Münster University of
Applied Sciences

Transmission Scheduling Optimizations for Concurrent Multipath Transfer

T. Dreibholz, R. Seggelmann, M. Tüxen, E. Rathgeb

Content

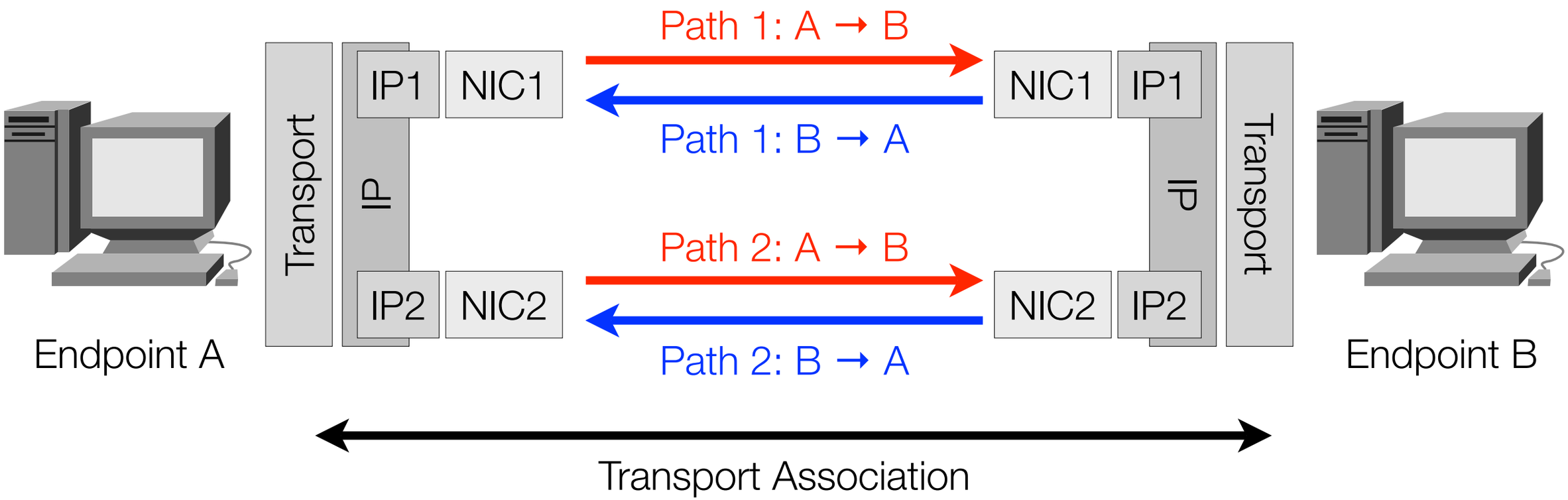
- Stream Control Transmission Protocol (SCTP)
- Streams & Multipath
- Scheduling Possibilities, Limitations & Algorithms
- CMT-aware Scheduling
- Measurements
- Conclusion & Outlook

Stream Control Transmission Protocol

- Connection-oriented and message based
- Reliable
- Multihoming
- Multistreaming
- Extensible packet format

Multihoming

- Multiple addresses per endpoint
- Primary path for transfer, other paths for increased reliability
- Change of primary path in case of failure



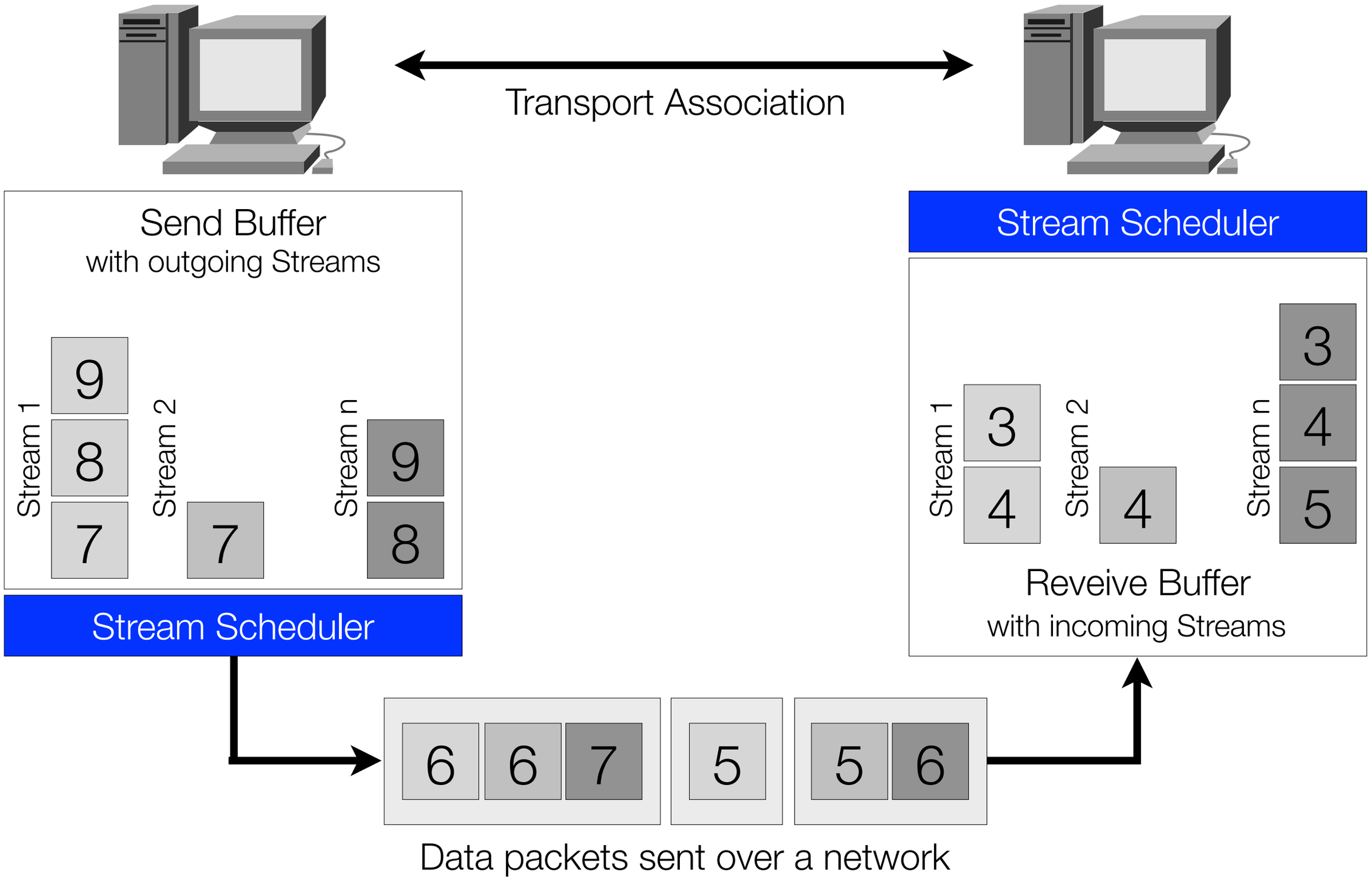
Concurrent Multipath Transfer

- Extension for SCTP
- Load sharing with multiple paths
- Possible throughput ideally is combined bandwidth of all paths
- NR-SACKs are used to reduce necessary buffer size

Streams

- Unidirectional separation of logically independent data
- Application assigns Stream Identifier
- SCTP maintains order only within a stream
- Message loss does not affect other streams
- Reducing the impact of Head-of-Line blocking

