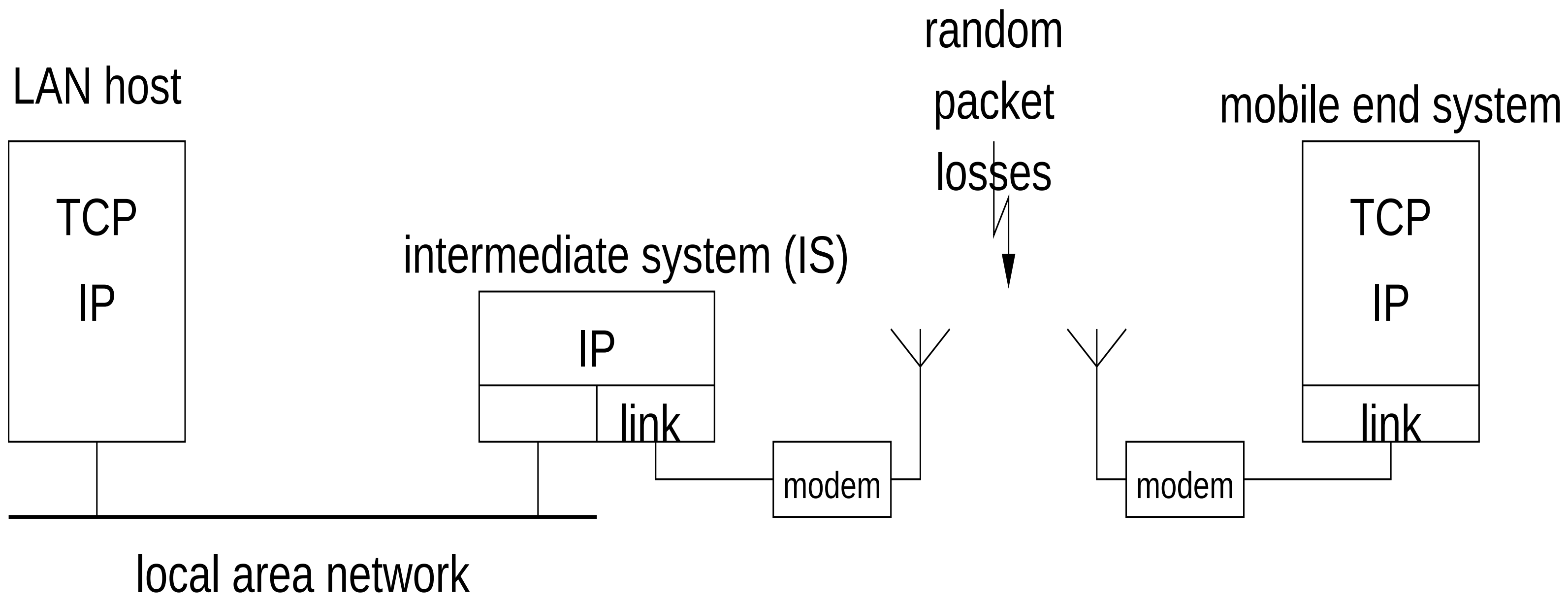
COMPARATIVE PERFORMANCE ANALYSIS OF VERSIONS OF TCP IN A LOCAL NETWORK WITH A WIRELESS MOBILE LINK

Anurag Kumar
Visiting Research Scholar, WINLAB, Rutgers Univ.
Permanent Affiliation: ECE Dept., Indian Institute of Science
Bangalore, INDIA

