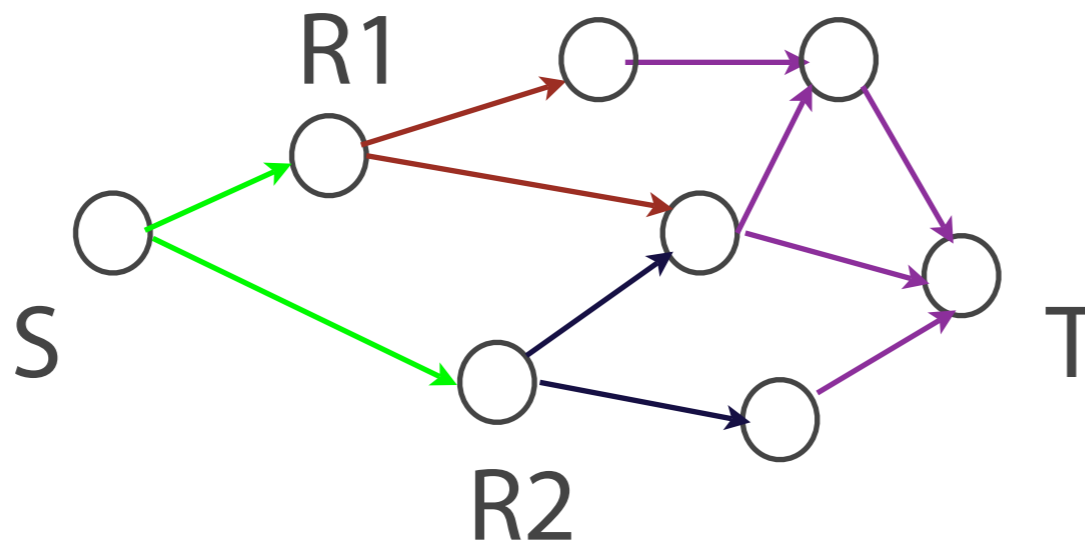
When to move to transmission of the next segment?

How many segments shall we transmit simultaneously?

Moving Towards the Next Segment

When to move to the next segment?