SCTP Streams



Scheduling Possibilities & Limitations

Sender scheduling

- Scheduling affects message sending sequence
- Bundling depends on the sequence
- Behavior on the wire can be influenced
- Buffers limit the preference of specific messages

Scheduling Possibilities & Limitations

Receiver scheduling

- Scheduling affect message delivery order
- Buffers limit the preference of specific messages
- No other influence

Scheduling Algorithms

- Common algorithms beneficial in specific scenarios

- **First-Come, First-Serve / Round-Robin**

  Simple, generic algorithms (standard in today's implementations)

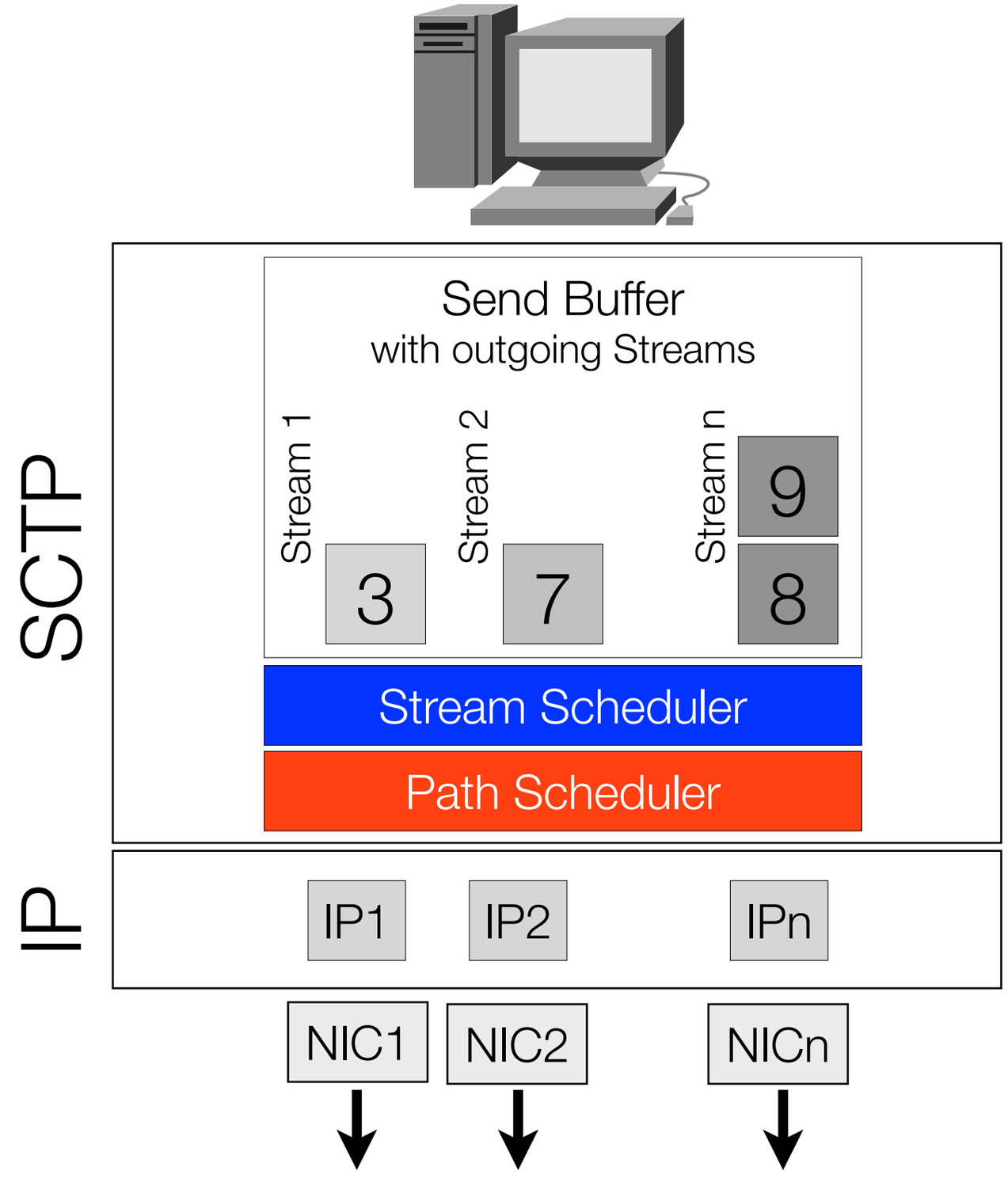
- **Fair Bandwidth**

  Equal bandwidth for each stream (Tunneling)

- **Priorities**

  Prioritization of specific messages (Signaling, Monitoring)

Stream & Path Schedulers



CMT-aware Scheduling

Possible optimizations

- Mapping of streams to paths

  ➜ mitigate reordering (delay optimization)

