A Trace-based Approach to Modeling Wireless Channels

Giao Nguyen

University of California at Berkeley

BARWAN Retreat, January 1997

Motivation

- Need to reproduce network conditions for comparative evaluation
  - wireless channel quality varies unpredictably
  - impact at the packet level not fully understood
- Modeling wireless channels
  - propagation model: computationally expensive
  - simple packet error model: inaccurate

Goal: technique to realistically model wireless channels

Motivation

- Wireless channel traces & models benefit
  - Evaluations of link-level reliable: ARQ, FEC
  - Handoff, header compression schemes
  - Transport: SNOOP-TCP, I-TCP

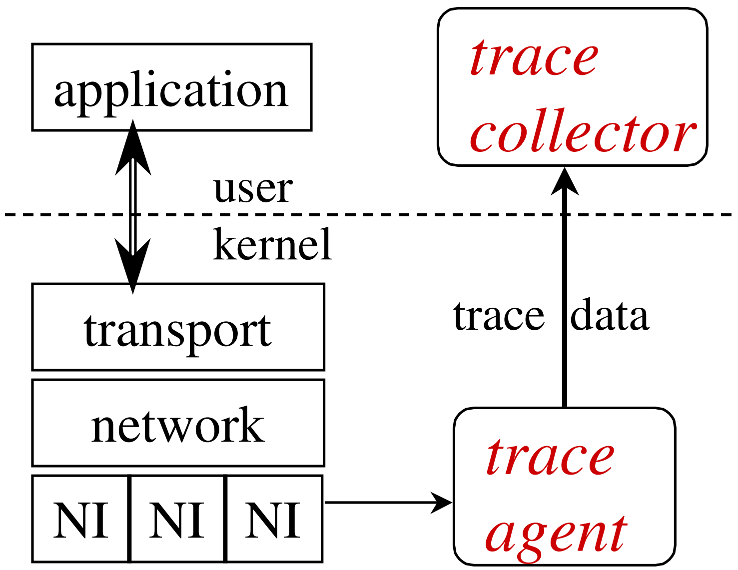
Outline

- Methodology
- Experimental Results
- Modeling
- Conclusions
- Future Work

Methodology

- Three phases
  - Trace Collection
  - Analysis and Modeling
  - Validation

Trace Collection



NI: network interface

- *trace agent*
  - in kernel (BSD/OS)
  - records packet and device-level information
    - e.g., sequence number, signal strength
- *trace collector*
  - user-level process
  - extracts data from trace agent buffer

