Scheduling analysis for Multi-Hop Wireless Mesh Networks in TDMA mode

Carmine Benedetto

Silvio Bianchi

Luca Giovanni Laudadio

Alessandro Pischedda

Introduction

ETT & WCETT evaluation

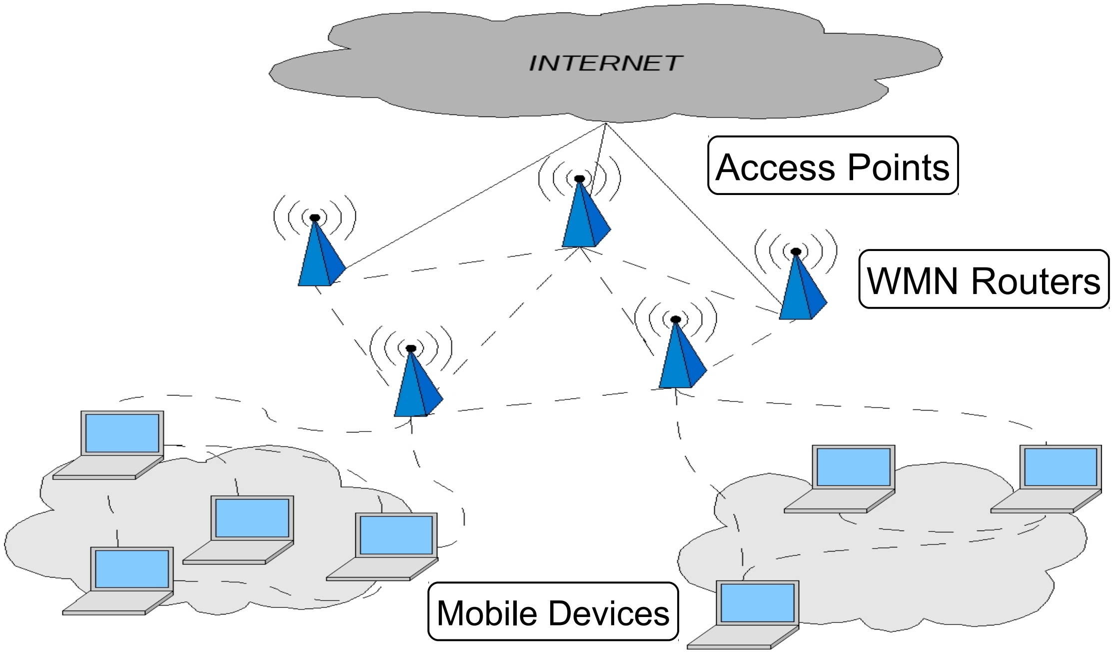
Reference paper

- ✓ ***Title:***

  Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks
- ✓ ***Authors:***

  Richard Draves, Jitendra Padhye, Brian Zill (Microsoft Research)

Mesh Networks TDMA



- The Wireless Mesh Networks core, represented by Wireless Mesh Routers, provides a multihop connectivity between mobile users and wired gateways (Acces Points)

- Time Division Multiple Access is a digital multiplexing technique in which the channel sharing is realized by dividing the access time to the channel among users

Scenario

- ✓ Topology: Grid Network
- ✓ Number of Nodes: 16
- ✓ Number of Flows: 15
- ✓ Simulation Duration: 800
- ✓ Simulation Warm-Up: 80
- ✓ Max Hops Number per path: 6
- ✓ Traffic Model: CBR (Constant Bitrate)

Metrics

- ✓ Average end-to-end Delay
- ✓ End-to-end Throughput
- ✓ Number of Hop per Flow

Metrics added in packet loss presence

- ✓ Packet Loss Ratio
- ✓ Average number of Retrasmissions
- ✓ Average number of Free Slots

ETT
Scheduling Policy

Overview

- ✓ ETT: Expected Trasmission Time
- ✓ Goal of the policy: having a high-throughput path between a source and a destination
- ✓ This policy selects the path with the minimum sum of the trasmission time for each link from source to destination

$$ETT_i = \frac{\text{packet size}}{\text{bandwidth}}$$

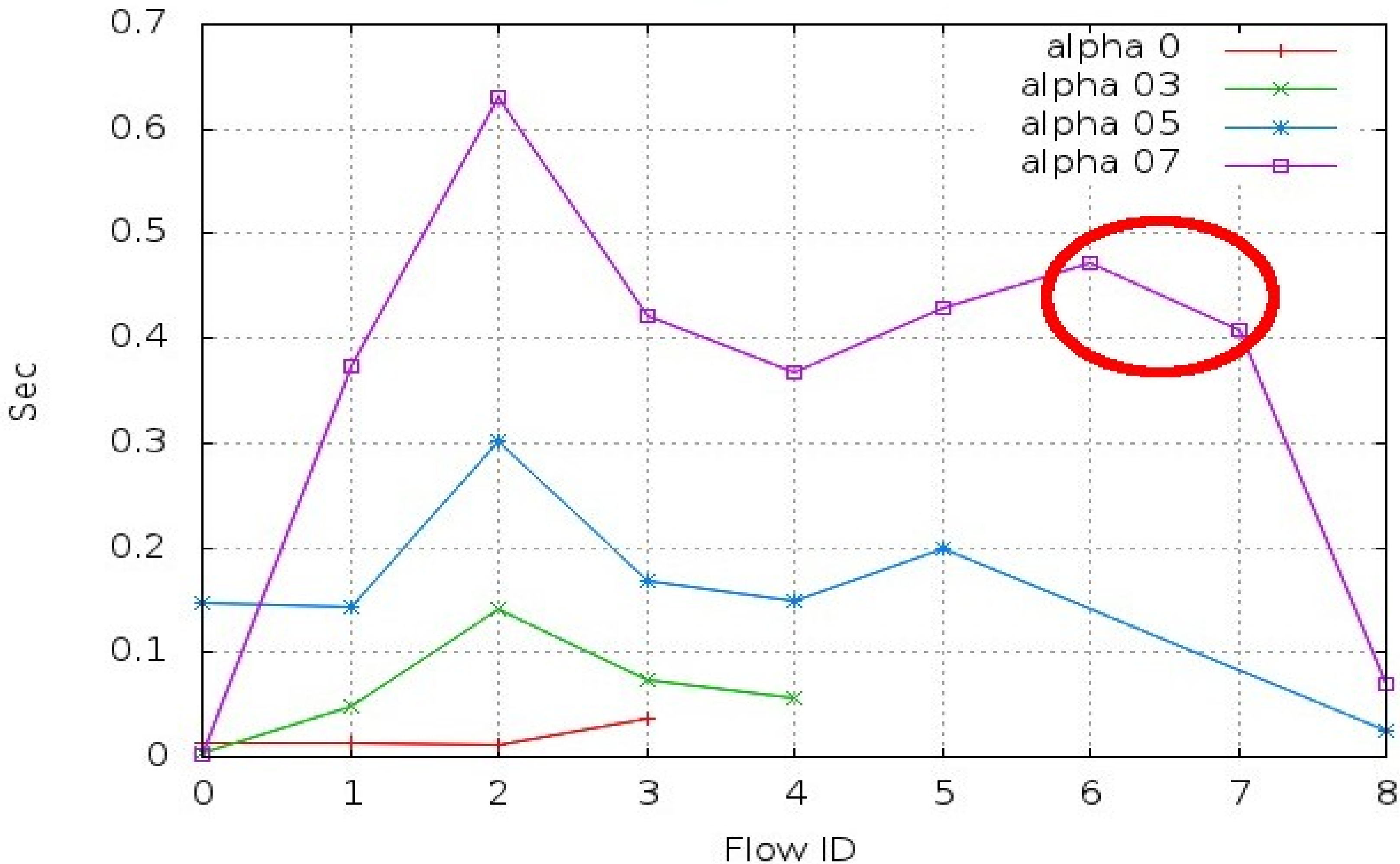
$$ETT = \sum ETT_i$$

End to end provisioning

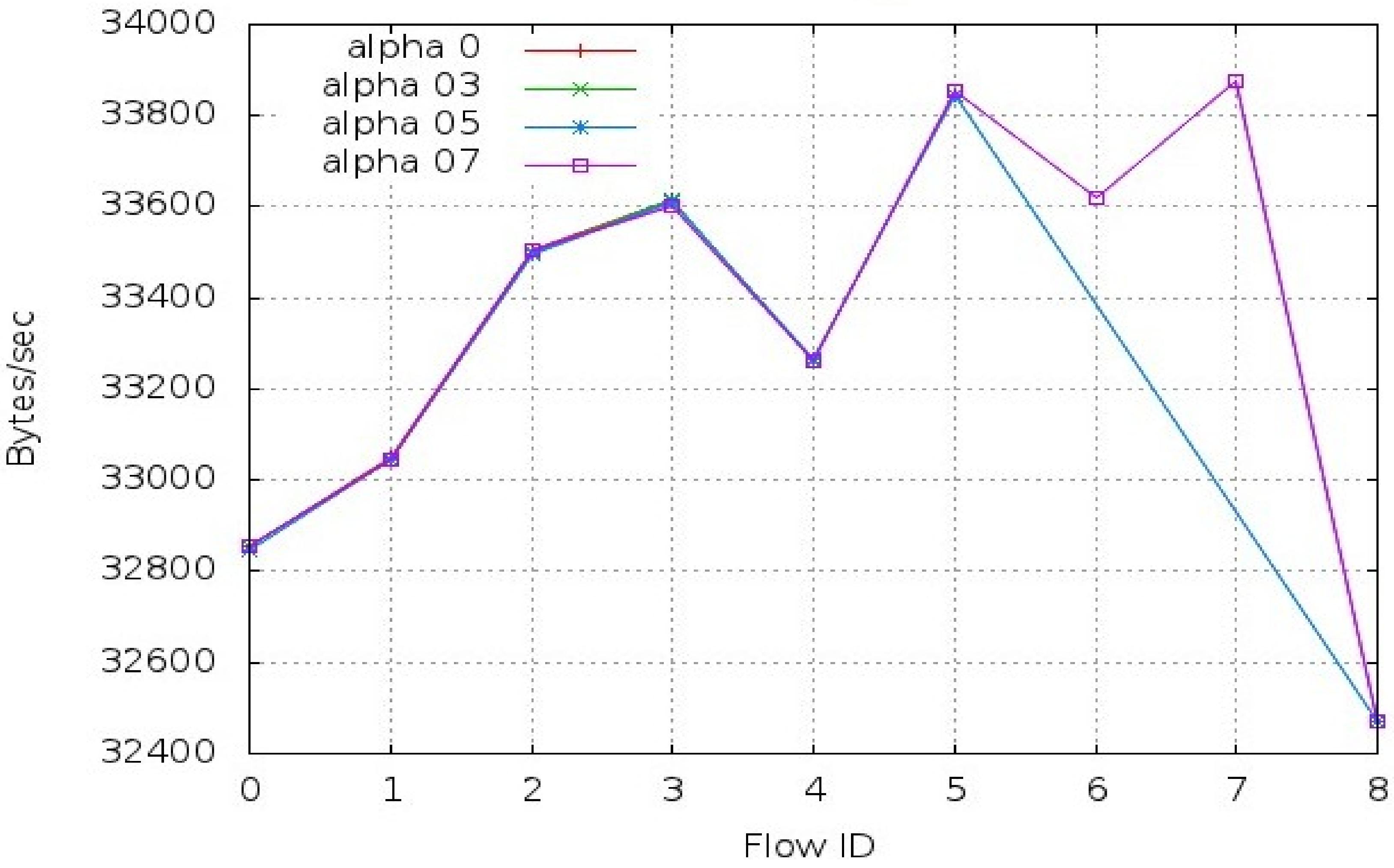
$$(1-a)*\text{peak rate}+a*\text{avg rate}$$

- ✓ $\alpha < 0.5$ less allocated flows
- ✓ $\alpha \geq 0.5$ more allocated flows