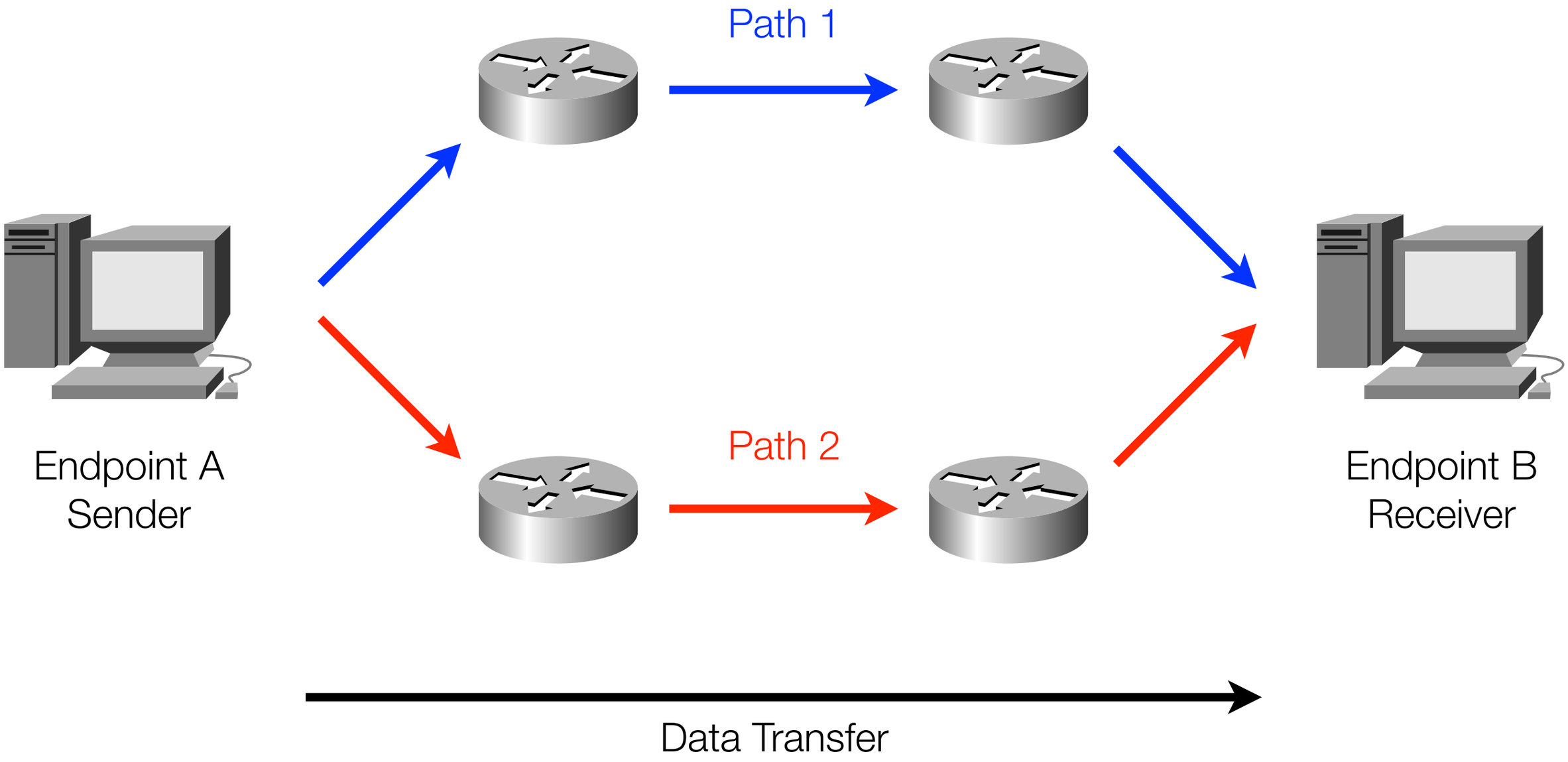
- Optimized distribution of streams

  ➜ fully utilize the paths (bandwidth optimization)

Measurements

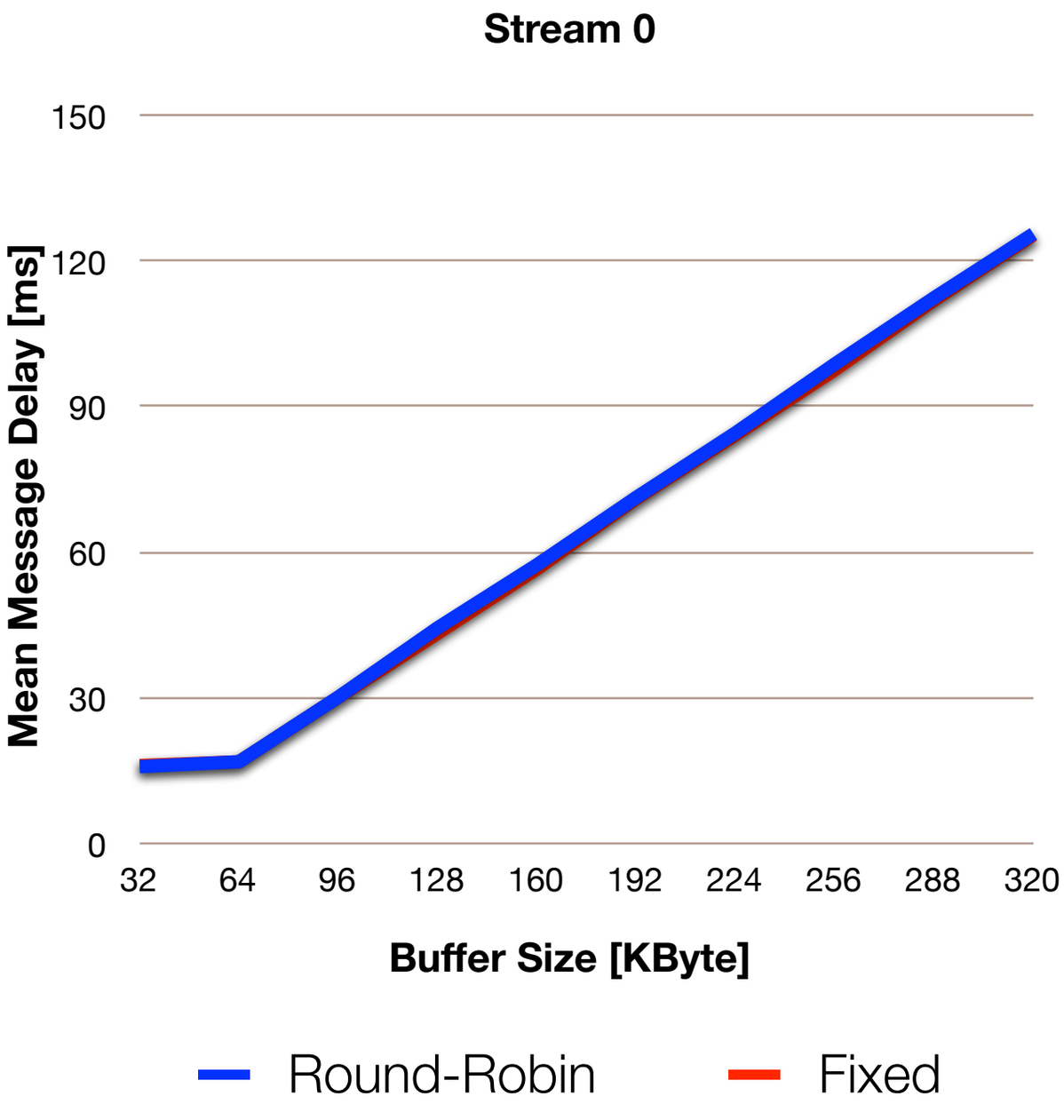
- A “fixed” scheduler with optimal scheduling
- Comparison with standard round-robin scheduling
- Simulation of several scenarios
- Sender Queue Info Option API Extension for data delivery decoupled per stream

