Improving the Performance of Reliable Transport Protocols in Mobile Computing Environments

Ramon Caceres (AT &T Bell Labs)

Liviu Iftode (Priceton University)

Presented by: Yuvraj Agarwal

Introduction

- Context
  - Microcellular networks
  - Mobile computers
- Motivation
  - Current protocols tuned for wired networks
  - Packet losses assumed to signal congestion
  - Protocols slow transmissions in response to losses
  - But wireless networks exhibit other types of losses

- What happens to active connections when hosts move ?

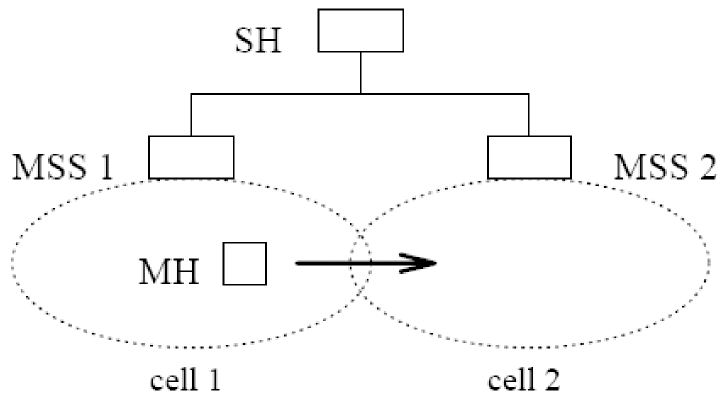
Summary

- Carried out a measurement study
  - TCP
  - Mobile IP
  - Wave LAN radios
- Found that motion degrades performance
  - Unacceptable interactive response
  - Significant drops in throughput
- Identified causes of performance degradation
  - Long waits for retransmission timeouts
  - Slow growth of transmission windows
- Devised an effective solution
  - Regains lost performance
  - Works over an internetwork

Outline

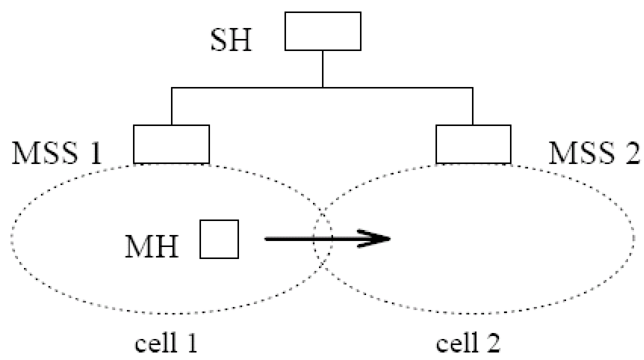
- Wireless networking test bed
  - Hardware and software components
  - Cellular handoff procedures
  - Experimental methodology
- Effects of motion
  - Loss of throughput
  - Pauses in communication
  - Packet losses
- Alleviating the effects of motion
  - Fast retransmissions
  - Improvements in latency
  - Improvements in throughput

Wireless Networking Testbed



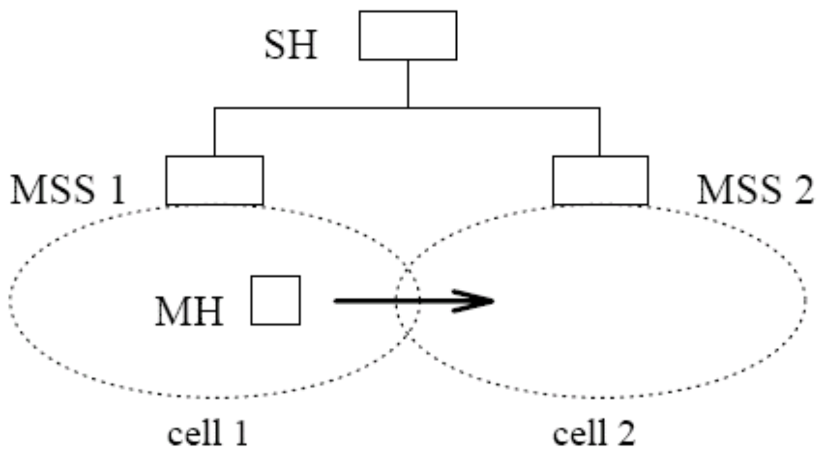
- Hardware
  - 2-Mbps WaveLAN radio network
  - 10-Mbps Ethernet
  - 50Mhz- 486 MHz PCs
- Software
  - Mach 3.0 kernel with Unix server
  - Columbia University’s Mobile IP
  - 4.3-BSD Tahoe TCP

