Optimizing TCP Performance over UMTS with Split Proxy

Liang Hu

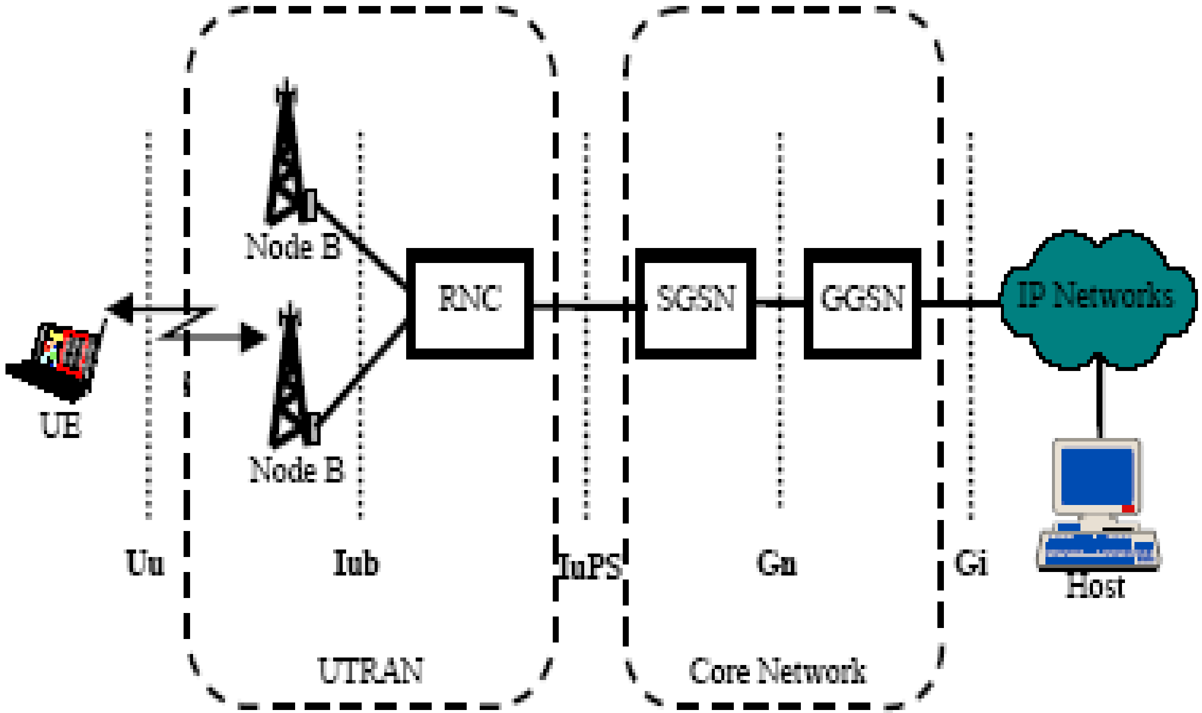
Network Group, COM Centre
Technical University of Denmark