NETWORK SCENARIO

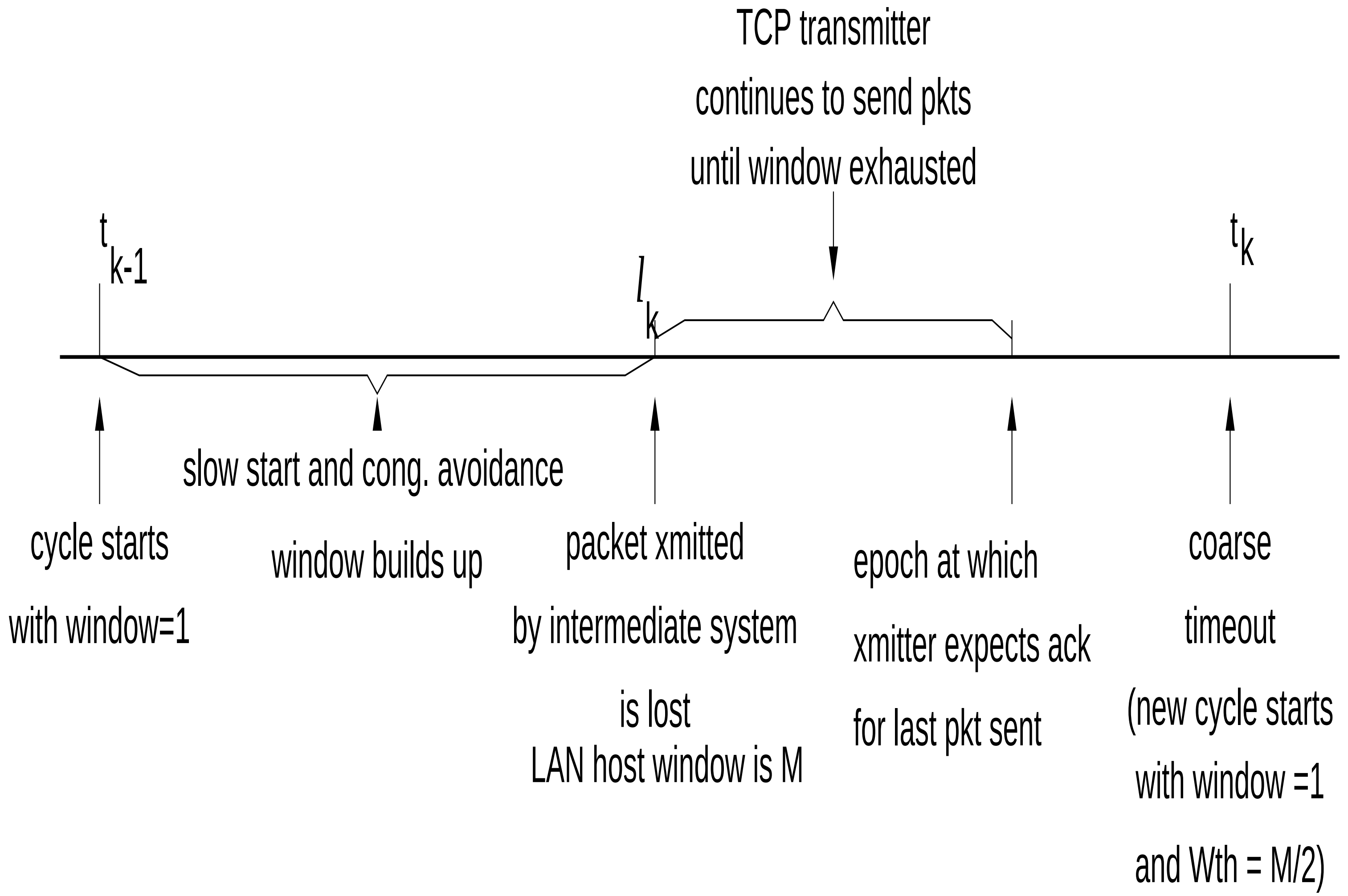


- Bulk data transfer from LAN host to mobile end system
- IS: router, integrated with base-station

WINDOW ADAPTATION: ORIGINAL ALGO.

- At transmitter, at time t
  - $W(t)$ = transmitter's congestion window
  - $W_{th}(t)$ = *slow-start* threshold
- When first ack for a sequence number received at t
  - if $W(t) < W_{th}(t)$, then $W(t^+) = W(t) + 1$
    * window doubles for each window-full of packets put through
    * *slow-start* phase; exponential window increase
  - if $W(t) \geq W_{th}(t)$, then $W(t^+) = W(t) + \frac{1}{W(t)}$
    * window increases by 1 for each window-full of packets put through
    * *congestion avoidance* phase; slow window increase
- If timeout occurs at t, $W(t^+) = 1$, $W_{th}(t^+) = \left\lceil \frac{W(t)}{2} \right\rceil$
- We call this original (Jacobson 1988) protocol **OldTahoe**
  * packet loss is always followed by a coarse timeout

