MPTCP Implementation: Use cases for Enhancement Opportunities

https://www.ietf.org/id/draft-palanivelanchetansenthil-mptcp-enhancements

A.Palanivelan (palanivelan.appanasamy@intl.verizon.com)
Chetan Harsha (chetan.harsha@intl.verizon.com)
Senthil Sivakumar (ssenthil@cisco.com)

Introduction

- The Intention of this proposal is to document Opportunities, where Enhancements to MPTCP can translate to more wider deployments

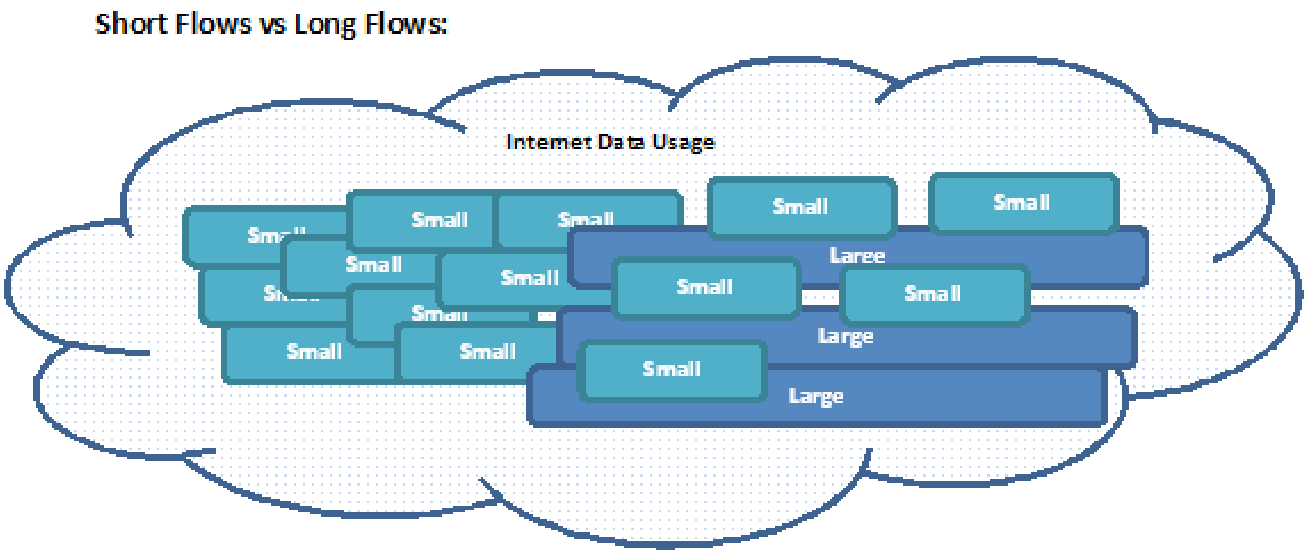
- The Scope of the use cases/problem statement(s) discussed is limited to impact on enduser experience only

What is this about?

- Document use cases (Draft-01)
  - Document test results (WG discussion)
- Design recommendations/solutions (new draft?)

Use cases

Short Flows vs Long Flows:



- Too many Small Flows => Higher Number of Transactions. But, much less Bandwidth Consumption
- Far Lesser Long Flows => Lesser Number of Transactions. But, higher Bandwidth Consumption

Use cases

- **Too many Small Flows => Higher Number of Transactions. But much less Bandwidth Consumption**
  - ***Can we achieve Low latency for short flows?***
    - Average completion of flow with mptcp can be higher than completion time without mptcp
    - With Bunch of Short Flows, MPTCP may negatively impact throughput
    - Even a single lost packet can force an entire connection to wait for an RTO.

- **Far Less Long Flows => Lesser Number of Transactions, But Higher Bandwidth Consumption**
  - ***Can we achieve higher Throughput for Long Flows?***
    - Without compromising on performance
  - ***How do we maintain Reliability?***
    - How do we manage tolerance to sudden and high bursts of traffic?

- Both long and short flows are important from the enduser perspective. We need to come up with appropriate definition and clear demarcation for short and long flows, from MPTCP Perspective. These need be dealt differently (Probably with multiple profiles).

Use cases

Application based Path selection and Adaptive buffering:

- **Elastic vs Inelastic Applications**
  - Different way of handling packets => Better Performance.

- **MPTCP performance is impacted …**
  - When the size of the receive buffer is limited.
  - Path with high RTT may result in the receive buffer size growing beyond the allowed maximum
  - Diversified RTT

Application based Path Selection and Adaptive Buffering can help with the above scenarios. Tweaking the buffer sizes based on the type of application and/or network condition can positively impact the flows.

Use cases

Path Selection Enhancements?

Usecases where MPTCP path selection can be enhanced:

- For High packet loss and High latency networks?
- Multiple profiles to dynamically switch (move across) the networks?
- Roaming scenarios

The best optimal path is ever changing in Internet. Frequent switching may cause unnecessary overheads and can impact performance. Enhanced yet controlled Path Selection and Path Switching can help get better performance out of the network.

Use cases

Optimal number of paths:

- **Controlling the number of subflows getting created…**
  - How many is too many?
  - Can this be controlled? What Inputs to Consider?
    - Based on Network Characteristics
    - Historic data (region wise)