Outline

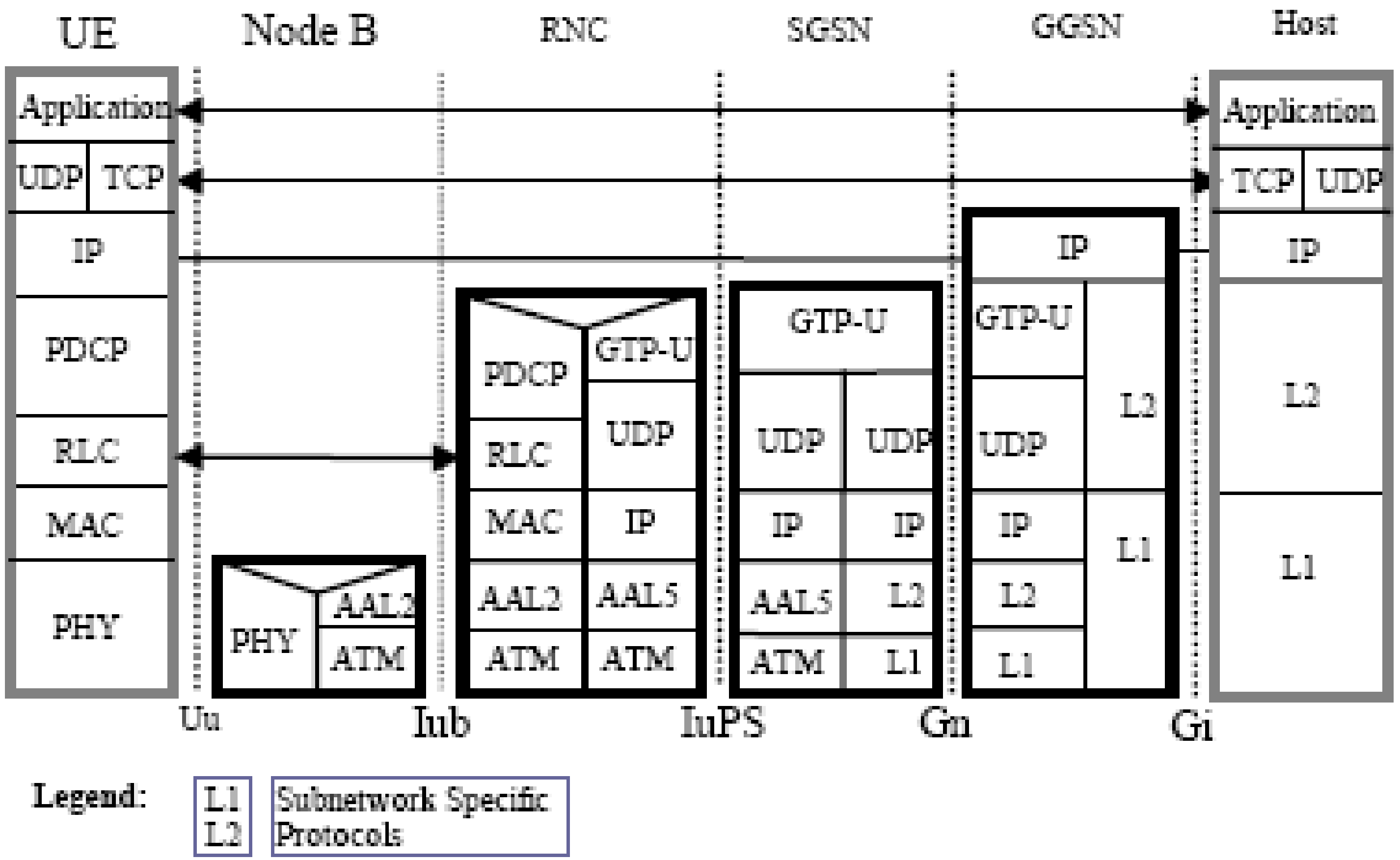
- Recall background knowledge
- Why study TCP over WCDMA
- Performance Evaluation
- Split TCP Proxy
- Future work and expected results

Background

Network Architecture in PS operation



Protocol Stack



Why study TCP over WCDMA

In WCDMA, TCP performance is harmed by:

- Large Bandwidth Delay Product (BDP)
  e.g. take more RTT to reach BDP during slow start, low utilization TCP pipe. BDP=4 kbytes or 24 kbytes when Delay=500 ms, Bandwidth=64 kbps or 384 kbps