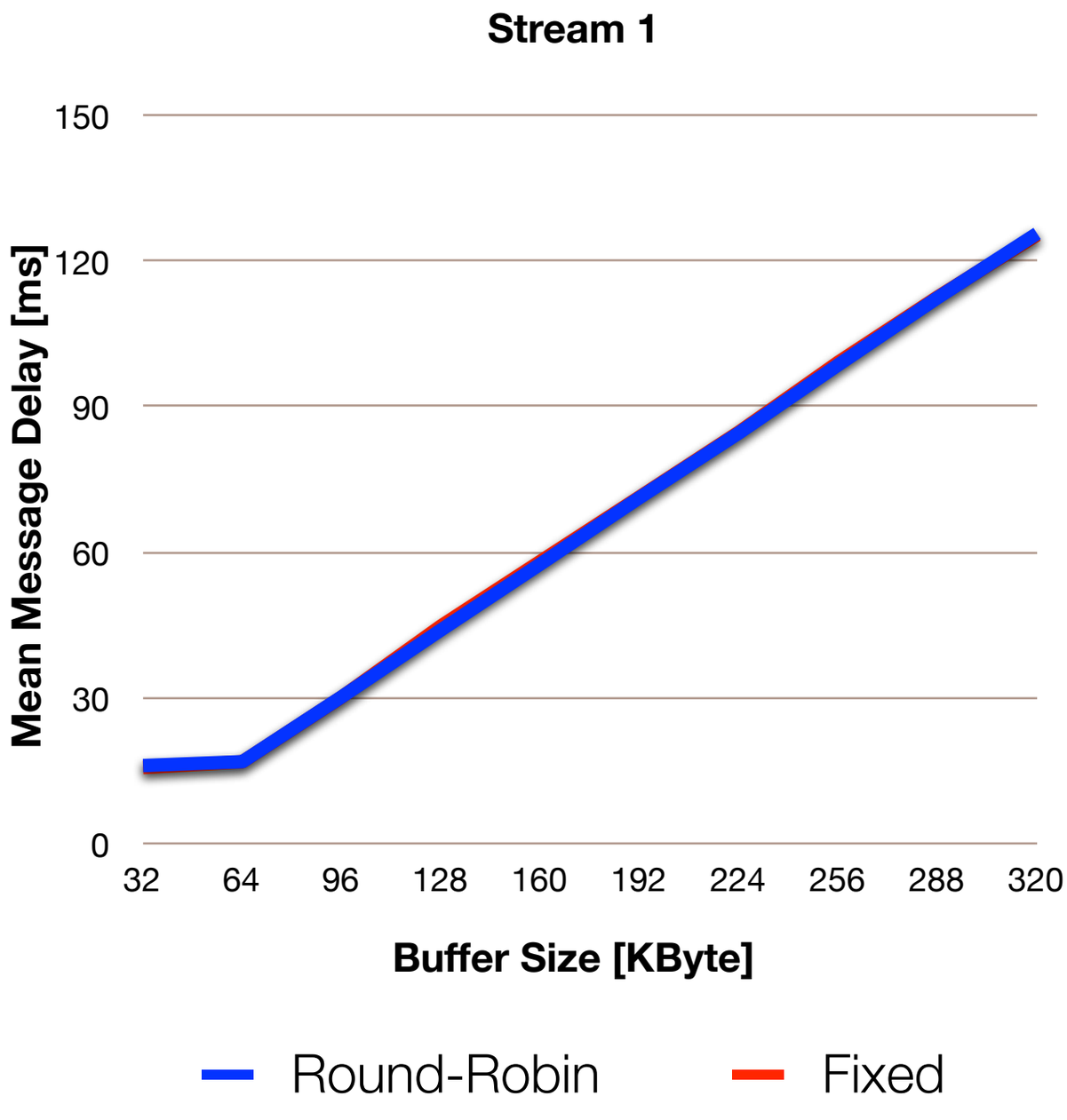
Measurement Setup



Measurement: Dissimilar Delay

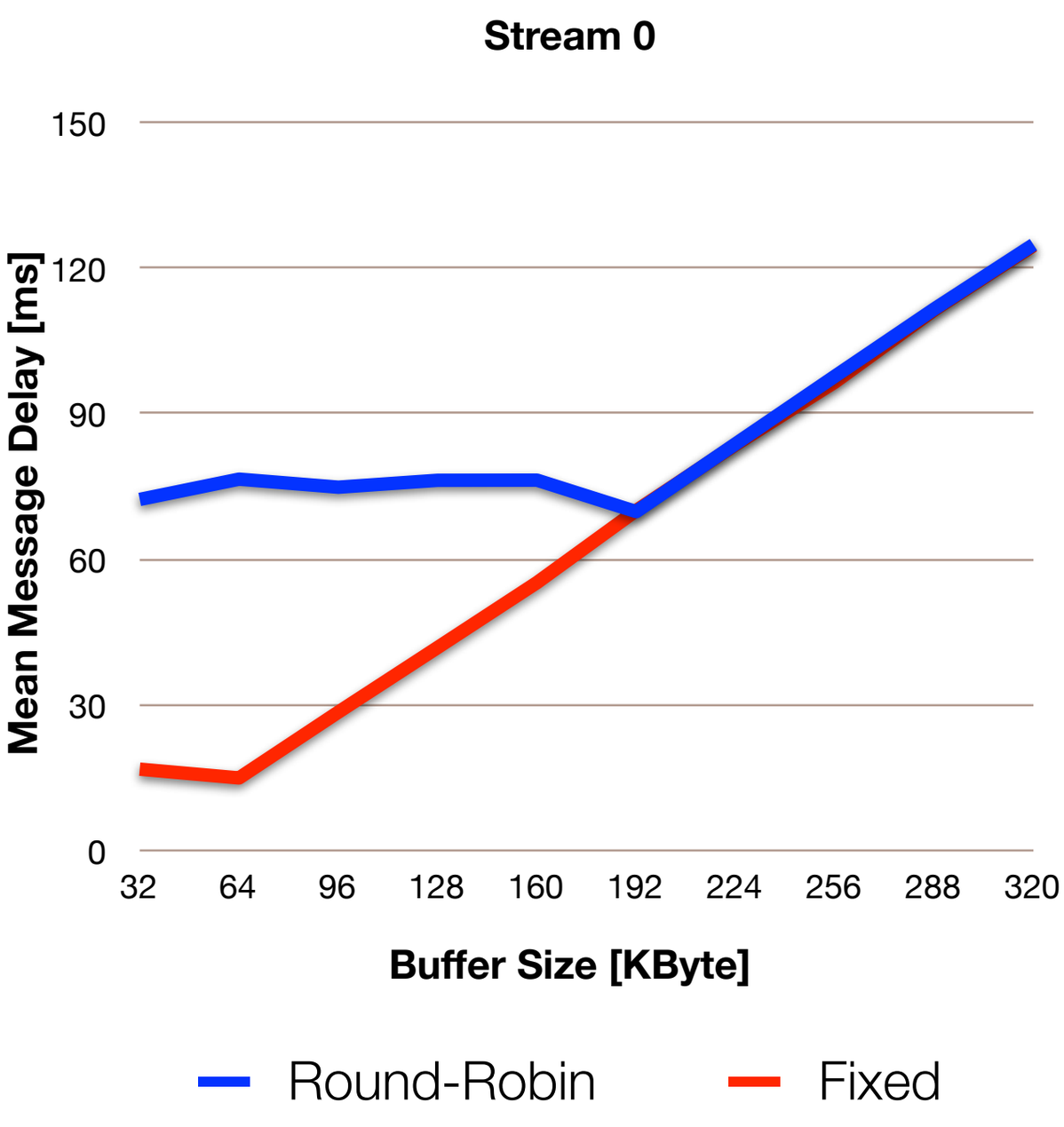
Scenario: 2 streams (saturated), 2 paths (10 ms and 10 ms)