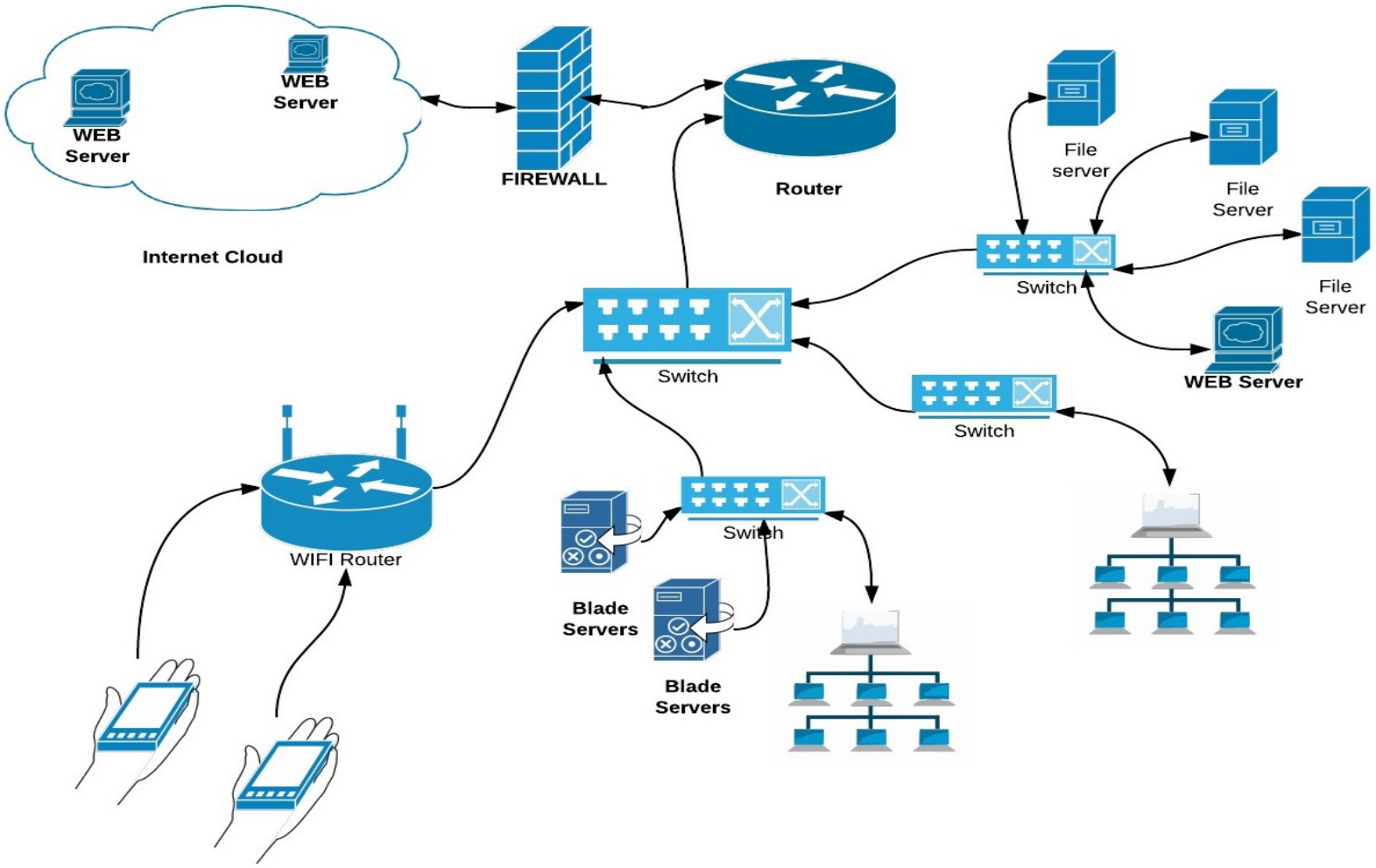
Scenarios covered

User experience (MPTCP enabled clients)

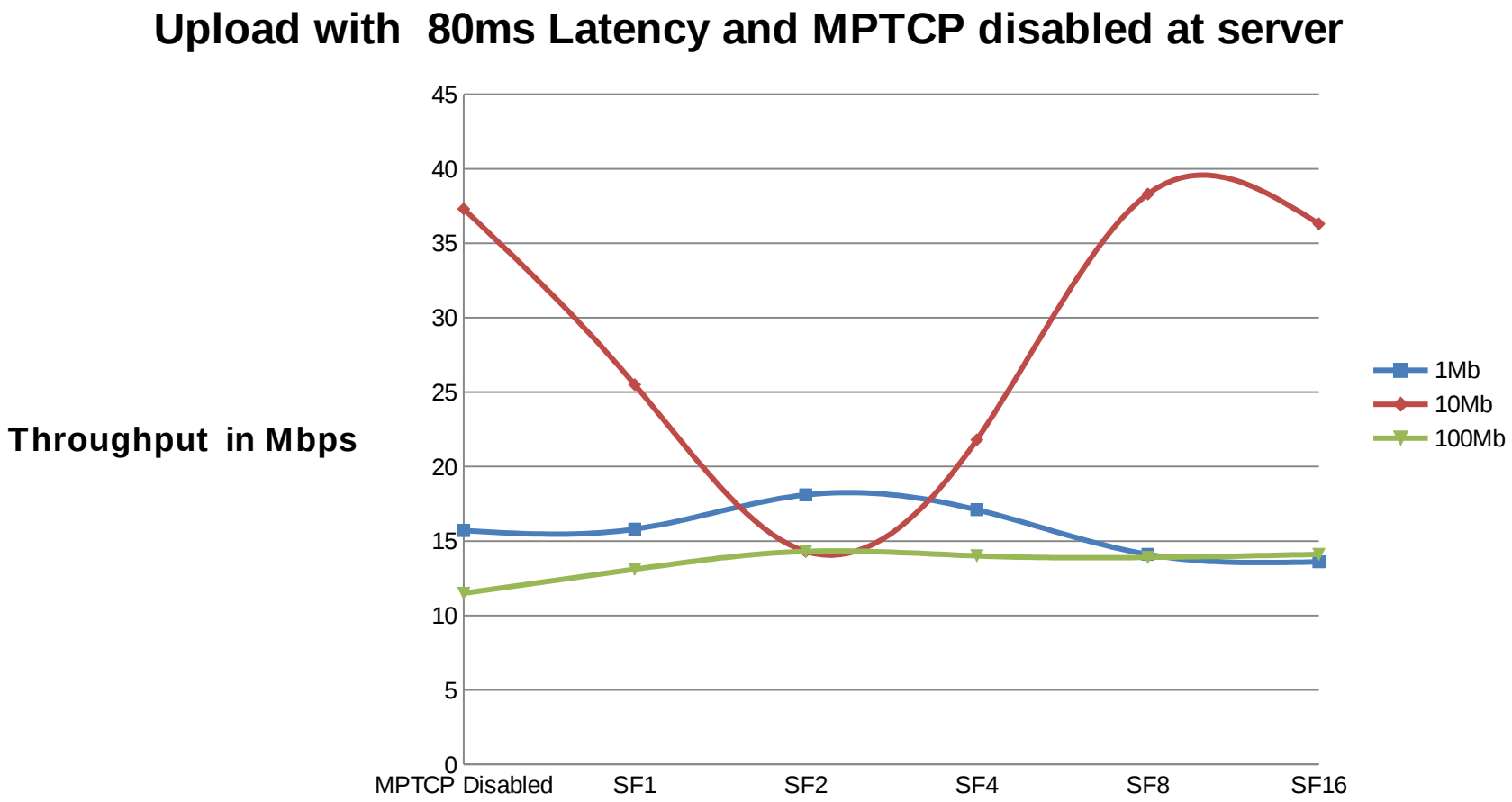
- Different RF conditions in upload/download
- Different types of content/applications
- Different number of subflows
- MPTCP enabled/disabled servers