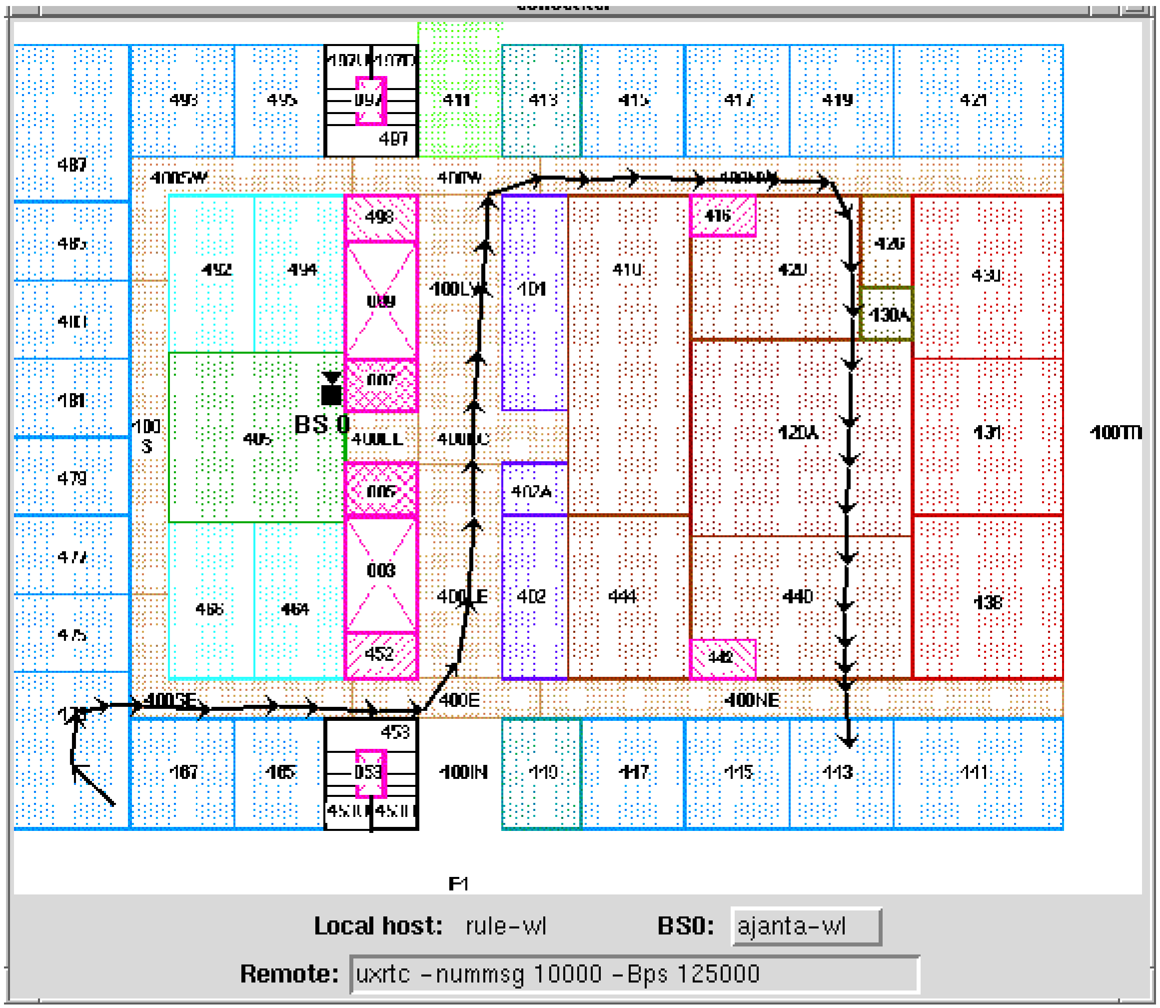
Experimental Conditions

- Workload
  - constant-rate UDP streams
  - packet size: 1400 bytes
  - data rate: 1.3-1.5 Mbps downstream and 0.9-1.1 Mbps upstream
- Physical Location
  - in-building: Soda Hall, UC Berkeley
  - stationary MH at different locations
  - moving MH

WaveLAN

- 2 Mbps radio, 902-928 Mhz ISM band
- No link-level error or loss recovery
- MAC: CSMA/CA
- Asymmetric bandwidth
  - 1.6 Mbps downlink
  - 1.2 Mbps uplink
- Signal strength information
  - signal level, noise level, and signal quality

