

*Multipath-Transfer  
mit dem  
Stream Control Transmission Protocol*

---

Dr. Thomas Dreibholz

Lehrstuhl Technik der Rechnernetze

Institut für Experimentelle Mathematik

Universität Duisburg-Essen

- Motivation und Einführung
- Ungleichartige Pfade
- Fairness
- Zusammenfassung

# Überblick: Motivation und Einführung

## ■ Motivation und Einführung

- Klassische TCP/IP-Datenübertragung
- Multi-Homing und Multipath-Transfer
- Herausforderungen

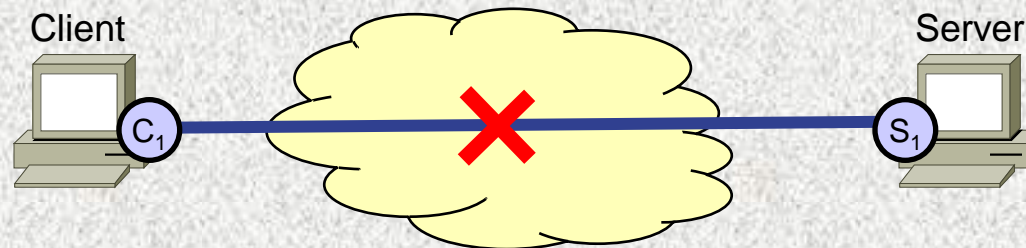
## ■ Ungleichartige Pfade

## ■ Fairness

## ■ Zusammenfassung

## TCP/IP-Verbindung

- Initiiert vom Client, Adresse  $C_1$  zum Server, Adresse  $S_1$



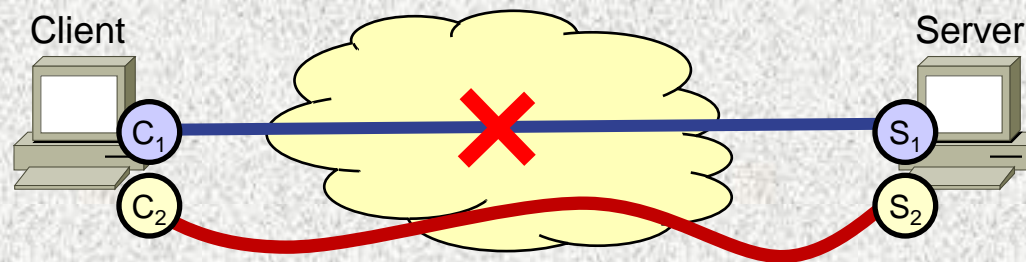
- Pfad zwischen  $C_1$  und  $S_1$  unterbrochen => Verbindung bricht ab
  - auch wenn alternative Pfade über andere Adressen möglich wären
- Kein (großes) Problem für klassische Anwendungen wie E-Mail und FTP
- Aber kritisch für neue Dienste
  - E-Commerce, Telemedizin, Telefonsignalisierung über IP-Netze,
  - und viele weitere

**Ziel: Ausfallsicherheit für kritische Anwendungen**

FTP: File Transfer Protocol; IP: Internet Protocol; TCP: Transmission Control Protocol