* *Recommendations to be implemented either at protocol level (Client Side Changes) and/or at ISP side. This is scoped for future work/discussions.*

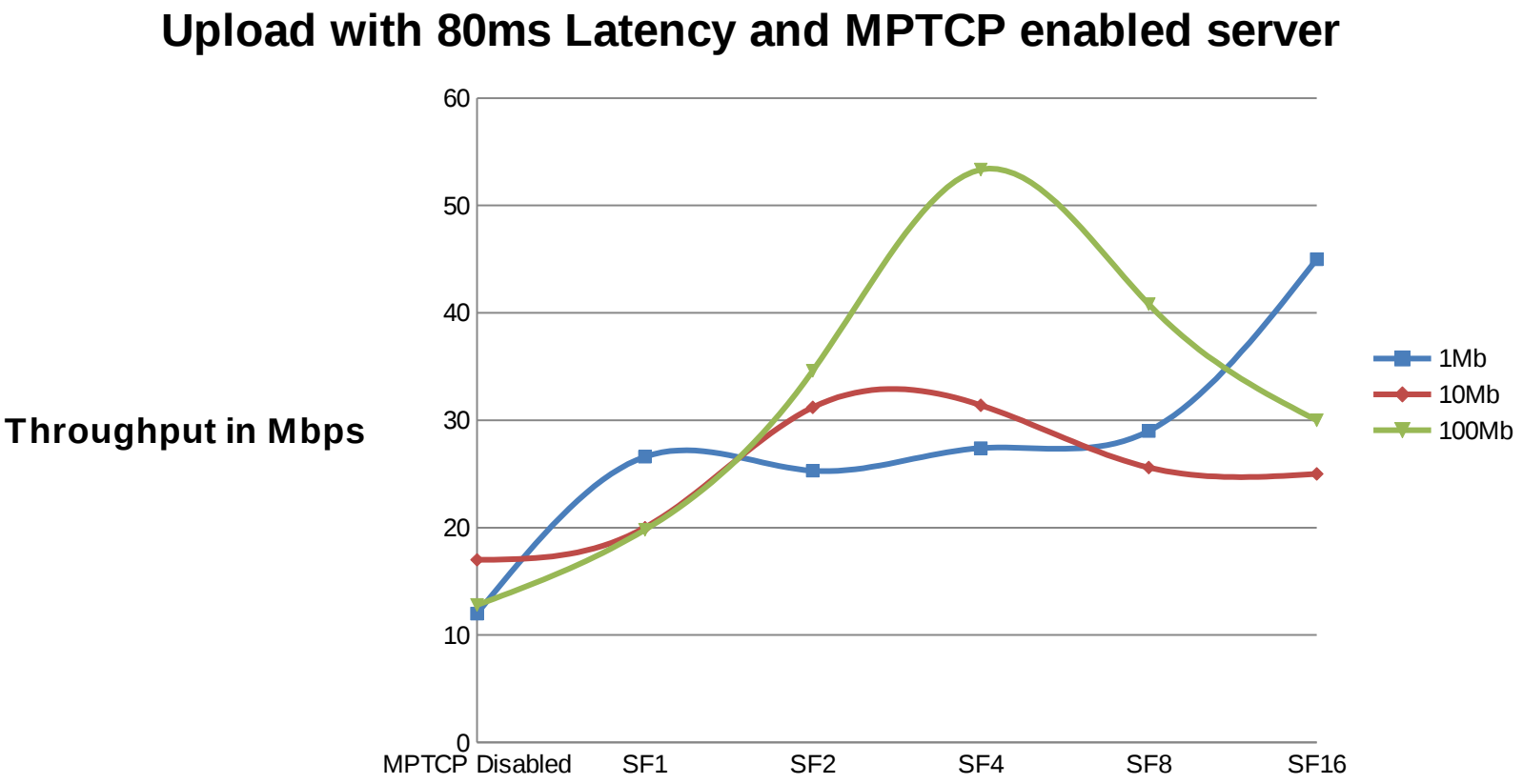
Network Topology



