Interfaces for Path Selection

Laurent Chuat, ETH Zurich

IETF 100, Singapore,
Path-Aware Networking Proposed RG (PANRG),
16 November 2017

ETH zürich

Parallelism / Redundancy

Goal: improve the reliability, availability, performance, and capacity of computer systems.

Examples:

- Multi-processor systems
- Multi-core CPUs
- Multi-disk (RAID) storage
- [...]

What about network paths?

Today's Network Paths

Heterogeneous links

- Optical fiber
- Pair of conductors
- Wireless
  - WiFi (802.11)
  - Cellular (3G, 4G, ...)
  - Personal Hotspot (Bluetooth + Cellular)

Tomorrow's Network Paths

Future networks

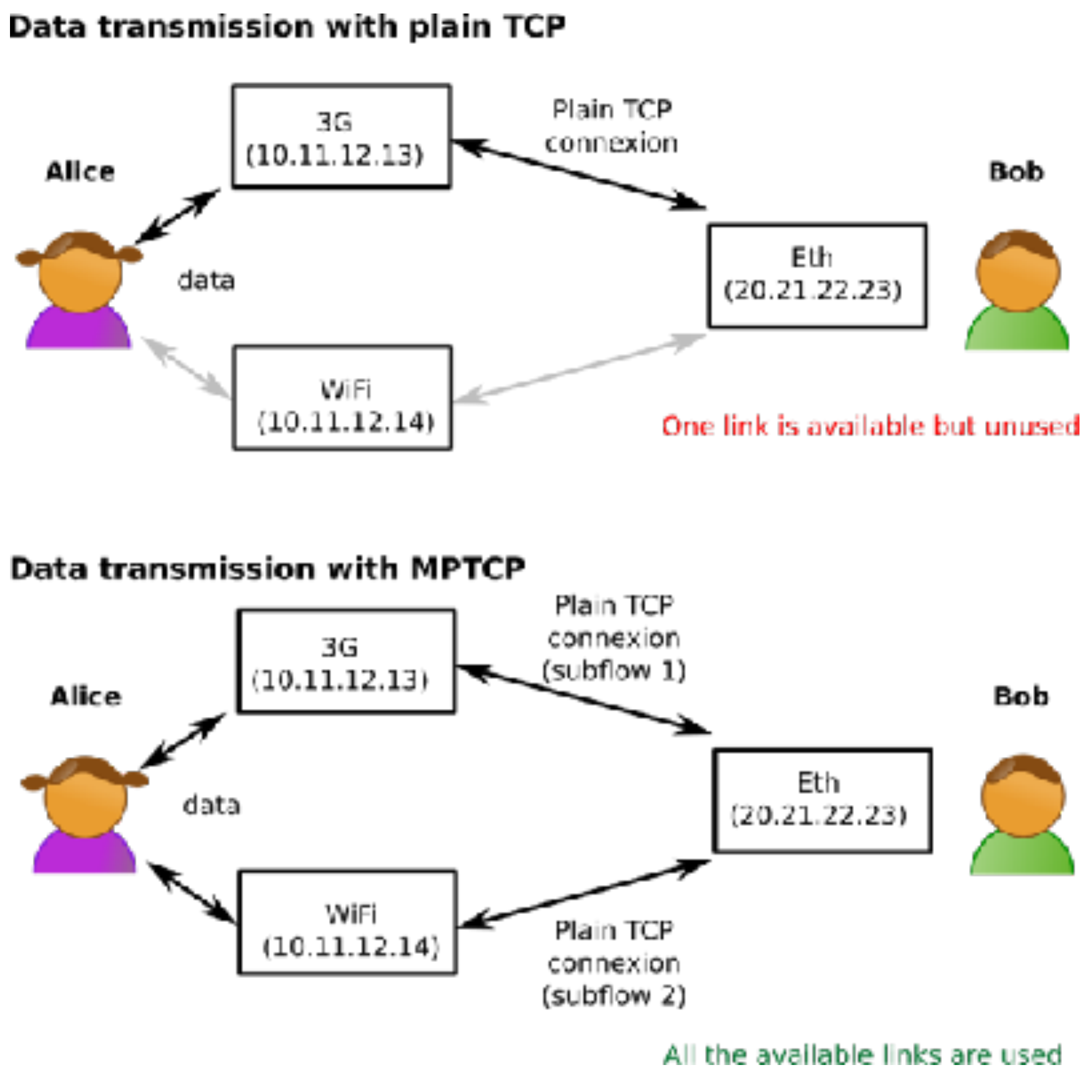
- Google's Loon project
- Facebook's Aquila project
- SpaceX's low-orbit satellites project