## ■ Offensichtlicher Ansatz: Multi-Homing

- Ausfallsicherheit durch weitere Pfade: mehr Interfaces und Adressen

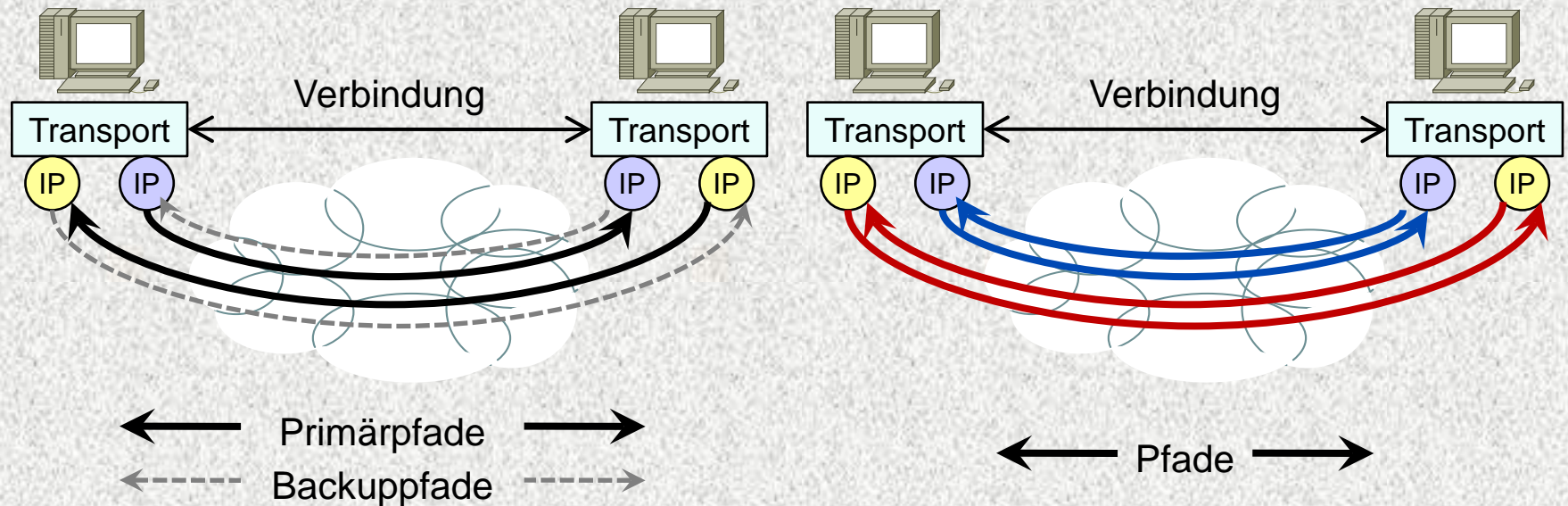


- TCP unterstützt kein Multi-Homing, aber Anwendung kann Verbindung zwischen C<sub>2</sub> und S<sub>2</sub> erneut aufbauen
- Erfordert Handlung von allen Anwendungen
  - oder ein leistungsfähigeres Transportprotokoll