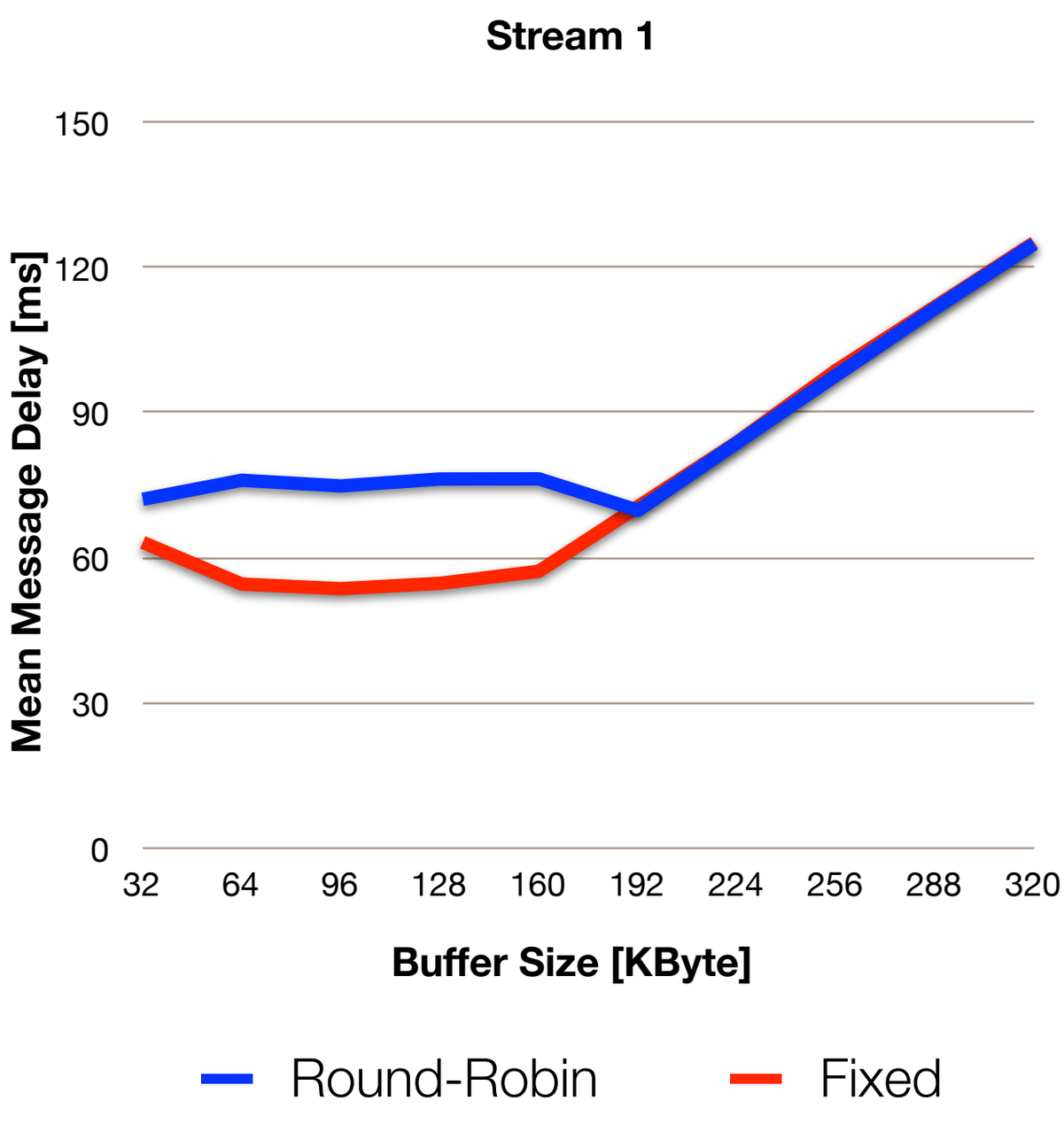




Measurement: Dissimilar Delay

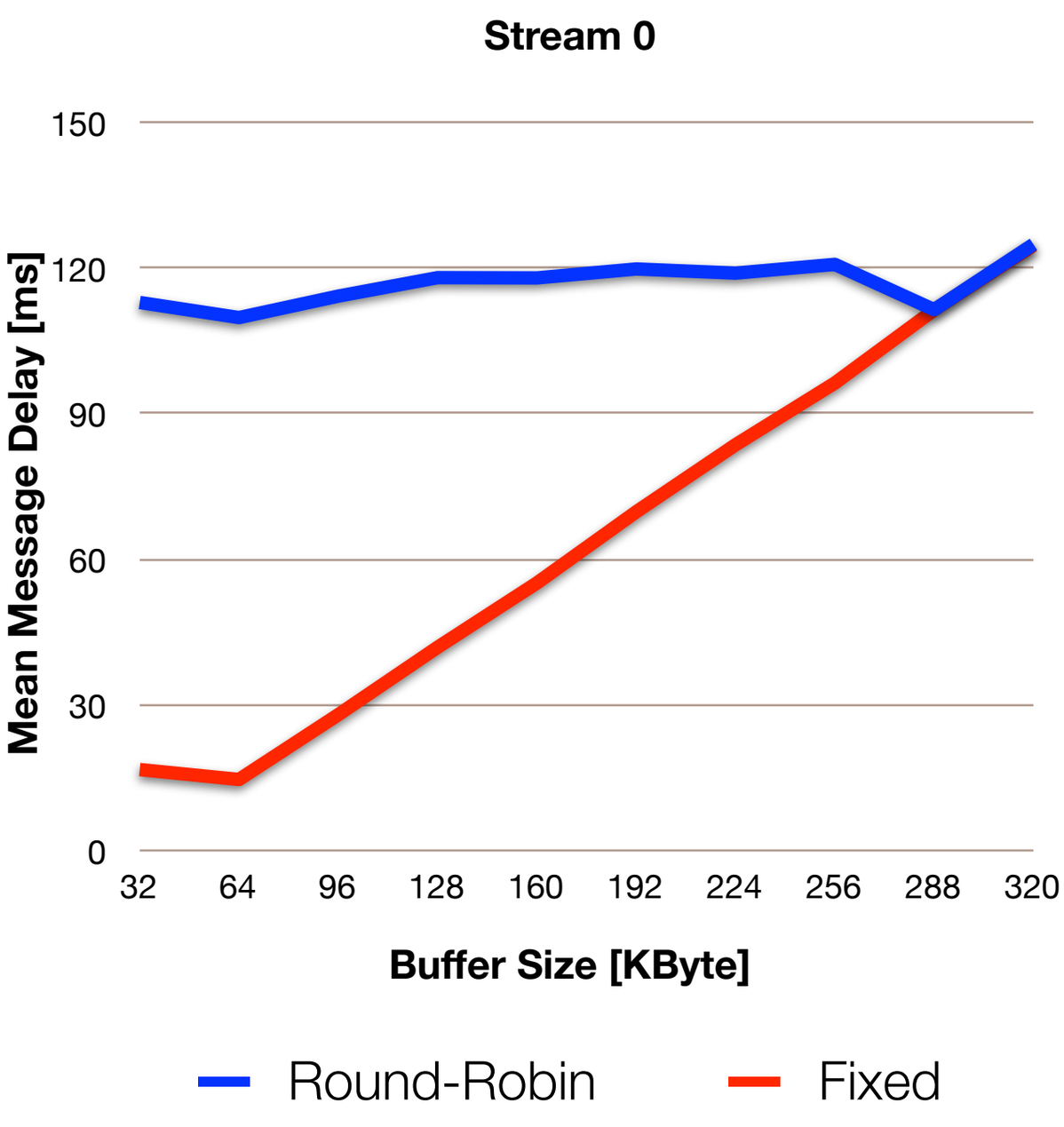
Scenario: 2 streams (saturated), 2 paths (10 ms and 50 ms)