- Higher Bit Error Rate in radio link
  - Packet loss rate could be up to 10%

  trigger non-congestion related TCP timeout or fast retransmit

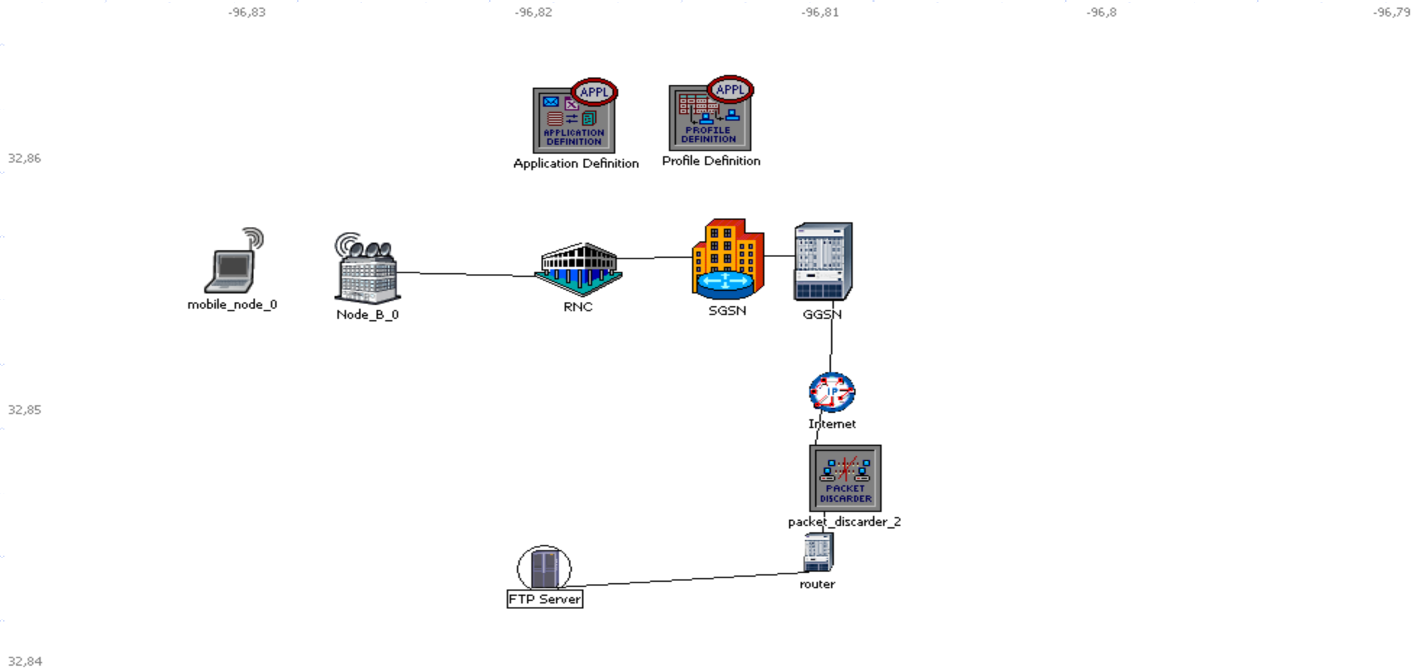
- Spurious Timeout or Spurious fast retransmit, e.g. due to packet re-ordering in handover

The focus of this work

- Optimal link from TCP performance perspective
  - High speed, low delay
  - no non-congestion-related packet losses
  - non-packet re-ordering
- Current solutions for improving TCP over WCDMA
  - ARQ at RLC layer to avoid packet loss due to radio link errors
  - RLC in order delivery to avoid Packet re-ordering
- Rather than impact of non-congestion related packet losses. this work focuses on the impact of Delay Bandwidth Product while incorporating the impact of Internet loss rate,

  It is expected that, in slow start phase, TCP performs well in small BDP scenarios (e.g. 4kbytes) while in large BDP scenarios (e.g.24 kbytes), the TCP pipe can be easily underutilized for several RTTs