Future Internet architectures

- SCION
- NEBULA
- NIRA

Multipath TCP (MPTCP)

- Extension of TCP, backward compatible
- Specifically designed to hide multipath communication specificities from the application.
- RFCs 6182, 6824, 6897



(Wikipedia)

Transport Protocols and Application Types

Reliable Transport (TCP) ➡ **Multipath TCP (MPTCP)**

- Web browsing
- File transfer
- [...]

Unreliable Transport (UDP)

- Request/Response: ➡ **Send along: best path / all paths**
  - Domain Name System (DNS)
  - [...]
- Real-time communication: ➡ **Possibly the most challenging**
  - Voice over IP (VoIP)
  - Teleconferencing
  - Gaming
  - [...]

The Berkley Sockets API

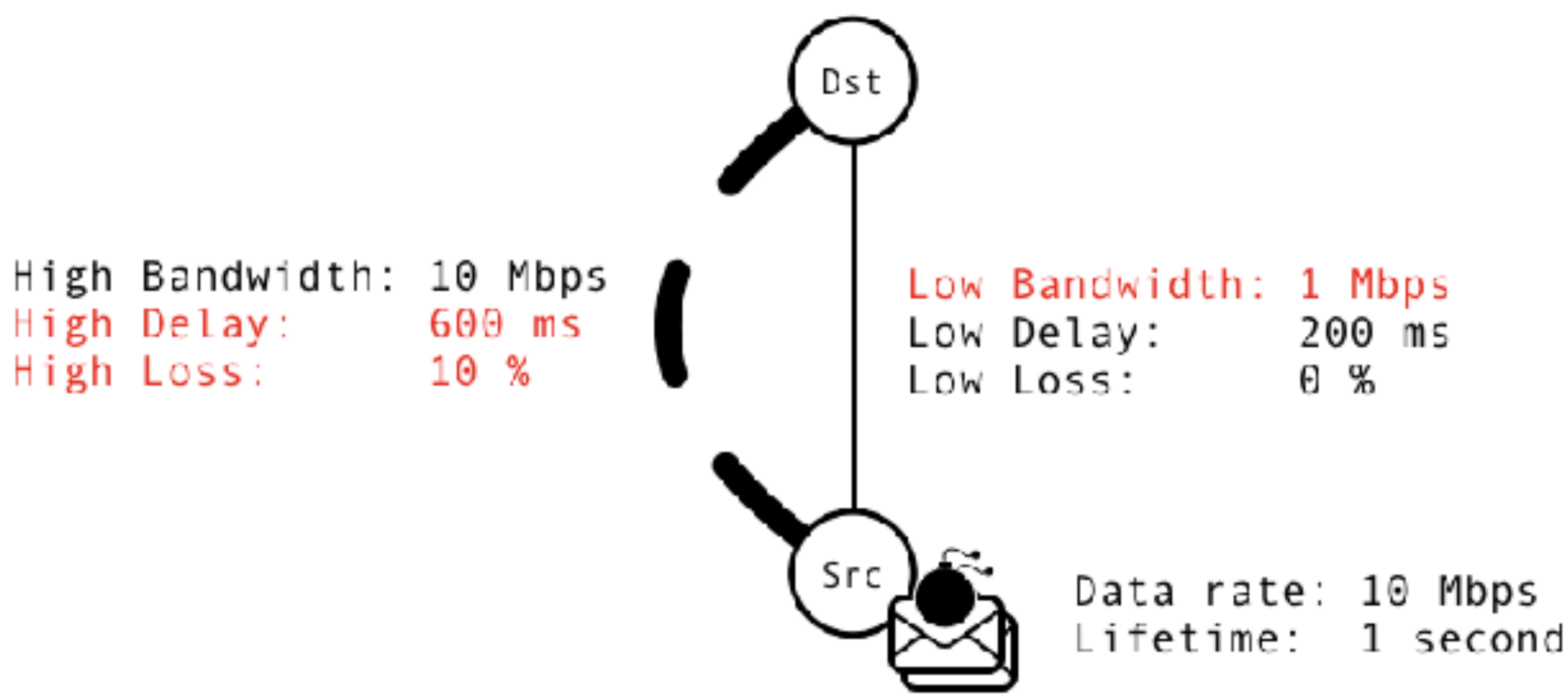
- Introduced in 1983, the Berkley sockets API was a revolution in simplicity.
- But it may be too simplistic for modern applications and networks...
- **Main abstractions:** `SOCK_STREAM`, `SOCK_DGRAM`
  - The network is not really a file, and datagrams do not offer much functionality.