OLDTAHOE TRANSMISSION CYCLE



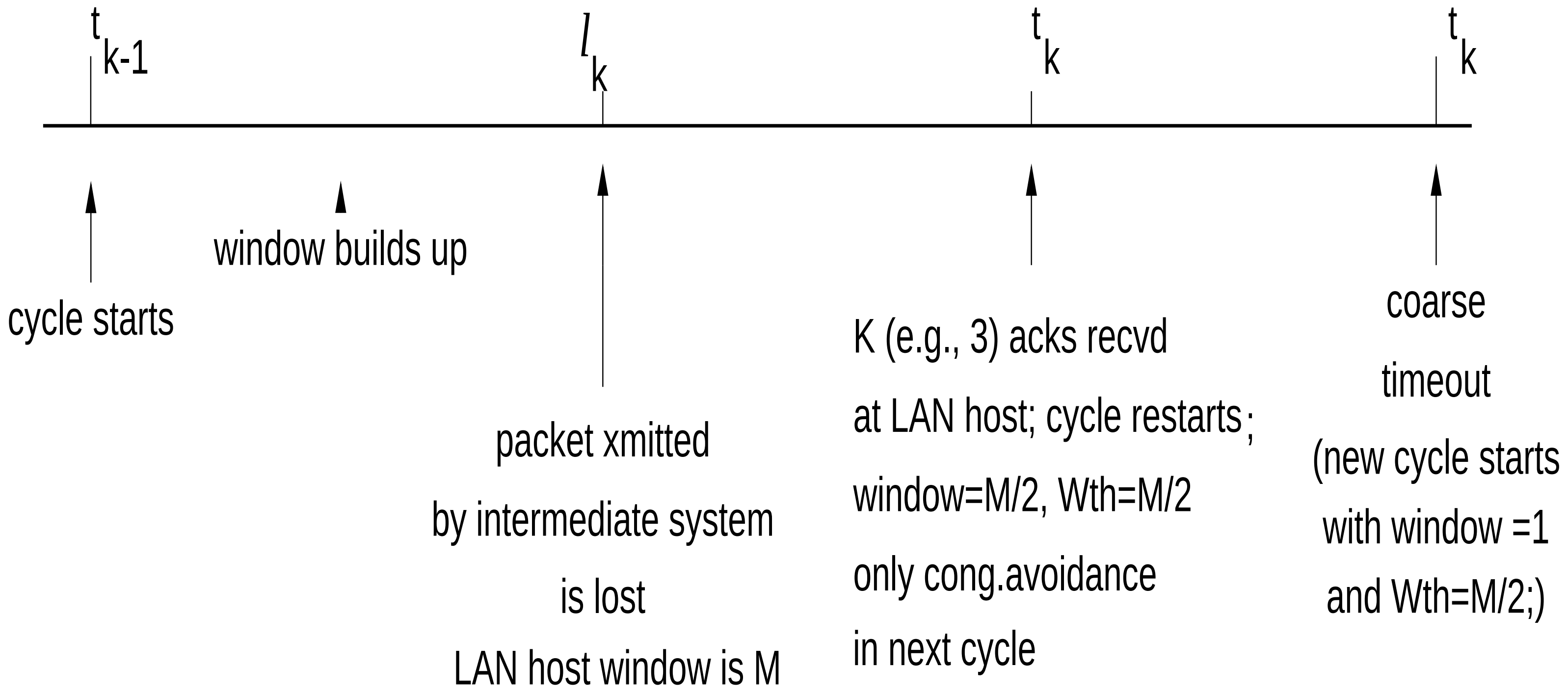
FAST-RETRANSMIT AND FAST-RECOVERY

- Fast-retransmit, after a packet loss
  - receiver continues to get subsequent packets
    - these are packets already in the pipe
    - or packets subsequently transmitted
  - rcvr generates duplicate acks, all indicating first lost pkt
  - receipt of multiple dup acks $\Rightarrow$ noncongestion loss
    - dup ack threshold K, e.g., =3
  - retransmissions begun; $W_{th}(t^+) = \left\lceil \frac{W(t)}{2} \right\rceil$
- Fast-recovery (following a fast-retransmit)
  - do not drop congestion window to 1,
  - but set it to $W_{th}(t^+)(= \left\lceil \frac{W(t)}{2} \right\rceil)$

