SCTP's Congestion Control and TCP-Friendliness

Irene Rüngeler, Münster University of Applied Sciences
Michael Tüxen, Münster University of Applied Sciences
Erwin P. Rathgeb, University of Duisburg-Essen

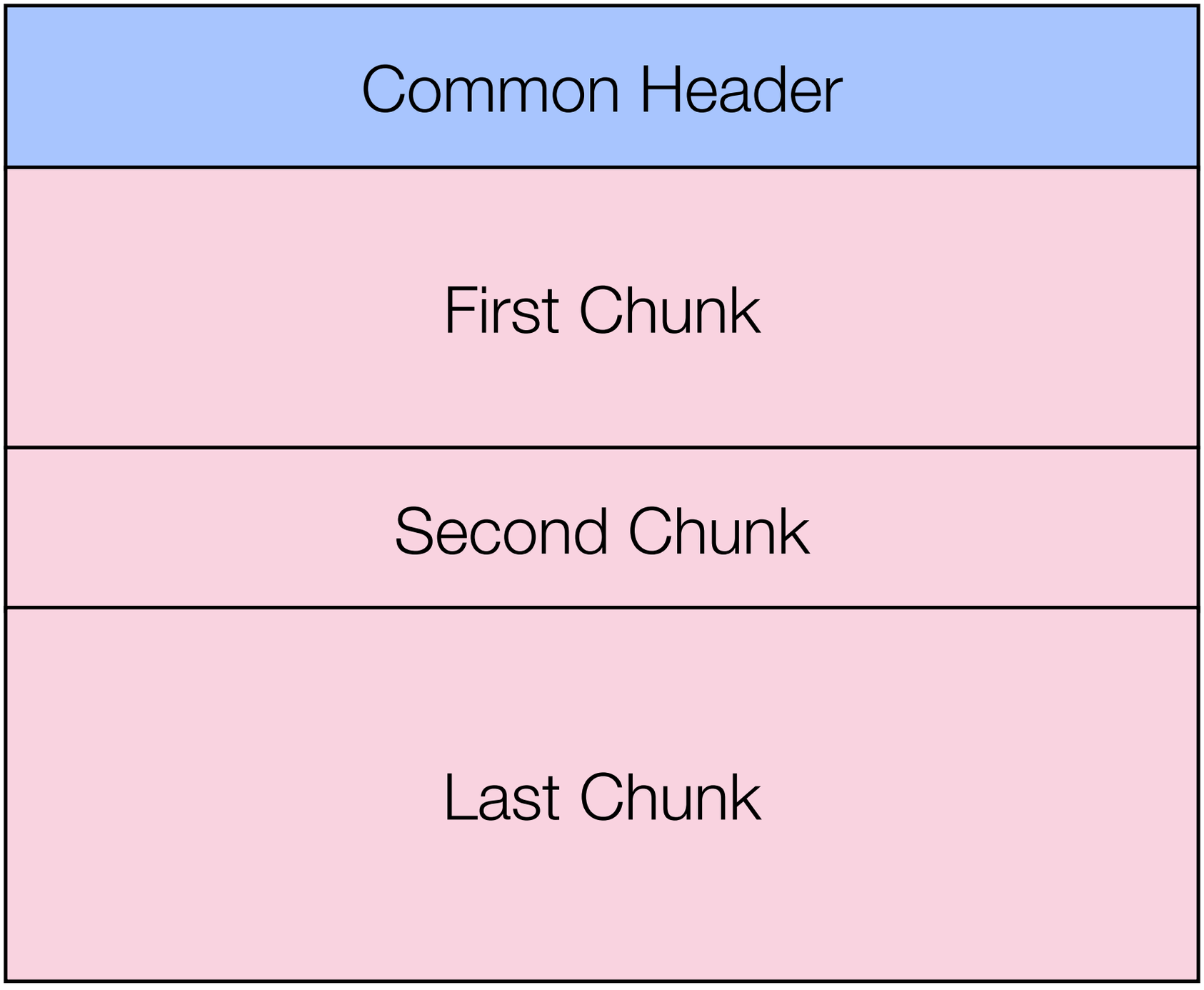
Outline

- SCTP and its packet format
- SCTP overhead for small user messages
- Analysis of the impact
- Conclusion