Cellular Handoff Procedures



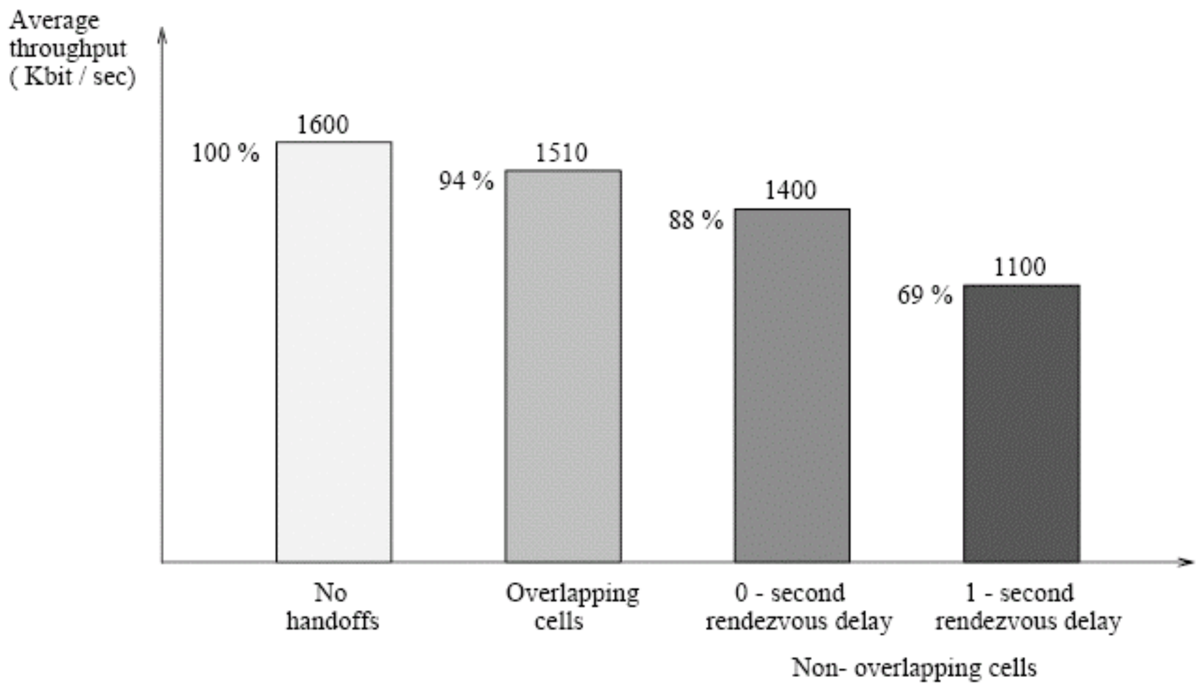
- Mobility Support Stations (MSSs) route packets to/from MHs
- MSSs broadcast periodic beacons
- MHs decide to switch based on signal strength / missing beacons
- Mobile IP handoff protocol
  - MH greets new MSS
  - MH changes its routing tables
  - New MSS changes its routing tables
  - MH informs new MSS of old MSS
  - New MSS notifies old MSS of handoff
  - Old MSS changes its routing tables
  - New MSS acknowledges handoff to MH

Experimental Setup



- Methodology
  - Establish TCP connection between SH and MH
  - Instruct MH to periodically change cells
  - Measure TCP performance
- Parameters
  - Overlapping vs non-overlapping cells
  - Rendezvous delay