What about `SOCK_SEQPACKET`?

- **Synchronous** (with async event notification)
- **Multipath** (but for *failover* only)
- **No path abstraction**
- Bound to the **Stream Control Transmission Protocol (SCTP)**, not extremely deployable in the open Internet today.

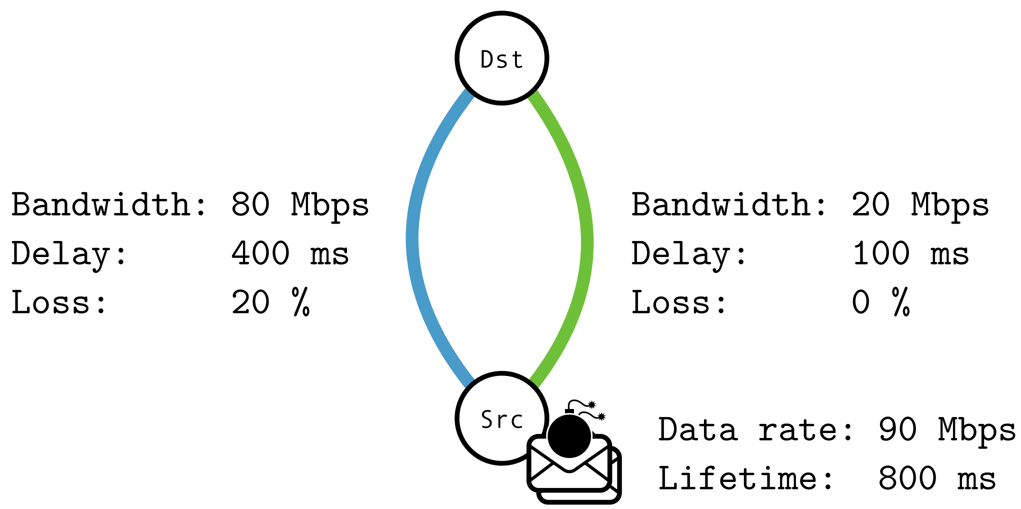
Where Current Models Fall Short: An Example

Goal: deliver as many packets as possible before their *deadline*.