### Ziele:

1. Transportprotokoll mit Multi-Homing-Unterstützung
2. Mehrere Pfade gleichzeitig nutzen





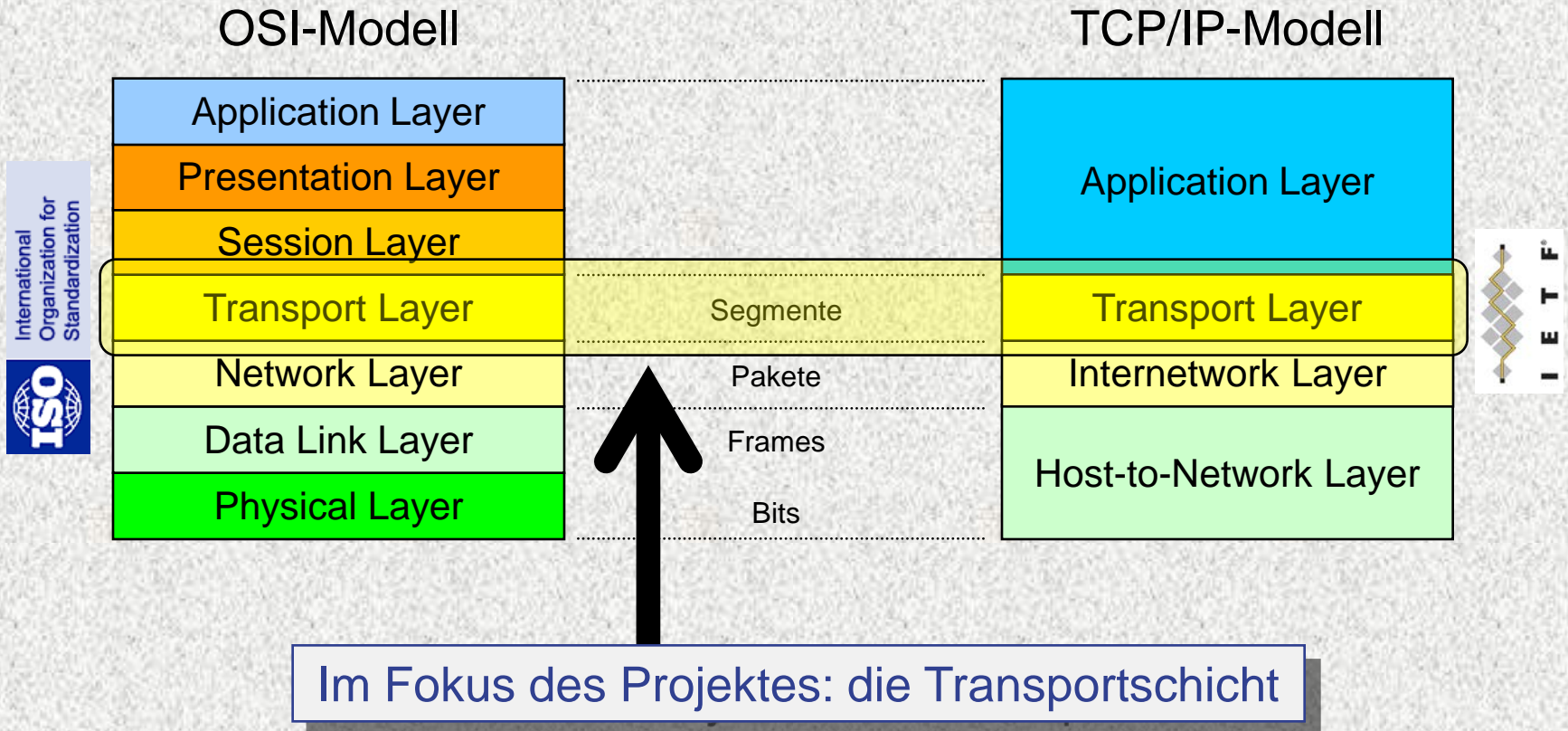
## ■ Multi-Homing

- Nur ein ausgewählter Pfad für Nutzdaten („**Primärpfad**“)
- Andere Pfade als Backup

## ■ Multipath-Transfer

- **Gleichzeitige** Verwendung **mehrerer Pfade** für Nutzdaten

# Projektfokus in OSI- und TCP/IP-Referenzmodellen



IETF: Internet Engineering Task Force; ISO: Int'l Organisation for Standardization; OSI: Open Systems Interconnection

# Multipath-Protokolle und deren Herausforderungen

## ■ Protokollerweiterungen

- **MPTCP**: Multi-Path TCP für TCP
  - Université catholique de Louvain, University College London
- **CMT-SCTP**: Concurrent Multipath-Transfer für SCTP
  - University of Delaware, WIDE Project

## ■ In IETF-Standardisierung; Thema aktueller Forschungsarbeiten

- Unterschiedliche Protokolle, aber die Probleme sind die gleichen!

### Zentrale Herausforderungen:

- **Nutzdatendurchsatz** muss mindestens so gut sein wie Nicht-Multipath-Protokoll über den besten Pfad
- **Fairness** bei Pfaden mit gemeinsamen Engpässen

## ■ Untersuchungen am Beispiel CMT-SCTP

SCTP: Stream Control Transmission Protocol



# SCTP – Stream Control Transmission Protocol

## ■ Definiert in RFC 4960

## ■ Eigenschaften

- Zuverlässige Übertragung mit Fluss- und Überlastkontrolle (wie TCP)
- **Nachrichtenorientiert** (statt Bytestrom wie bei TCP)
- Nachrichtenauslieferung:
  - Unordered (ohne Reihenfolgeerhaltung)
  - Ordered (mit Reihenfolgesicherung; wie TCP)
- **Multi-Homing**
  - Primärpfad je Richtung
  - Andere Pfade als Backup und für Paketwiederholungen
- **Multi-Streaming**
  - Multiplexing mehrerer Datenströme über eine Verbindung
  - Verhindert „Head-of-Line Blocking“

## ■ Protokollerweiterungen

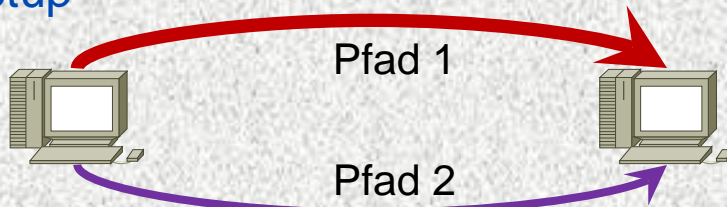
- z.B. unzuverlässige Übertragung, Mobilität, Sicherheit, CMT-SCTP, ...