Choosing Alpha (1/2) - Average Delay

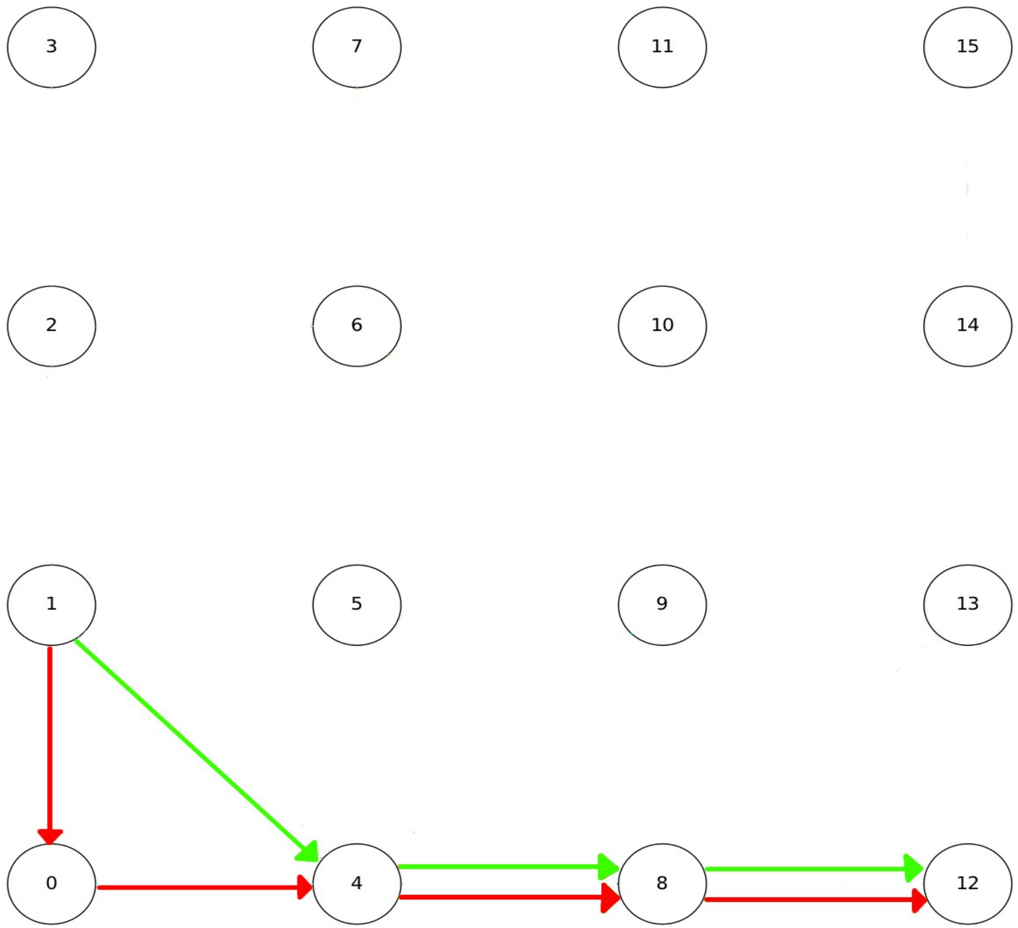


Choosing Alpha (2/2) – Throughput



ETT vs SP

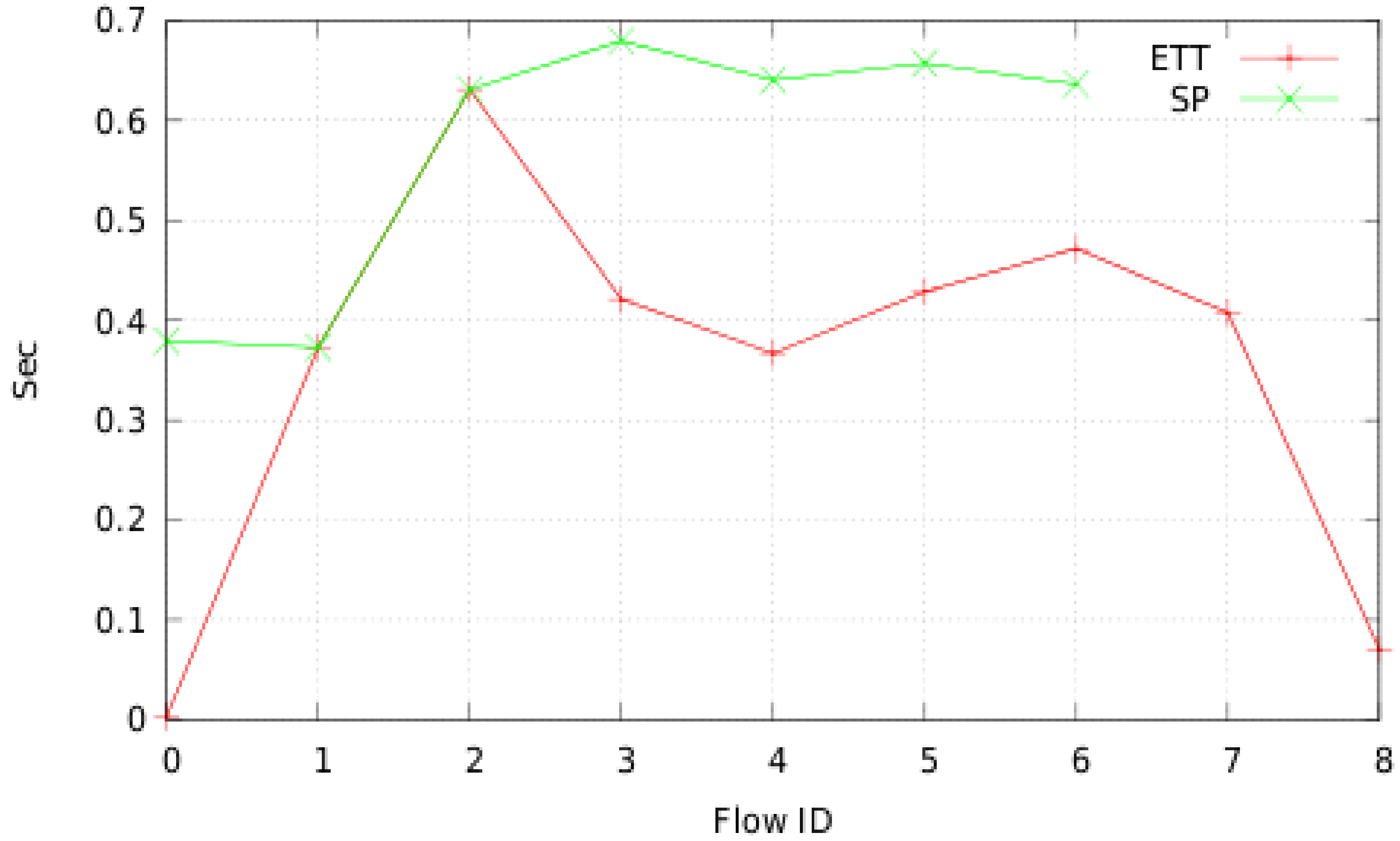
The shortest path isn't always the best!



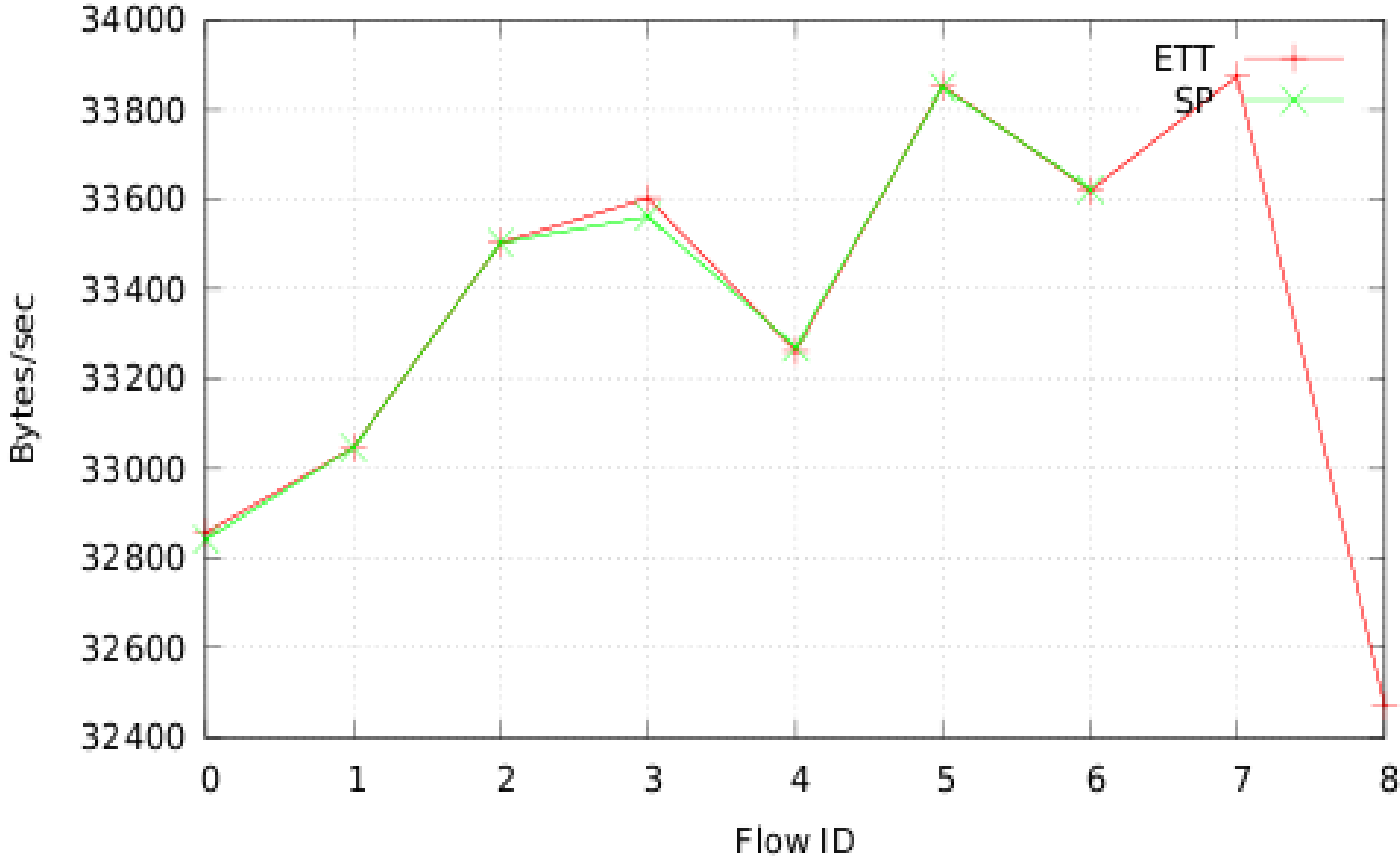
SP path

ETT path

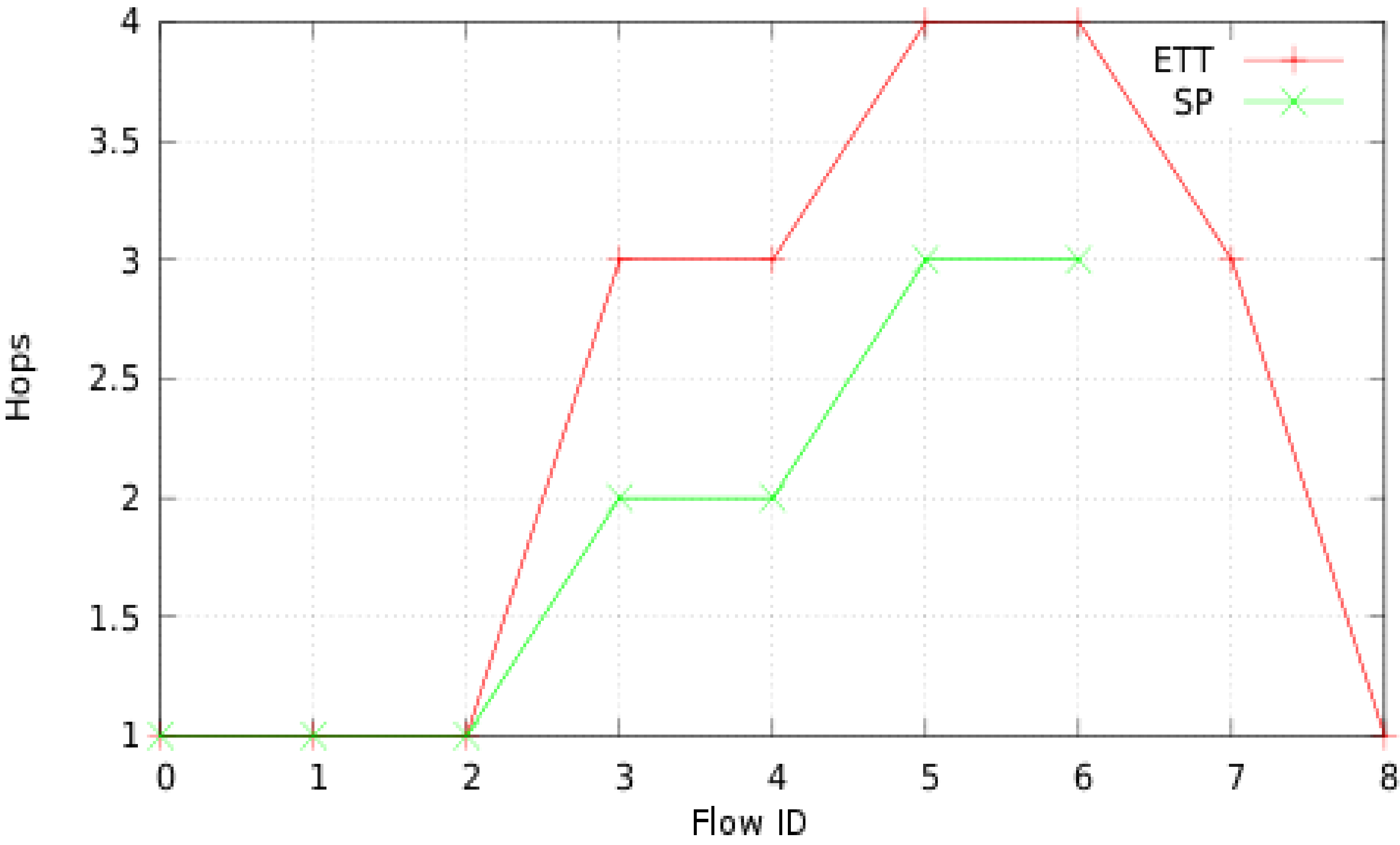
ETT vs SP – Average Delay



ETT vs SP - Throughput



ETT vs SP – Flow Hops



Error Introduction

For a more realistic approach:

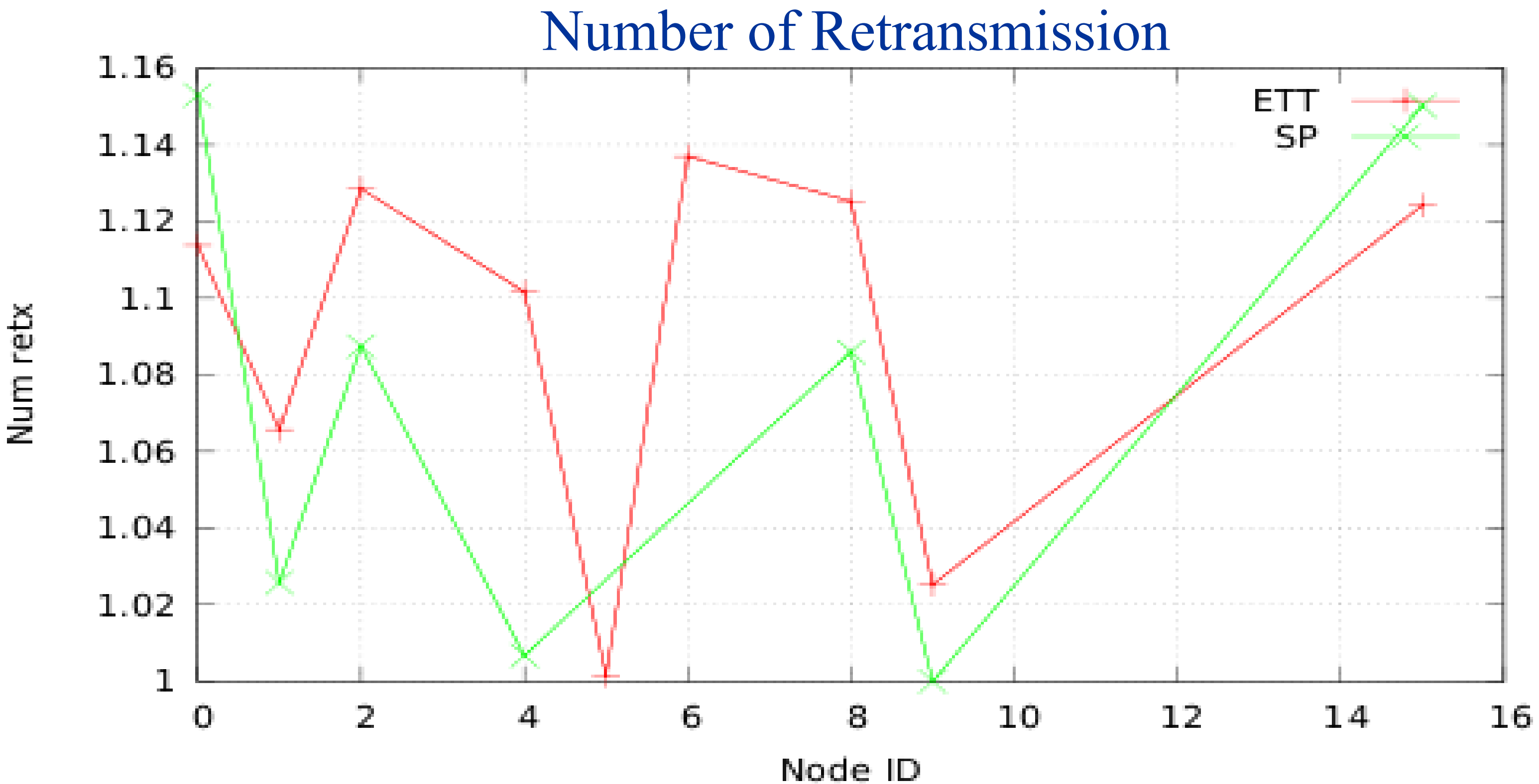
- ✓ Collision enabled
- ✓ Packet loss enabled
- ✓ Retransmission enabled
- ✓ Over-provisioning