NEWER VERSIONS OF TCP

- **Tahoe**: implemented in some operating systems (OSs)
  - does fast-retransmit
  - but, after a loss, window always dropped to 1
- **Reno**: implemented in some OSs
  - does fast-retransmit, but is conservative
  - resends only the first missing packet
  - expects additional dup acks to send more packets
  - if recovers all lost pkts, window is not dropped to 1
- **NewReno**: recent proposal; not available yet in standard OSs
  - does fast-(optimistic)-retransmission
    - window is not dropped to 1

TCP CYCLE WITH FAST RETRANSMIT



RELATED PREVIOUS ANALYTICAL WORK

- Mishra, Sanghi, Tripathi (1993)
  - identify cyclic behaviour of OldTahoe
  - loss window Markov chain analysed
  - no study of fast recovery protocols
- Lakshman, Madhow (1994)
  - study multiple TCP connections in a bottleneck link
  - wide area network model
  - cyclic evolution model used; approximate analysis
  - no study of fast recovery and coarse timeouts

CONTRIBUTIONS OF OUR WORK

- Developed an analytical model for TCP over a local network
- Several versions of TCP can be analysed
  - details of basic TCP congestion control
  - fast-retransmit, fast-recovery
  - coarse timers, minimum timeout
- Fading model for wireless mobile link
  - effect of signal-to-noise ratio
  - effect of vehicle speed
  - effect of link protocol

ANALYSIS APPROACH

- $\{U_k, k \geq 1\}$: sequence of congestion windows at loss instants
  - Markov chain
- Markov renewal reward process
  - number of good packets is the reward in each cycle
- For $ver \in$ {OldTahoe, Tahoe, Reno, NewReno }
  - C_{ver} := mean cycle time for a TCP version
  - U_{ver} := stationary value of process $\{U_k\}$
- Throughput T_{ver} is given by (p: packet loss probability)

$$T_{ver} = \frac{\text{mean reward per cycle}}{\text{mean cycle length}} = \frac{\frac{1-p}{p} + (1-p)(EU_{ver} - 1)}{C_{ver}}$$

- Numerator is a slight overestimate, since some good packets may be resent

MARKOV CHAIN $\{U_k\}$

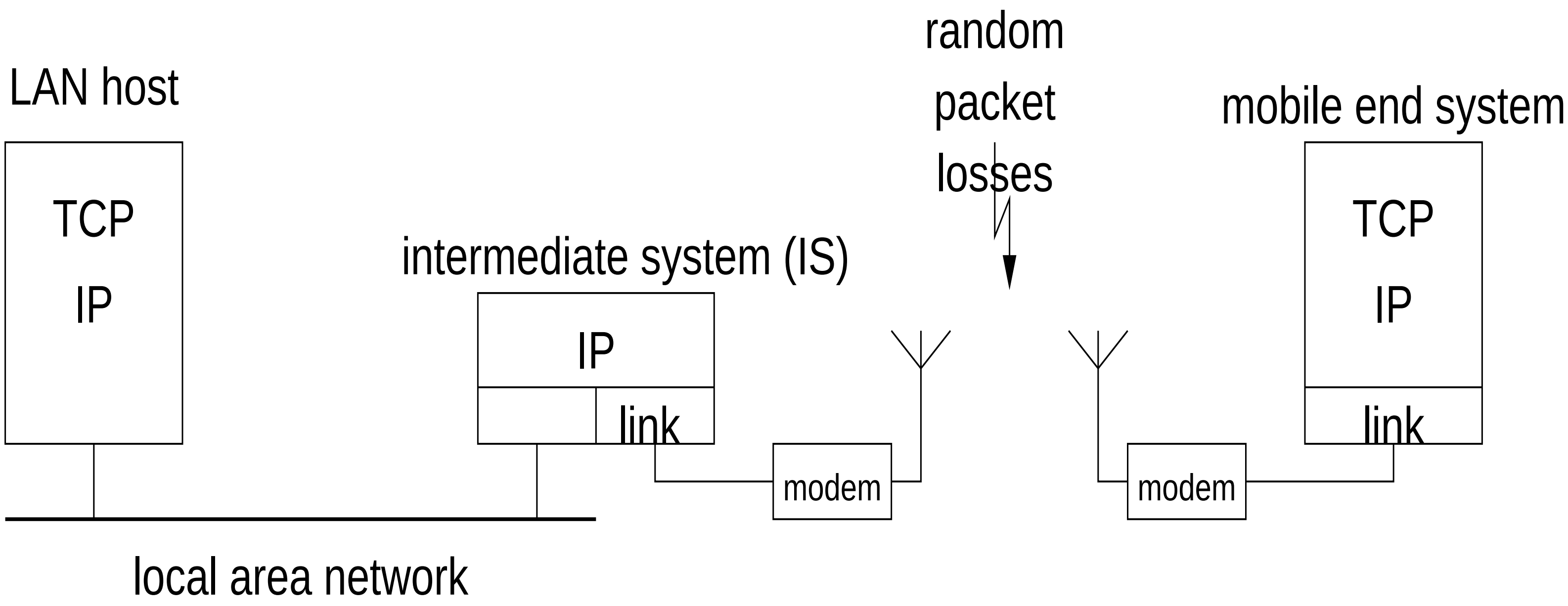
- If loss occurs at window $M, 1 \leq M \leq W_{max}$, let
  - $f_M :=$ probability of fast recovery
  - depends on TCP version
- Transition probabilities for $\{U_k\}$ (Bernoulli loss model)
  - given that $U_k = M$, and $m := \lceil \frac{M}{2} \rceil$
  - $U_{k+1} =$