# Überblick: Ungleichartige Pfade

- Motivation und Einführung
- Ungleichartige Pfade
  - Einführung
  - Problem des Buffer Blockings
  - Unordered-Übertragung
  - Ordered-Übertragung und weitere Verbesserungen
- Fairness
- Zusammenfassung

# Herausforderung von CMT-SCTP über ungleichartige Pfade

## Setup

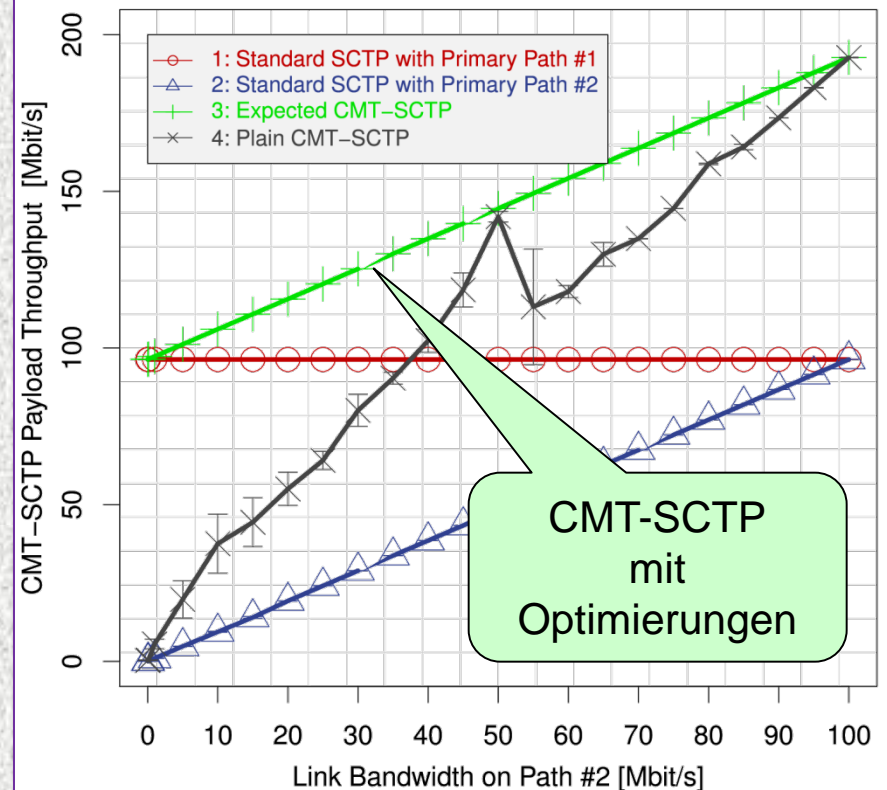


- 2 Pfade;  
100 Mbit/s, 1 ms, 0% Verlust
- Saturierter Sender,  
Unordered-Übertragung
- Bandbreitevariation auf Pfad 2

## Simulationsergebnisse

- Sctp, Primärpfad über Pfad 1 ✓
- Sctp, Primärpfad über Pfad 2 ✓
- Erwartet für CMT-SCTP 😊
- Ursprüngliches CMT-SCTP ☹

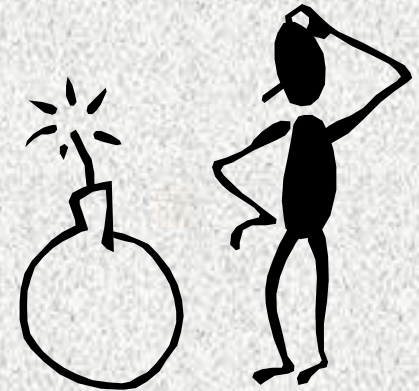
## Nutzdatendurchsatz



# Effizientes CMT-SCTP über ungleichartige Pfade

## ■ Komplexität durch **Interaktion** verschiedener Mechanismen

- Überlastkontrolle
  - Congestion Window und Slow-Start Threshold
  - Burst Mitigation
- Verwaltung von Sende- und Empfangspuffern
- Handhabung von Paketwiederholungen
  - Timer-Based Retransmission
  - Fast Retransmission
- Quittungsmechanismus
  - Kumulative Quittungen
  - Selective Acknowledgements (SACK) und deren Widerrufbarkeit
- Optionen für Nachrichtenauslieferung (unordered, ordered)



## ■ Idee:

- Problem in Teilprobleme zerlegen
- Lösung zunächst für Unordered-Übertragung ...
- ... und dann für Ordered-Übertragung



- Beobachtung: Blockierung durch vollen Sende- oder Empfangspuffer
- Zerlegung in Teilprobleme => einzeln verstehen und lösen
- **Send Buffer Blocking (SBB)**
  - Transmission-Induced SBB
    - Zu viele Daten auf einem Pfad gepuffert, kein Platz mehr für weitere Daten auf anderen Pfaden
  - GapAck-Induced SBB
    - Selective Acknowledgement ist vom Empfänger widerrufbar
    - Pufferplatz blockiert durch Warten auf kumulative Quittung
- **Receive Buffer Blocking (RBB)**
  - Advertised-Window-Induced RBB
    - Platz im Advertised Receiver Window durch einen Pfad fast komplett belegt => zu wenig Platz für anderen Pfad
  - Reordering-Induced RBB
    - Warten auf fehlende Daten zur Reihenfolgewiederherstellung