Why over-provisioning? (1/2)

- ✓ Avoid asymptotic situation
- ✓ Give more resources to a flow

It is straightforward that less flows are accepted

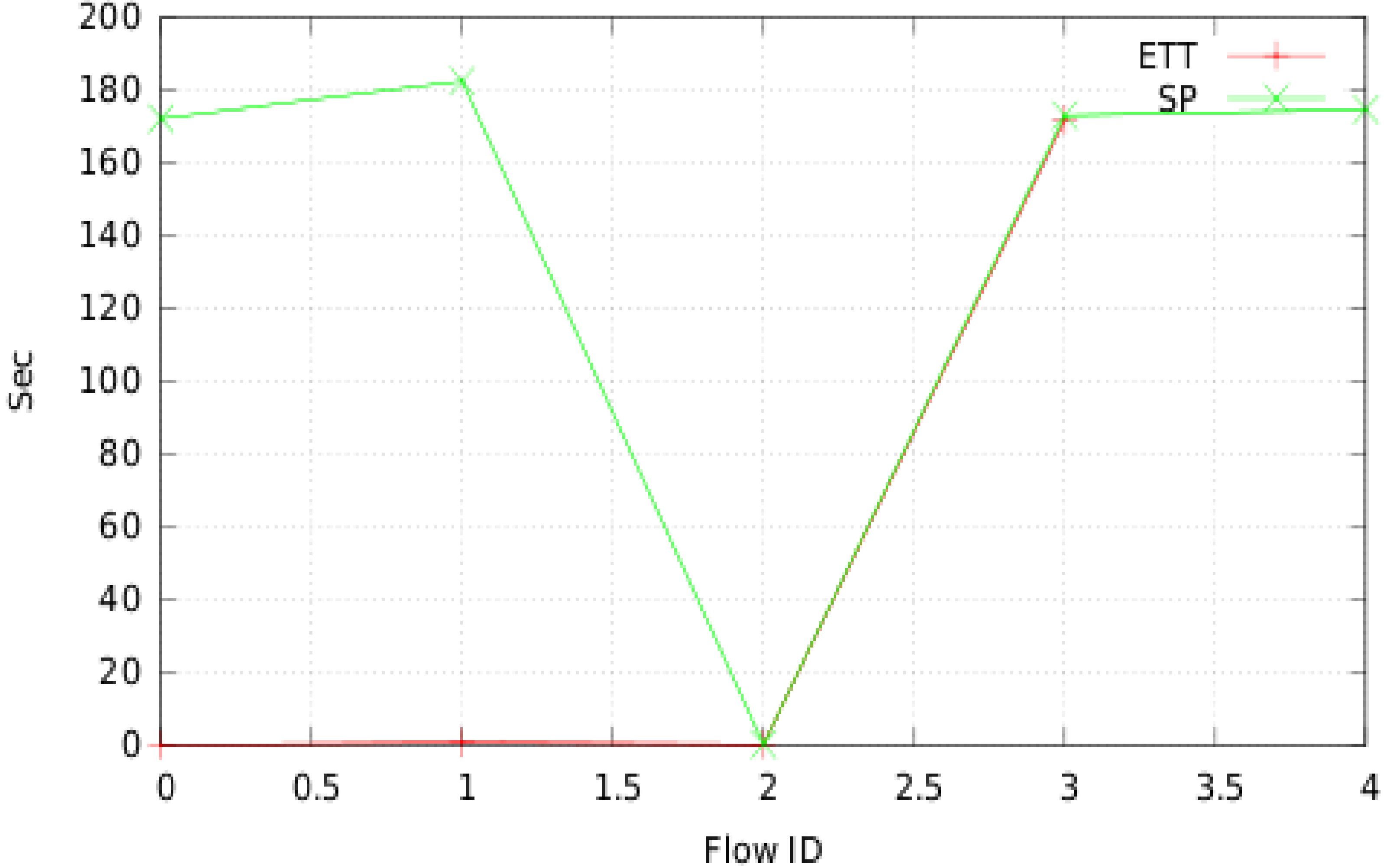
Why over-provisioning? (2/2)



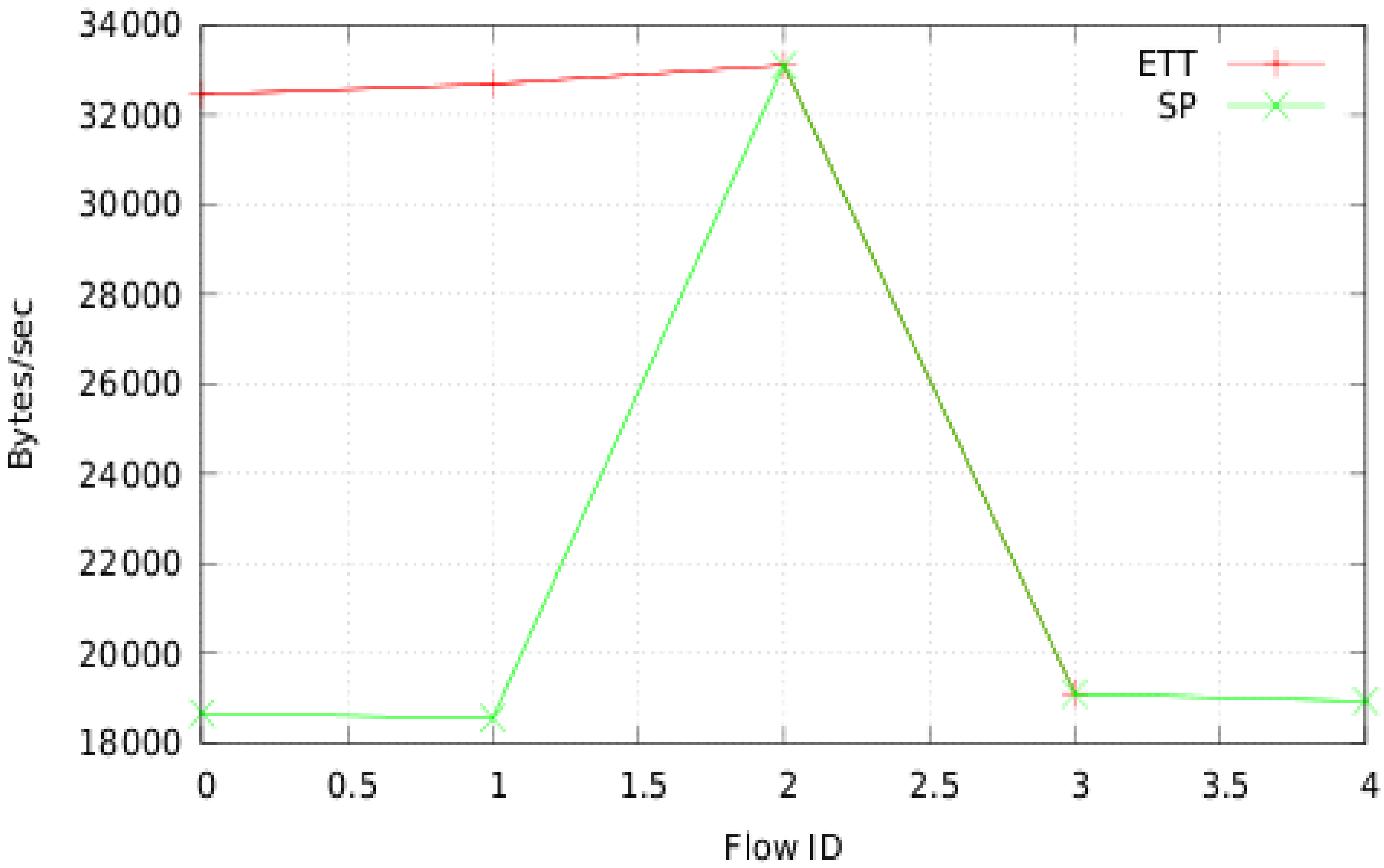
$$OverProv = ProvisioningRate * OPCoeff$$