$$\begin{cases} j & \text{w.p. } \overline{f}_M \overline{p}^{j-1} p & , 1 \leq j \leq m-1 \\ m+k & \text{w.p. } \overline{f}_M \overline{p}^{d(k)}(1-\overline{p}^{m+k}) + f_M \overline{p}^{h(k)}(1-\overline{p}^{m+k}) & , 0 \leq k \leq W_{max}-1-m \\ W_{max} & \text{w.p. } \overline{f}_M \overline{p}^{d(W_{max}-m)} + f_M \overline{p}^{h(W_{max}-m)} & \end{cases}$$

where

$$d(k) = (k+1)((m-1) + \frac{k}{2}) \quad \text{and} \quad h(k) = d(k) - (m-1)$$

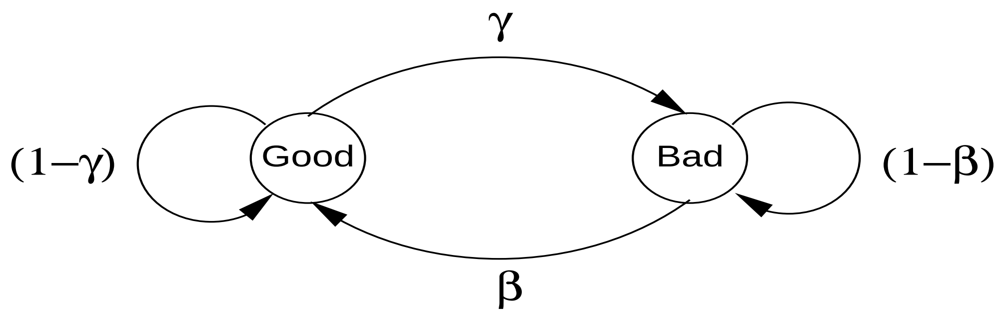
LINK PROTOCOLS



- TCP packets fragmented into smaller link packets
  * e.g., TCP packet = 1536 bytes = 12 $\times$ 128 byte link pkts
- In case of error only smaller packets need to be resent
- Each link packet attempted N times (e.g., 3)

LOSS MODELS FOR WIRELESS LINK

- Without fading: independent TCP packet loss (probability p)
- Rayleigh fading: Markov loss model



  - model transitions:
    * Low speeds (slow fading): TCP packet boundaries
    * High speeds (rapid fading): link packet boudaries
- Markov parameters from Rayleigh model, speed, Eb/No, etc.