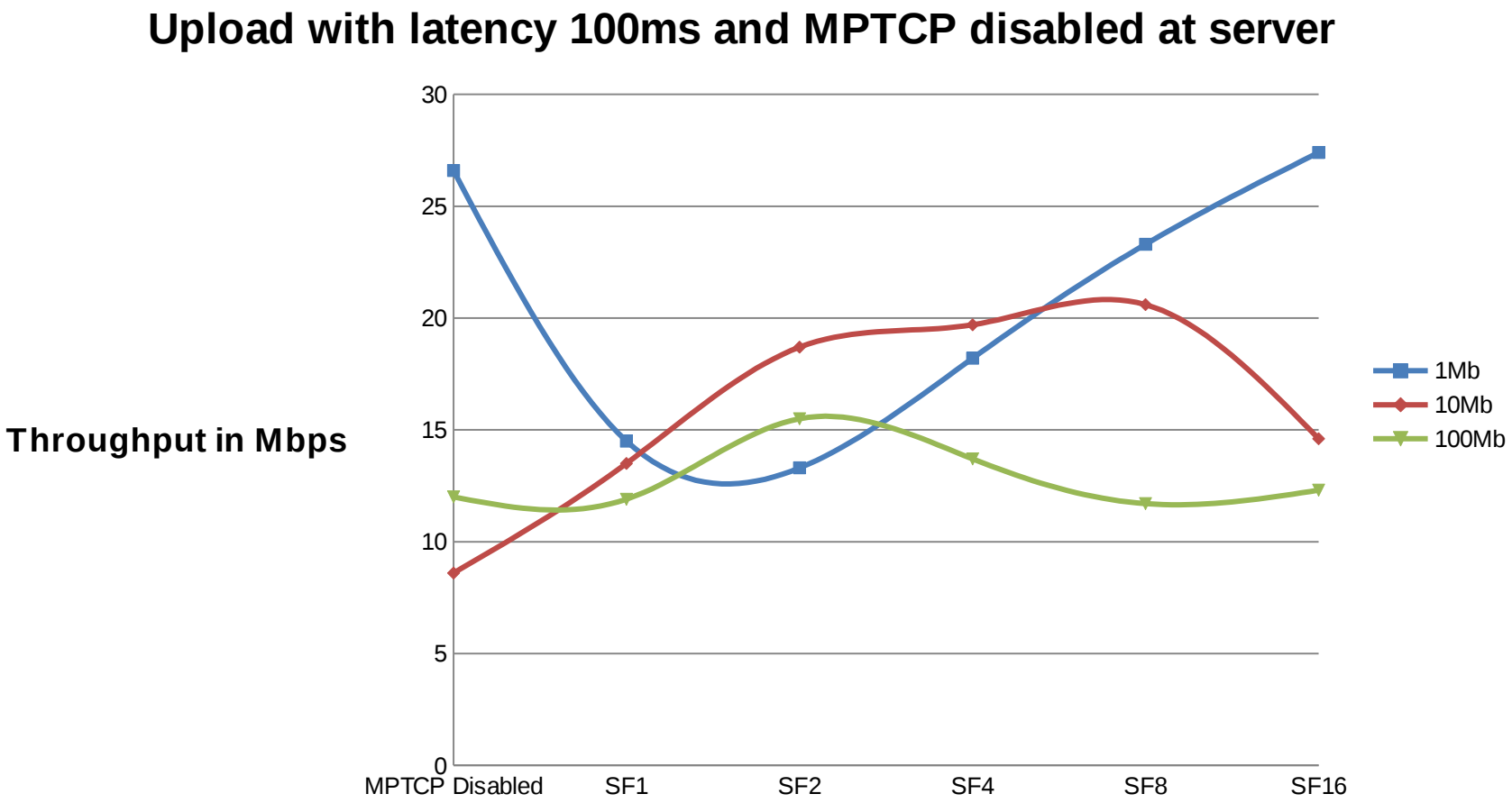
Upload with 80ms (3G mid) latency MPTCP disabled at server



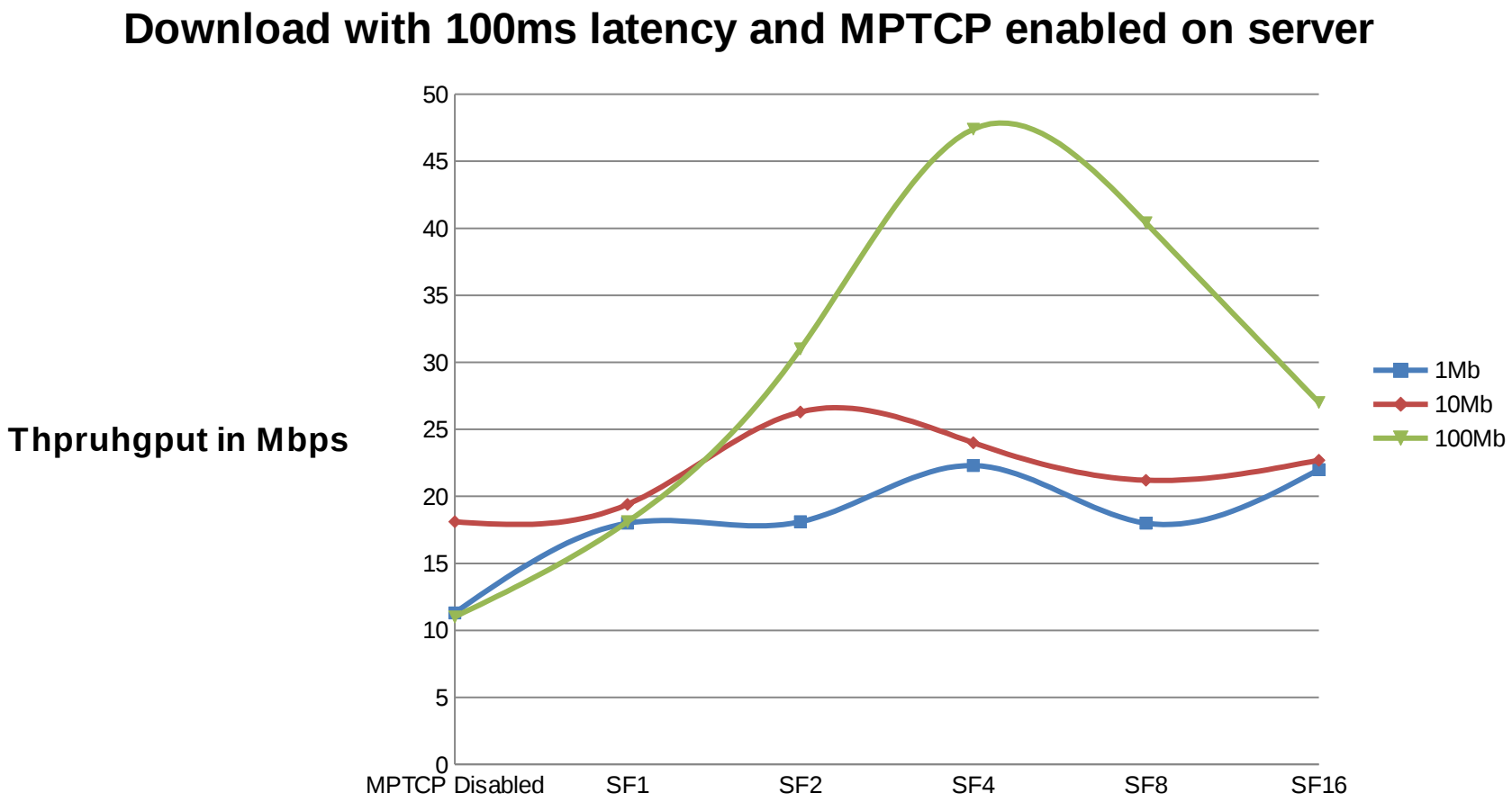
Upload with 80ms (3G mid) latency MPTCP enabled at server