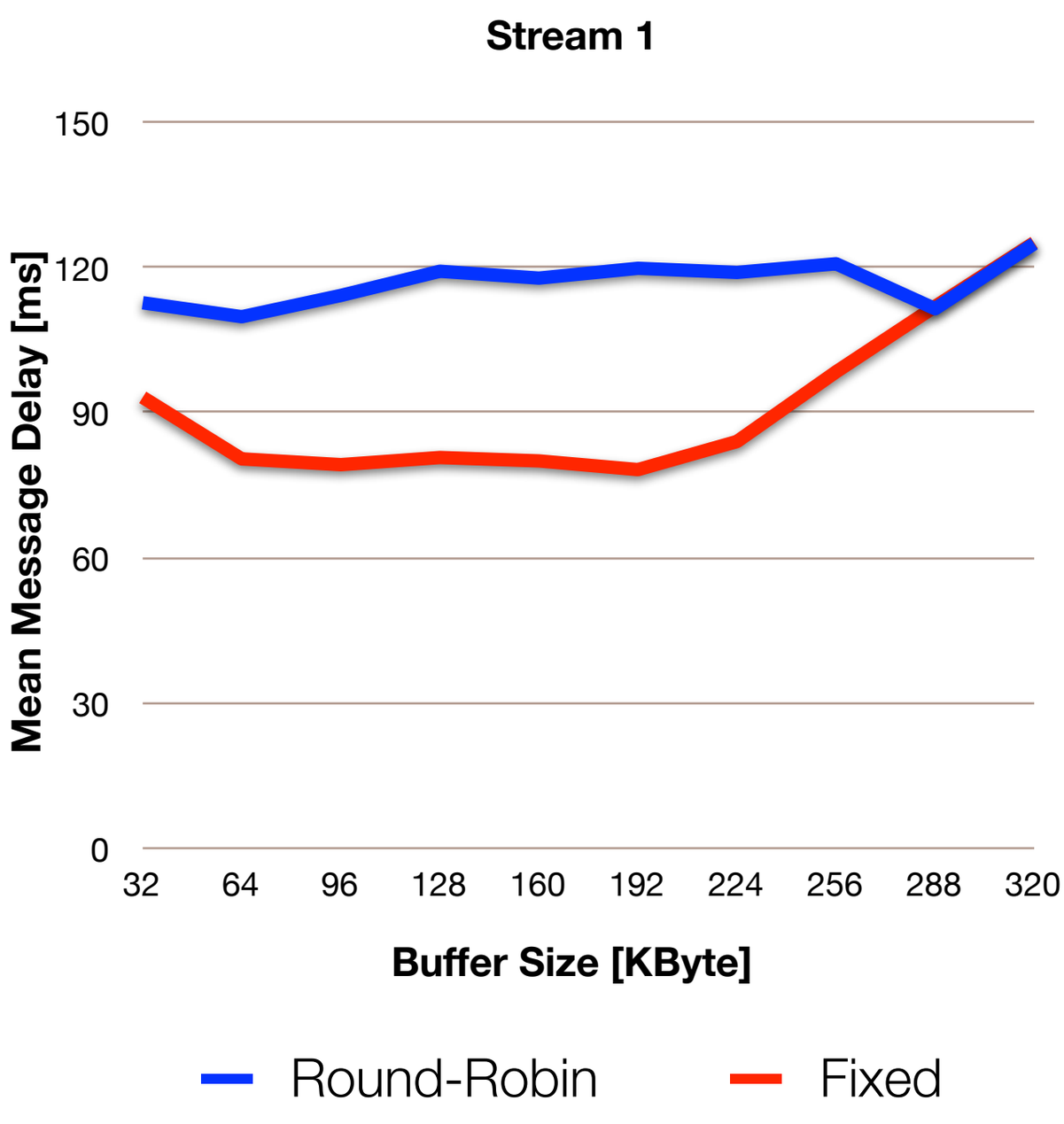




Measurement: Dissimilar Delay

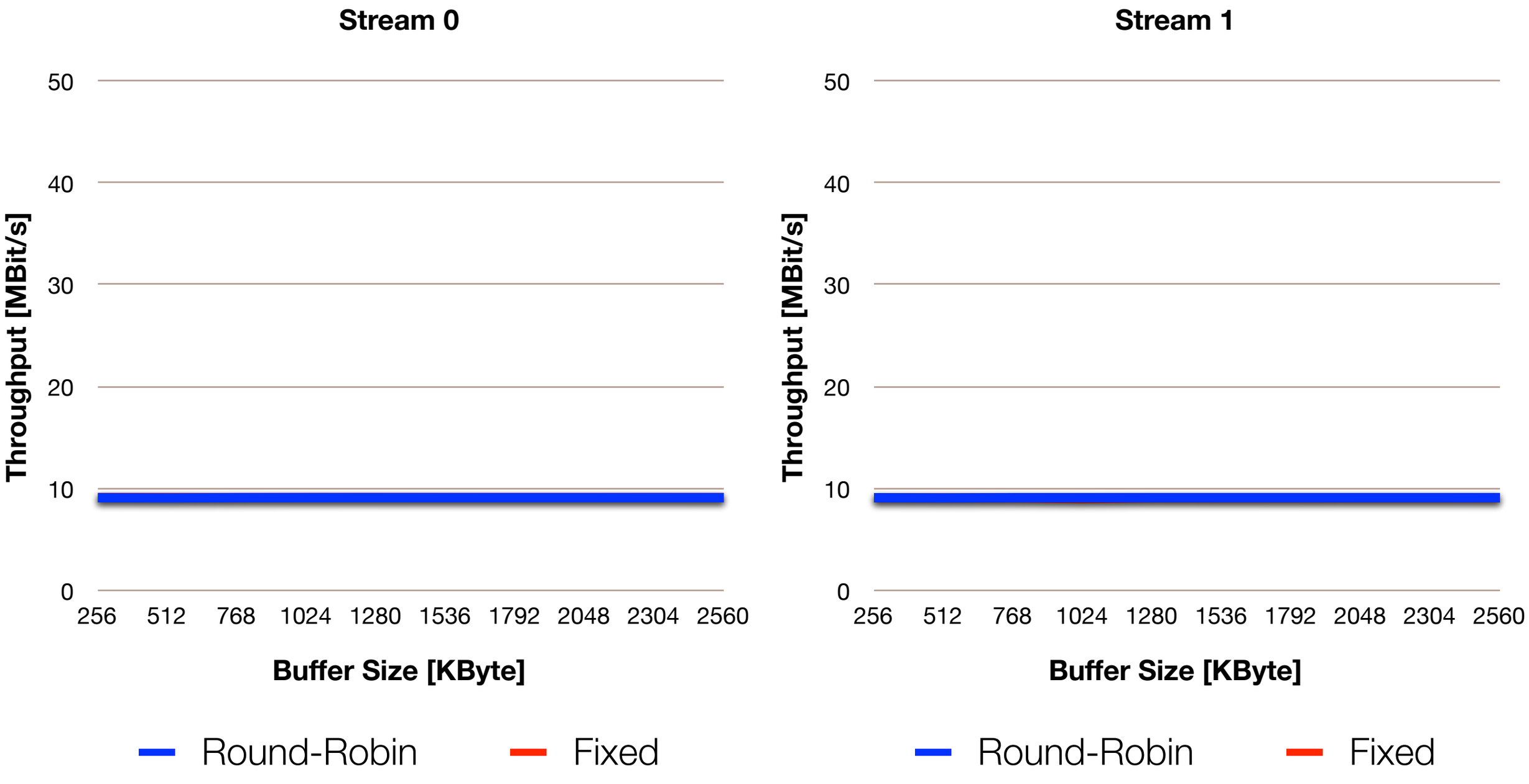
Scenario: 2 streams (saturated), 2 paths (10 ms and 75 ms)