Move to next if downstream nodes receive sufficient data



Different downstream nodes have different rates

Lower rate—later time to receive all data in a segment

Moving Towards the Next Segment

Move if **all downstream nodes receive all data in this segment**

Wait for the slowest node—might be too late

Move if **one downstream node receives all data in this segment**

Might degrade to single-path routing

Move if downstream nodes

Receive data in proportion to their data rate

The node with highest rate receives all data

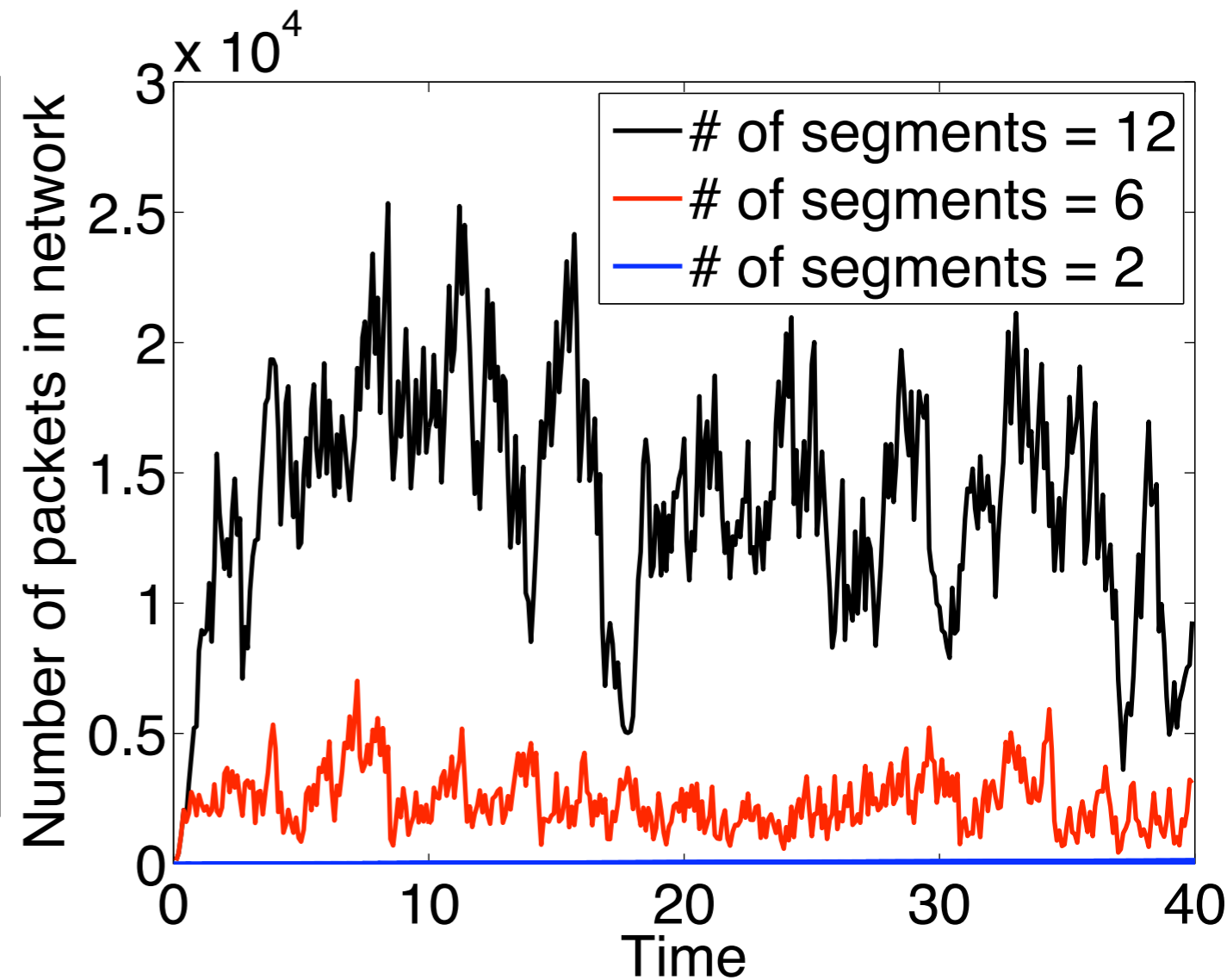
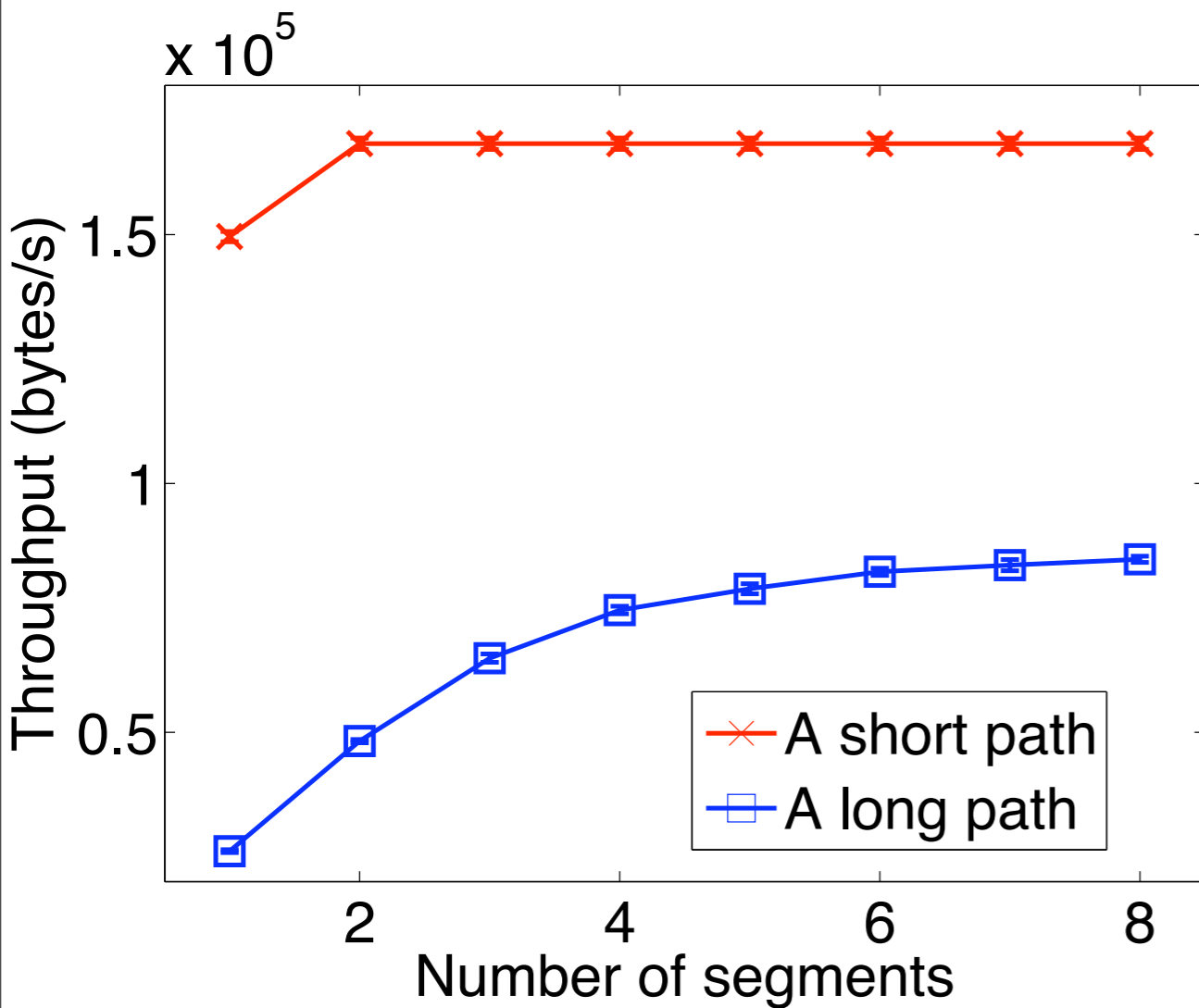
Local heuristic, not globally optimal