Performance Evaluation



Application: FTP

File Inter-Request time: exponential with mean=30s

Simulation Parameter

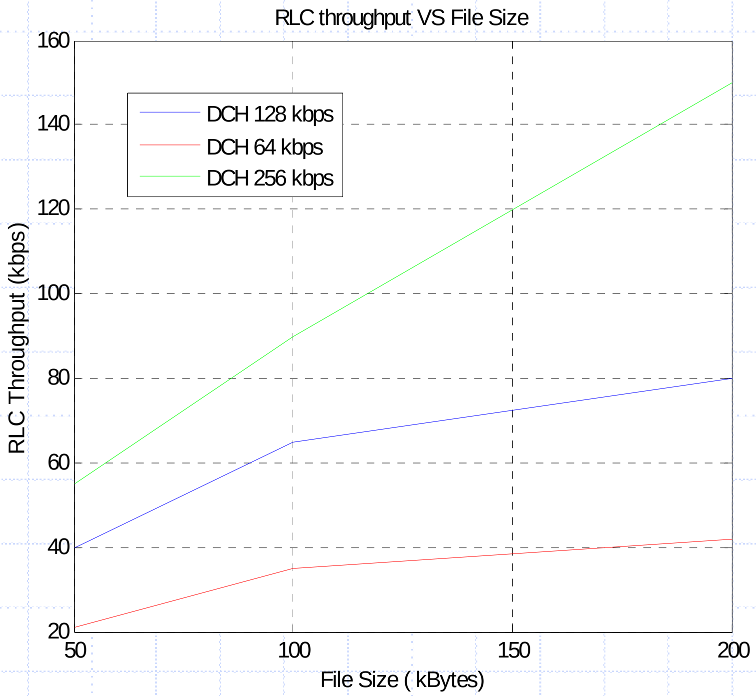
Parameter	Parameter	Description
Physical Layer	Transport Channel Type	DCH
	TTI	10 ms
	Channel Bite Rate	Constant=[64 128 256] kbps
	Closed Loop Power Control	Ideal (Independent erasures Block)
	Outer Loop Power Control	BLER Target=10%
RLC layer	Operation Mode	Acknowledged
	PDU Delivery	In-Sequence
	PDU Size	320 bits
	RLC_Tx Window Size	1024 PDUs
	RLC_Rx Window Size	1024 PDUs
	SDU Discard Mode	After MaxDAT
	MaxDAT	10
	Polling Mechamism	Not used
	Timer Status Prohibit (milliseconds)	60

	Timer STATUS Periodic (milliseconds)	100
PDCP Layer	TCP/IP Header Compression	Header Fully Compressed
TCP layer	Version	Reno
	MSS	536 B
	awnd	32768 B
	Initial cwnd	1 MSS
	Initial ssthresh	awnd
	Initial RTO	3 sec
	Maximum RTO	60 sec
	Minimum RTO	1 sec
	Duplicated ACKs for fast retransmit	3
	Delay of internet and UMTS CN	100 ms (one way)
Application Layer	FTP	300 kb

Goal 1 : Impact of the file size

Simulation Results

- In high bit rate DCH channel, small file size (e.g. web page) degrade RLC throughput dramatically. (200% when file size from 200k to 50k)
- In low bit rate DCH channel, RCL throughput is only slightly affected by file size