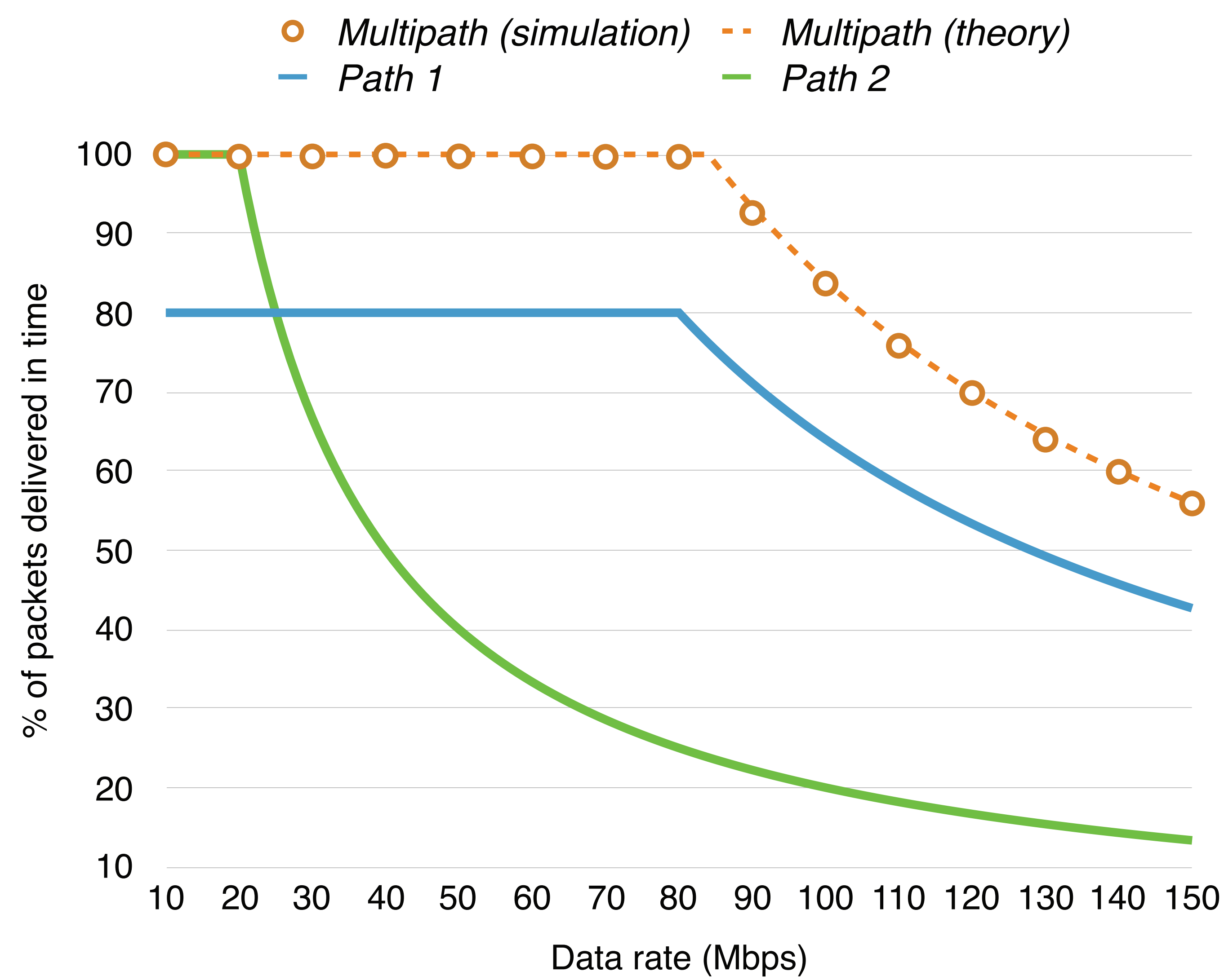


Sending initially the data along the left-hand path and retransmitting lost packets along the right-hand path, we can expect to deliver **100% of the packets** before their deadline.

Network Performance Gain Through Linear Optimization



$$
\begin{array}{ll}
\text{maximize} & p^{\mathsf{T}} \cdot x \\
\text{subject to} & A \cdot x \leq q, \\
 & B \cdot x = 1, \\
\text{and} & x \geq 0
\end{array}
$$



Deadline-Aware Multipath Communication: An Optimization Problem. L.Chuat, A. Perrig, Y. Hu, DSN 2017.