Loss in Throughput…

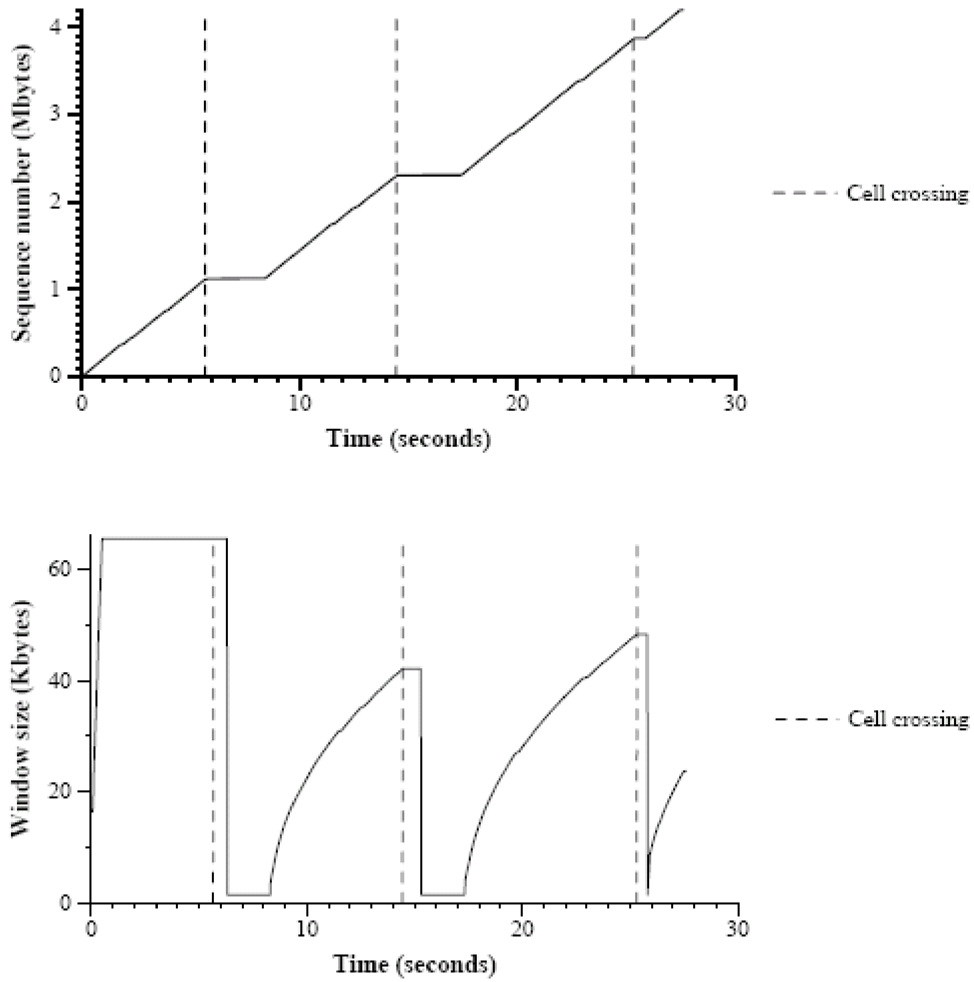


- Parameters
  - 4 Megabyte bulk data transfer
  - 1 second beaconing interval
  - Handoffs occur every 8 beacons (8s)