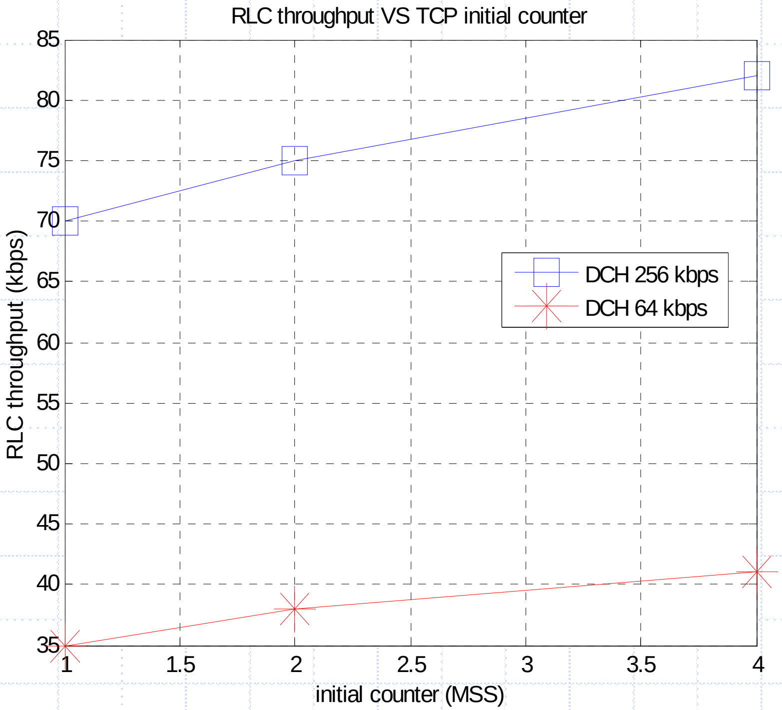
Reasons:

For a given time interval, the largrer file size, the less slow start phases

Goal 2: Impact of slow start initial counter

Simulaton Results:

- A large initial counter gives a higher RLC throughput
- In case of large bit rate DCH, the performance gain is larger than the case of small bit rate channels (32% VS 19%)



Reasons:

- A larger initial counter can fill the TCP pipe more quickly during TCP slow start phase

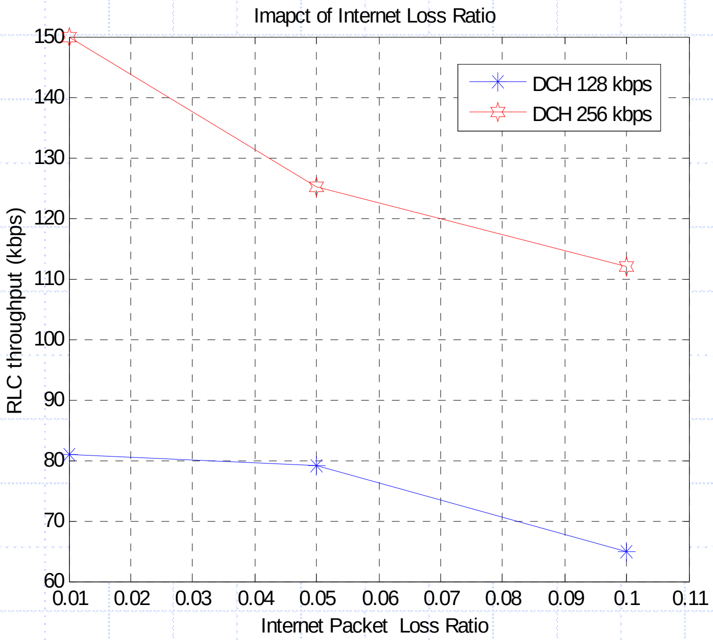
Goal 3: Impact of loss rate in the Internet

Simulaton Results:

- For high bit rate radio link, the impact of internet loss rate is significant to the RLC throughput.
- For low bandwdith radio link, the impact of the Internet loss rate is minor.

Reasons:

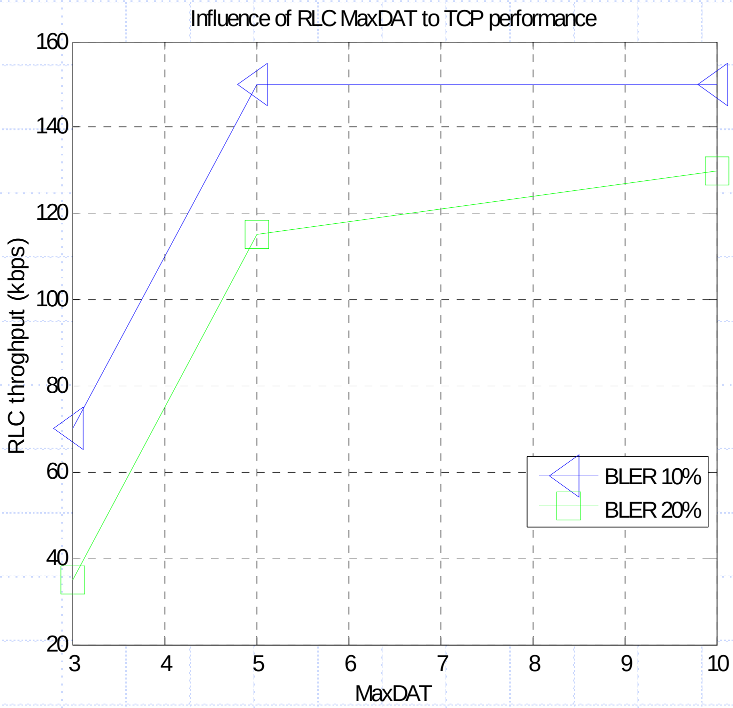
- In larger BDP scenario, the RLC throughput degration (due to fast retransmit triggger by packet loss in Internet) is larger.



Goal 4: Impact of MaxDat at RLC

Simulation Results

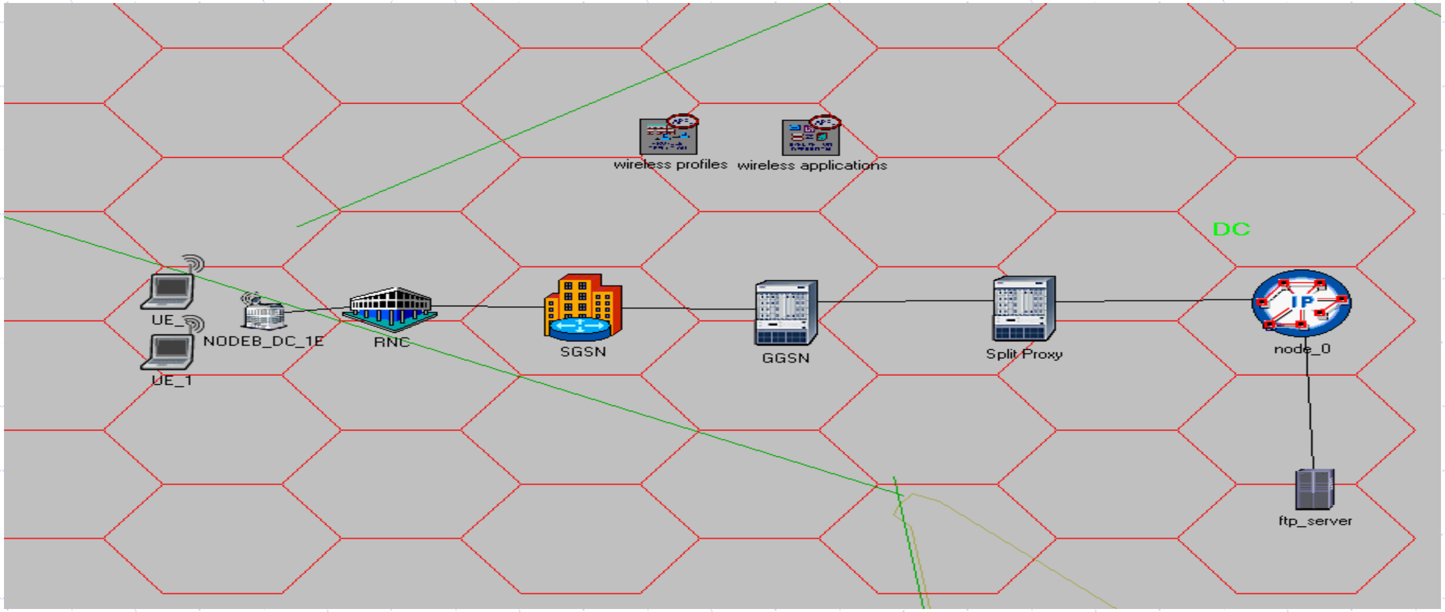
- For a given BLER at link layer, there is a MaxDAT value : above this value the optimal RLC thoughput can be achieved;
- However, a too large MaxDAT may introduce long PDU transmission delay to transport layer and trigger TCP spurious timeout