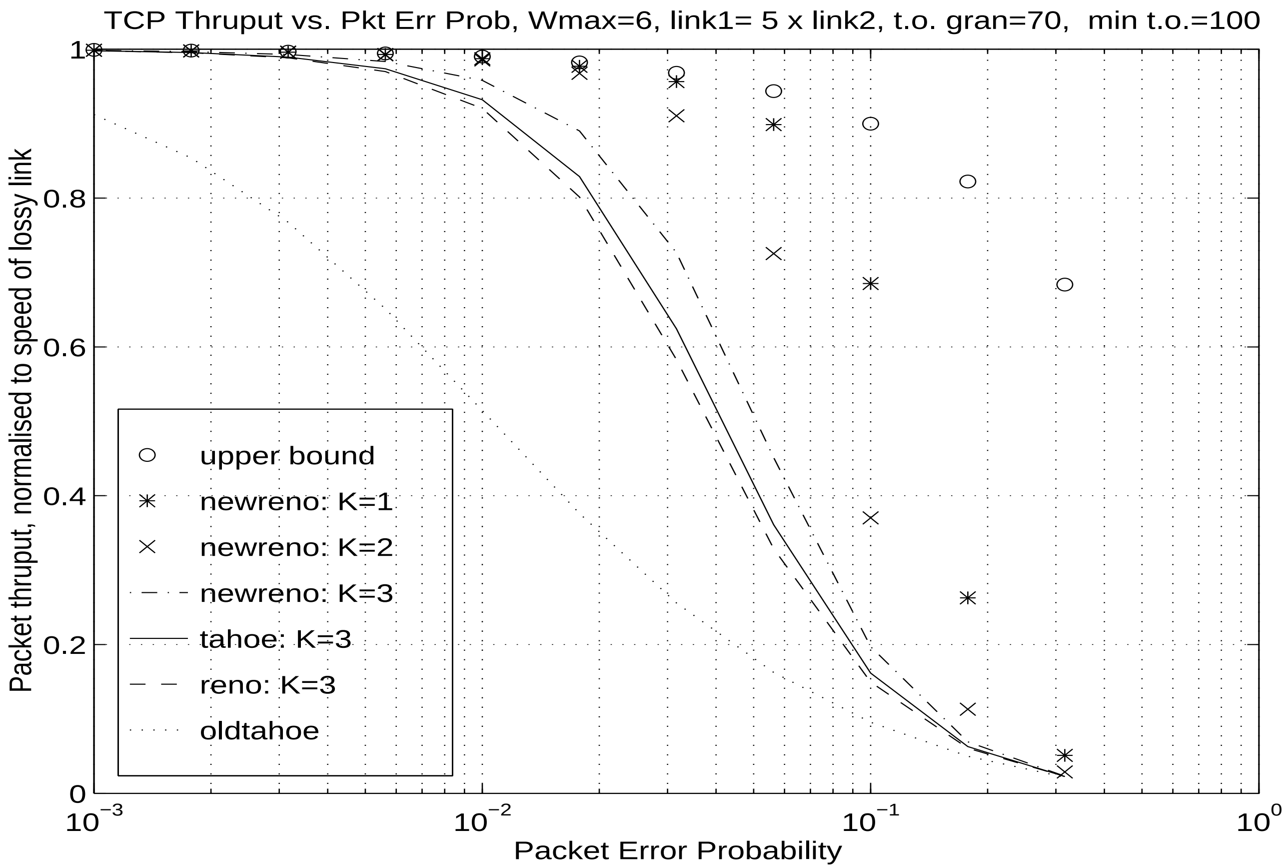
NUMERICAL RESULTS

- 1.5Mbps wireless link
- 1400byte packets $\Rightarrow$ 7.5ms transmission time
  - time is normalised to this transmission time
- LAN host can source data at 7.5Mbps $\Rightarrow \lambda = 5\mu$
- 500ms. timer tick $\Rightarrow rto\_gran = 70$
- 750ms. minimum timeout $\Rightarrow rto\_min = 100$
- $W_{max} = 6, 24,$ or 48

  $\Rightarrow$ receiver buffer sizes of 8kB, 32kB, or 64kB
- For fading model: carrier frequency, $f_c = 900MHz$