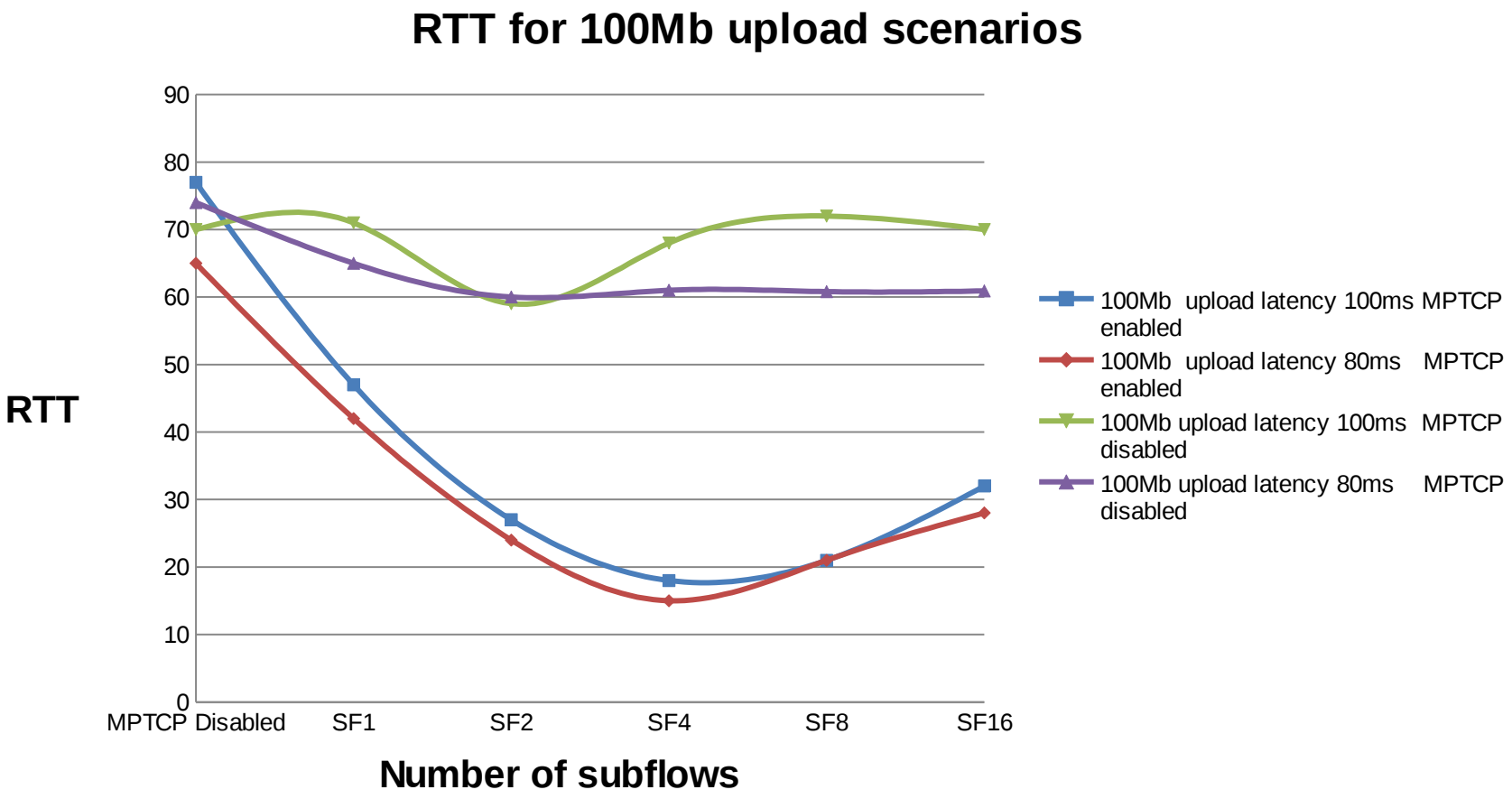
Upload with 100ms (3G far) latency MPTCP disabled at server



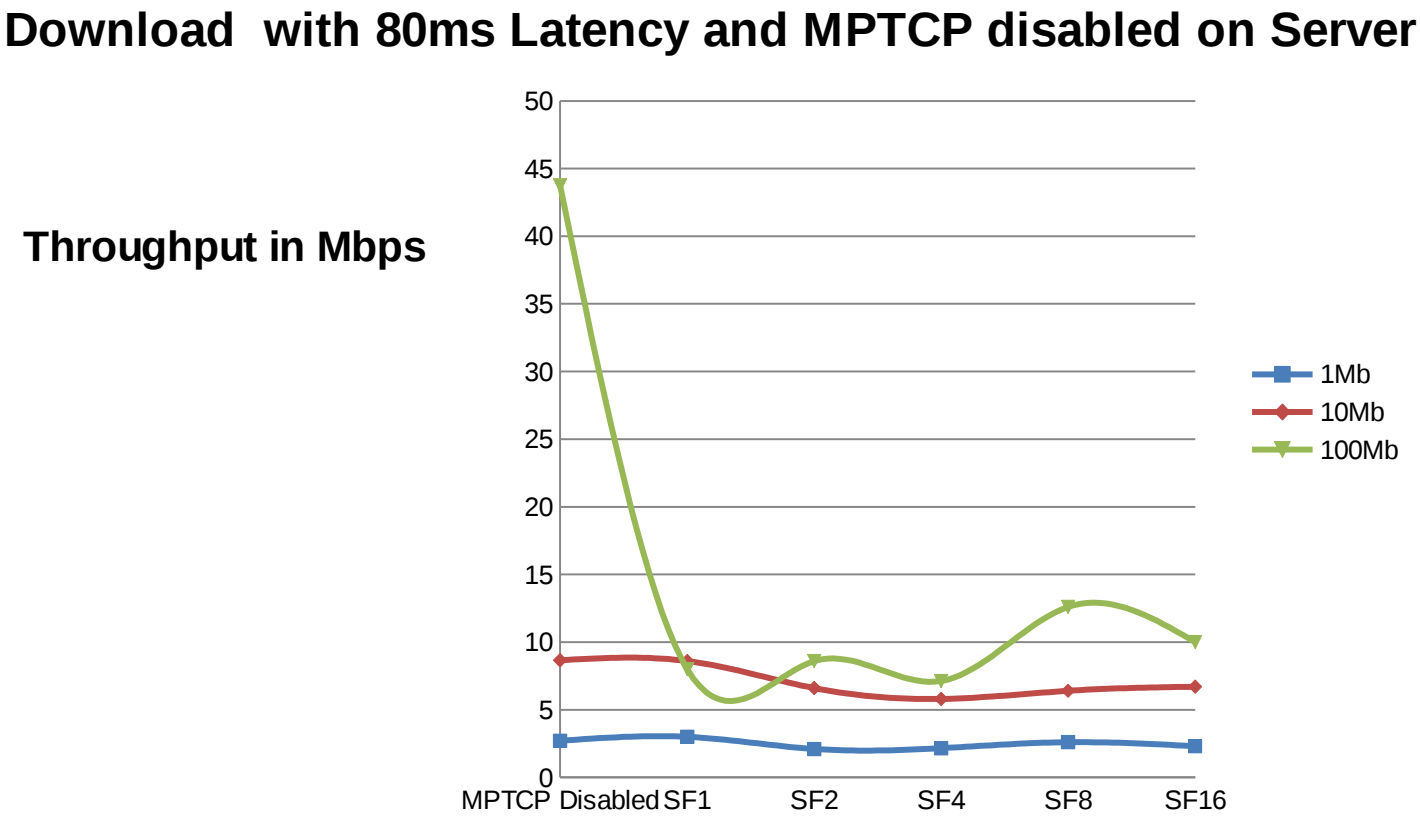
Upload with 100ms (3G far) latency MPTCP enabled at server