What is the Best Number of Segments?

A fixed number cannot handle different data flows

Too small — not sufficient data in network, throughput suffers

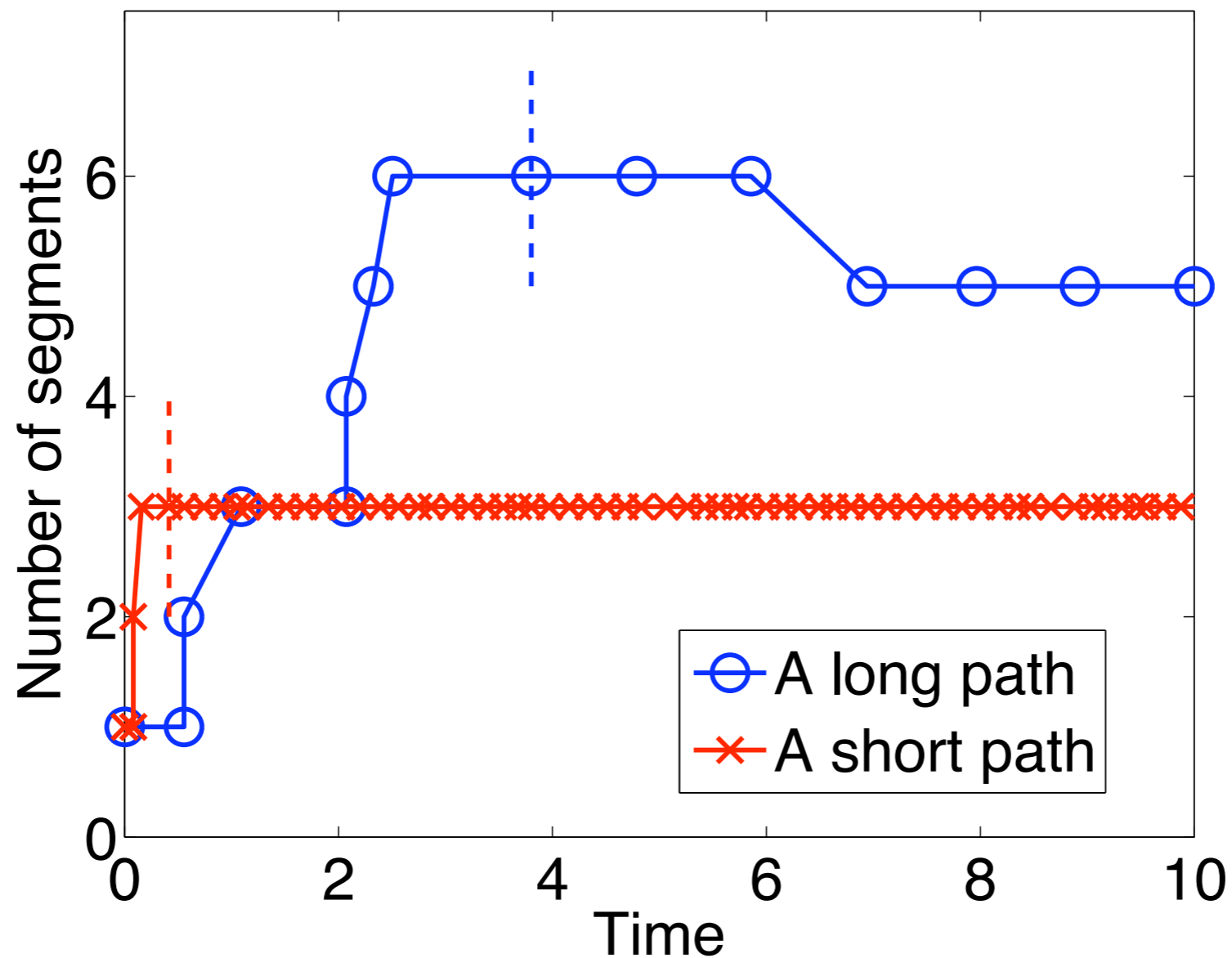
Too large — too much data in network, buffer overflow



