Optimizing Mobile Communication using a TAPS system

Per Hurtig, Karlstad University

ICCRG Meeting — Prague — 2017-07-17

Outline

• An Ossified Internet
• NEAT — A Transport Services (TAPS) Implementation
• Example: NEAT for Mobile Communication
• Conclusions
An Ossified Internet

• The design of the Internet is a tremendous success
  – scaled from a few users to global use in under 40 years

• In particular, TCP over IP is one of Internet’s greatest success stories
  – permits reliable, congestion controlled data transmission over “any” link

• However, TCP over IP has contributed to ossification
  – works good ⟹ everyone uses it
  – everyone uses it ⟹ infrastructure adapts
  – adapted infrastructure ⟹ no room for innovation
NEAT – A Transport Services (TAPS) Implementation

• NEAT enables the use of transport services rather than protocols
  – i.e., an implementation of TAPS
  – example services may include e.g., reliable transfer, multi-path communication

• NEAT maps application requirements to services
  – if asked for e.g. ”low latency” NEAT will try to create such a service
  – the mapping is transparent to applications

• NEAT tries to fight ossification by
  – providing a more expressive API;
  – using local and remote info to make well-informed decisions;
  – using Happy Eyeballs to ignore the existence of middle-boxes
NEAT – Service Example

{"low_latency": {"value": true}}
NEAT – Service Example

{"low_latency": "{value": true}"}

{ "uid": "low_latency", 
  ...
  "match": { "low_latency": { "value": true }},
  "properties": {
    "transport": { "value": "TCP" },
    "TCP_NODELAY": { "value": false } }
}
Example: NEAT for Mobile Communication

• Multi-path transports can enable mobile communication
  – handover/failover between interfaces
  – transmission over multiple interfaces (load-balancing/resilience)

• MPTCP is an IETF solution for multi-path transport
  – focus on compatibility (to deal with ossification)

• MPTCP is not optimally designed for mobile communication
  – general transport protocol
  – path-management and default settings are not suitable

• Can NEAT build good mobile transport services using MPTCP?
  – Let’s see...
Example: NEAT for Mobile Communication

- What if?
  - Costly to set up?
  - Use bad links?

Client

WLAN

3G/4G

Internet

Server
Experimental Setup

Client
- LTE
  ~20 ms RTT
- WLAN
  ~150 ms RTT

MONROE node
(running Linux with NEAT and MPTCP)

WLAN KAU

3G/4G Operator X

Internet

Server

Regular server
(running Linux with MPTCP)
Experimental Setup

**Experiment:** The client downloads a set of files from the server using TCP, MPTCP, and NEAT.

**TCP:** Only WLAN interface is used.

**MPTCP:** Both WLAN and LTE are used.

**NEAT:** Uses service based on TCP or MPTCP depending on file size and quality of WLAN/LTE.

MONROE node (running Linux with NEAT and MPTCP)

Regular server (running Linux with MPTCP)
Costly to set up?

- Short flows do not gain from multi-path
- Significant gain for long(er) flows
- NEAT is able to select the correct protocol for its transport service
  - How is this done?
NEAT – Policy Manager

properties:
\[ P_X = 100\text{KiB} \]

policy:
if \( P_X \leq 100\text{KiB} \) or \( C_Y < 4\text{G} \): TCP

metadata:
\( C_Y = 4\text{G} \)
Use bad links?

WLAN and LTE

Protocol
- TCP
- MPTCP
- NEAT

Relative download time

Size [KiB]

0.0
0.5
1.0
1.5
2.0
2.5

1 10 100 1000 10000

WLAN and 3G

Size [KiB]

1 10 100 1000 10000

neat
What if?

• The choice of “initial” interface is very important

• In this experiment, WLAN quality was very poor
  – measured in library during exam week

• NEAT used link quality metrics to configure protocol
Conclusions

• NEAT is an implementation of TAPS
  – composes transport services based on application requirements

• NEAT fights ossification by
  – considering application requirements
  – using all available information (both local and remote) to make that happen
  – making sure to get through obnoxious networks

• This presentation exemplified the use of NEAT, in a “mobile” scenario
More material on NEAT

- Library [https://github.com/NEAT-project/neat](https://github.com/NEAT-project/neat)
- Project [https://www.neat-project.org](https://www.neat-project.org)