Congestion Avoidance and Control

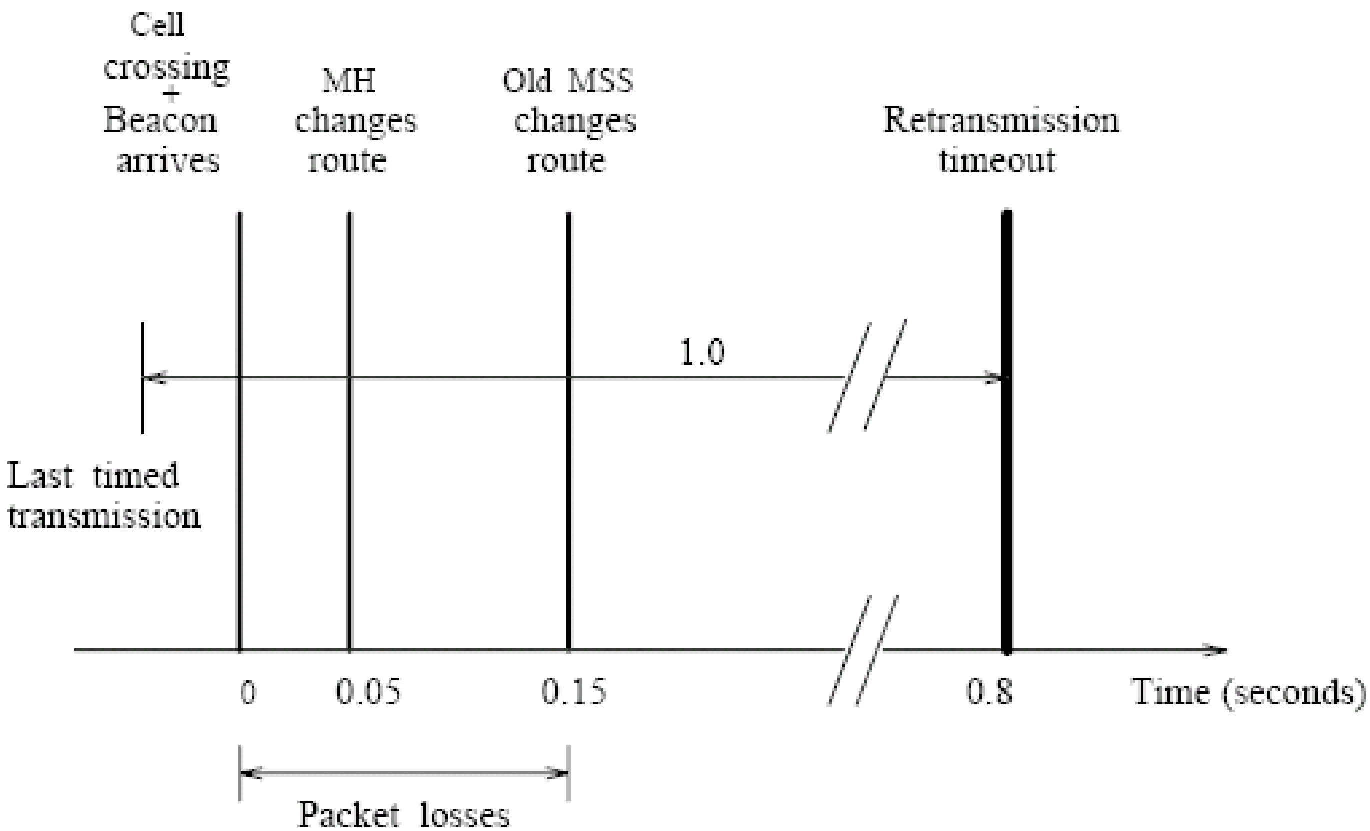
- Delay estimators
  - Maintain average of roundtrip delay = Davg
  - Maintain average of variation in delay = Vavg
- Exponential retransmission backoffs
  - Set retransmission timer Tout = Davg + 2* Vavg
  - Assume timeout signals packet loss
  - Retransmit lost packet after timeout
  - Double timeout with every consecutive timeout
- Slowstart algorithm
  - Drop window size to 1 on every timeout
  - Double window with every ACK until threshold
  - Add 1 to window with every ACK after threshold
  - Set threshold to 1/2 of window size at time of loss

Pauses in Communication…



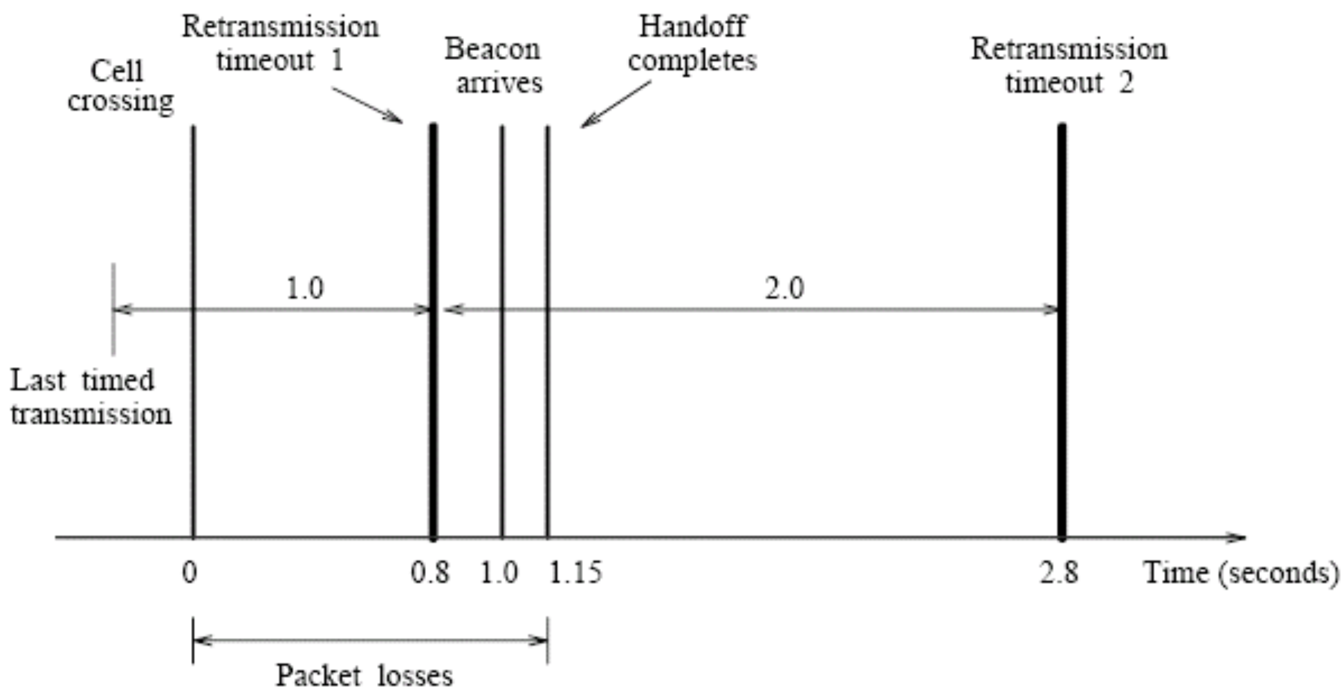
- Cell crossings with a 1 second rendezvous delay