## ■ Send Buffer Splitting

- $n$  Pfade => max.  $1/n$  des Sendepuffers je Pfad ausstehend
- Löst: Transmission-Induced SBB

## ■ Receive Buffer Splitting

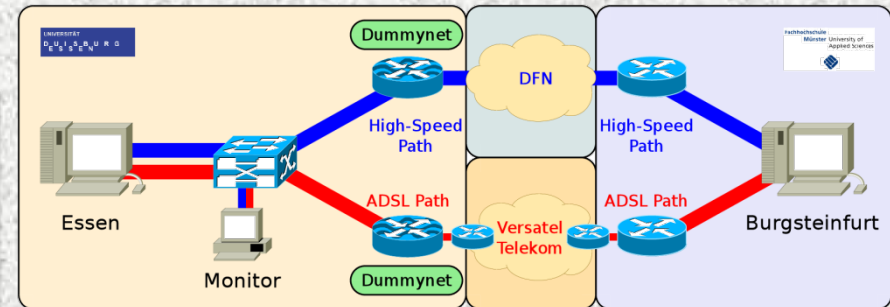
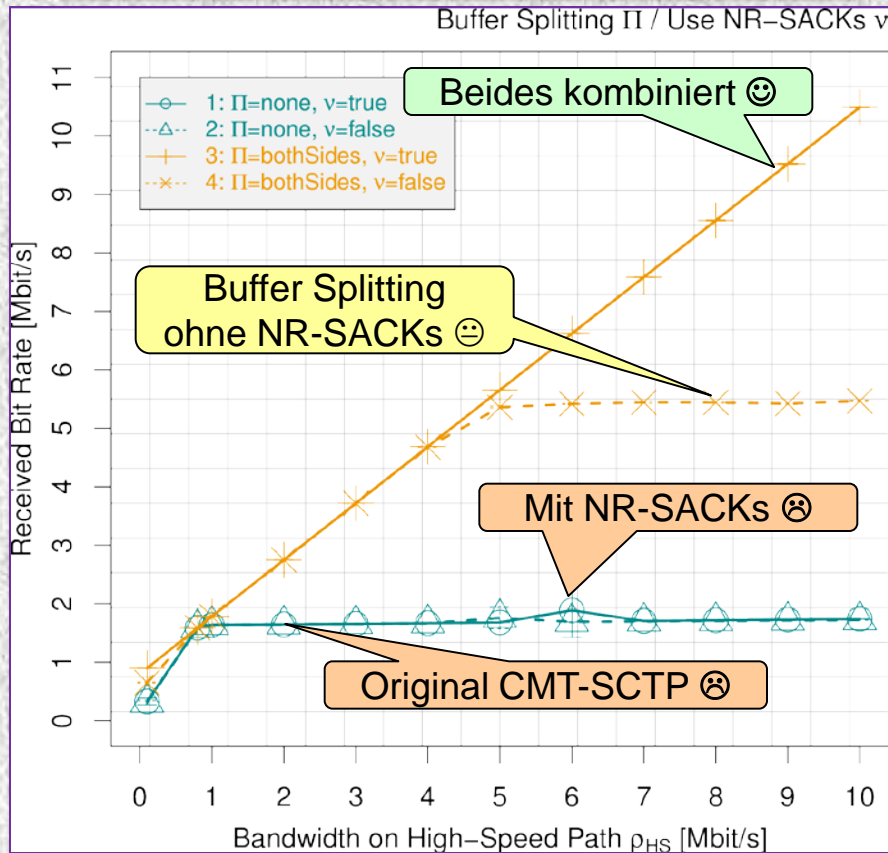
- $n$  Pfade => max.  $1/n$  des Empfangspuffers je Pfad ausstehend
- Mechanismus komplett auf Senderseite implementierbar
- Löst: Advertised-Window-Induced RBB

## ■ Non-Renegable SACK (NR-SACK)

- Nicht-widerrufbare SACKs
- Ursprünglich: Mechanismus zum Sparen von Sendepufferplatz für Standard-SCTP (University of Delaware)
- Segment NR-SACK'ed => kann aus Sendepuffer entfernt werden
- Löst: GapAck-Induced SBB