Reasons:

- Interaction between of RLC retransmission and TCP retransmission: RTT_TCP=(n*RTT_RLC+C);
- RLC layer Error Recovery is preferred than TCP end-to-end Error Correction

WCDMA network architecture with Split TCP proxy



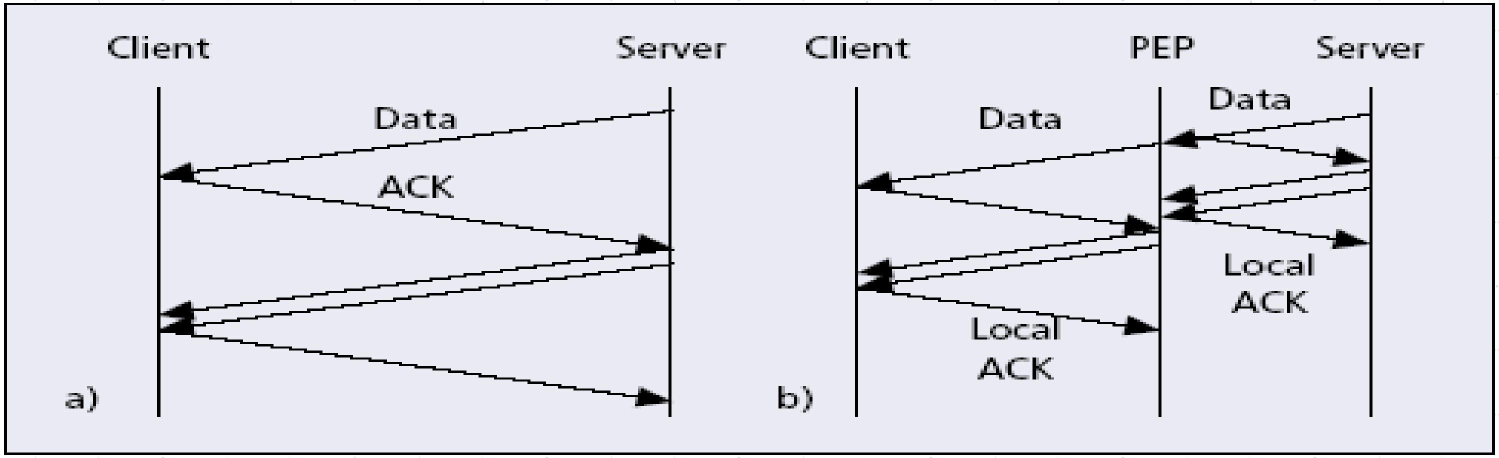
Split TCP Proxy to improve large BDP

- **Local retransmission for Fast Error Recovery is not the case in WCDMA**

  In WCDMA, RLC layer ARQ is assumed to provides a reliable data delivery and hide unsuccessful transmissions of wireless links. Typical BLER 10% can be tolerant by RLC layer.

- **What is the case in WCDMA?**
  - Local ACK (pipelining two TCP connections) at Proxy to cope with large BDP
  - Larger initial congestion window for the TCP connection from Proxy towards UE e.g. up to maximum 10 MSS!