Packet loss during handoff



- Cell crossing with 0 rendezvous delay

Packet loss during handoff



- Cell crossing with 1 rendezvous delay

Problems caused by motion/mobility

- Packet losses
  - Hosts leave cells without warning (?)
  - Beaconing period limits how soon handoff begins
  - Two packet exchanges necessary to change routes
  - Routing inconsistencies cause futile transmissions
  - Up to a full window of data and ACKs are lost
- Transport level reacts badly to motion
  - Backoffs freeze TCP for >= 0.8 seconds
  - Slow-start throttles TCP for another >= 1 second
- How do we fix these problems
  - Hide motion from transport level ?
  - Adapt transport level to react better to motion ?

Hide Motion?

- Should strive for smooth handoffs
  - Make before break handoff protocols
  - More frequent or continuous beacons
  - Use signal strength to anticipate cell crossings
  - Buffer data in MSS while handoff completes
- But microcellular networks make it difficult
  - Many small cells
  - Little or no overlap between cells
- Microcellular networks are desirable for three reasons
  - Efficient reuse of electromagnetic spectrum
  - Low power consumption by mobile transceivers
  - Accurate location information

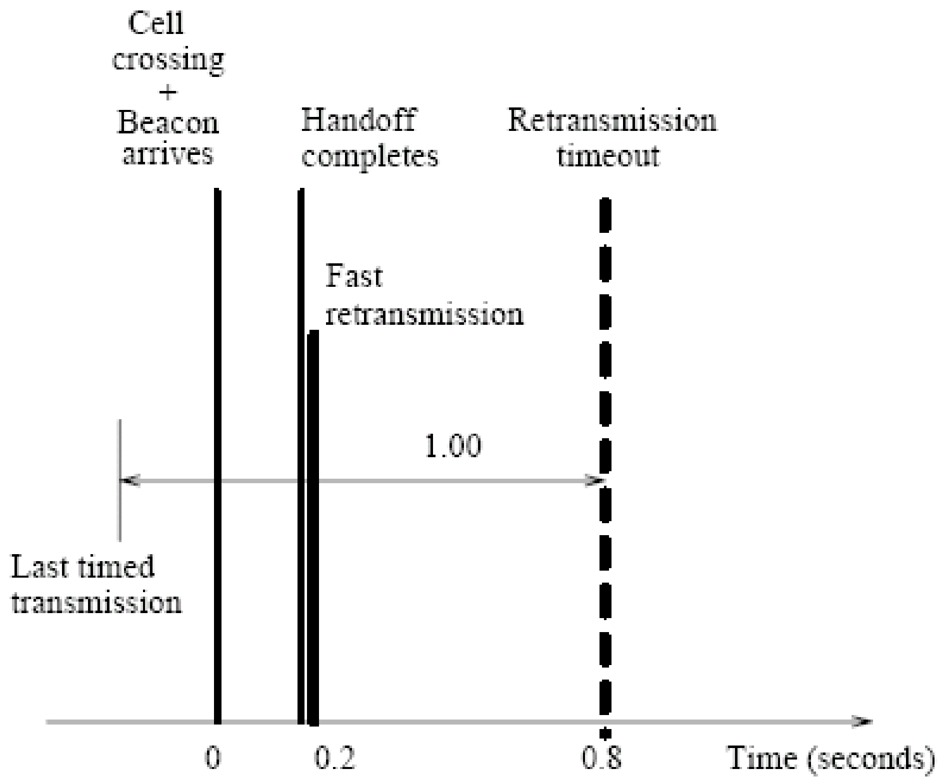
Fix Re-transmission Timers ?