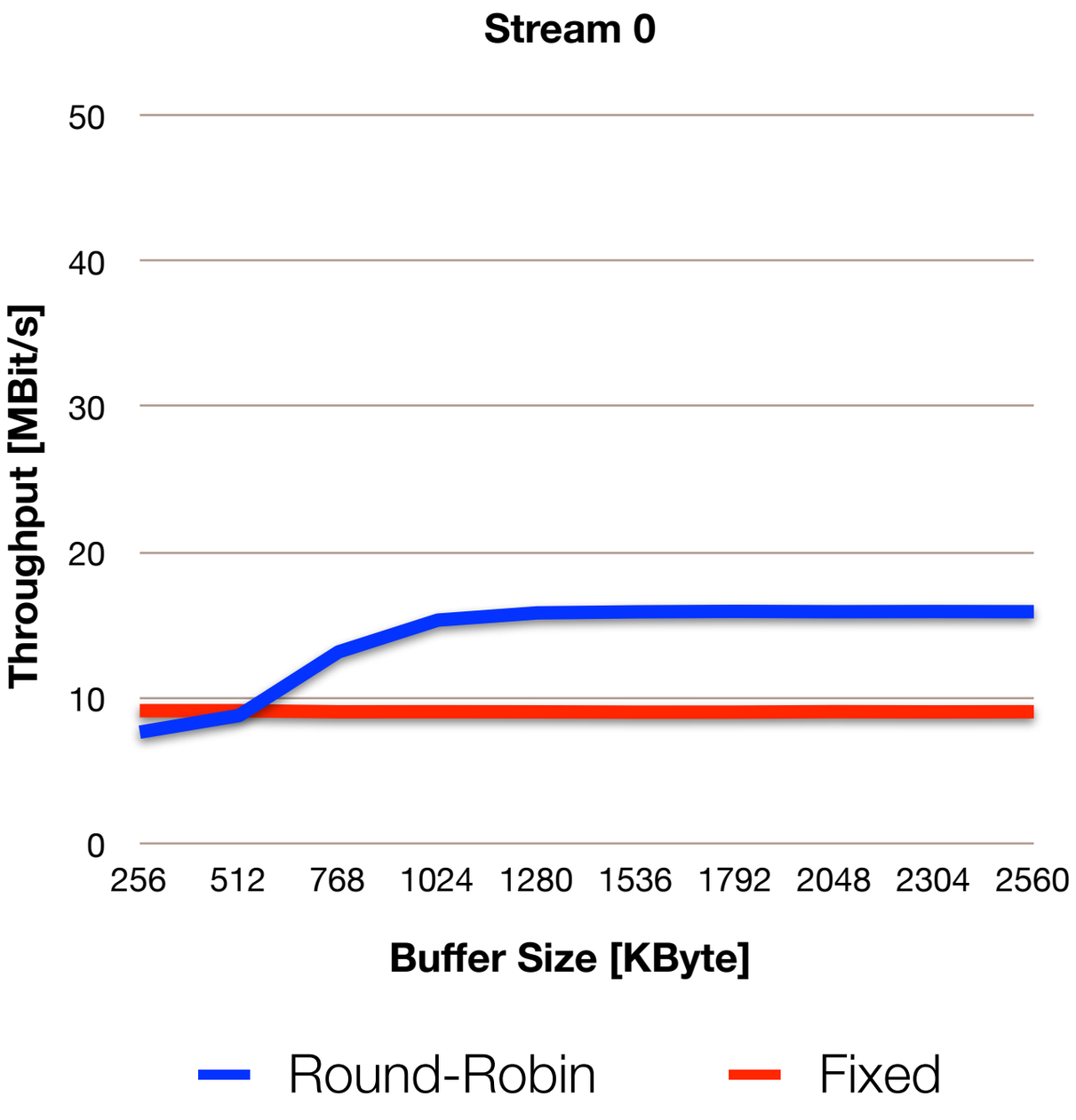
Measurement: Dissimilar Bandwidth

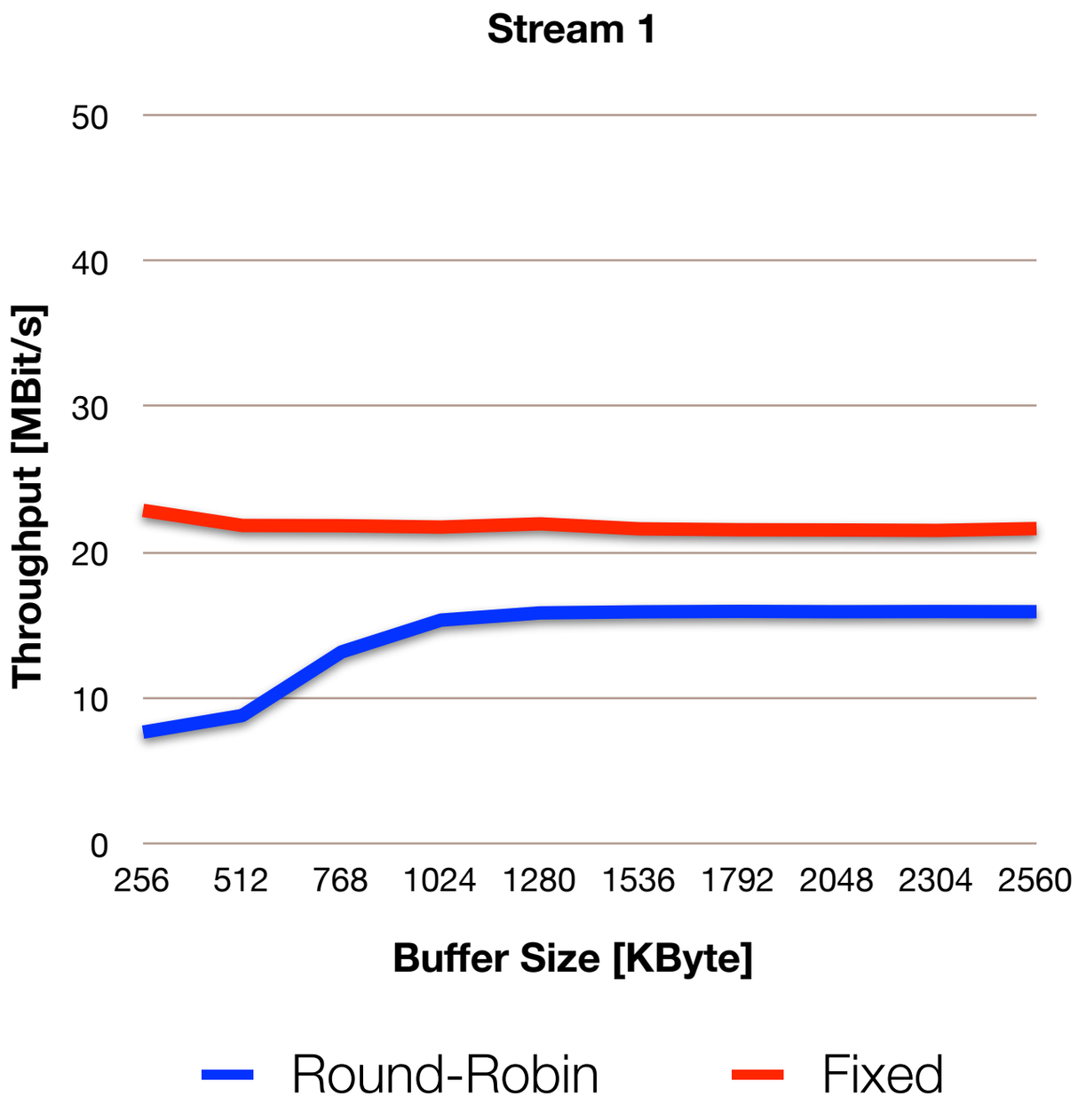
Scenario: 2 streams (saturated), 2 paths (10 MBit/s and 10 MBit/s)