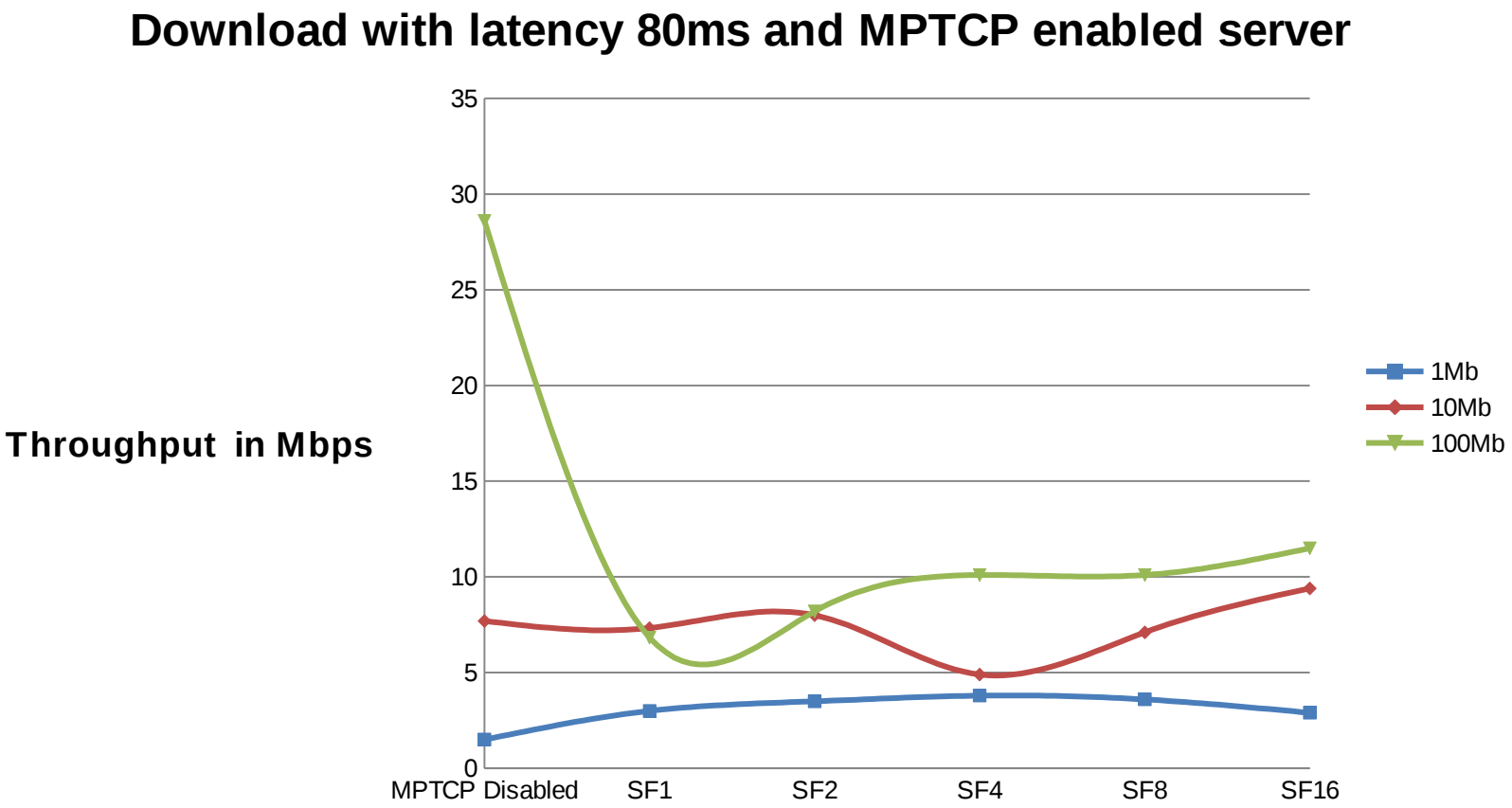
Upload Latency Variations



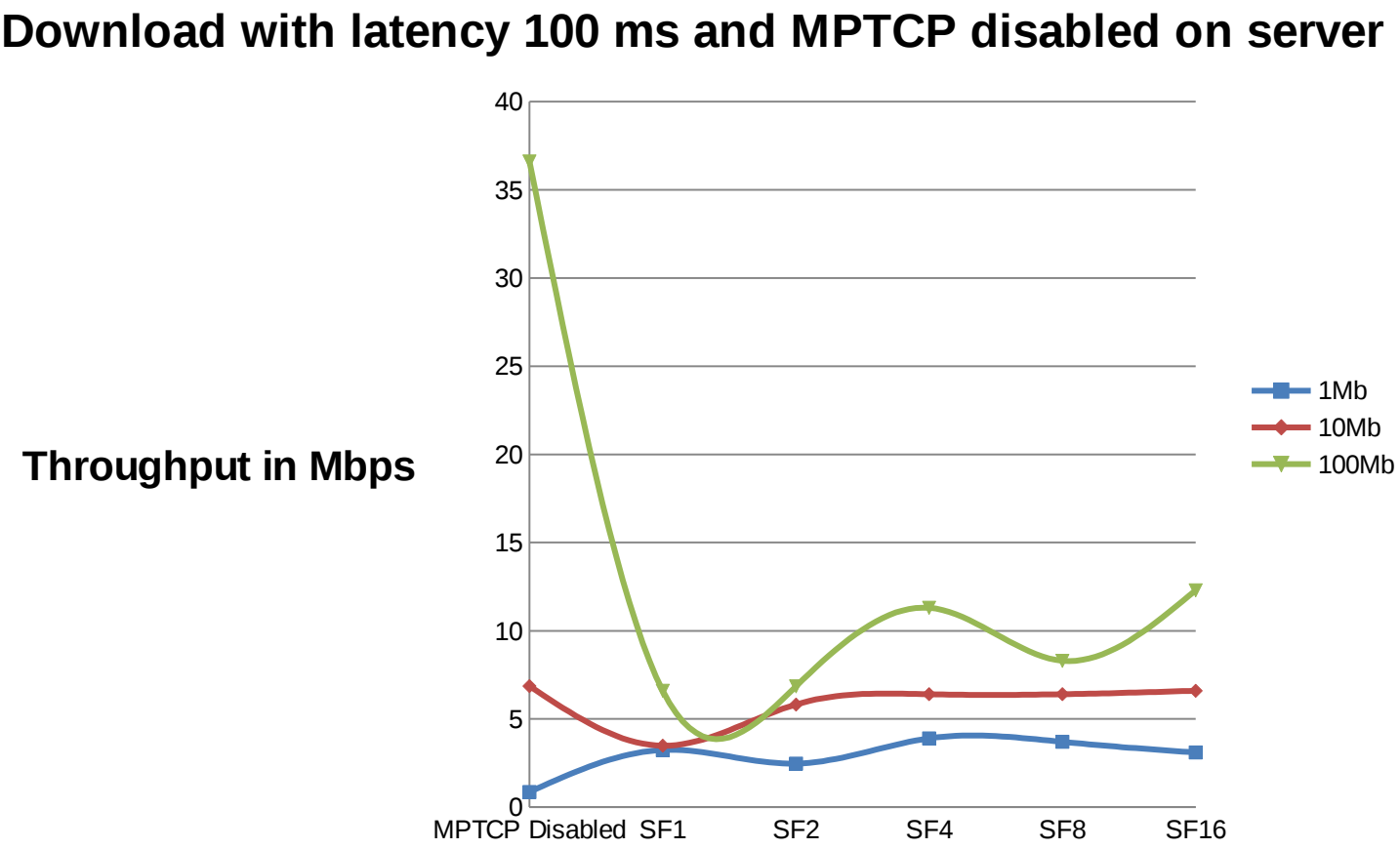
Download with 80ms 3G(mid) latency