$$OPCoeff = 2$$

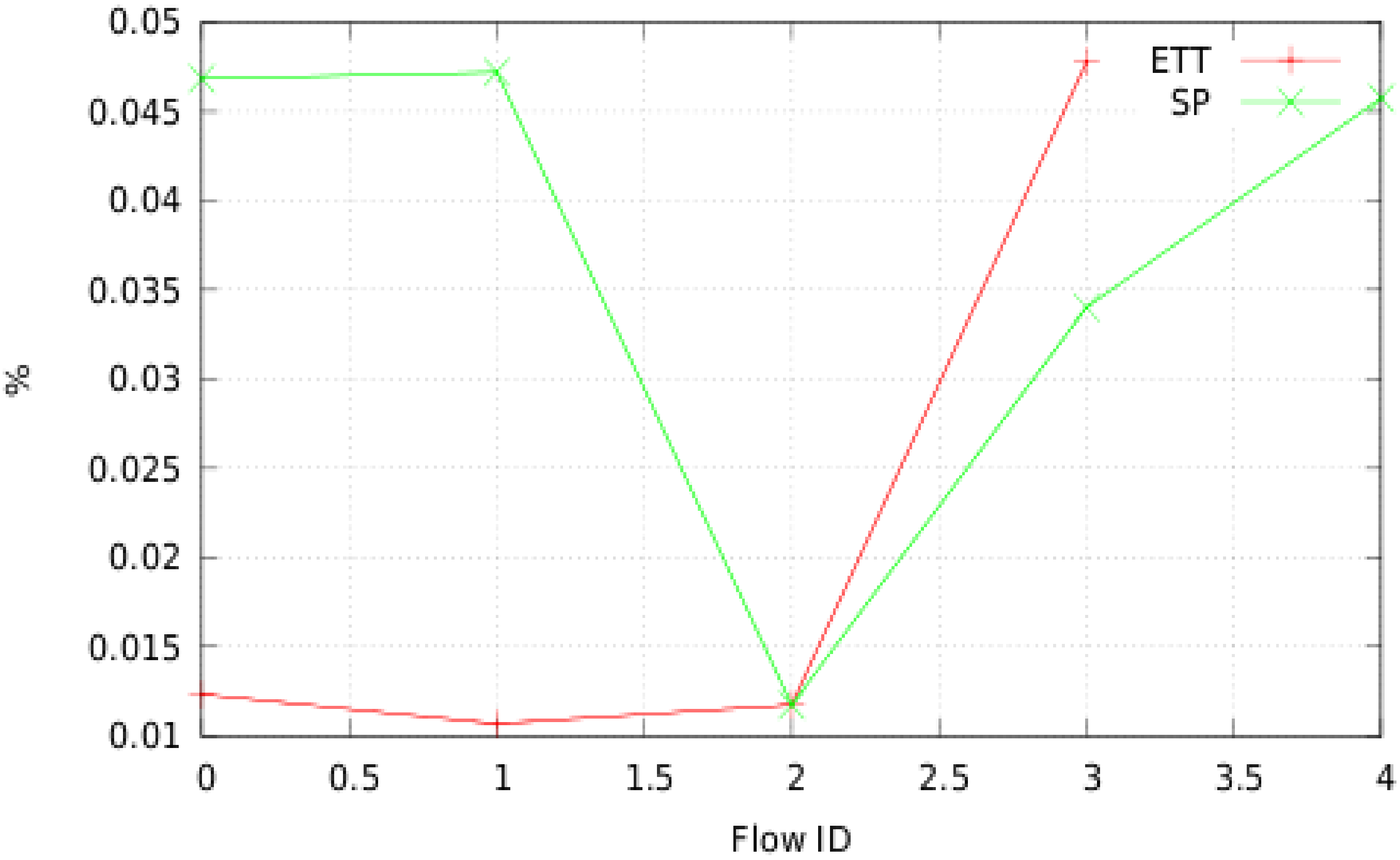
ETT vs SP – Average Delay



ETT vs SP - Throughput



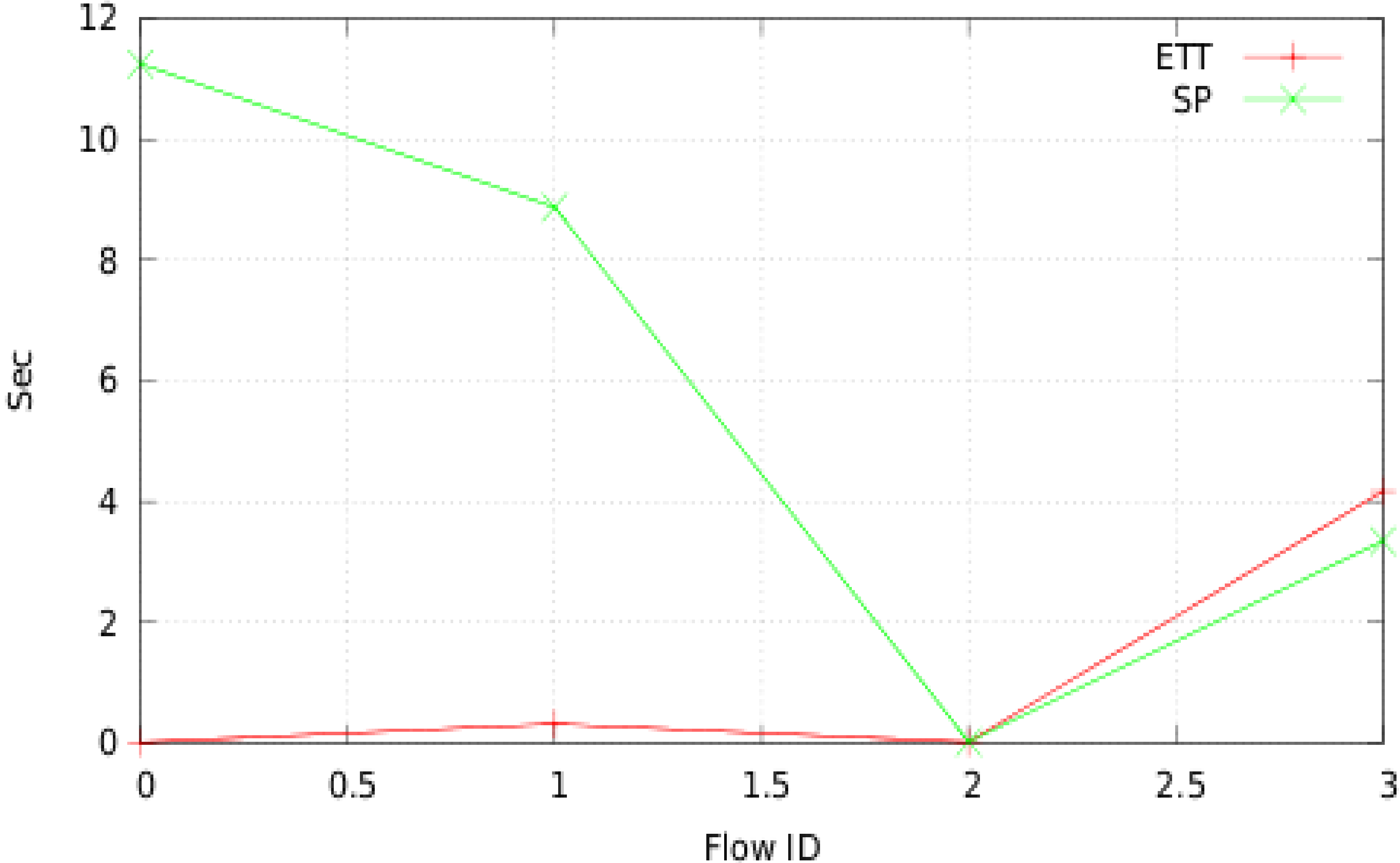
ETT vs SP – Packet Loss Ratio



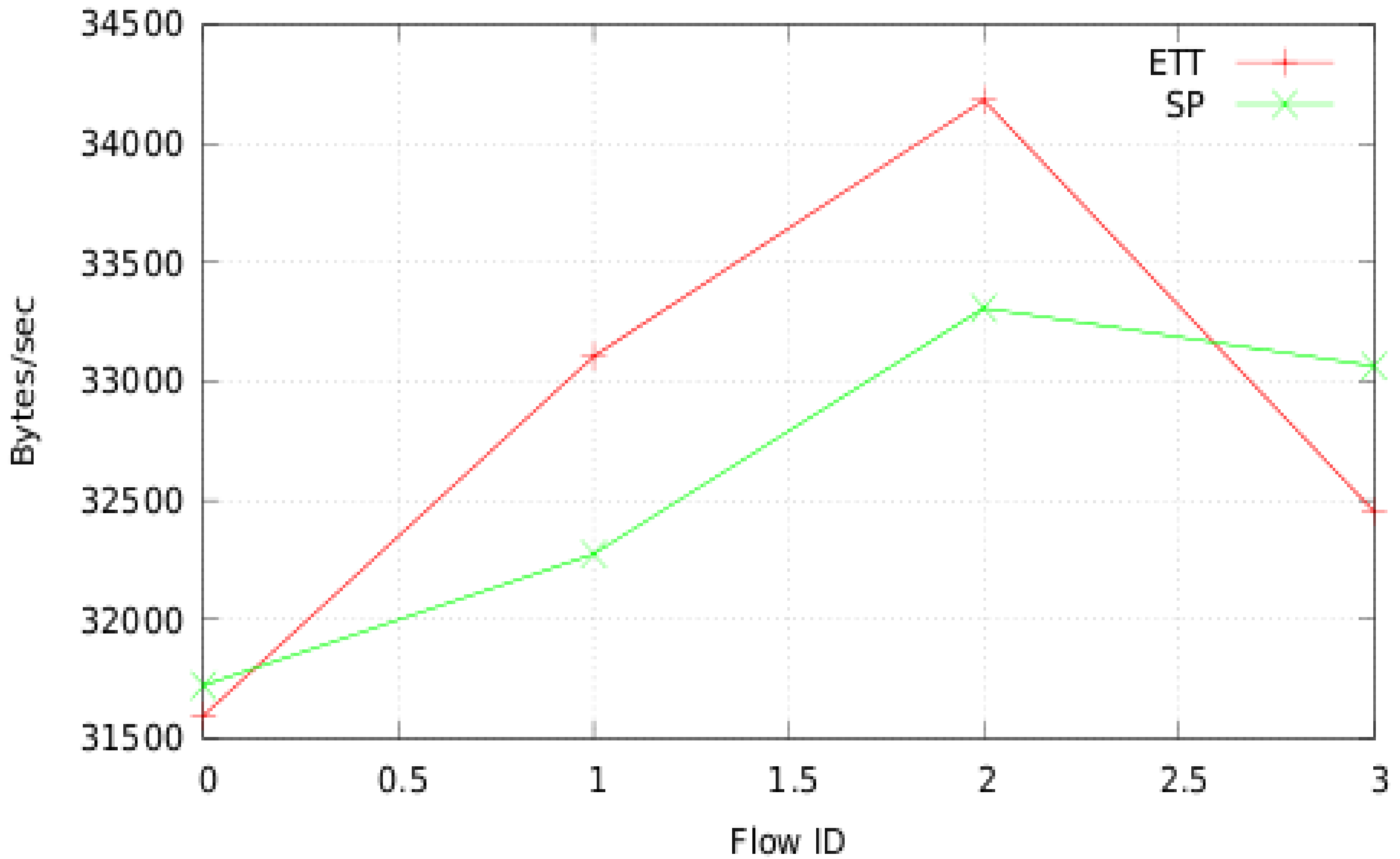
Note about alpha

- ✓ According to the goal of ETT policy (hight throughput), for the most of the alpha values, ETT works better than SP
- ✓ To avoid a possible negative effect due to the additional flow allocated by SP ,we have considered a particular alpha value (0.5)
- ✓ In this way we have the same number of allocated flows for both policies
- ✓ Fair balance between peak-rate and average-rate

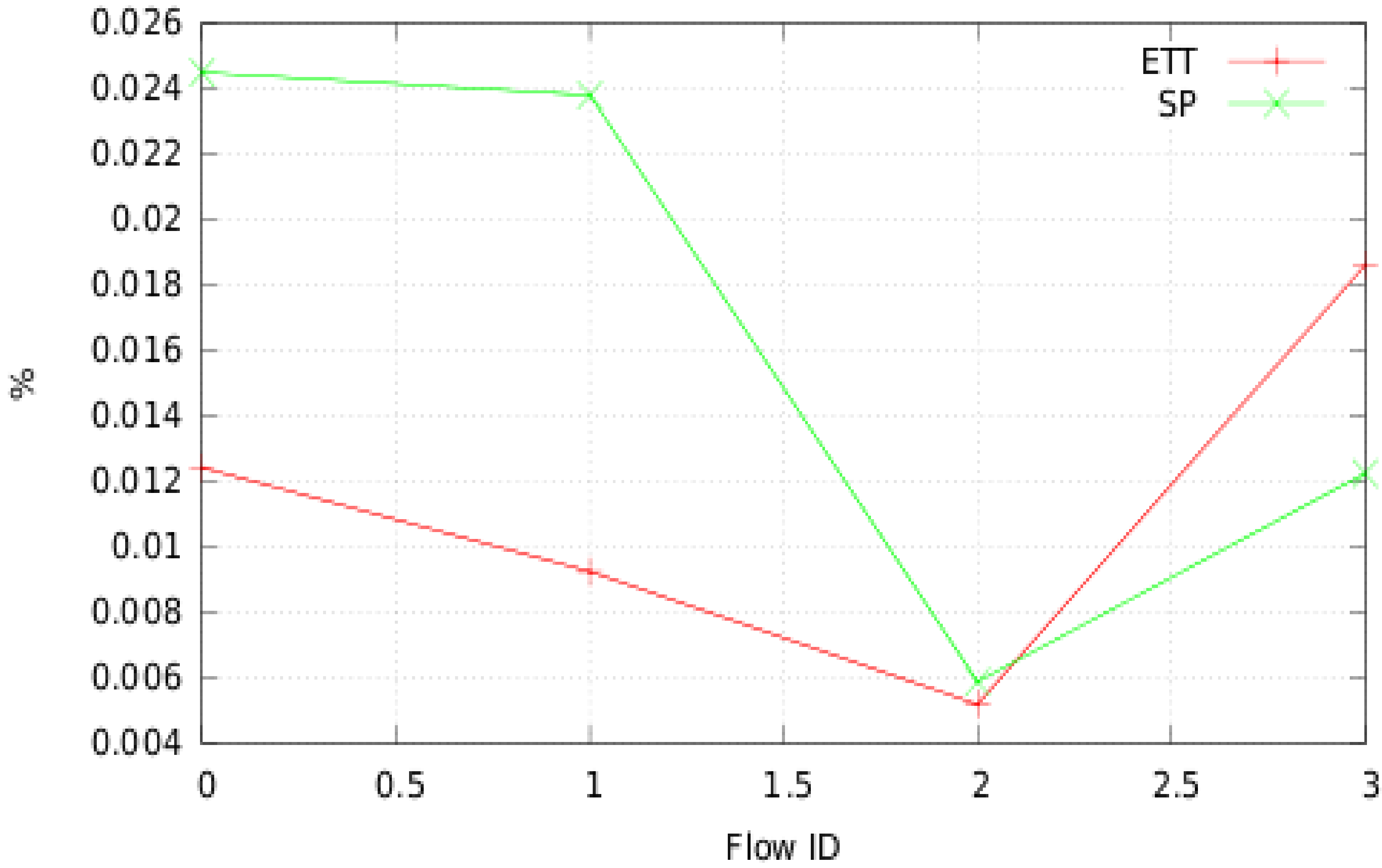
ETT vs SP – Average Delay



ETT vs SP - Throughput



ETT vs SP – Packet Loss Ratio



Conclusions

- ✓ In the ideal case ETT works clearly better than SP for different values of alpha (alpha greater than 0.3)
- ✓ Considering the more realistic scenario, ETT results better than SP in terms of its purpose
- ✓ We think the choice of ETT preferable to SP when we need a good service in term of delay and throughput

WCETT Scheduling Policy

Overview (1/2)

- ✓ WCETT: Weighted Cumulative ETT
- ✓ Goal of the policy: increase the ETT performance using more wireless cards and so different channels
- ✓ This policy favors paths that are more channel-diverse

Overwiev (2/2)

$$WCETT = (1-b) * \sum ETT_i + b * \max X_j$$

$$X_j = \sum ETT_i \qquad 1 \le j \le k$$

X_j is the sum of transmission time of hops on channel j

PROBLEM: *the simulator doesn't support multi-channel*

~~WCETT~~

~~Scheduling Policy~~

WCSLOT

New Scheduling Policy

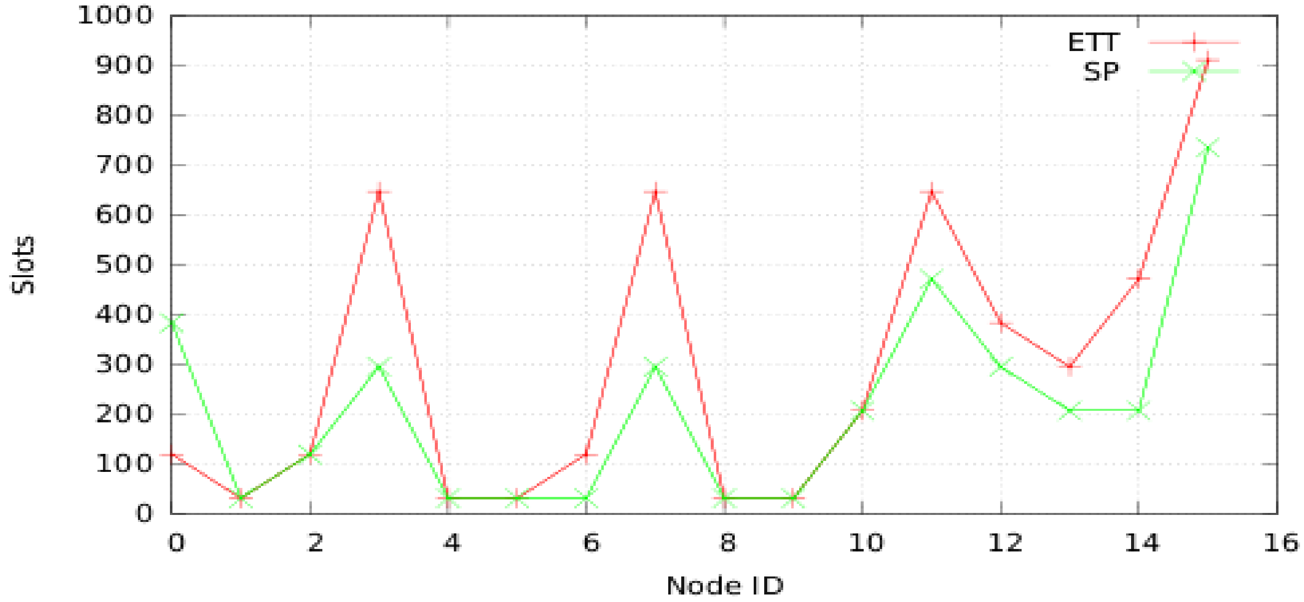
Idea for a new scheduling policy (1/2)