Schematic TCP transfer with and without PEP



Assume the RTT from PEP to Client is larger than Server to PEP

- Without PEP, the server has to wait for the response time from the client before increase the sending window
- With PEP, pipelining for the two TCP connections can be effectively implemented

Simulation Parameters

Compare the effects of with and without Proxy

- RLC throughput as a function of file size (0-1000k bytes) on various DCH channels

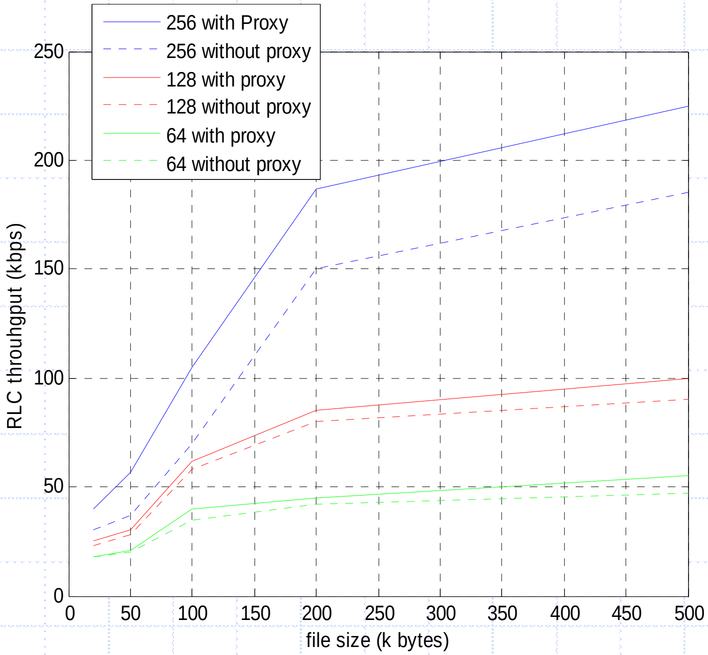
  (64kb/s,128kb/s,384kb/s)

- RLC throughput as a function of BLER (0%-15%)

Parameter	Setting
Transferred file size	50 Kbytes
TCP maximum segment size	1460 bytes
TCP initial window	Three segments
TCP initial window in Proxy	3,5,10 segments
Packet loss rate(internet)	1%
Delay of the internet and UMTS CN	70 ms (one way)
RLC RTT	80 ms
RLC payload size	320 bits
RLC status prohibit timer	90 ms
Transmission time interval (TTI)	10 ms
Data rates on air link	64,128,384 kb/s
BLER on radio link	10%

Simulation Results

- Proxy gives more performance gain for high bit rate DCH
- Proxy gives more performance gain for larger file size both during slow start and fast retransmit



Thank you ☺

Question and Answer

Some Limitations of using MATLAB

- simplify the modeling of the effects of the radio channel, e.g. constant DCH bit rates, uncorrelated block erasures in the channel decoding of the physical layer
- simplify the effects of the Core networks e.g., delays and packet errors, even though the main impact is caused by the radio link
- Not efficient when running very large campaigns of simulations

Basic types of traffic classes in UMTS

- **Conversational class** -> real-time connection, performed between human users, really low delay, nearly symmetric, e.g., speech
- **Streaming class** -> real-time connection, transferring data as a steady and continuous, low delay, asymmetric, e.g., video
- **Interactive class** -> non-real-time packet data, response requested from other end-user, reasonable round-trip delay, e.g., Web browsing
- **background class** -> non-real-time packet data, no immediate action expected, less sensitive to delivery time, e.g., e-mail

Reference: Harri Holma and Antti Toskala, ”WCDMA for UMTS” second edition

Schemes of Improving Performance of TCP over Wireless Networks

- Link-layer protocols
  - Link-layer retransmission to cope with radio transmission error e.g. RLC ARQ

- End-to-end protocols

  1. Use timestamp in ACK to cope with spurious timeout

  2. Use a novel technique to differentiate the congestion-introduced segment losses from other types of losses at the sender

- Split-connection protocols
  - Split each TCP connection between the sender and the receiver into two separate connections