Towards More Powerful Sockets

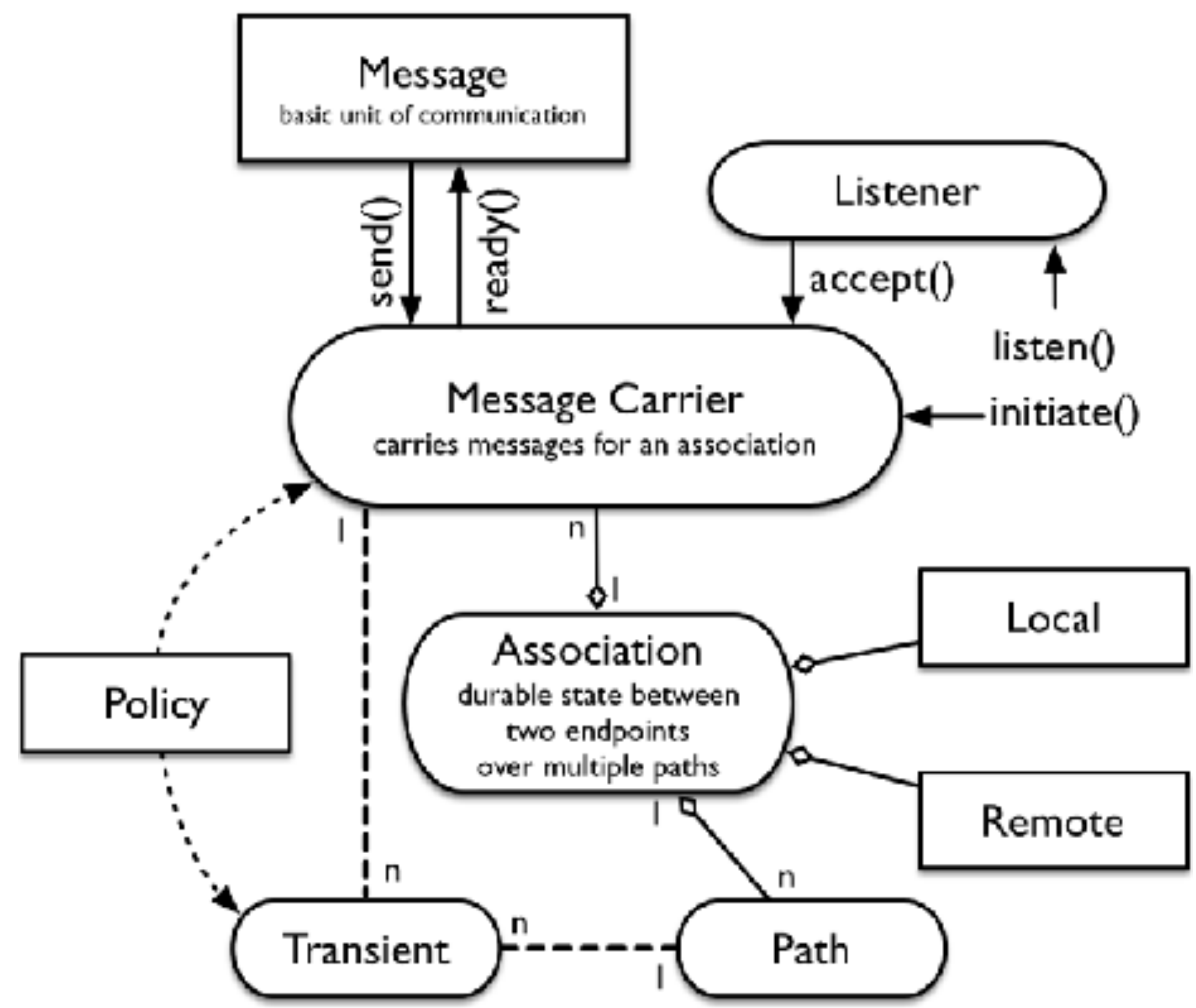
- **Main objective:** defining an expressive but simple path-aware socket API that is language, protocol, and architecture independent.
- **Main questions:**
  - **Knobs:** What should the application tell the transport layer about its requirements? So that the transport can make appropriate decisions, such as:
    - Packet-to-path assignment
    - Wait/Send/Retransmit/Drop

  - **Dials:** What should the application be able to learn about the transfer of its messages?

Post Sockets (Trammell et al.)

Goal: transport-, platform-, and language-independent API for present and future transport protocols, which supports dynamic selection of stacks.

- **Multipath** environment
- **Asynchronous** reception
- **Long-lived** associations
- **Message** abstraction



Message Properties

- **Lifetime / Partial Reliability:** period during which the transport protocol should attempt to deliver the message. After this period, the message should be discarded.
- **Priority ("Niceness"):** express which messages should be delivered first.
- **Dependence:** specify whether other messages must be delivered beforehand.
- **Idempotence:** safe to send in situations that may cause the message to be delivered more than once.
- **Immediacy:** do not wait to combine this message with other messages or parts thereof.

Message Properties: Lifetime / Partial Reliability

TCP (stream) vs. UDP (datagram) is often a false dichotomy.

Partial reliability: packets with a lifetime/deadline may be retransmitted, but only for a limited period. This is particularly useful for real-time communication.

Existing transport protocols:

- Partial Reliability extension of the Stream Control Transmission Protocol (PR-SCTP)
- Deadline-Aware Datacenter TCP (D^2TCP)

Message Properties: Priority ("Niceness")

- **Niceness** is represented by an unbounded non-negative integer and is the **inverse of priority**.
- By default, *niceness = 0*, i.e., the highest priority.
- This inversion has convenient properties:
  - Priority increases as both niceness and lifetime decrease.
  - High priority by default, can be reduced arbitrarily.