- ✓ Use resources in an optimized way
- ✓ Analysis of the Free Slots metric which gives us an index of the available resources on the specific node
- ✓ Goal: "spread" the traffic as much as possible on the various nodes of the network

Idea for a new scheduling policy (2/2)

This is the graphic that gave us the idea of using slots for the realization of a new protocol

Average Number of Free Slots



NOTE: simulation performed with the alpha parameter set to 0.5, with error disabled

Research Phase

- ✓ ***Research activity*** with the purpose of finding a new scheduling policy

- ✓ Various steps and different implementations of the algorithm to reach the final solution

WCSLOT
First Step

Overview

- ✓ Solution that considers ETT and the free slots

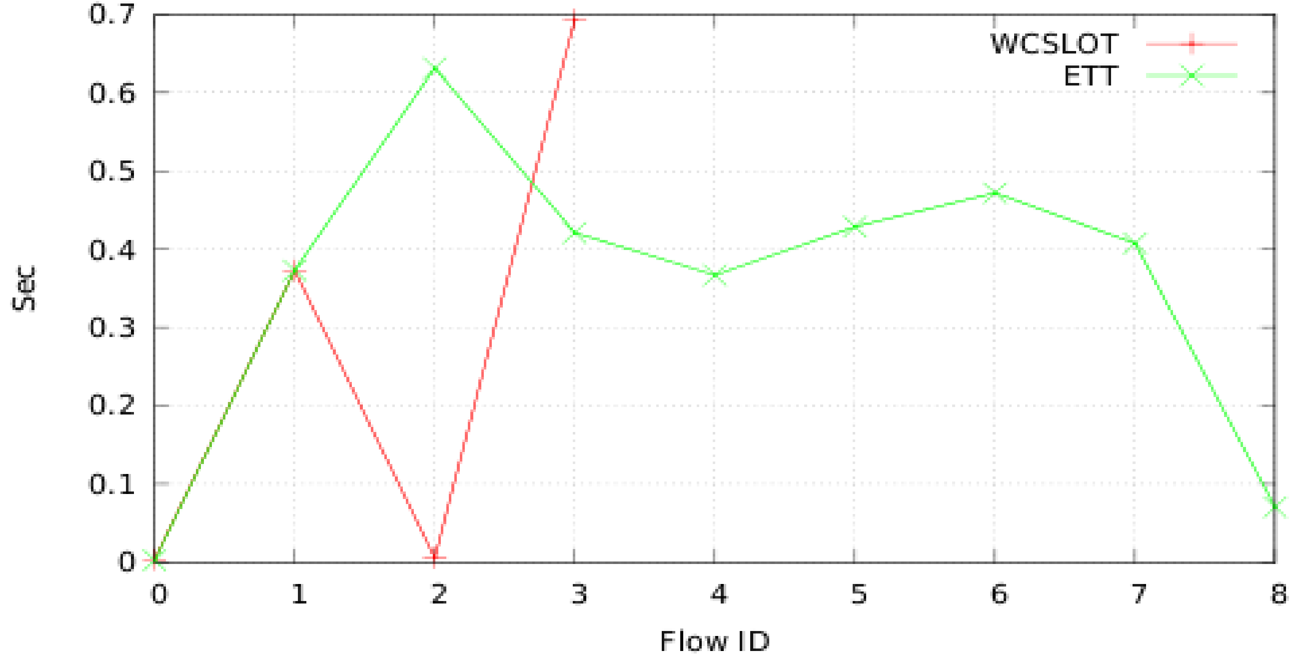
$$WCSLOT = (1 - b) * ETT + b * NumFreeSlots$$

NumFreeSlots is the sum of the free slots along the path calculated node by node

Considerations (1/2)

- ✓ The number of allocated flows is much less than those allocated by ETT

Average Delay



NOTE: simulation performed with the alpha and beta parameters set to 0.7, with error disabled

Considerations (2/2)

- ✓ The cause of this non-performing results is that the two components of the WCSLOT formula can't be compared because their orders of magnitude are completely different

WCSLOT
Second Step

Overview

- ✓ Solution that considers only the free slots, normalized for the number of nodes

$$WCSLOT = \frac{NumFreeSlots}{NumNodes}$$

NumFreeSlots is the sum of the free slots along the path calculated node by node

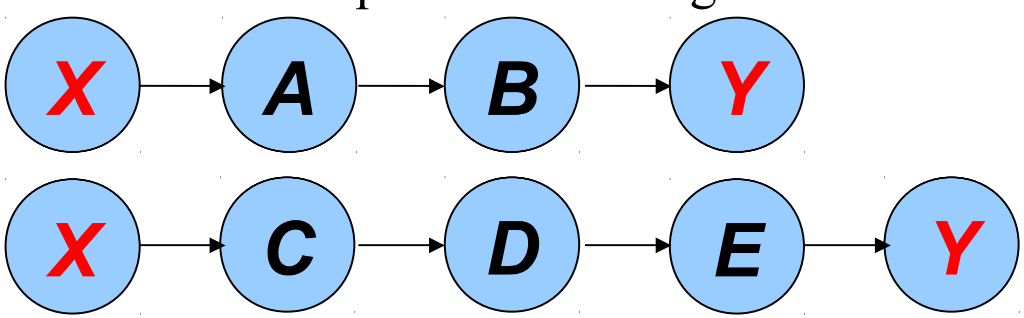
NumNodes is the number of nodes traversed to build the path

- ✓ The scheduler will choose the path with the maximum WCSLOT value

Worst Case Scenario

NODE	FREE SLOTS
A	3
B	11
C	5
D	7
E	5

Possible paths connecting a source *X* and a destination *Y*:



WCSLOT = 4.5

WCSLOT = 4.25

Considerations

- ✓ The bottleneck results in performance worse than expected

WCSLOT
Final Implementation

Overview

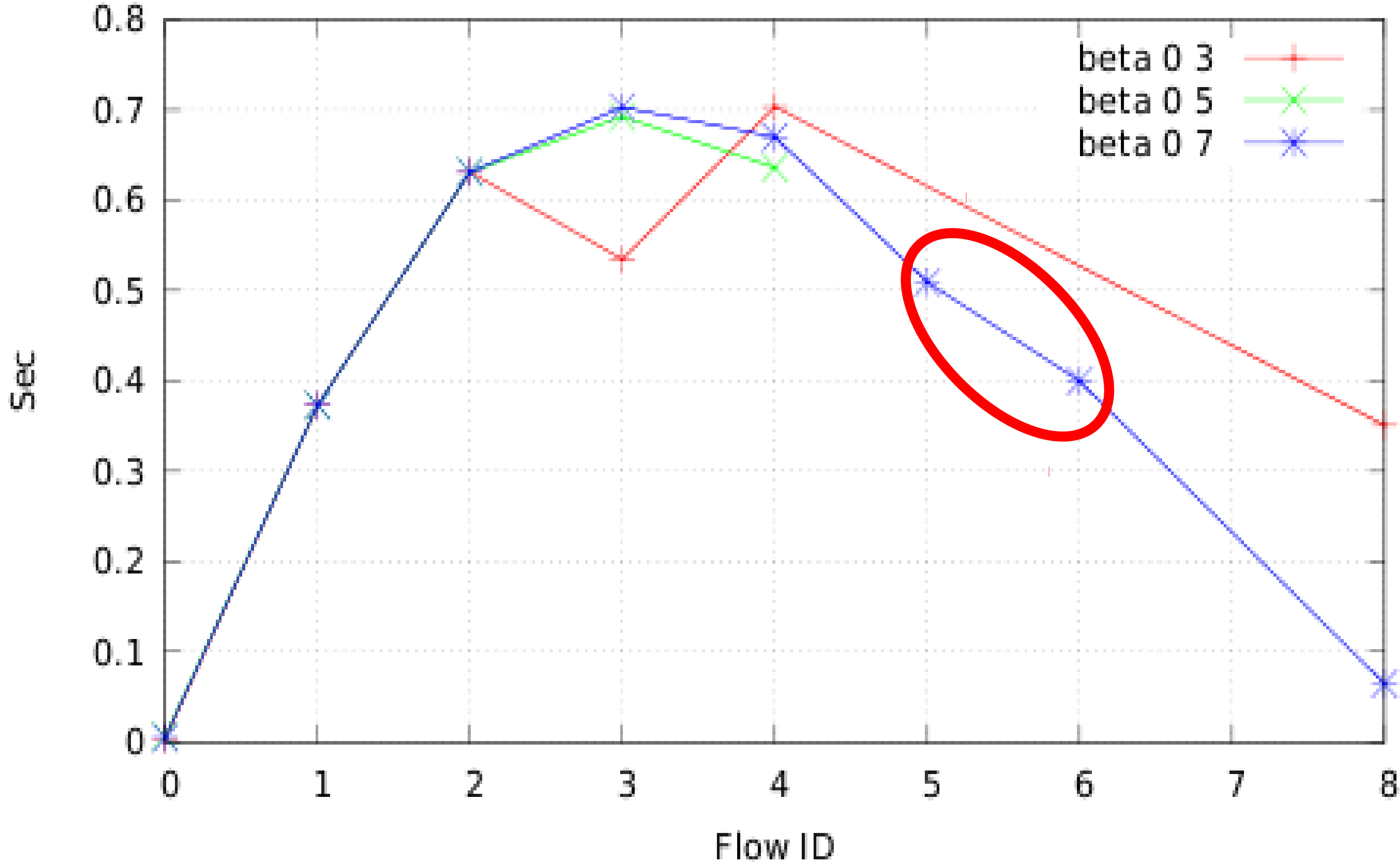
- Solution that considers the free slots, normalized for the number of nodes and the minimum number of free slots on a node belonging to the path

$$WCSLOT = (1-b) * \left(\frac{NumFreeSlots}{NumNodes} \right) + b * NumMinFreeSlots$$

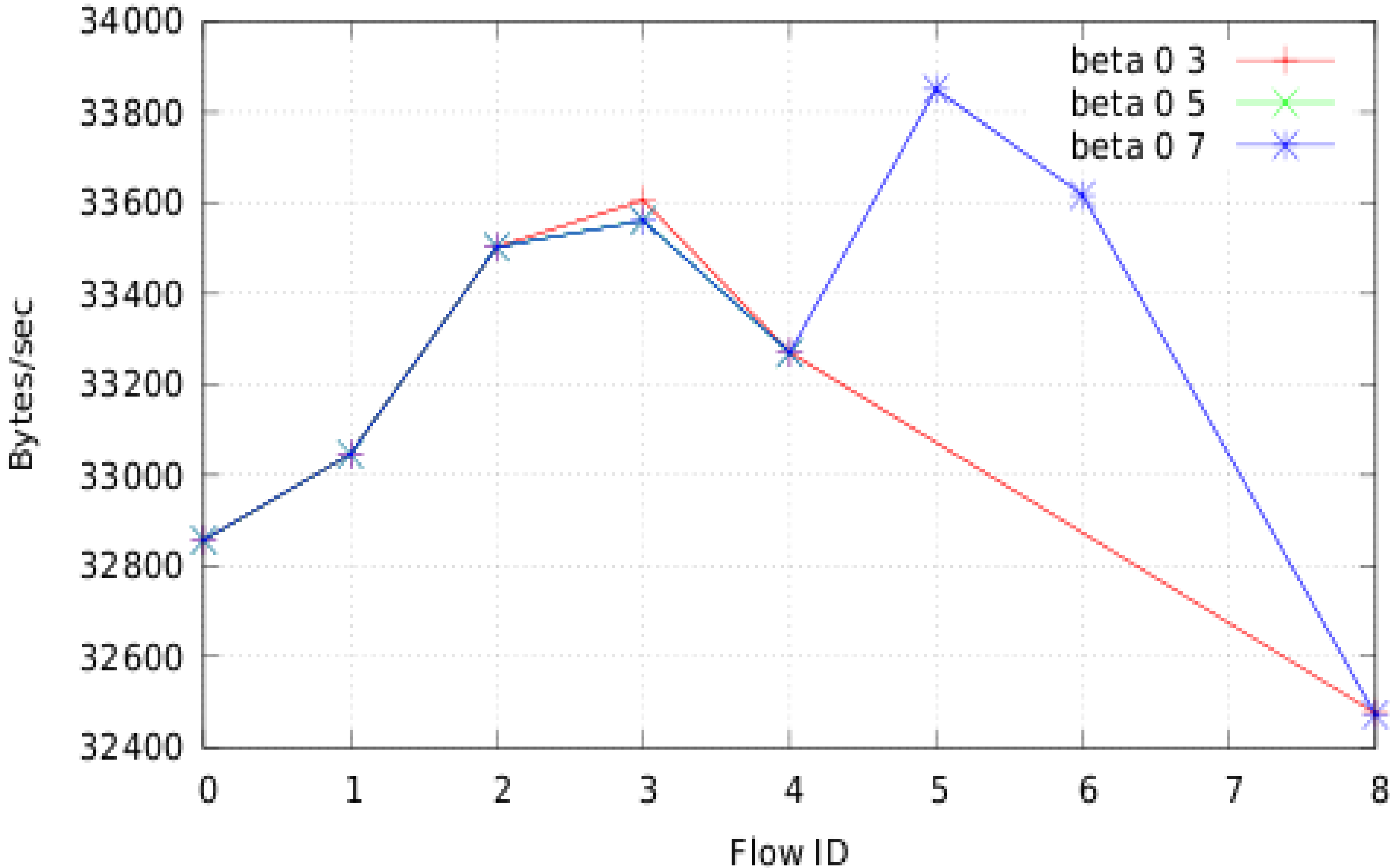
NumMinFreeSlots is the minimum number of free slots on the node considering all the nodes that compose the path

