TCP performance over links with adaptive modulation and fast link retransmissions

Stefan Alfredsson

25 May 2005
Background and purpose

- The Wireless IP project is putting forward a 4G system proposal...
- ... but how will it interact with the existing infrastructure (TCP/IP)?
- Purpose of studies: examine the effect of different phys/link layer design decisions on upper layers, give performance expectations
Agenda

• Emulation overview and setup
• Emulator validation
• Experiments with TCP over fix + adaptive modulation, with varying link ARQ and AM switching levels
• Looking forward
Emulation overview

Sender → Internet → Receiver

GW

Delay queue, throttling, packet drop

WIPEMU

Pred. chan, choose mod.

Real chan, apply errors
## WIPEMU - Validation

| Modulation | Bits/symbol | Bytes/frame | Application throughput | Ideal throughput | App/ideal | TCP/IP header | App+header/ideal |
|------------|-------------|-------------|------------------------|------------------|----------|--------------|----------------|-----------------|
| BPSK       | 1           | 13,5        | 19516                  | 20250            | 0,964    | 0,027        | 0,991          |
| 4-QAM      | 2           | 27          | 38952                  | 0                | 0,962    | 0,027        | 0,989          |
| 8-QAM      | 3           | 40,5        | 57268                  | 60000            | 0,954    | 0,027        | 0,981          |
| 16-QAM     | 4           | 54          | 77552                  | 81000            | 0,957    | 0,027        | 0,984          |
| 32-QAM     | 5           | 67,5        | 95109                  | 100500           | 0,946    | 0,027        | 0,973          |
| 4-QAM      | 1           | 115090      | 121500                 | 0,947            | 0,027    | 0,974        |
| 128-QAM    | 7           | 94,5        | 136605                 | 141000           | 0,969    | 0,027        | 0,996          |
| 256-QAM    | 8           | 108         | 156347                 | 162000           | 0,965    | 0,027        | 0,992          |

*Notes:* Bits/symbol * 108 symbols/frame / 8

*Notes:* Bytes/frame * 1500 frames/s
Emulation parameters

- **Channel parameters**
  - 75 km/h, prediction horizon = 2 ms, pred. error NMSE 0.1 @ 16 dB mean SNR
  - 12 tap Jakes fading model + AR(1) shadow fading with 4 dB var.
  - Original WIP system; 108 data symbols per frame, 1500x25 frames per second (time x freq) (5 Mhz -> 25 x 200 kHz OFDM)
  - Most parameters can be changed easily (channel, time x freq, modulation + switching levels, queue size, …, !num_users, !scheduling)

- **Single user, one channel**
  - \[ C = 1500 \times 1 \text{ frame/s} \times 108 \times \log_2(\text{modulation\_order}) / 8 \text{ byte/s} \]
Fix modulation (at ~ 16 dB)
Obtaining better performance

• Use adaptive modulation
  – Optimize for max throughput
  – Optimize for target BER
  – Uncoded / coded system
Adaptive modulation, non-coded

Transmission of 3 Mb bulk data

- TCP, AM, PerfPred, Levels for NMSE 0.0
- TCP, AM, PredErr, Levels for NMSE 0.1
- TCP, AM, PredErr, Levels for TBER $10^{-3}$
- TCP, AM, PredErr, Levels for TBER $10^{-4}$
- TCP, AM, PredErr, Levels for TBER $10^{-5}$
Adaptive modulation, coded
(not reliable results!)

Transmission of 3 Mb bulk data

Maximum allowed number of link layer retransmissions

Throughput (Bps)
Modulation level distribution

<table>
<thead>
<tr>
<th>Modulation Level</th>
<th>BPSK</th>
<th>4-QAM</th>
<th>8-QAM</th>
<th>16-QAM</th>
<th>32-QAM</th>
<th>64-QAM</th>
<th>128-QAM</th>
<th>256-QAM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:
- **NMSE0.1**
- **TBER3UC**
- **TBER4UC**
- **TBER5UC**
- **TBER5C**
Analysis / Discussion

• Target BER assumes cut-off level, or use modulation+coding that preserves BER constraint below BPSK limit ( < about 11-14 dB in this case)

• When below cut-off, another user will instead be scheduled for that time-frequency bin

• No suitable user -> transmit anyway?

• Is end-user throughput a good metric? (may be unfair comparison as seen earlier; a cut-off would produce even lower throughput!)
Looking forward

• WIPEMU – Work In Progress Emulator
  – Adjusting BER calculation to account for coding gains
  – Channel scheduling
    • For single and multi-user
  – Going multi-user to really see effects of Target BER constraints
Thanks for your attention

Questions, comments?