MPTCP disabled at server



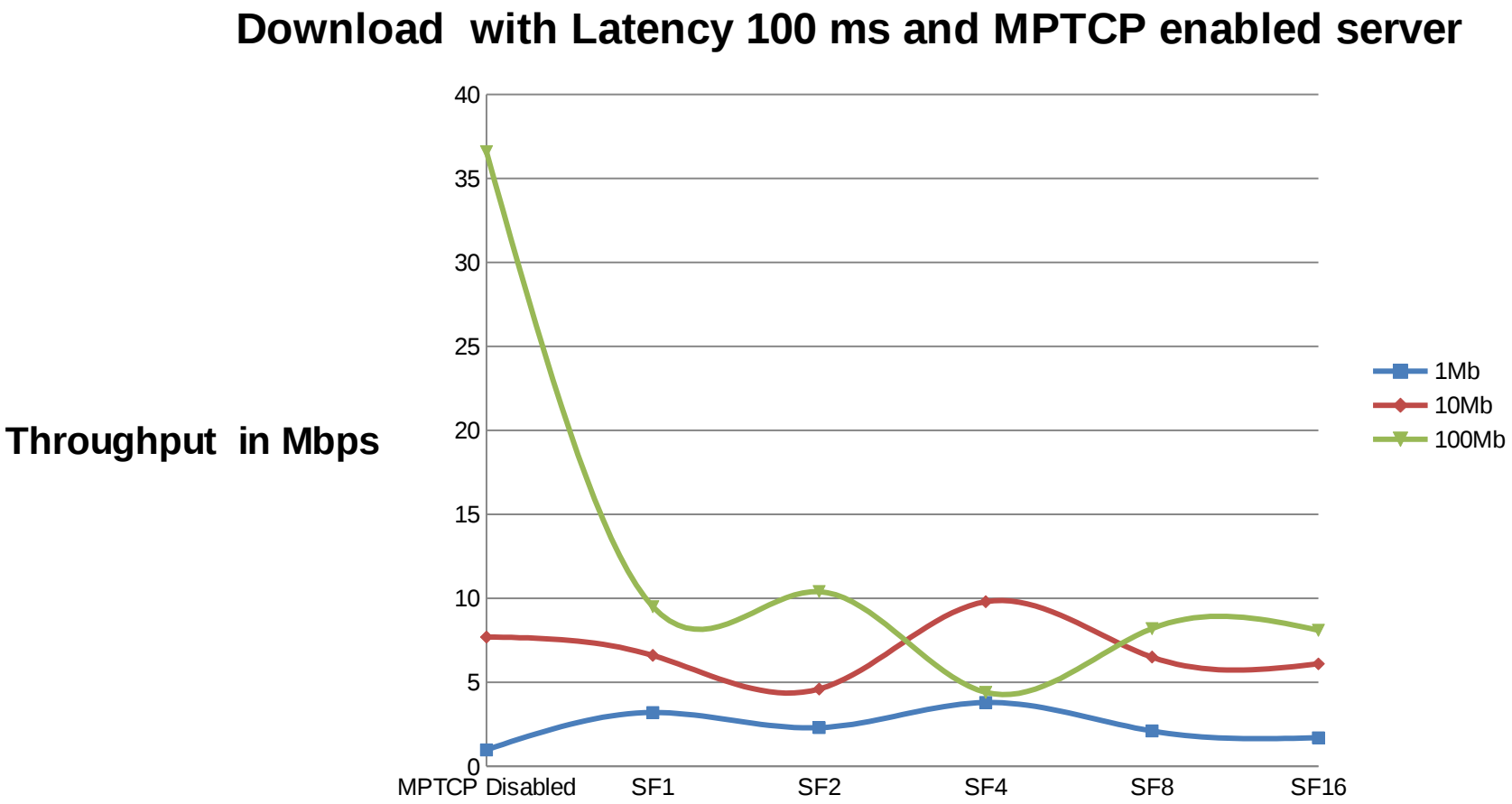
Download with 80ms (3G mid)latency MPTCP enabled at server