- The scheduler will choose the path with the maximum WCSLOT value

Choosing Beta (1/2) - Average Delay

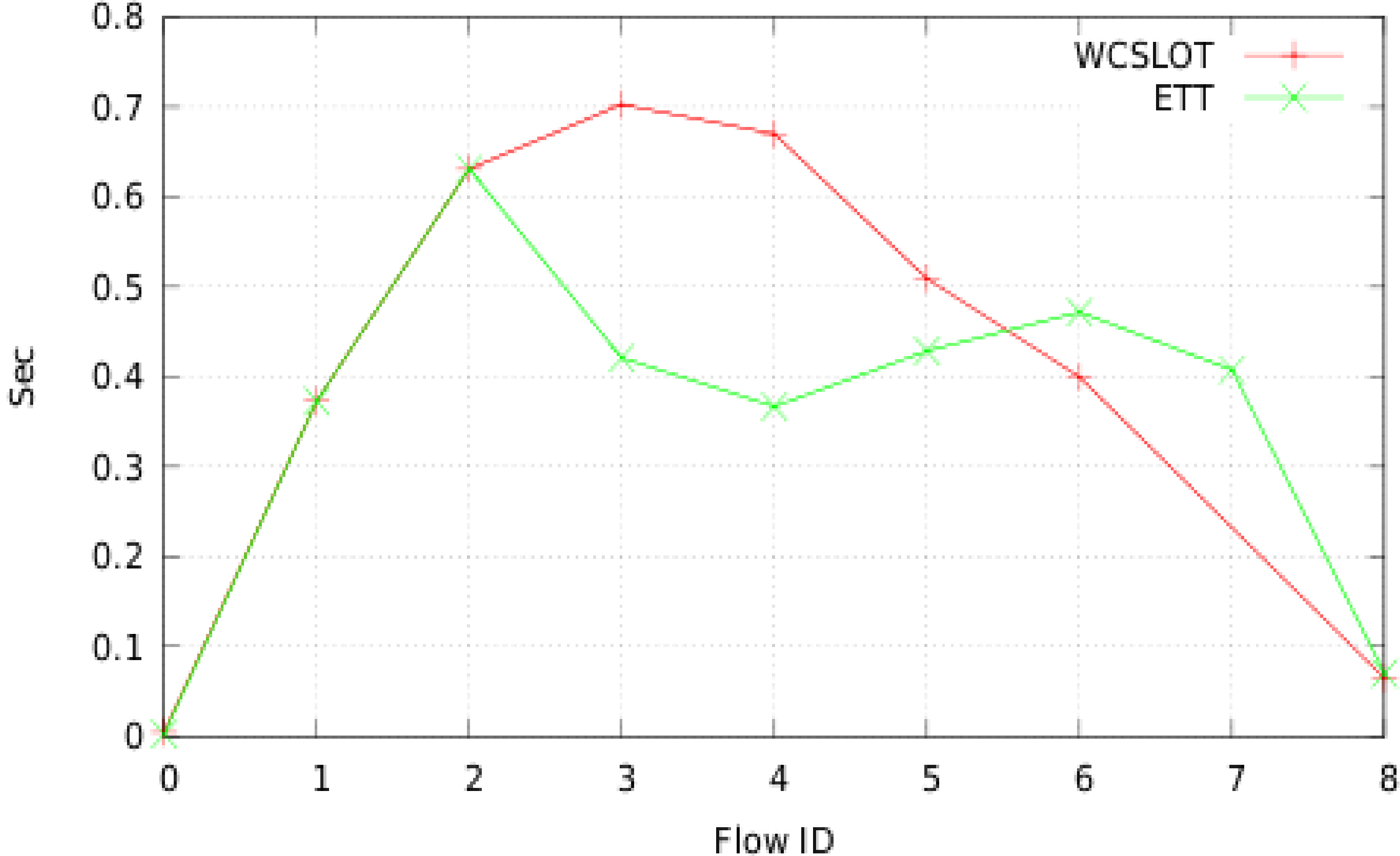


Choosing Beta (2/2) - Throughput

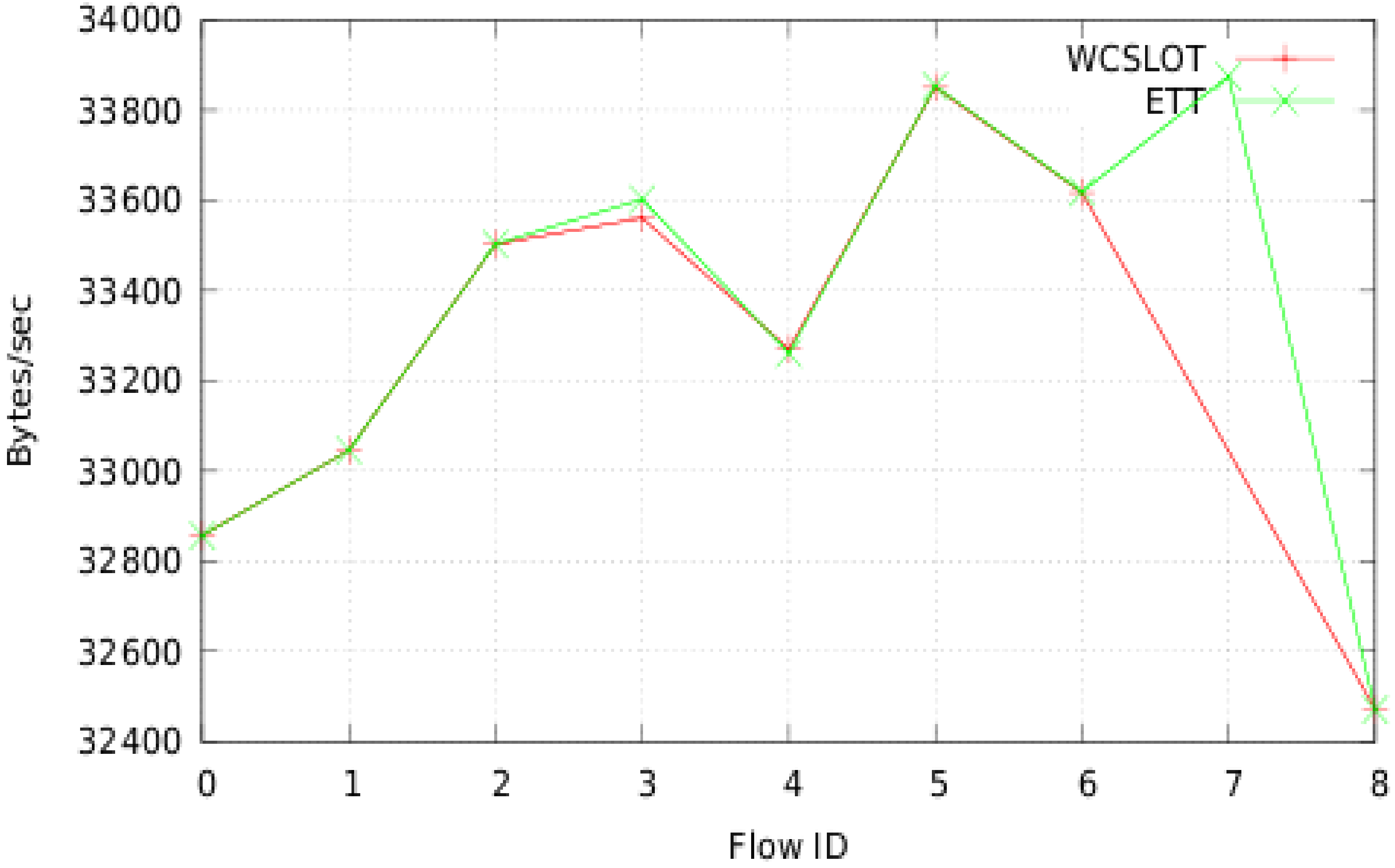


WCSLOT vs ETT

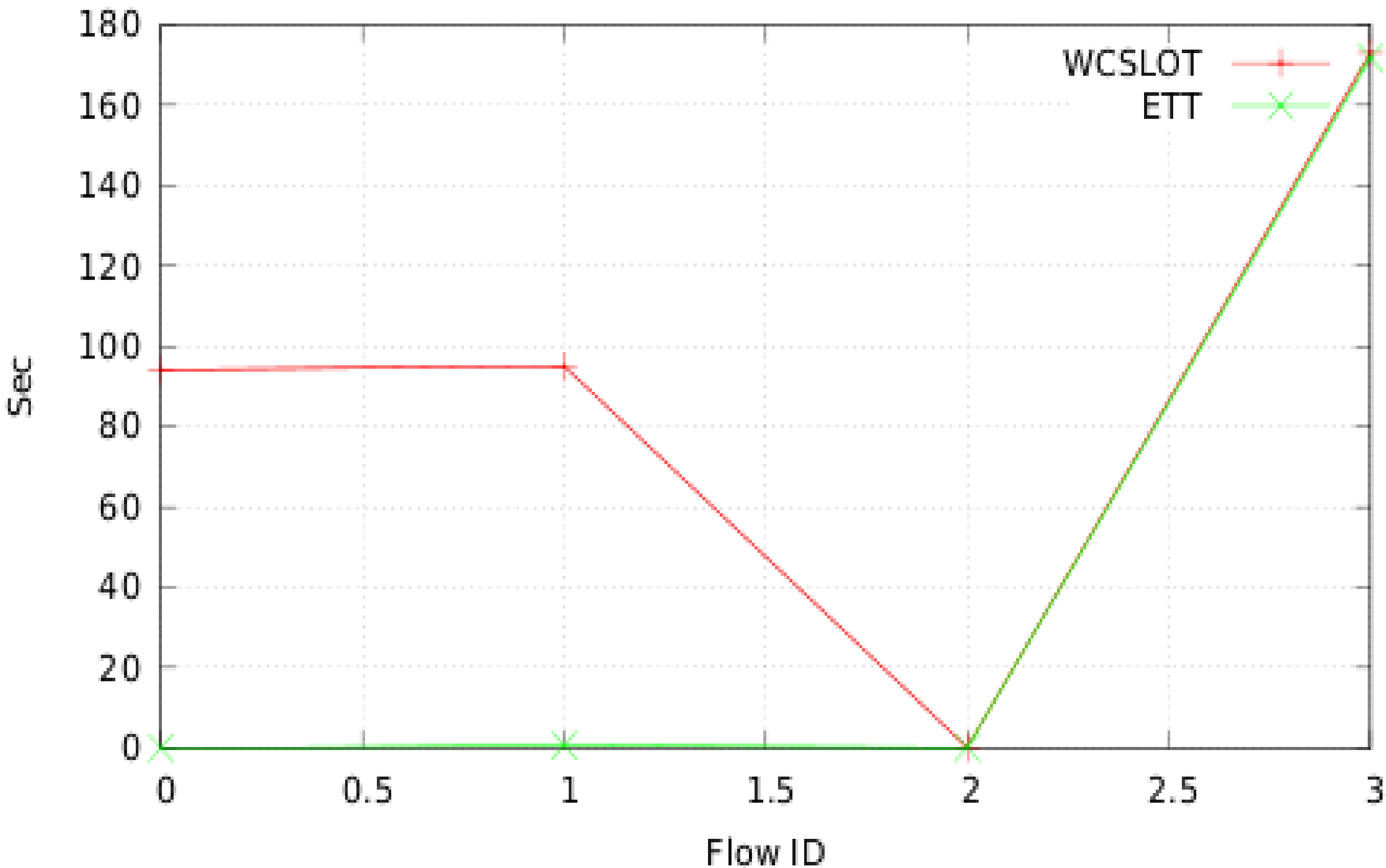
WCSLOT vs ETT – Average Delay



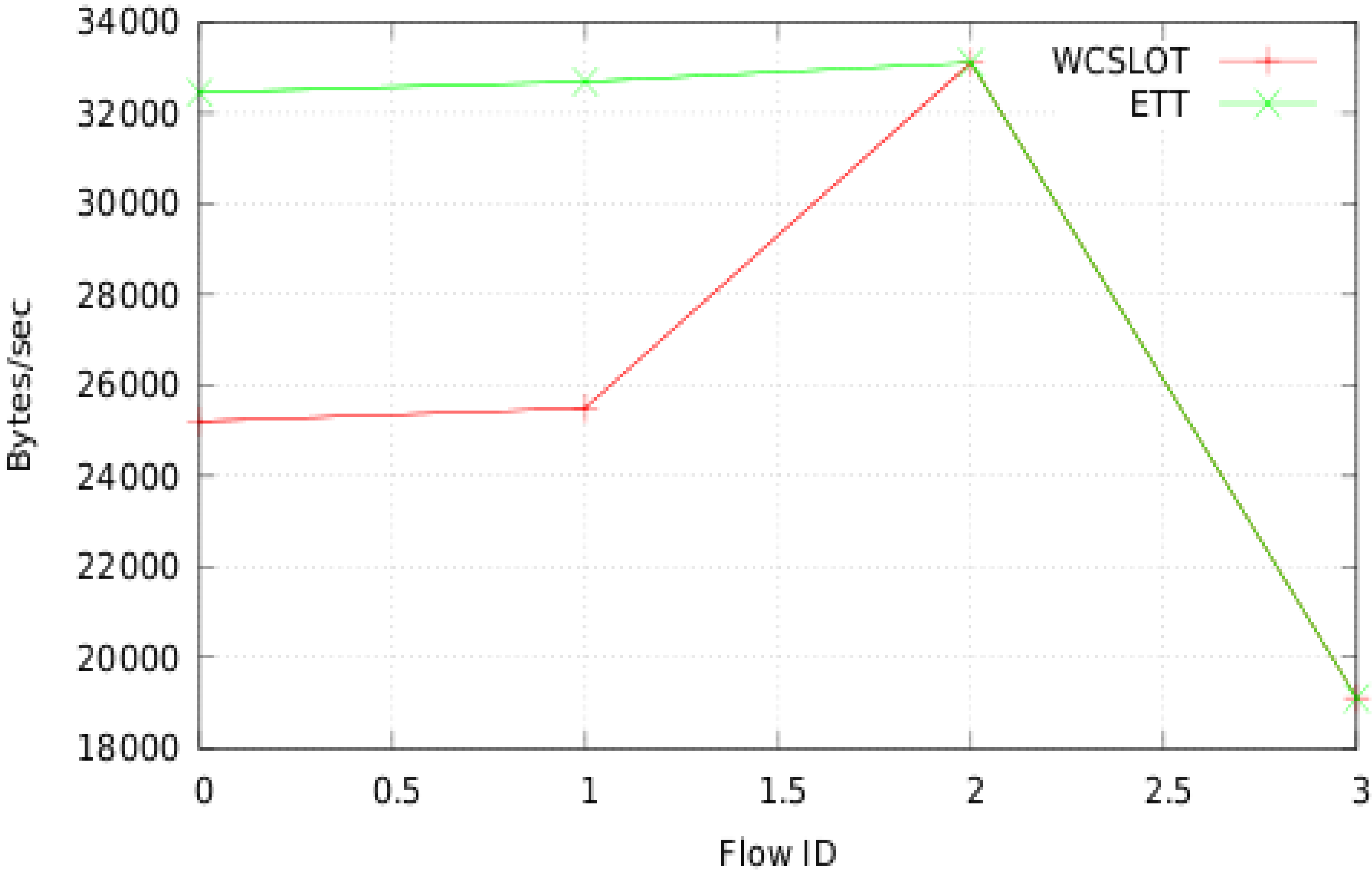
WCSLOT vs ETT – Network Throughput



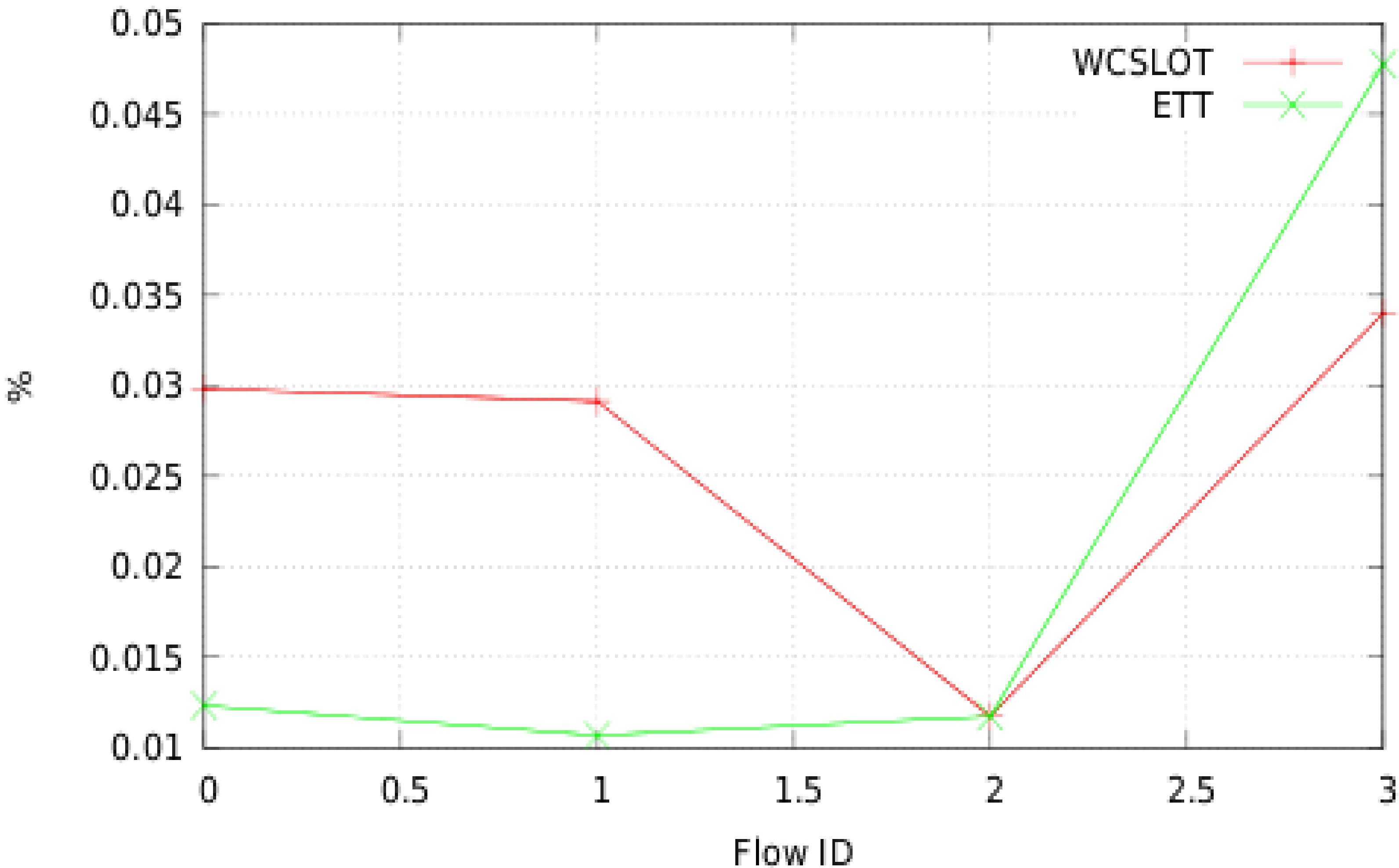
WCSLOT vs ETT – Average Delay (Error Enabled)



WCSLOT vs ETT – Throughput (Error Enabled)