- Long pauses are due partly to long timeout values
  - TCP implementations use coarse timer
  - Minimum timeout values are 0.6 to 1 seconds
- Actual roundtrip times are much smaller
- But more accurate timers may not help
  - Multiple timeouts during handoffs
  - Multiple retransmissions before routes are consistent
  - Retransmission backoff grows exponentially causing more delay
  - Slow-start threshold decays exponentially
  - Other spurious retransmissions

An end-to-end Approach

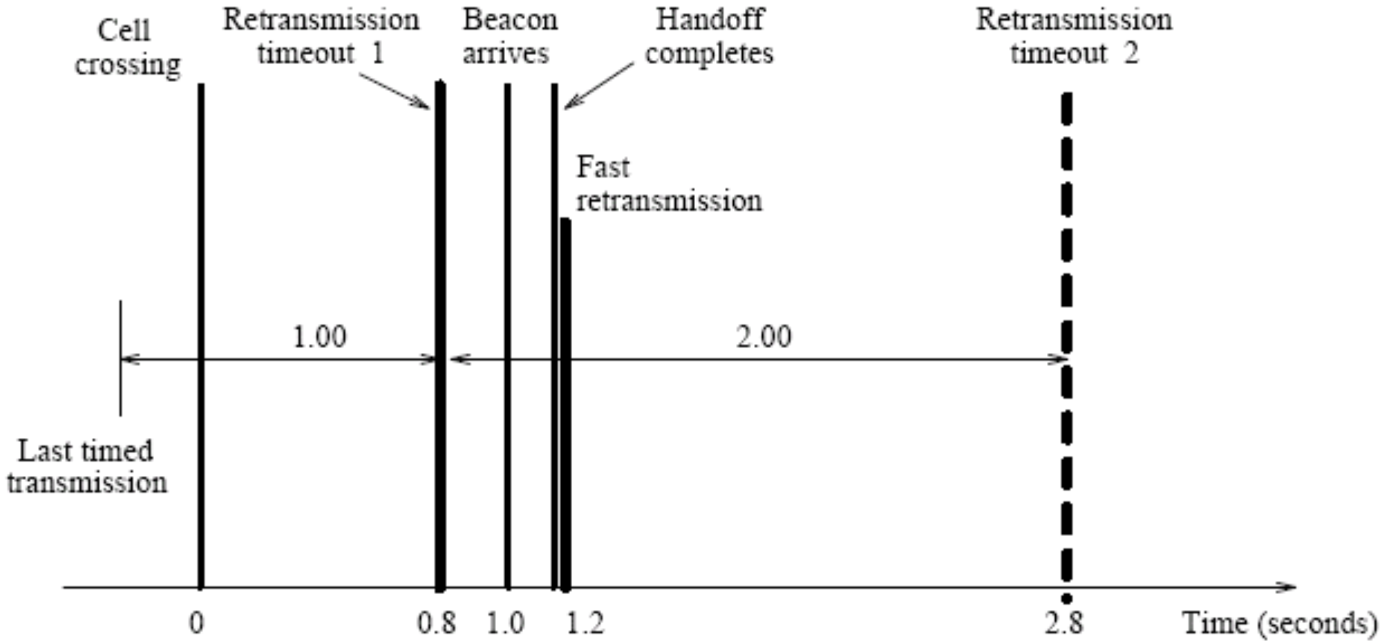
- Resume transport-level communication after handoffs
  - Signal transport level when routes settle
  - Invoke fast retransmission procedure
- Fast retransmission procedure
  - Retransmit earliest unacknowledged packet
  - Drop transmission window
  - Initiate slow start
- Advantages
  - Avoids waiting for retransmission timeout
  - Avoids congesting the new cell
  - Requires minimal changes to end systems
  - Requires no special support from network
  - Preserves end-to-end semantics