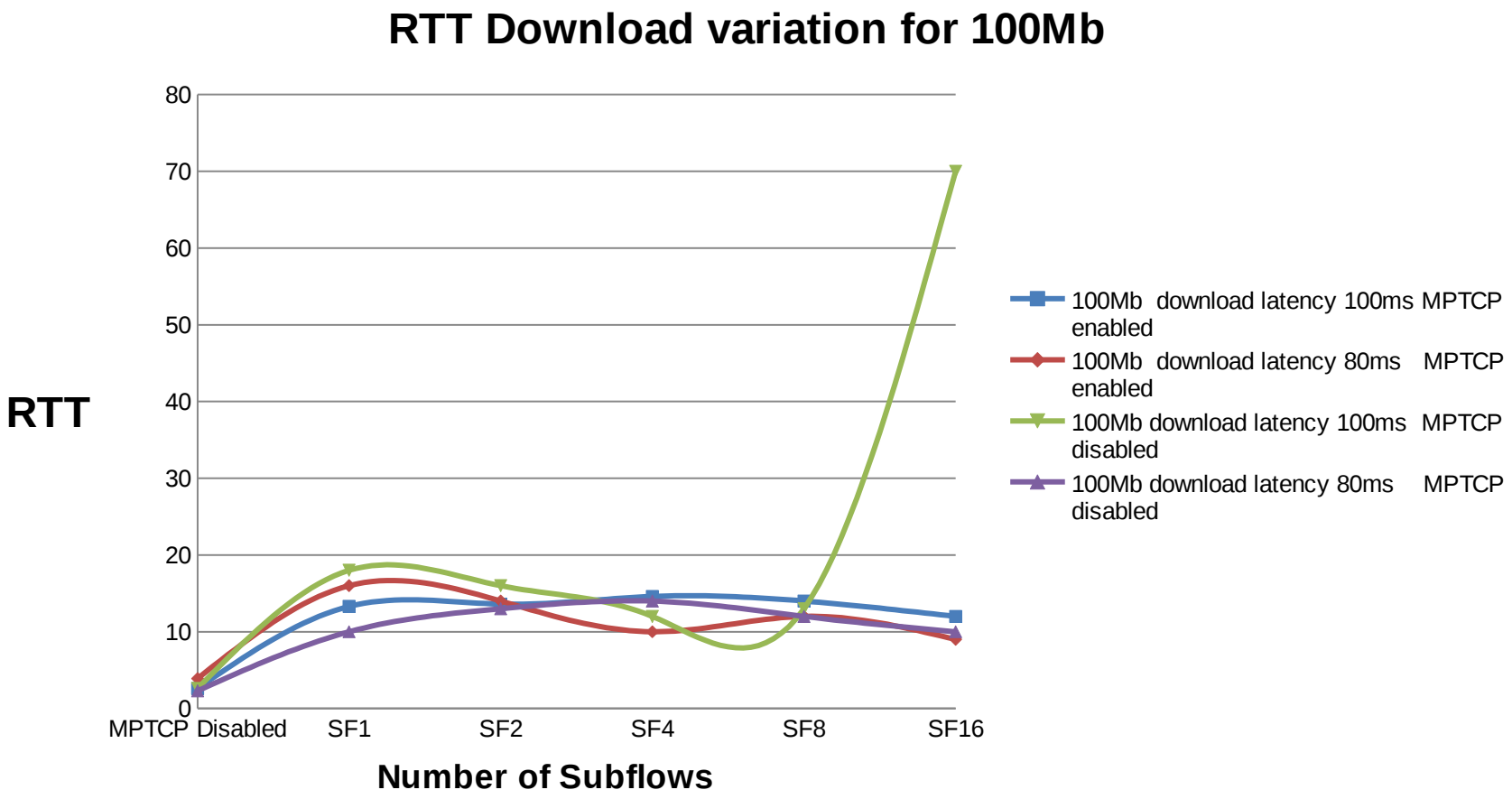
Download with 100ms (3G far) latency MPTCP disabled at server



Download with 100ms (3G far)latency MPTCP enabled at server



Download Latency Variations



Conclusion

- Throughput optimization Improves end user experience. Pulling MPTCP into the context, Our observations Point to
  - Need for multiple profiles
  - Need for optimized profiles – based on internet usage pattern and data
  - Need for some protocol stack level changes *(Open for future discussions)*

Work In Progress and Next Steps

More tests … (by IETF-95)

- Through proxy
- Varied applications (elastic vs inelastic)
- Change in behavior with v6
- MPTCP stack parameters variations/tweaks

Draft updates… (by IETF-95)

- With Test Results, Analysis and Summary

Discuss/Review with Working Group

Ask for WG Adoption (Informational category)

Work on Design recommendations/Solutions draft (Experimental category)

THANK YOU

Chetan Harsha
(chetan.harsha@intl.verizon.com)

A.Palanivelan
(palanivelan.appanasamy@intl.verizon.com)

Senthil Sivakumar
(ssenthil@cisco.com)