## ■ Kombination der Mechanismen für Unordered-Übertragung; Veröffentlicht auf IEEE GLOBECOM 2010

## Nutzdatendurchsatz



### Messung in realem Internet-Testbett:

- Pfad 1: Glasfaser –  $x$  Mbit/s, 2ms
- Pfad 2: DSL – 0.8 Mbit/s, 28ms
  - Lange Queue => Buffer Bloat!
- Erwartet: Durchsatz  $\sim x+0.8$  Mbit/s

### Kombination von Buffer Splitting (beidseitig) und NR-SACK löst Probleme bei Unordered-Übertragung!

### Veröffentlicht auf IEEE PAMS 2011

## ■ Ordered-Übertragung

- Problem des Reordering-Induced RBB
- Tritt bei Unordered-Übertragung natürlich nicht auf

## ■ Chunk Rescheduling

- Grundidee: Unterbrechung durch fehlende Pakete verhindern
- Präventive Paketwiederholung des ersten Pakets, sobald Sende- oder Empfangspufferanteil eines Pfades zu 50% blockiert
- Max. 1 Paket/RTT => geringer Overhead
- Veröffentlicht auf IEEE GLOBECOM 2011

## ■ Predefined Stream Mapping

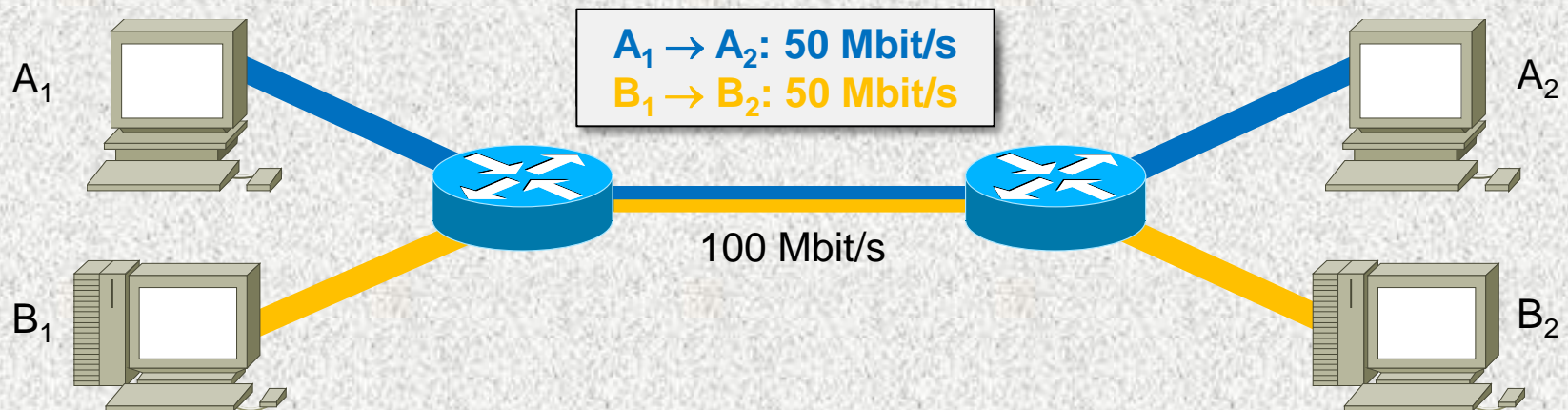
- Mapping von Streams auf Pfade => weniger Reordering
- Beispiel: Secure Shell (SSH)
  - Shell-Stream auf langsamen Pfad mit kleiner Verzögerung
  - X11-Forwarding-Stream auf Pfad mit hoher Bandbreite
- Veröffentlicht auf PFLDNeT 2011

- Motivation und Einführung
- Ungleichartige Pfade
- Fairness
  - Fairness bei nicht-disjunkten Pfaden
  - Resource Pooling
  - Ausblick
- Zusammenfassung



## ■ SCTP-Überlastkontrolle analog zu TCP

- AIMD-Verhalten mit **Congestion Window** (*cwnd*) und **Slow-Start Threshold** (*ssthresh*)
- Faires Verhalten gegenüber TCP-Verbindungen