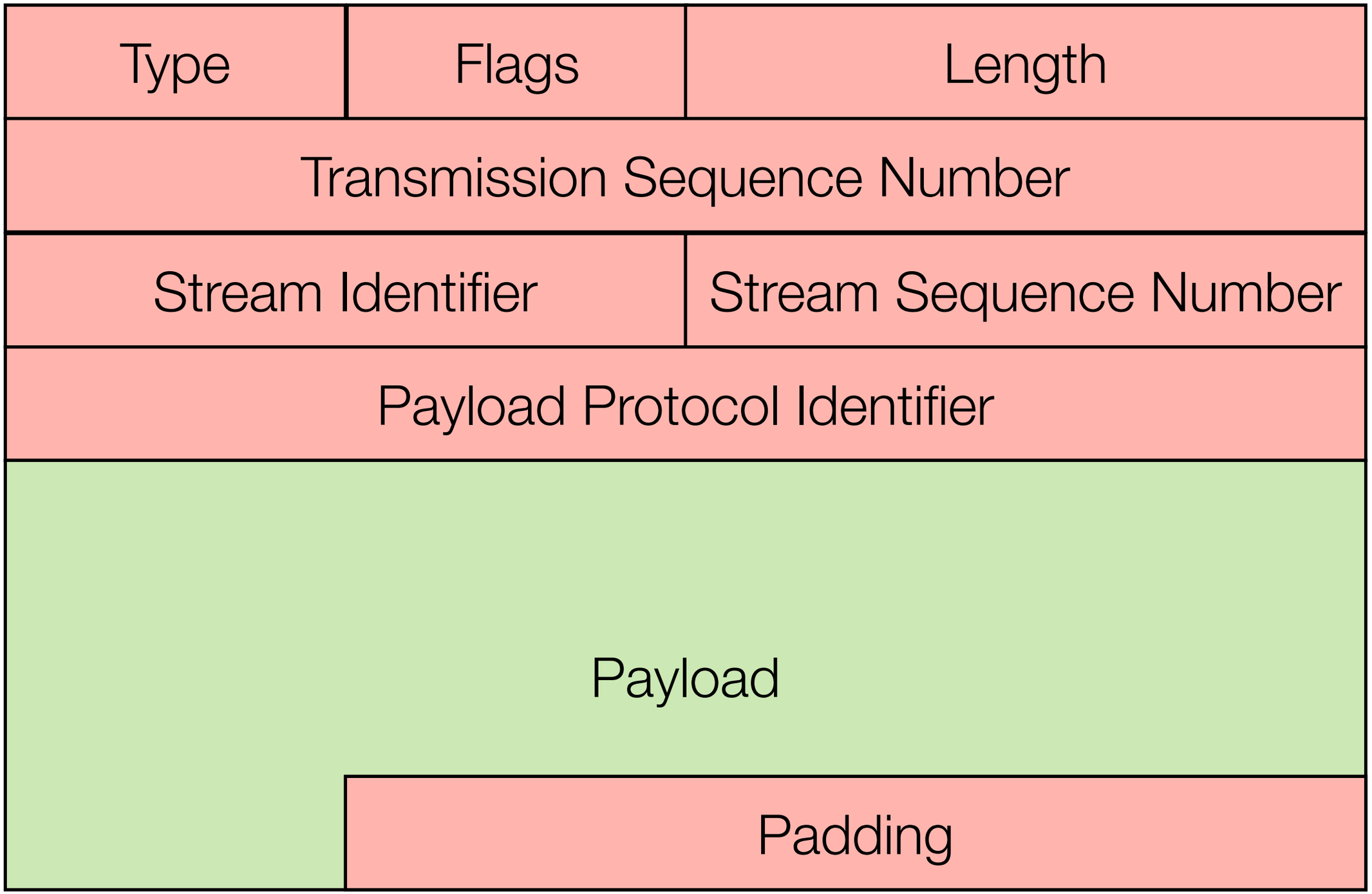
SCTP

- Message oriented, unicast transport protocol.
- Supports fragmentation and reassembly of user messages.
- Supports bundling of multiple small user messages into on SCTP packet.
- Supports multiple streams. In-sequence delivery is only supported with each stream.
- Supports multi-homing.
- Provides flow and congestion control.