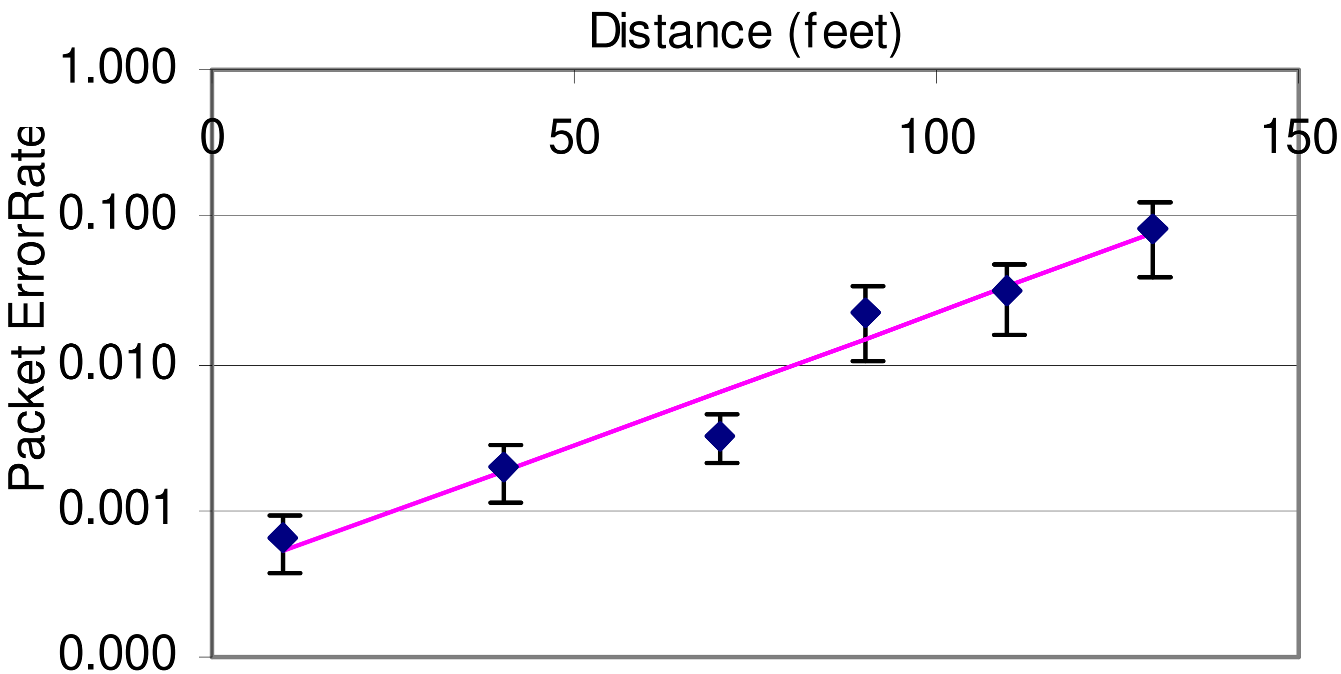
Layout of building



Analysis and Modeling

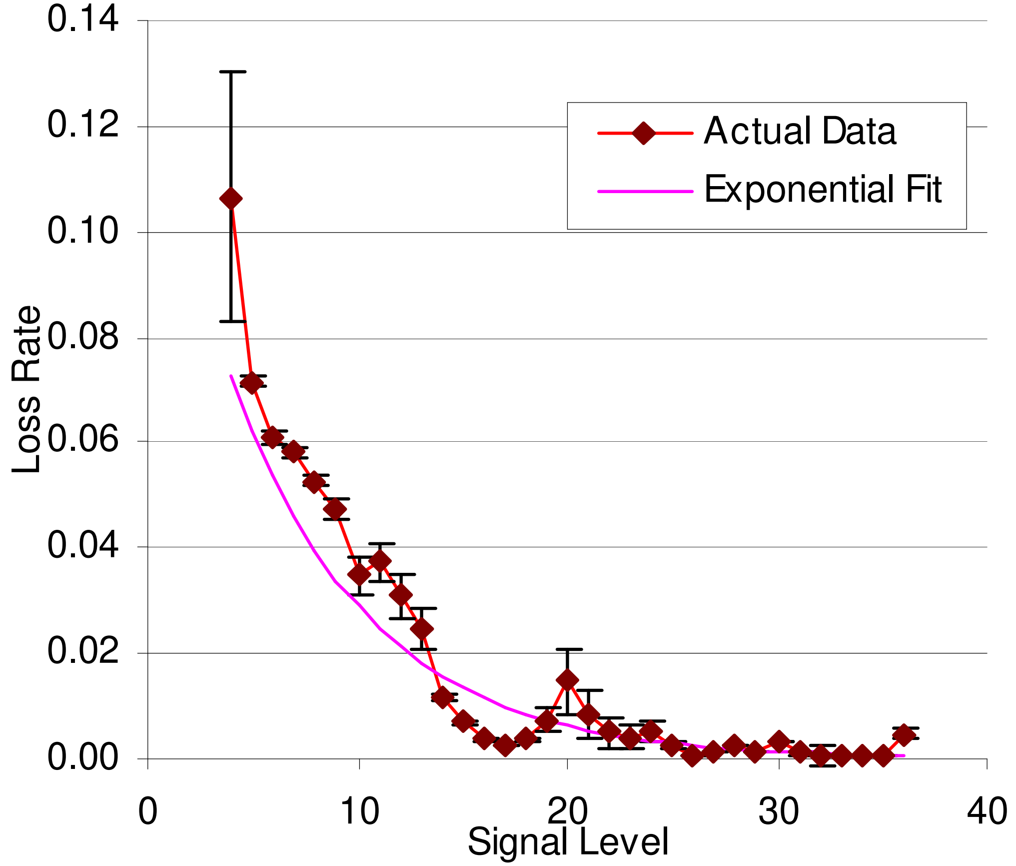
- Analysis
  - extract packet-level error information
  - analyze correlation of errors with:
    - distance, signal level, data rate, packet size
- Modeling
  - create empirical distribution of errors
  - derive mathematical expressions to fit the distribution