Fast Retransmission Approach



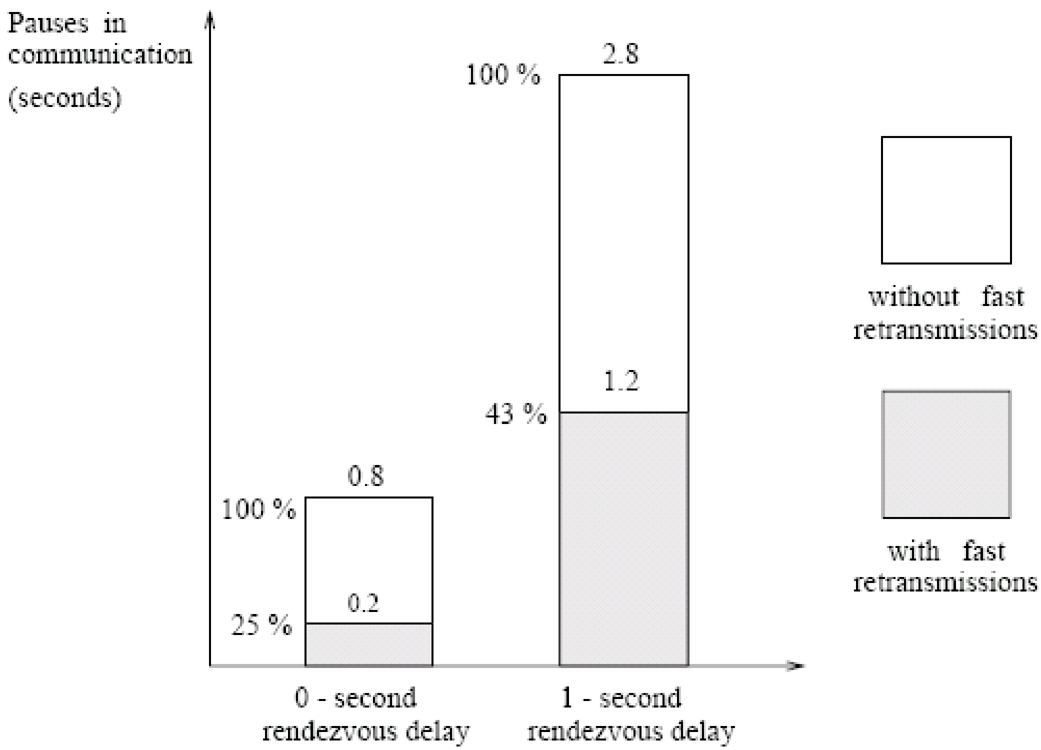
Cell crossing with a 0-second rendezvous delay

Fast Retransmission Approach (cont)