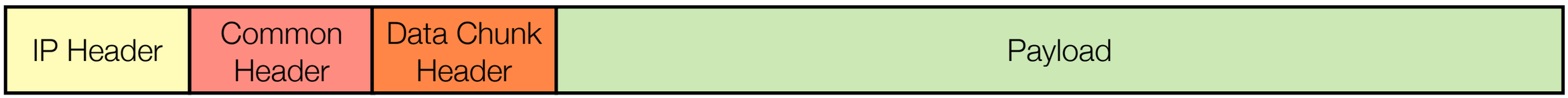
SCTP Packet Format



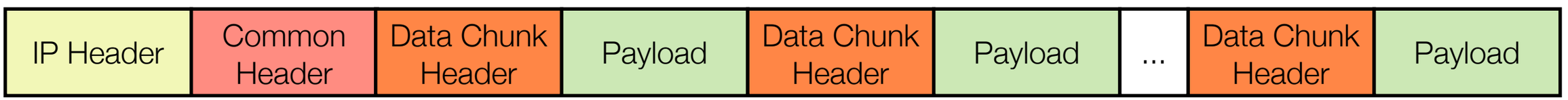
SCTP DATA-Chunk Format



Overhead Example



One chunk with 1436 Bytes: 3% Overhead

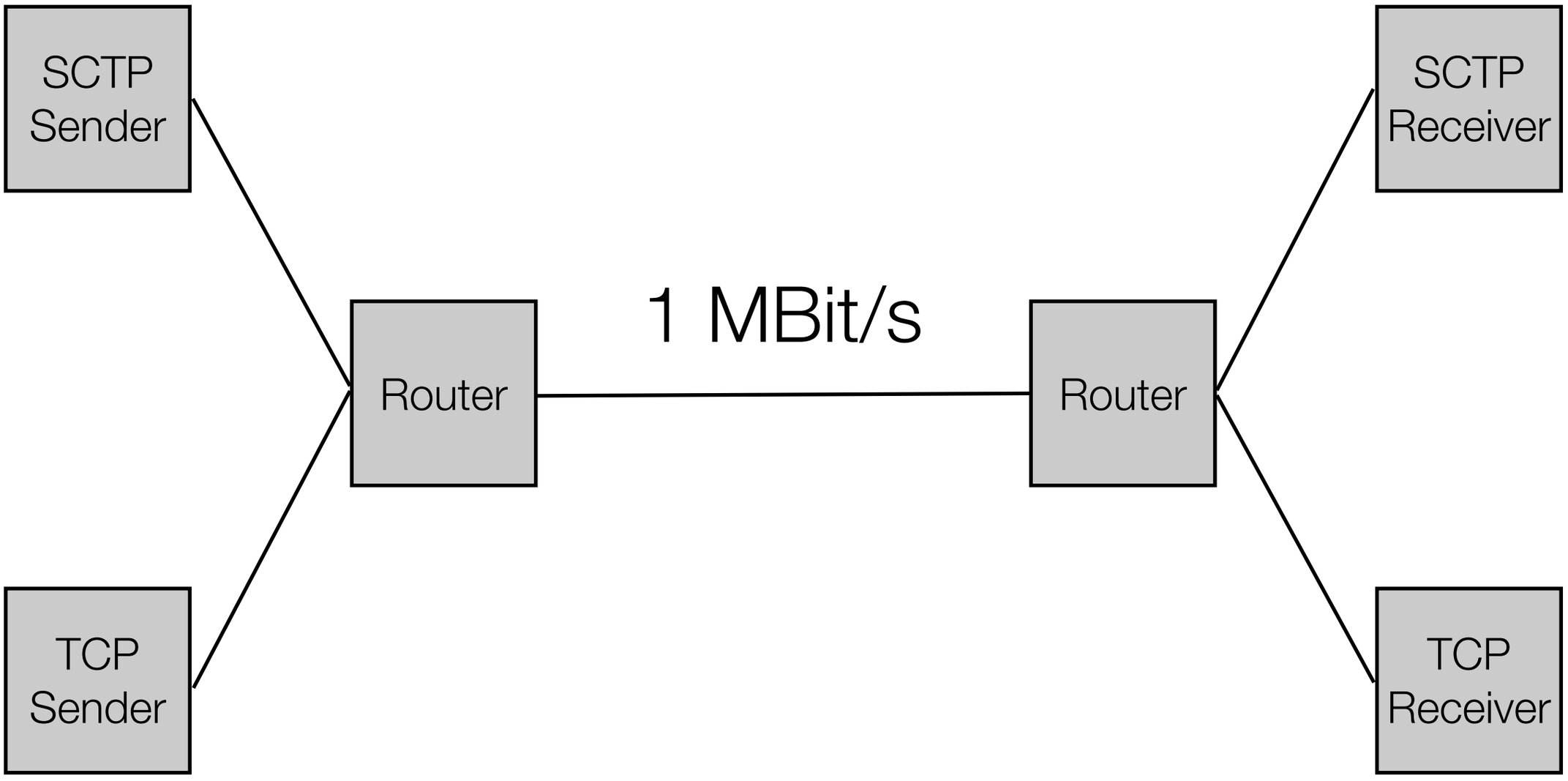


33 chunks with 28 Bytes (total 924 Bytes): 37% Overhead

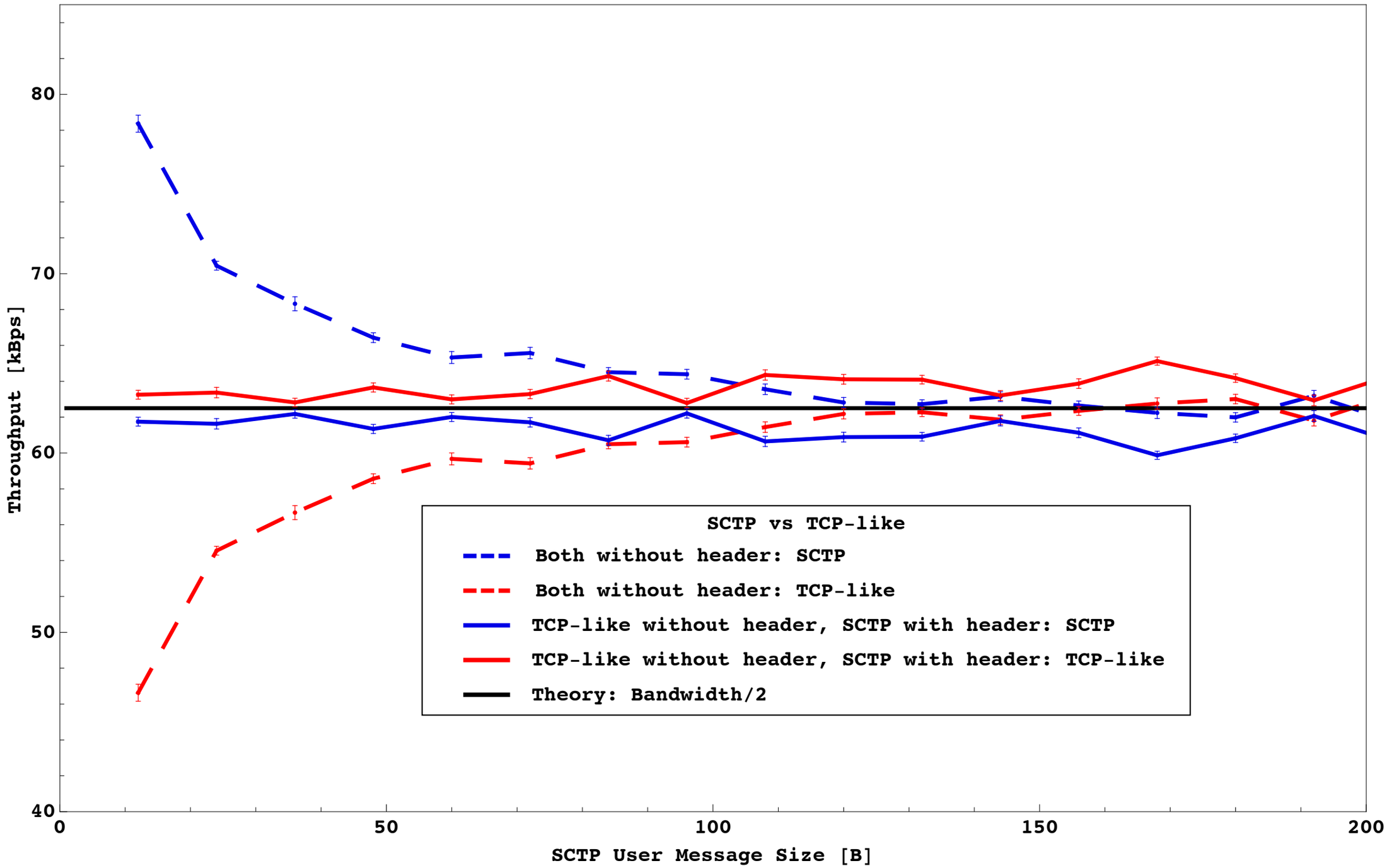
SCTP’s Congestion Control

- Being TCP-friendly was a very important design goal.
- SCTP’s congestion control is basically the same as TCP’s one.
- However, SCTP’s congestion control incorporates most of TCP’s advanced features (SACK, ABC, increased initial congestion window, ...).
- Additionally, it incorporates a mechanism for burst mitigation.
- SCTP specification mainly refers to TCP specifications and uses similar terms.
- The question is: What is the number of outstanding bytes?