Packet Error vs. Distance

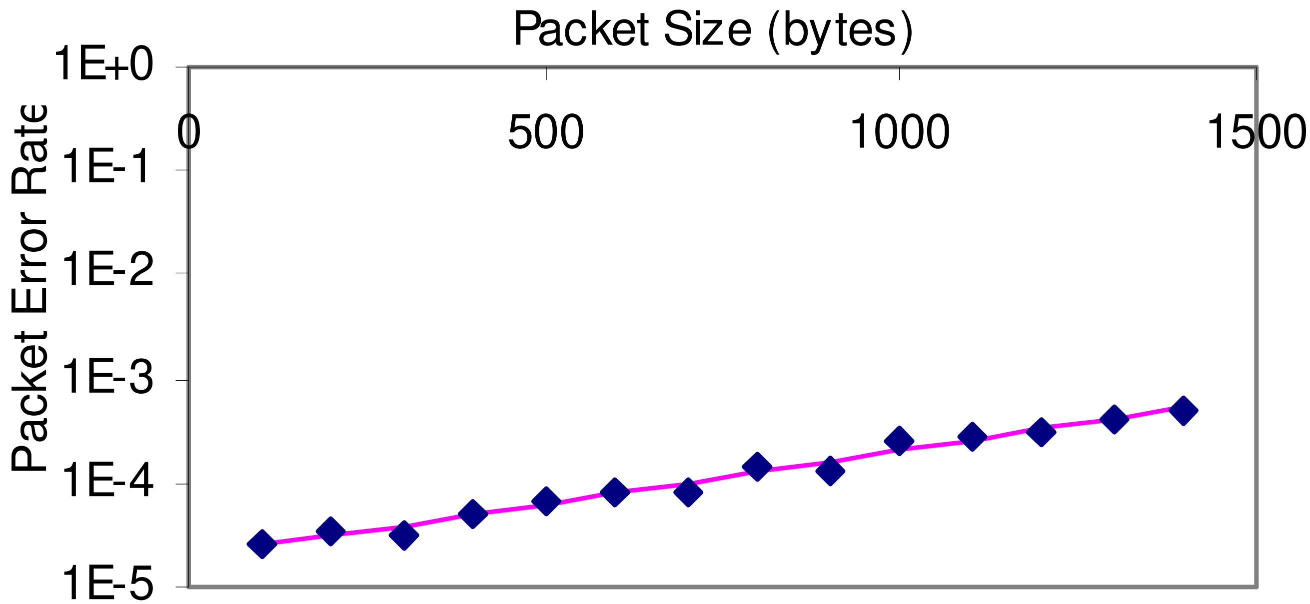


- Packet error rate doubles for every increment of ~20 ft
- Average packet error rate: 2.0%