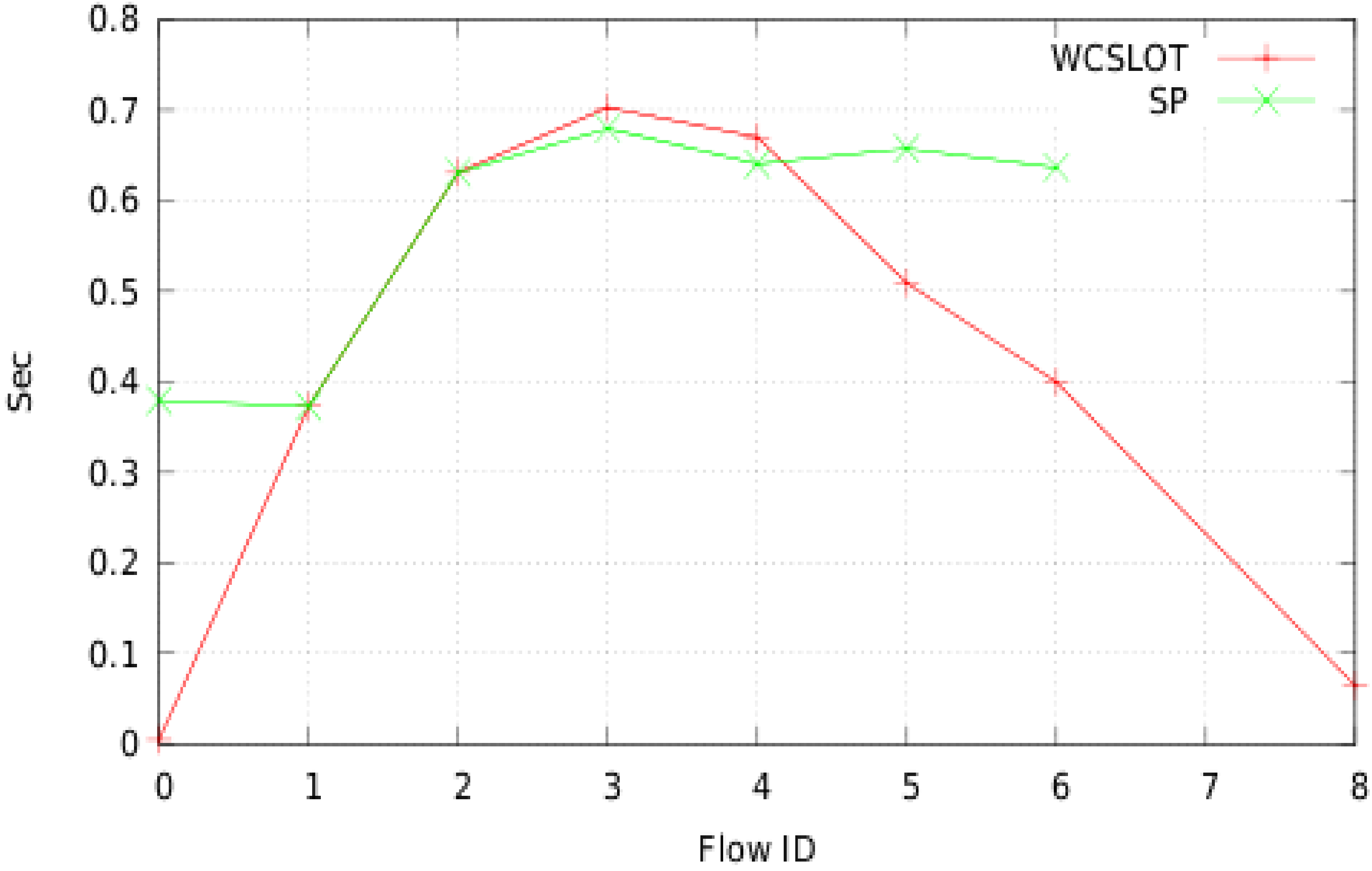


WCSLOT vs ETT – Packet Loss Ratio

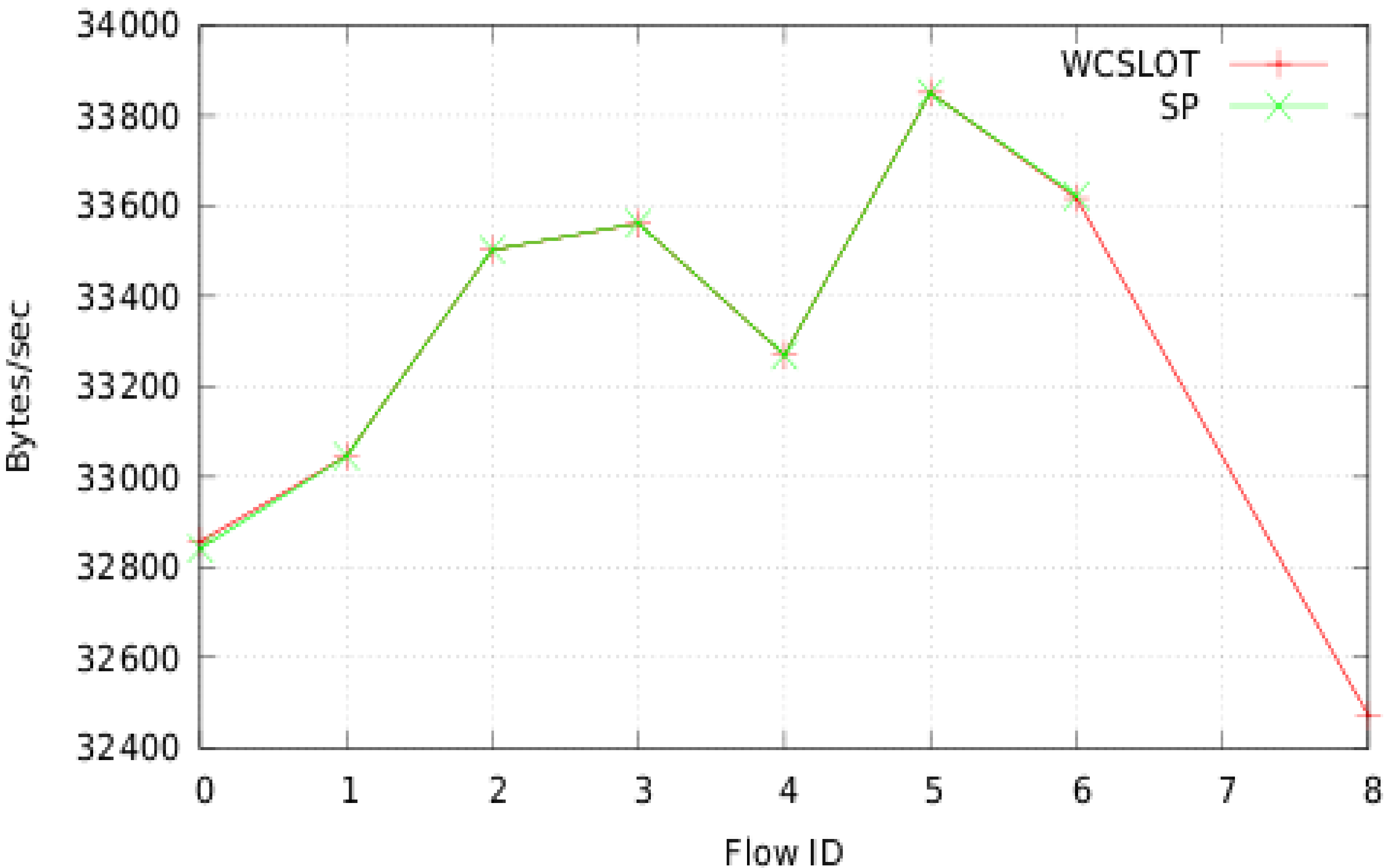


WCSLOT vs SP

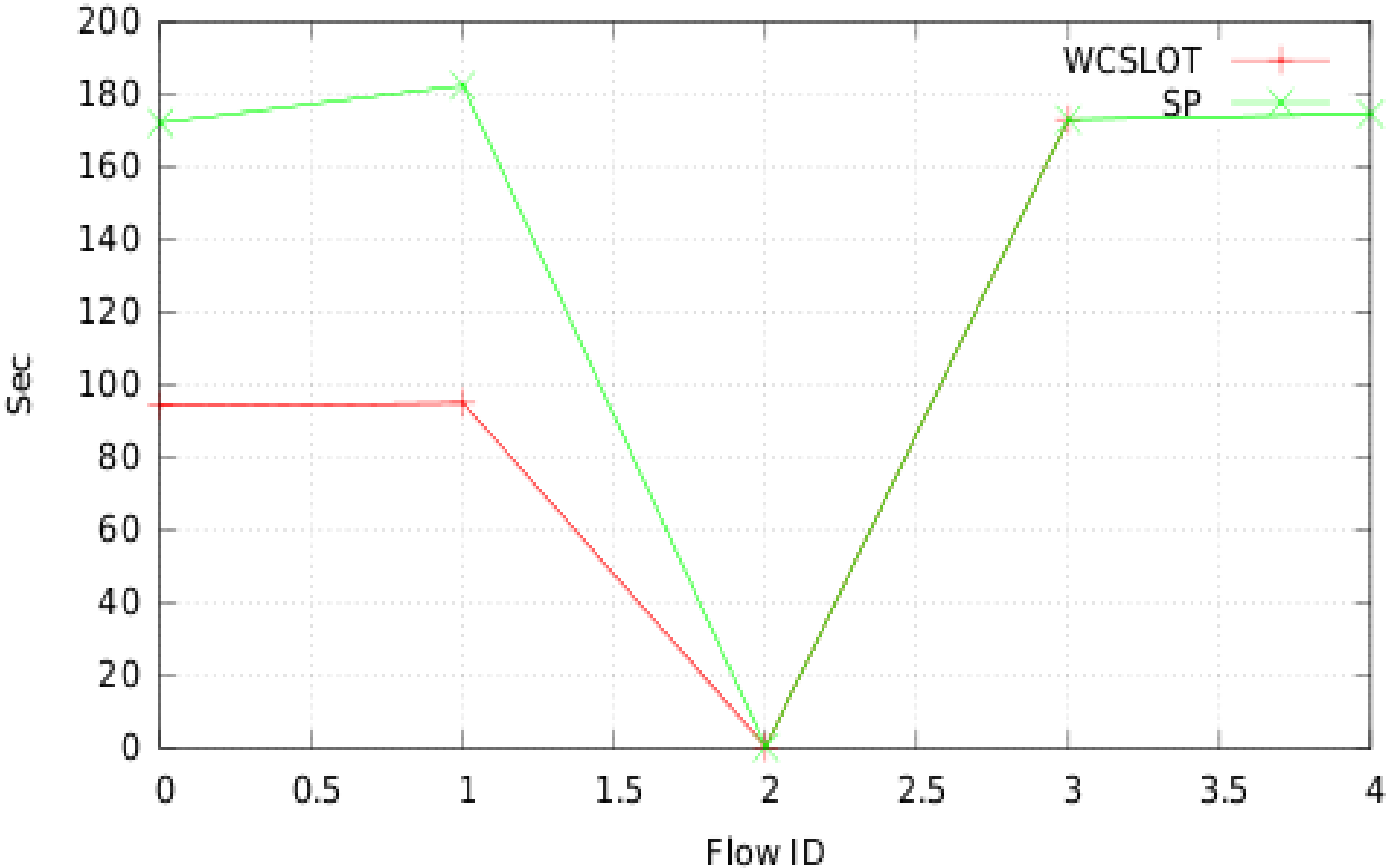
WCSLOT vs SP – Average Delay



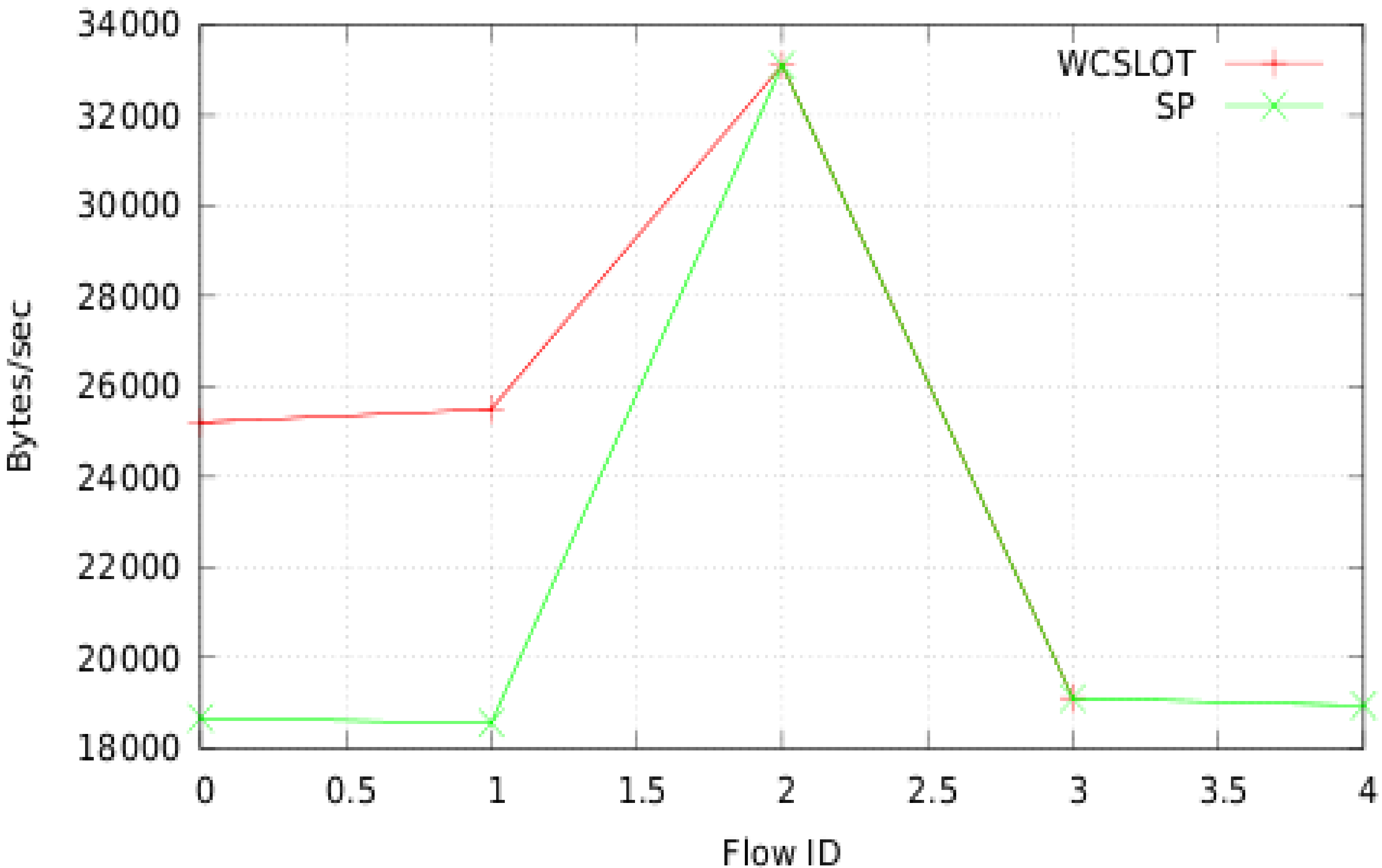
WCSLOT vs SP - Throughput



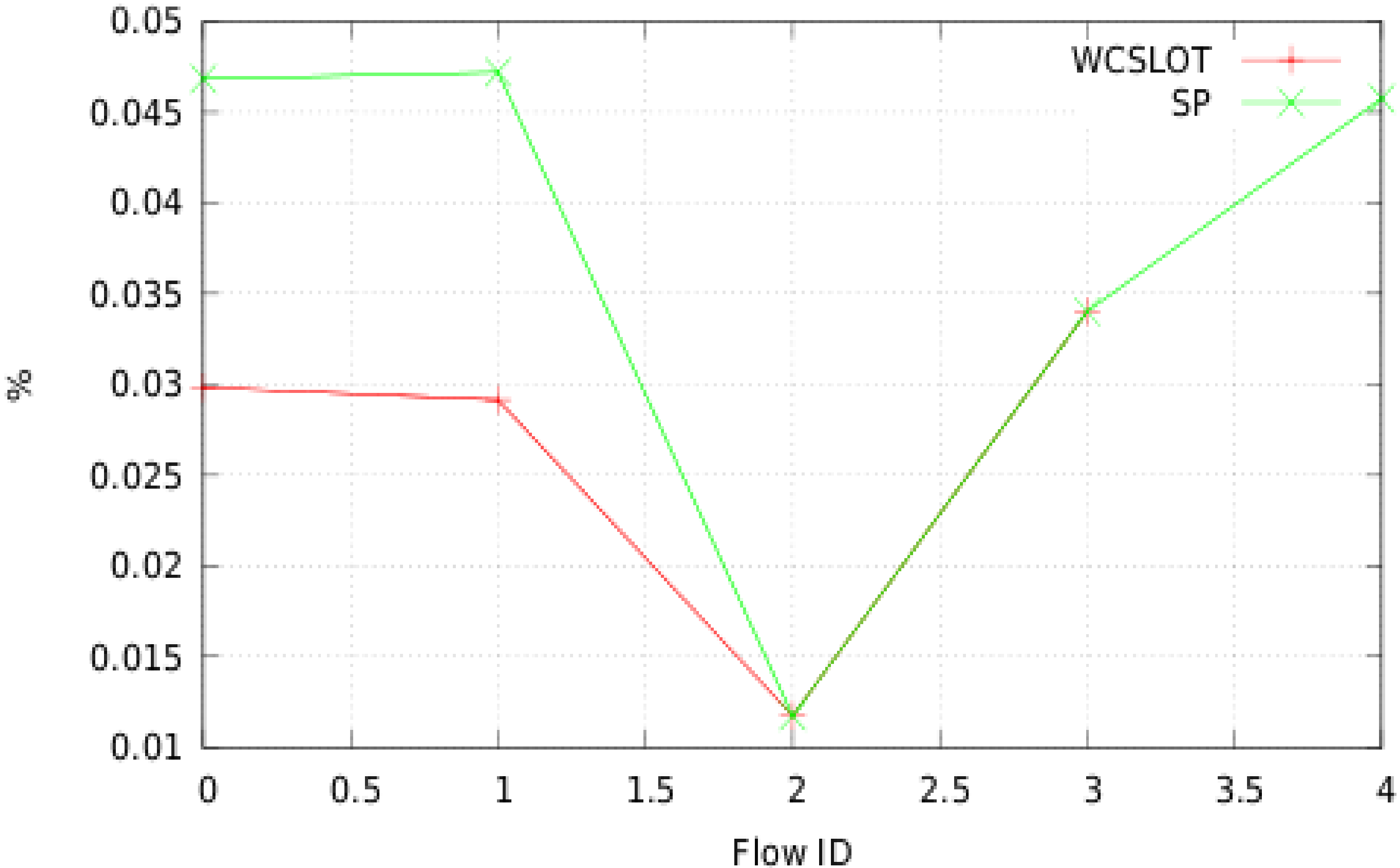
WCSLOT vs SP – Average Delay (Error Enabled)



WCSLOT vs SP – Throughput (Error Enabled)



WCSLOT vs SP – Packet Loss Ratio



Detail

Slots Allocation Policies

- **FIRST_NO_INT (default):**

  the first slots that are available locally, but are also marked as available by the neighboring nodes, are allocated (if this is possible, otherwise the first available slots are allocated)

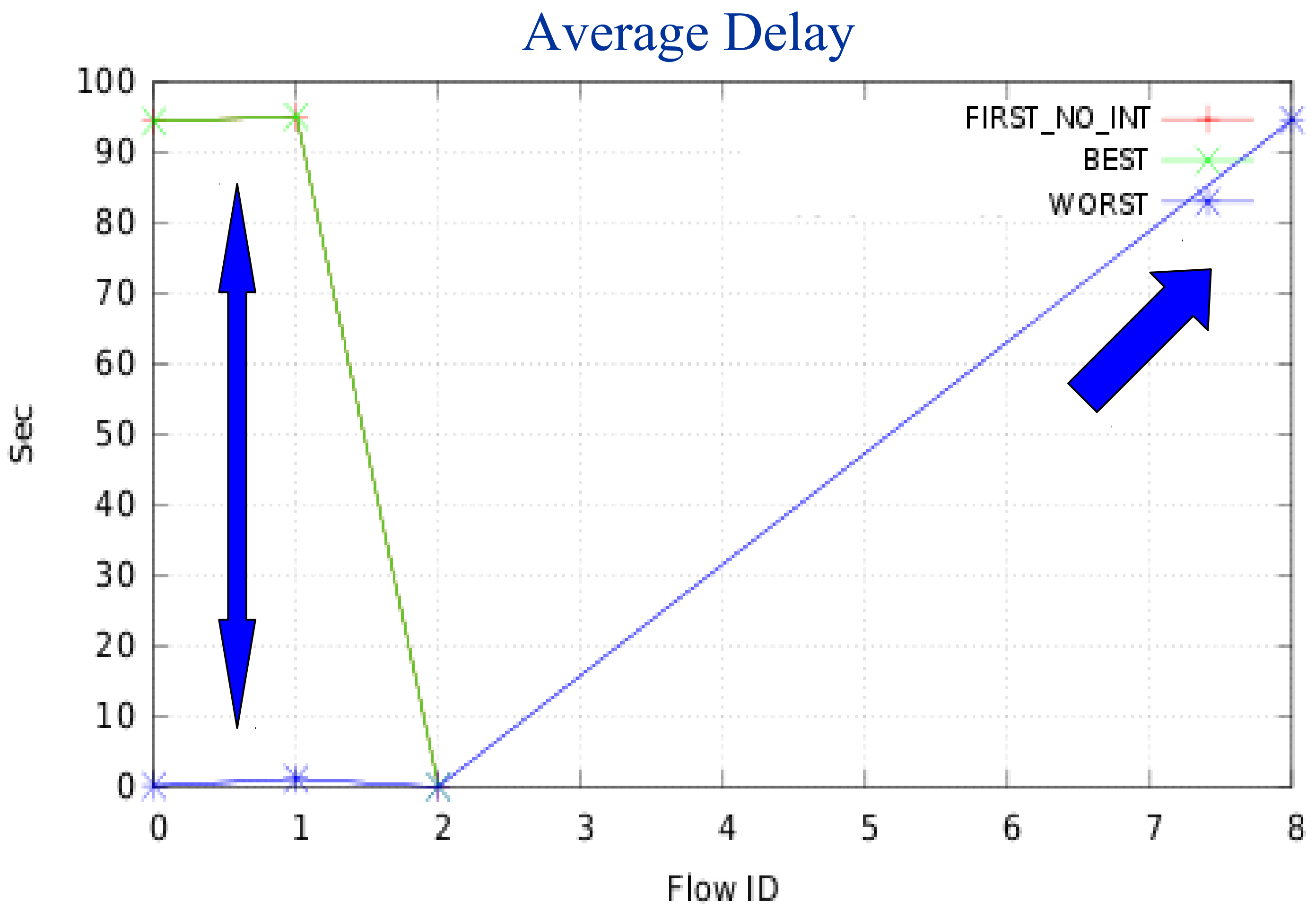
- **BEST:**

  slots are allocated early in the first group of free slots that best fit the plot (this is the same research policy of the free space for a new file in the file system)

- **WORST:**

  unlike BEST, the largest possible group of slots among the group of free contiguous slots is allocated

WCSLOT Best Case Ever



Conclusions

- ✓ WCSLOT works worse than ETT

Our considerations:

- ✓ The additional flow allocated by SP not be should considered as a preferential factor (the resulting delays are not admissible in a real context)
- ✓ WCSLOT works better than SP
- ✓ The grid scenario isn't enough for an exhaustive analysis. Would be best to consider other scenarios
- ✓ ETT can be improved only using multi-radio, multi-channel