Measurement: Dissimilar Bandwidth

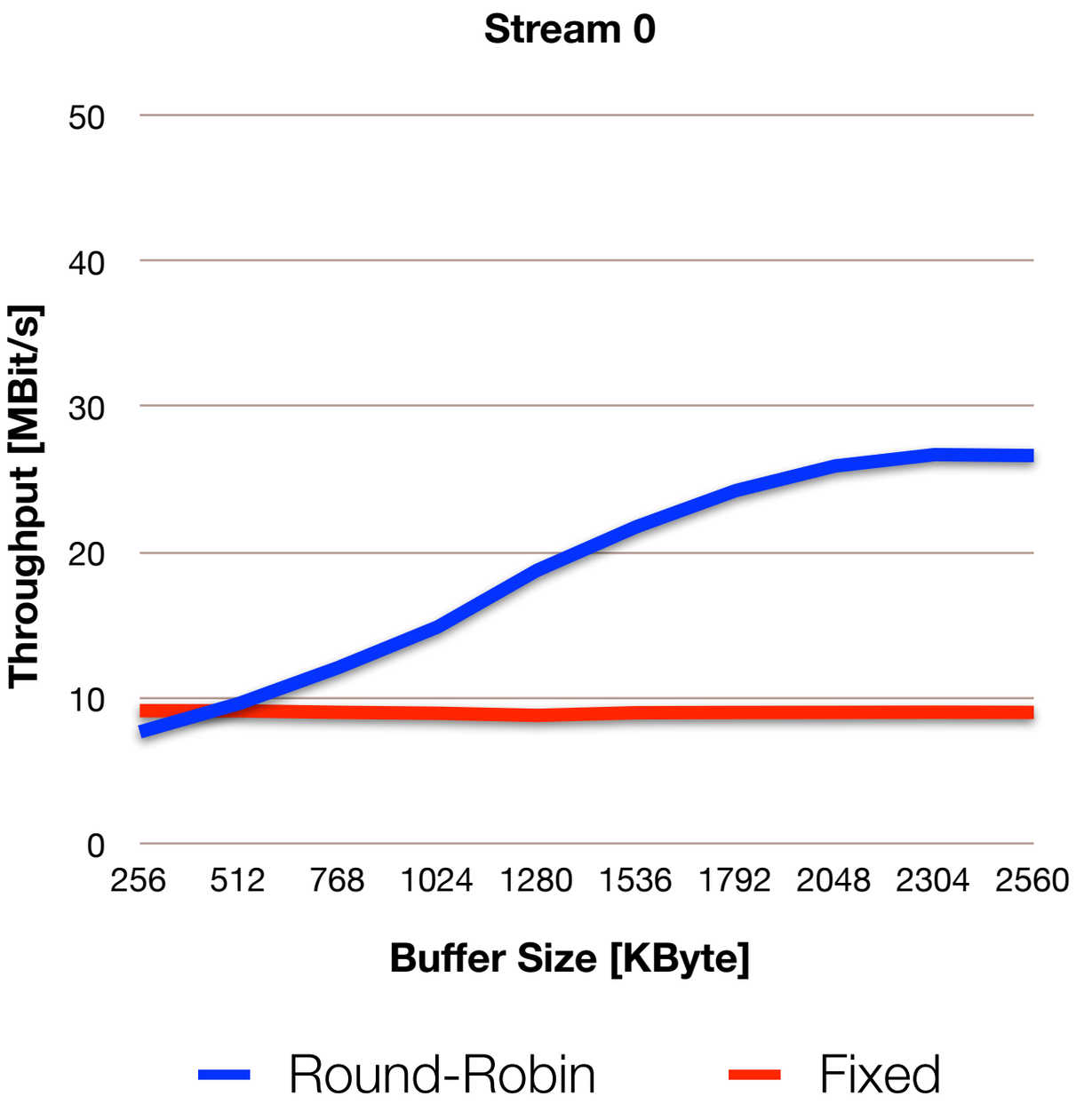
Scenario: 2 streams (saturated), 2 paths (10 MBit/s and 25 MBit/s)

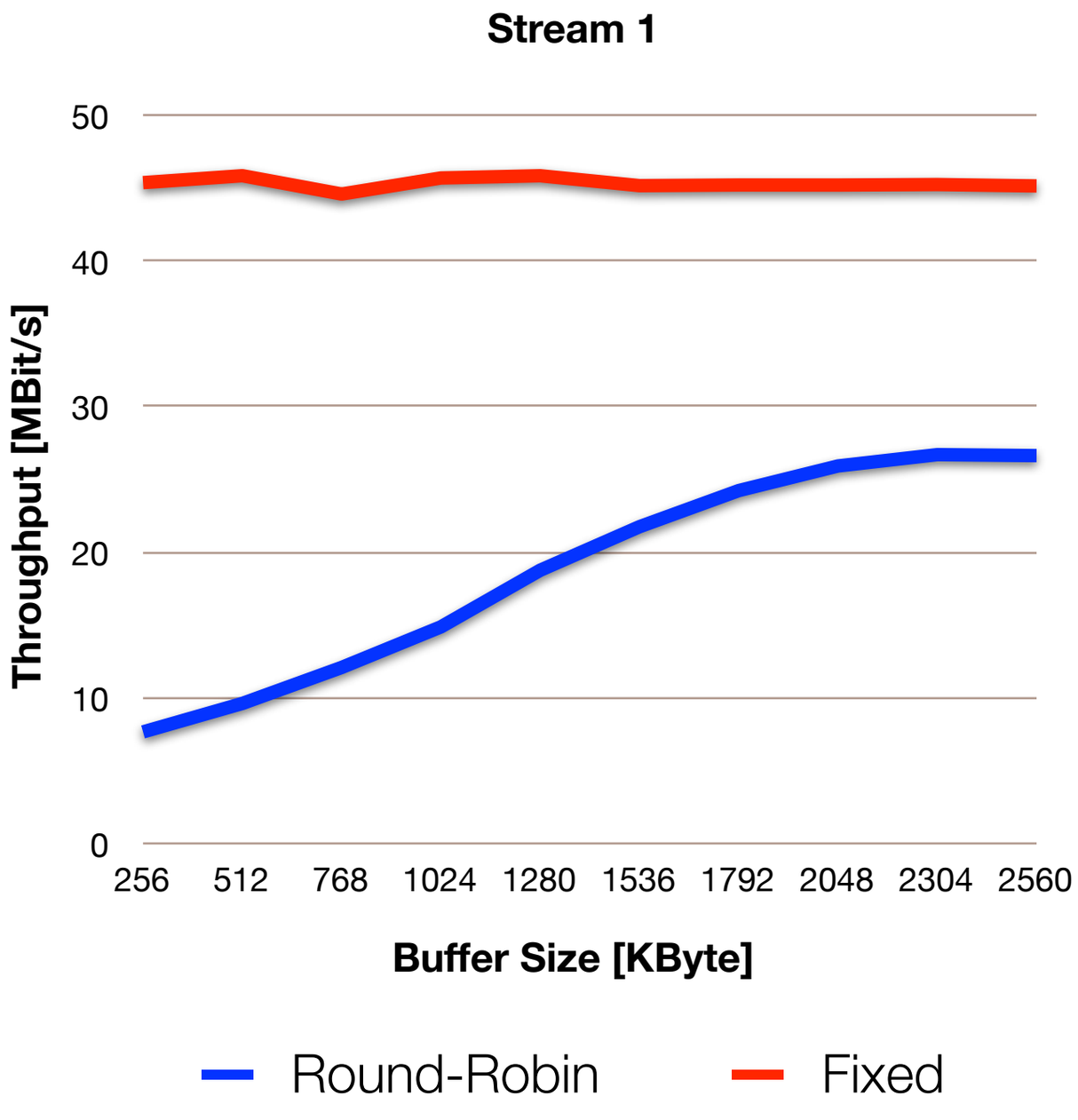




Measurement: Dissimilar Bandwidth

Scenario: 2 streams (saturated), 2 paths (10 MBit/s and 50 MBit/s)





Conclusion & Outlook

- Mapping streams to path reduces necessary buffer size and delay
- Optimized distribution of streams fully utilizes the available bandwidth
- Combining stream and path scheduling has notable benefits
- In the future: Development of dynamic CMT-aware scheduling

Thank you!