Packet Error vs. Signal Level



Packet Error vs. Packet Size

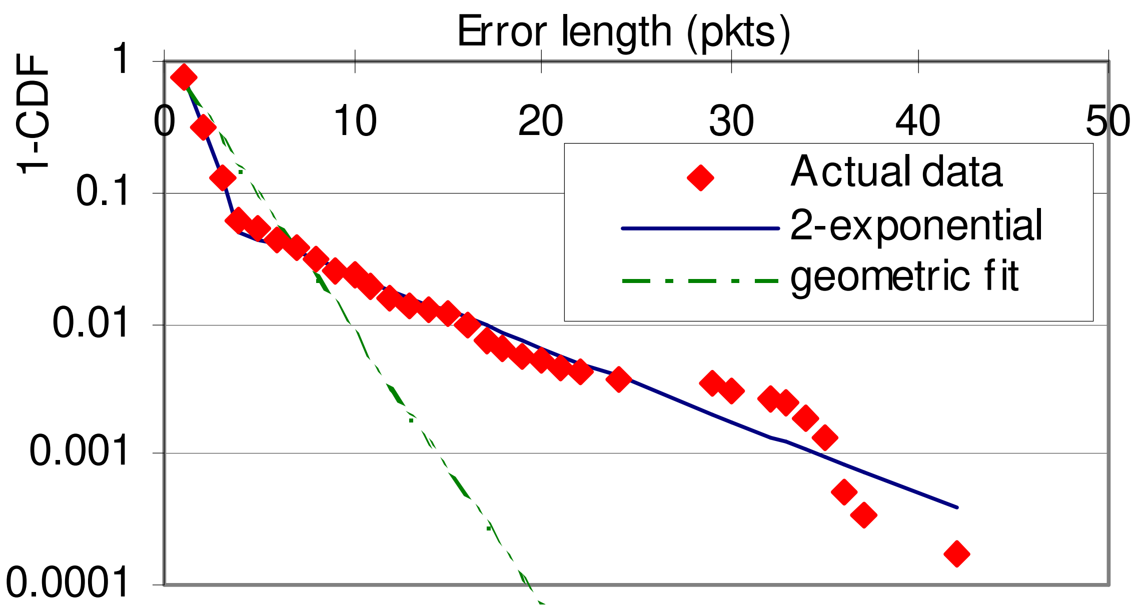


- Packet error rate doubles for every 300-byte increment in packet size

Modeling

- 2-state Markov model
  - states correspond to error- and error-free periods
  - duration of each state is geometrically distributed
- Our empirical model
  - error- and error-free distributions
  - piece-wise distribution
    - breaks down distributions into small number of segments and fit each with different expressions

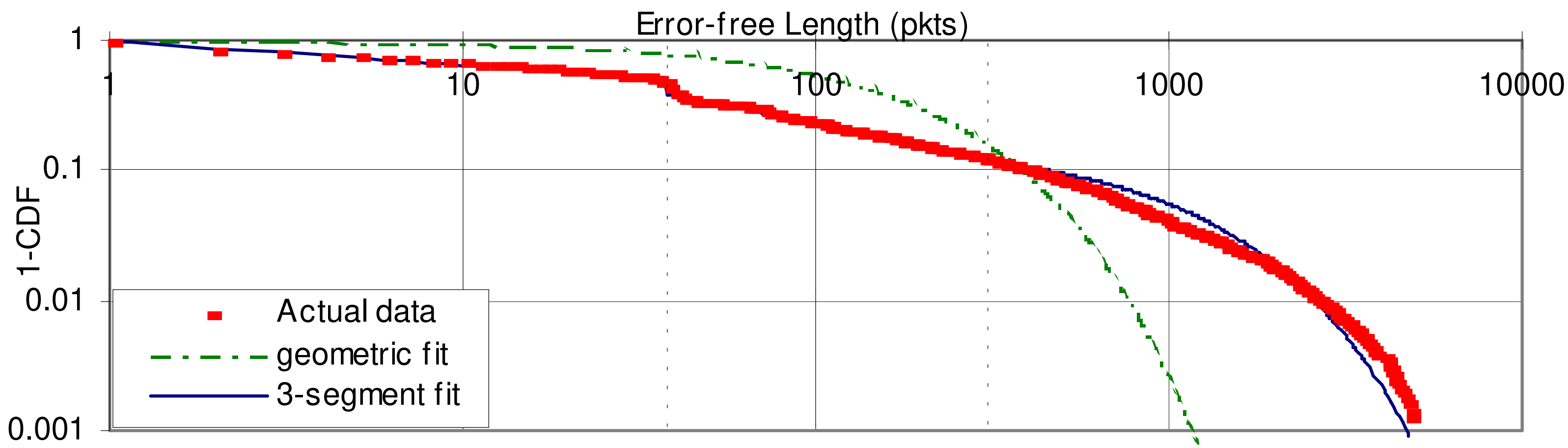
Modeling: Error-length



error-length: consecutive packets lost in a burst

- 90% of all error bursts is between 1-3 packets
- a single expression does not fit the entire distribution well
- piece-wise fit: two exponential curves

Modeling: Error-free length



Error-free length: consecutive packets received without errors

- 3 distinct regions
- shorter error-free bursts are not exponentially distributed
  - error periods are correlated

Validation

- Simulate packet errors in network simulator *ns*
  - 3 models: uniform, 2-state Markov, and empirical
  - replay errors from traces
- Compare simulated TCP performance
  - base case: simulations driven from error traces

Simulation Results

	Throughput (Mbps)	relative difference
Uniform	1.5134	0.213429
Markov	1.4054	0.126841
Improved	1.3071	0.047995
Trace	1.2472	0.000000

- simpler models overestimate TCP throughput
  - underestimate the burstiness of errors
  - TCP performance is sensitive to bursty errors

Conclusions

- Trace-based approach
  - can be used for wide range of networks
  - useful for evaluating network protocols and applications
- Modeling WaveLAN channel
  - Uniform and 2-state Markov do not model the observed burstiness well
  - piece-wise empirical model is more accurate and still reasonably simple to implement

Future Work

- Realistically *emulate* wireless networks using traces
- Evaluate mobile systems
  - transport protocols: TCP, HTTP, etc.
  - file systems, web proxies, etc.
- Investigate the interactions of wireless systems operating in the same frequency band
  - e.g., Metricom and WaveLAN