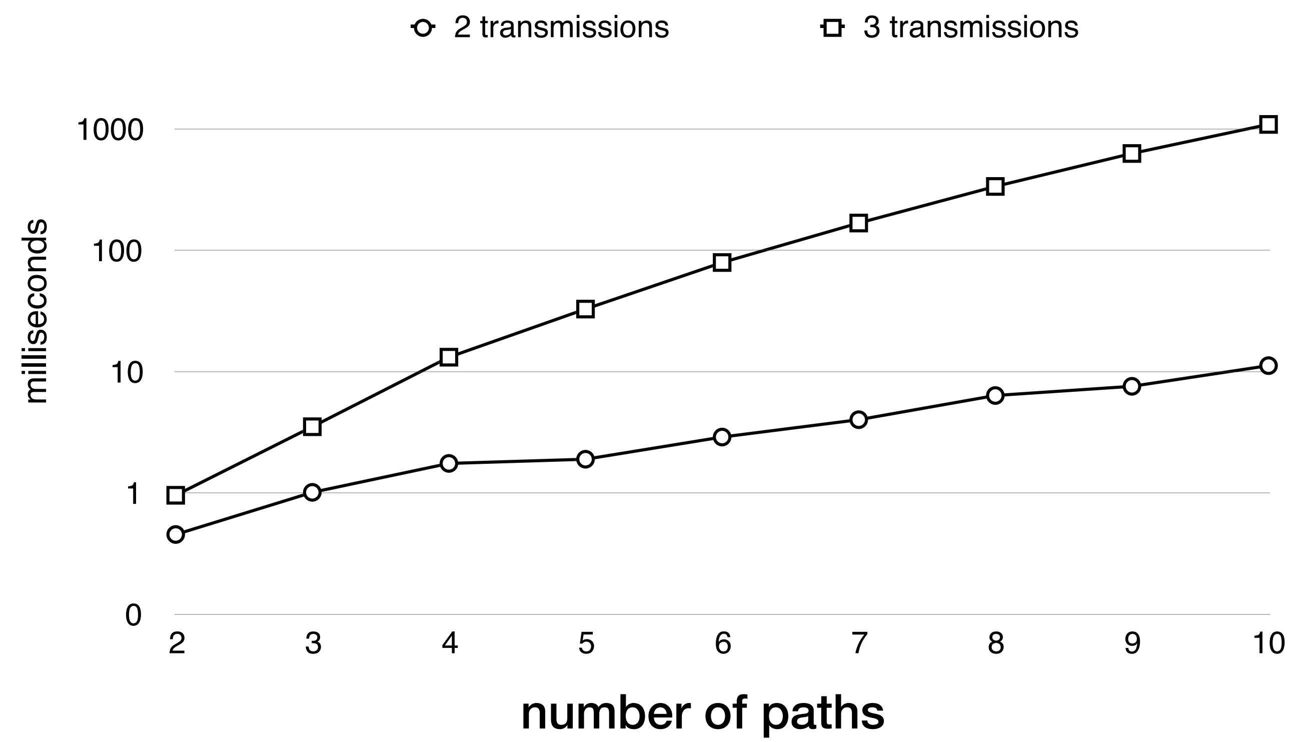
Message Properties: Dependence

- A message may have "antecedents", i.e., other messages that must be delivered first.
- Can be combined with lifetime and niceness to determine when to send which message down which path.
- **Example:** A web page should preferably be delivered before embedded media.

Policies

- An application may require, or prefer to use, certain features of the transport protocols. It may also prefer paths/interfaces over others.
- Reasons for defining policies: *Cost*, *Performance*, *Security/Privacy*
- **Multiple domains:**
  - application policy
  - user policy
  - system policy
- **Example 1:** WiFi might be preferred over LTE when roaming, due to cost.
- **Example 2:** An application might require that its messages do not go through certain ASes, for security reasons.

Hard Problems Ahead...



Bad news:

- Optimally assigning packets to paths is a hard problem.

Good news:

- The problem must not necessarily be solved for each packet.
- Not finding the optimal solution does not mean that we cannot find a good solution (with heuristics).
- ***A more expressive, unified sockets API could drive and focus new multipath research.***

Questions?

Laurent Chuat, ETH Zurich
(laurent.chuat@inf.ethz.ch)