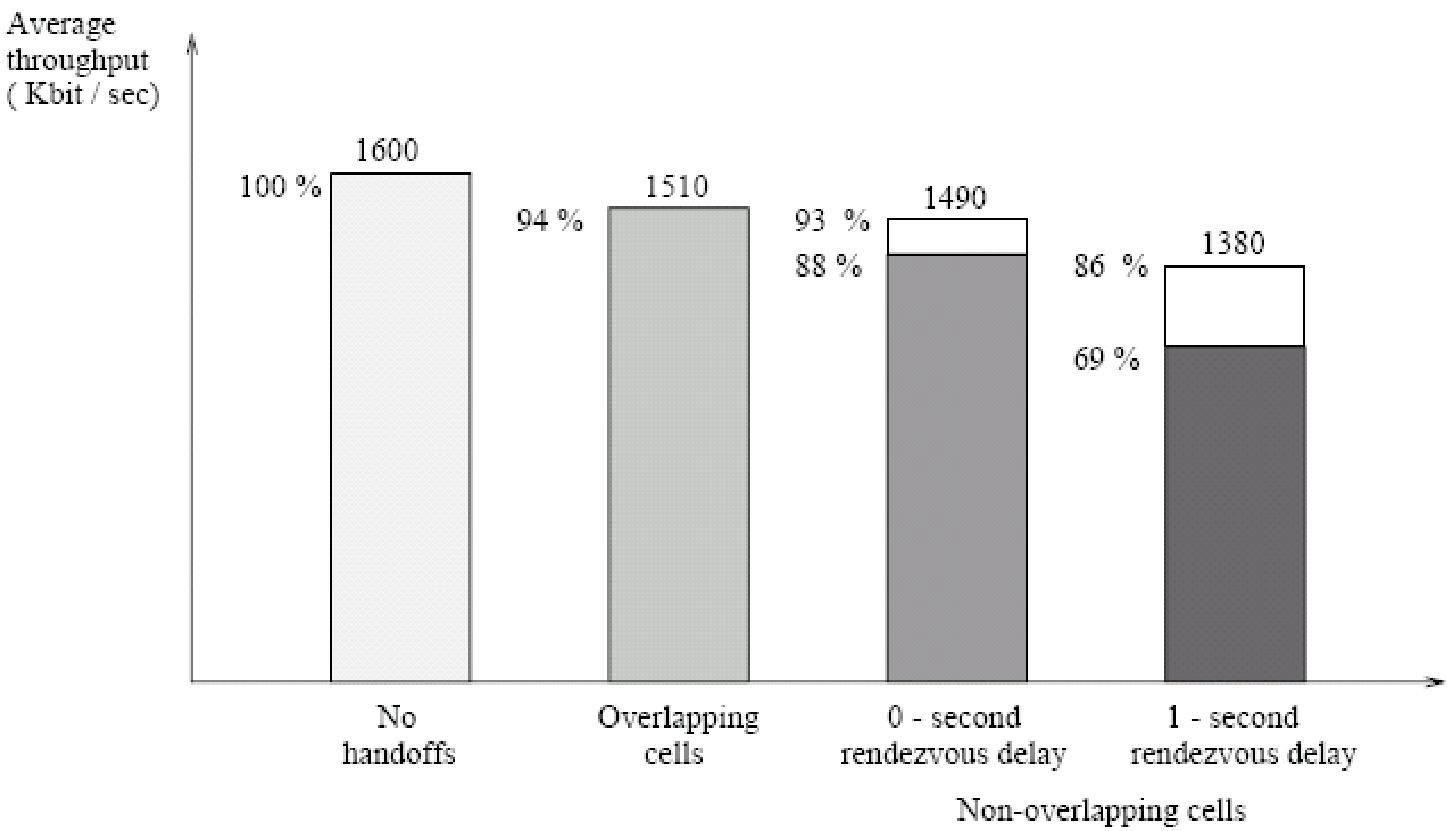
Cell crossing with a 1-second rendezvous delay

Improvements in Latency



Transmitter on the MH

Improvements in Throughput



Transmitter on the MH

Conclusion

- Measured the effects of motion
  - Unacceptable interactive response
  - Significant throughput degradation
- Identified the underlying causes
  - Cell crossings cause packet losses
  - Backoffs freeze communication for >= 0.8 seconds
- Proposed an effective end-to-end solution
  - Uses existing fast retransmissions
  - Reduces latency to acceptable levels (200ms – 300ms)
  - Significantly improves throughput
  - Preserves end-to-end semantics
  - Avoids network congestion
  - Works over the Internet

Discussion…

- “Micro-Cellular” networks as motivation (one handoff every 8s)…
- Are the long delays a function of poor implementation ?
  - Sec C. MH leaves cell without warning ?
  - Sec C. “The MH doesn’t know it has moved, thus does not update routes until new beacon arrives” ?
  - Sec D. “Mobile-IP is user level code” ?
- Beacons – 1s beacon interval pretty high (100ms for 802.11)
  - Would have been good to see intermediate results for other periods

- Would the performance be better if link-layer retransmits are used ?
  - 802.11 uses link-layer re-transmits (upto 8?)
  - Not really, as link-level retransmits are not effective for mobility losses
- 802.11 improvements to have better handoff
  - Proactive/passive scanning for APs, 100ms beacon intervals
  - Link-layer ACKs, Inter Access Point Protocol (IAPP)
  - Explicit Re-association packets on performing handoff, gratuitous ARPs

Wireless Transmission errors

- Wireless links exhibit high bit error rates
  - Ambient noise multipath interference
  - 10 ^ (-6) for radio/infrared vs 10 ^ (-12) for optical ber
- These errors also cause transport level problems
- But link-level retransmissions may not help
  - Competing retransmission timers
  - Transport level may time out first
- Other possible solutions
  - Selective retransmissions
  - Forward error correction