Probe the Number of Segments

Utilize the idea of TCP flow control

Adjust the number based on end-to-end feedback



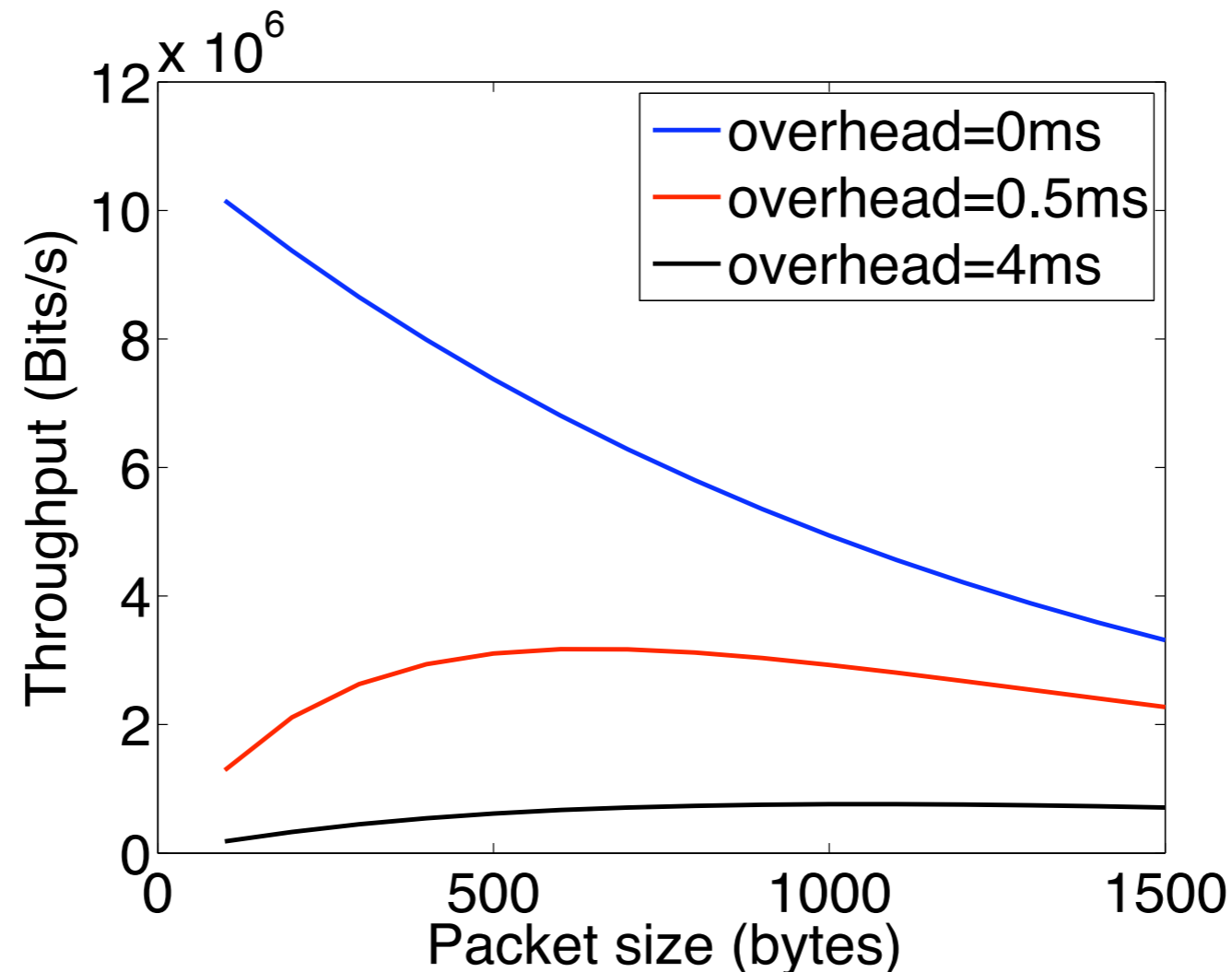
Smaller Packet Sizes—Higher Throughput

Minimum transmission unit is symbol instead of packet

Given constant symbol errors

Smaller packet sizes — lower packet loss rate

But higher proportion of overhead



Concluding Remarks

Allow multiple segments in transmitting at the same time

Use small packet size to increase throughput

Higher throughput in large-scale networks

5 hops — 1.5 times

18 hops — 3.5 times

Future work

Global optimal algorithm

[Http://iqua.ece.toronto.edu/~ylin](http://iqua.ece.toronto.edu/~ylin)

or

Google Yunfeng Lin