## ■ Bandbreite auf dem Engpass wird fair auf alle Verbindungen verteilt

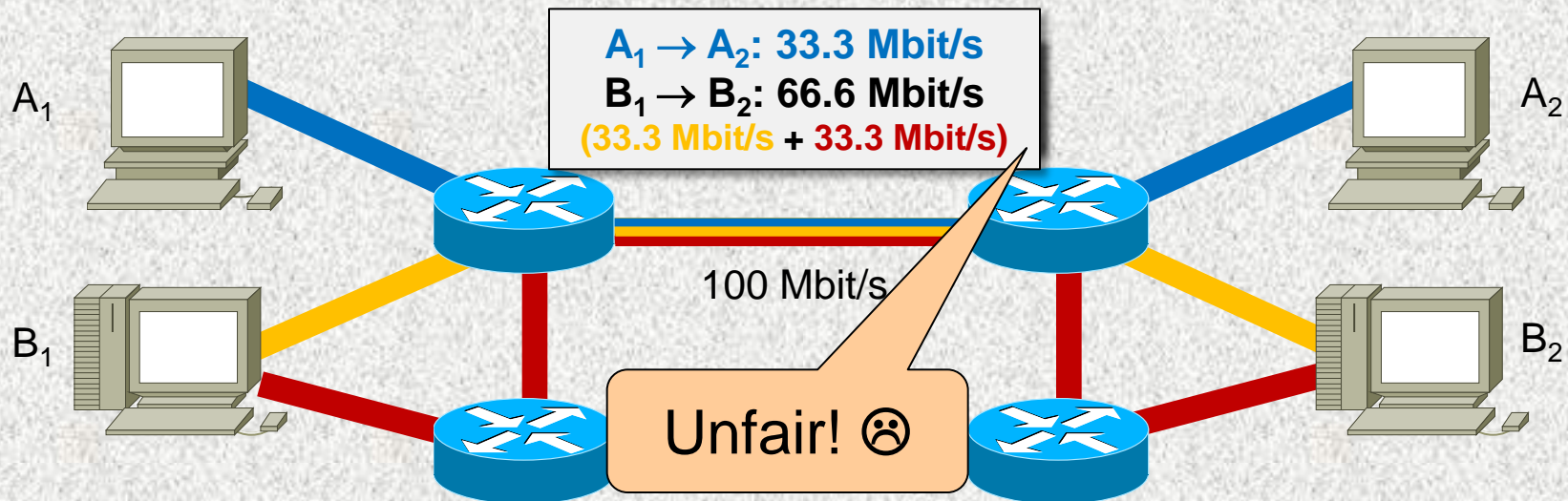
AIMD: Additive Increase Multiplicative Decrease



# Herausforderung bei CMT-SCTP über nicht-disjunkte Pfade

## ■ CMT-SCTP-Überlastkontrolle

- **Behandelt jeden Pfad unabhängig**
- Funktioniert erwartungsgemäß, unter Annahme von **disjunkten Pfaden**



## ■ Gemeinsamer Engpass (engl. shared bottleneck)

- $n$  Pfade über gleichen Engpass -> wie  $n$  Nicht-CMT-Verbindungen
- **Unfaire Bandbreitenverteilung!**
- **Garantien** für disjunkte Pfade sind im Internet **nicht möglich**

## ■ Resource Pooling (RP)

- „*Making a collection of resources behave like a single pooled resource*” [WHB09]
- Idee: **AIMD-Überlastkontrollen** der einzelnen Pfade **koppeln**

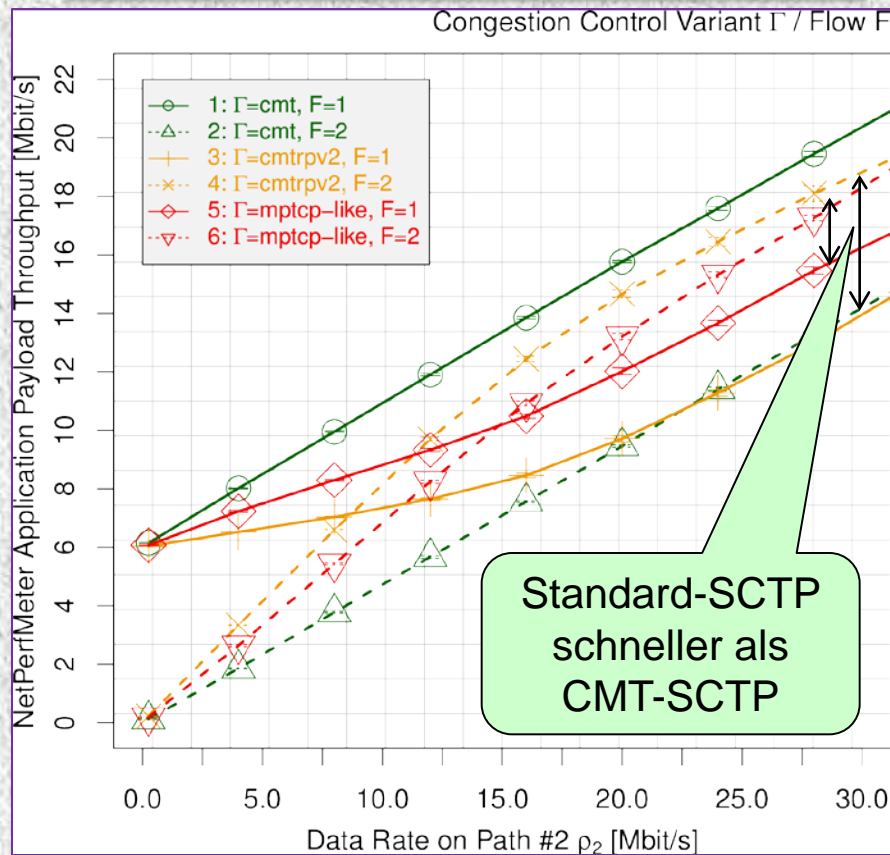
## ■ Neue AIMD-Überlastkontrollverfahren mit RP:

- **CMT/RPv1** – skaliert mit ssthresh-Verhältnis
  - Funktioniert, sofern ssthresh-Verhältnisse vergleichbar
  - Bei ungleichartigen Pfaden ggf. nicht der Fall
- **CMT/RPv2** – skaliert basierend auf Bandbreiten (-> cwnd/RTT)
  - Funktioniert auch für ungleichartige Pfade
- **MPTCP-like** – MPTCP-Ansatz auf CMT-SCTP übertragen
  - Cwnd-Senkung wie Standard-SCTP
  - Cwnd-Erhöhung skaliert mit Aggressivitätsfaktor

## ■ Veröffentlicht auf IEEE ConTEL 2011 und IEEE AINA 2010

# Was passiert bei ungleichartigen, disjunkten Pfaden?

## Nutzdatendurchsatz



### Ziele von RP:

1. CMT-SCTP-Bandbreite mindestens wie für Standard-SCTP über besten Pfad
2. Fairness bei gem. Engpass
3. Last auf weniger ausgelasteten Pfad verschieben

### Szenario:

- CMT-SCTP (Flow #1; —)
- Standard-SCTP (Flow #2; - - -)
- Zwei disjunkte Pfade:
  - P1: 6.25 Mbit/s; P2: x Mbit/s
- Original, CMT/RPv2, MPTCP-like

### CMT/RPv2- und MPTCP-like-Verfahren sind fair gemäß den gesetzten Zielen!

- Aber: Ist das wirklich „fair“?
- Diskussion von „Multipath-Fairness“ als zukünftige Arbeit!

# Überblick: Zusammenfassung

- Motivation und Einführung
- Ungleichartige Pfade
- Fairness
- Zusammenfassung



## ■ FH Münster in Burgsteinfurt

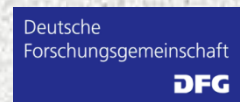
- **DFG-Projekt zu SCTP**
  - Zwei Phasen
  - Kooperation mit der University of British Columbia, Vancouver/Kanada
- Beiträge zur **IETF-Standardisierung**
  - Kooperation mit Huawei, Cisco und anderen

## ■ Weitere Forschungsaktivitäten

- BMBF German Lab (G-Lab)
  - Multipath-Übertragung im **Future Internet**
  - Special Interest Group: **SIG Multipath**
- **PAMS Workshop** (auf IEEE AINA 2011 und 2012)
- Open-Source-Werkzeuge -> Sichtbarkeit des Projektes

## ■ Hainan University in Haikou, Hainan/China

- Förderantrag zur SCTP-Pfadhandhabung





## ■ Multipath-Transfer

- Aktuell diskutiert in IETF und Forschung (MPTCP, CMT-SCTP)
- Probleme sind unabhängig vom Protokoll
  - Performanz
  - Fairness

## ■ Performanz bei ungleichartigen Pfaden

- Probleme -> Zerlegung in Teilprobleme und deren Lösung
- Buffer Splitting, Chunk Rescheduling, Predefined Stream Mapping

## ■ Fairness

- Durch geeignete Überlastkontrollverfahren handhabbar

## ■ Ergebnisse wurden in IETF-Standardisierung eingebracht

## ■ Ausblick auf laufende und zukünftige Arbeiten

- Weitere Untersuchungen im Testbett -> evtl. auf G-Lab basierend
- Verallgemeinerung der Fairness-Definition für Multipath-Transfer