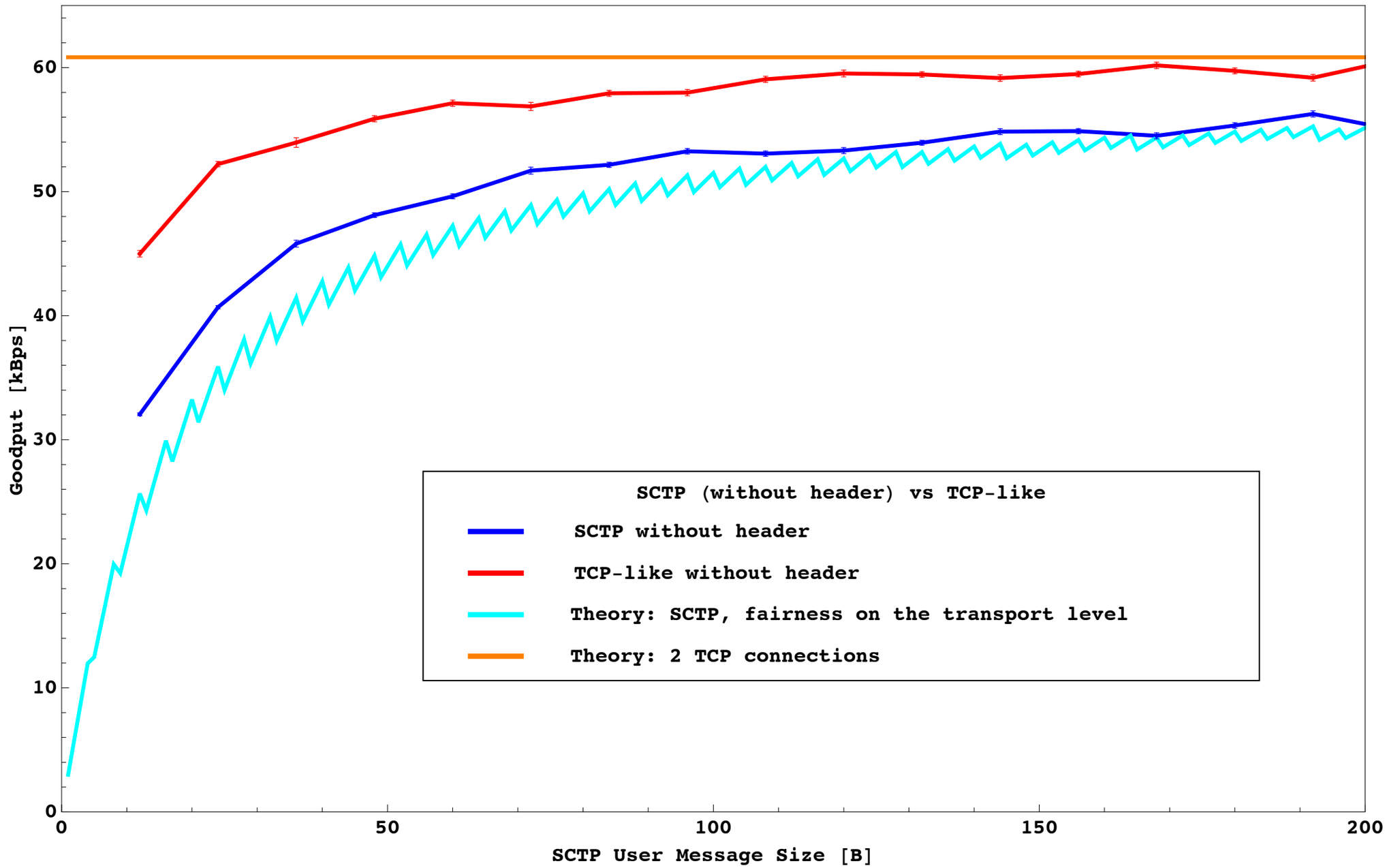
Simulation Setup



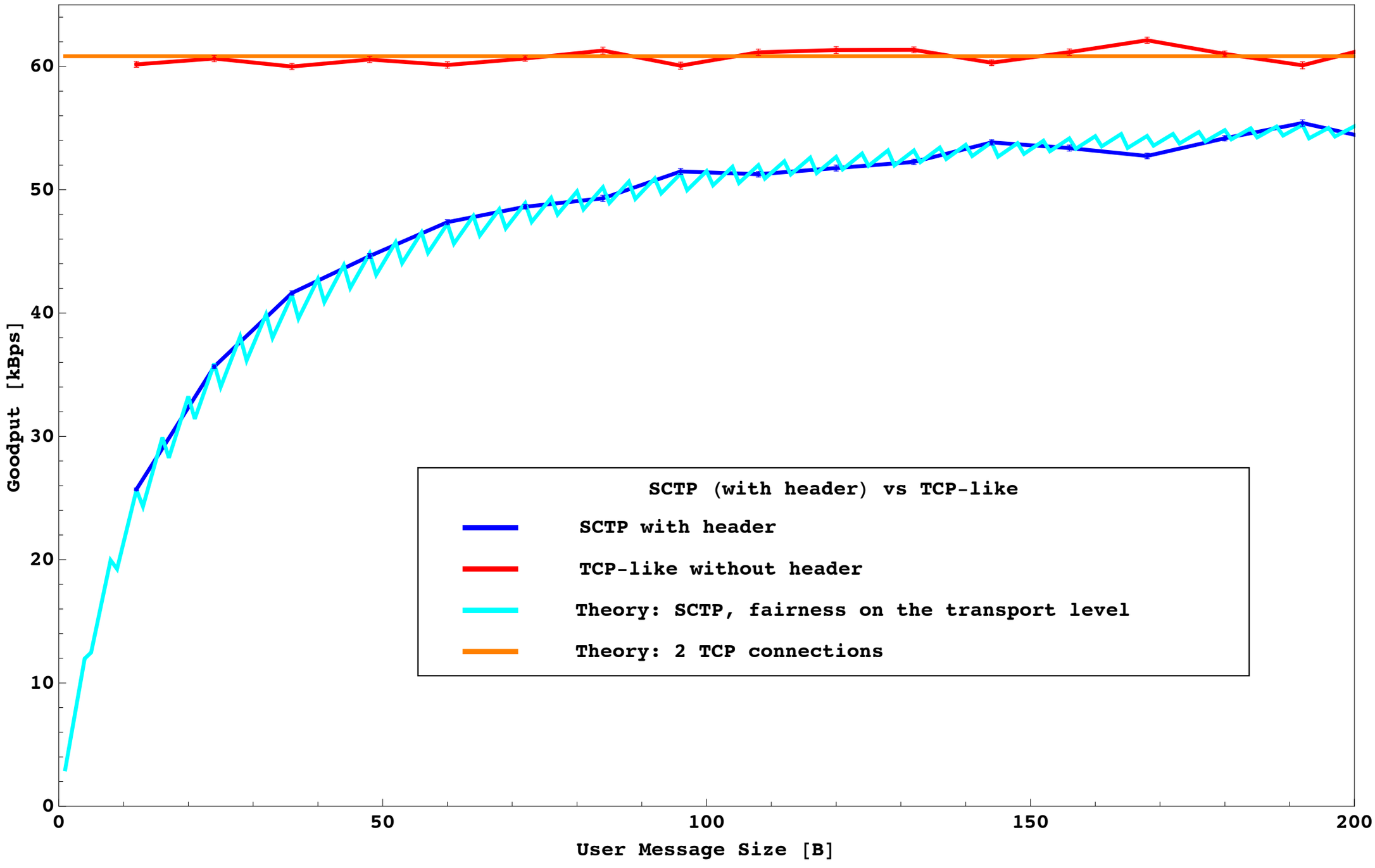
How much Bandwidth do TCP and SCTP get?



How much Bandwidth do the Applications get?



How much Bandwidth do the Applications get?



Conclusion

- For SCTP's congestion control taking the DATA chunk overhead into account matters.
- For TCP-friendliness the overhead SHOULD be taken into account.
- A clarification for RFC 4960 should be added.
- The same result applies to other protocols having a varying overhead.