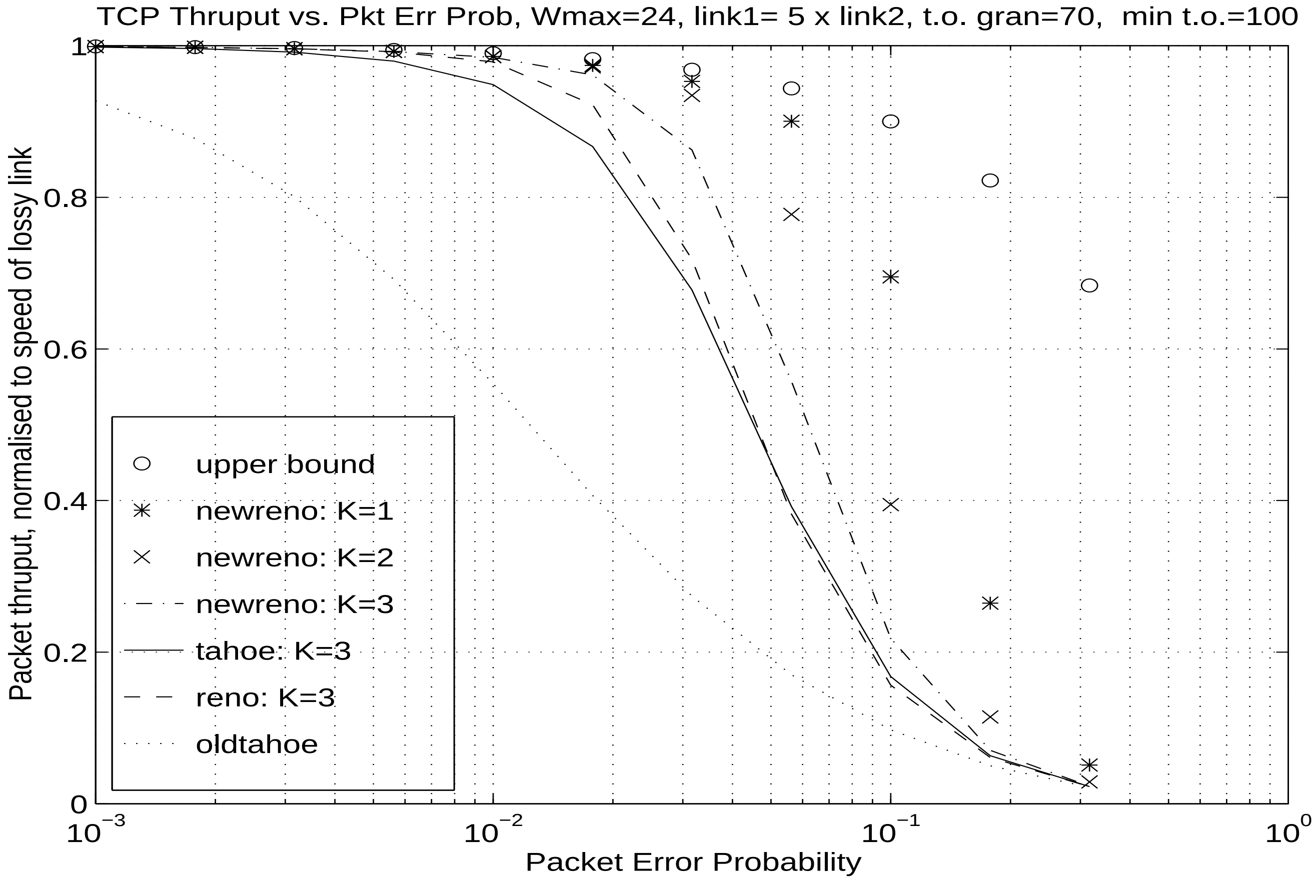
WITHOUT FADING: THRUPUT vs. p

- $W_{max} = 6, \lambda = 5, K = 3, rto\_gran = 70, rto\_min = 100$



WITHOUT FADING: THRUPUT vs. p

- $W_{max} = 24, \lambda = 5, K = 3, rto\_gran = 70, rto\_min = 100$

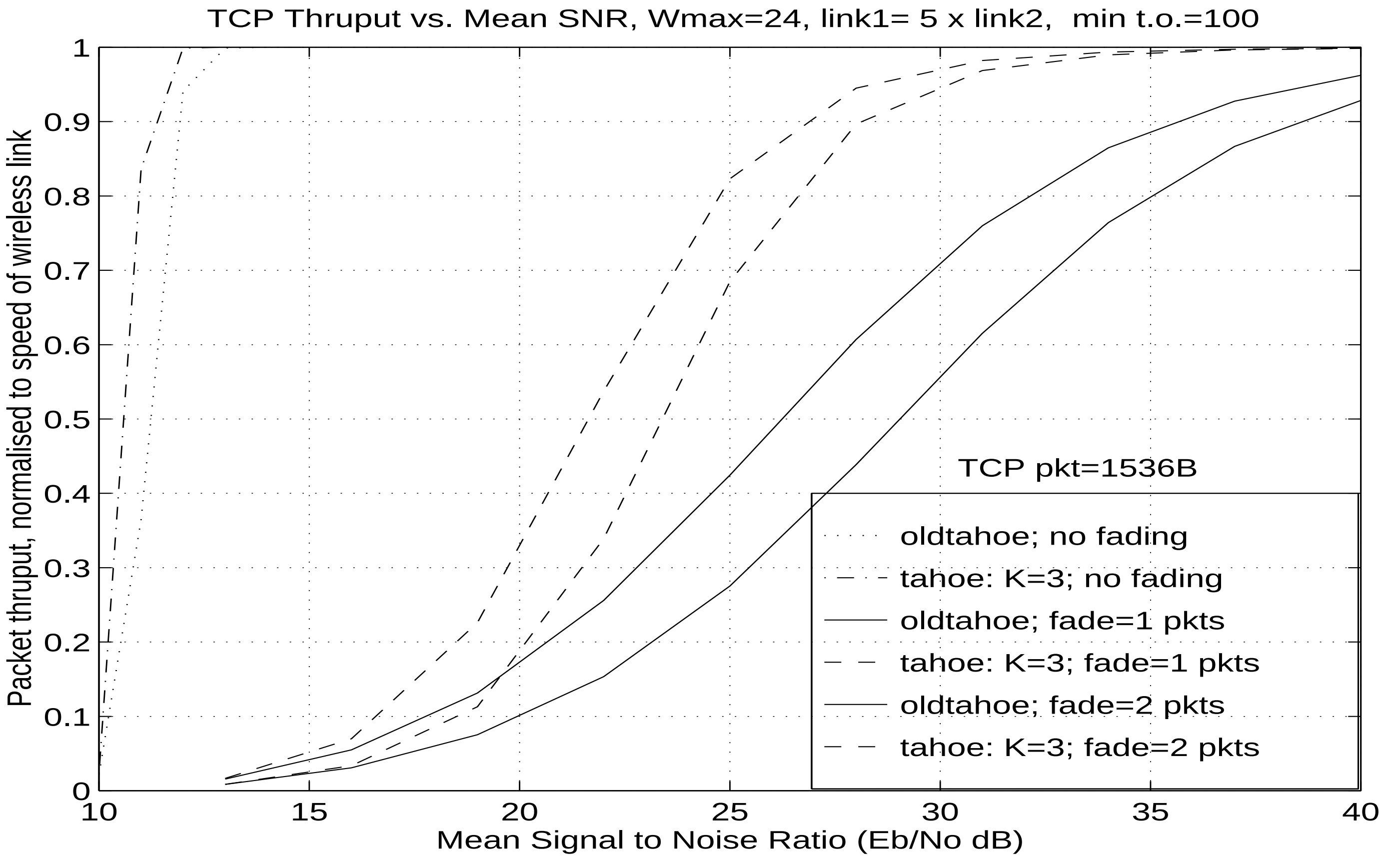


OBSERVATIONS

- OldTahoe yields poor throughput even for p as low as 0.001
- Fast-retransmit and fast-recovery improve thruput significantly
  - even then thruput degrades for $p > .01$
- Reno is not better than Tahoe for
  - for small W_{max}, or
  - for $p > .05$
- NewReno benefits most from fast-recovery
  - reducing K to 1 results in significant improvement
    * similar to one of the proposals in TCP-Vegas
- All versions have similar poor performance for $p > 0.1$

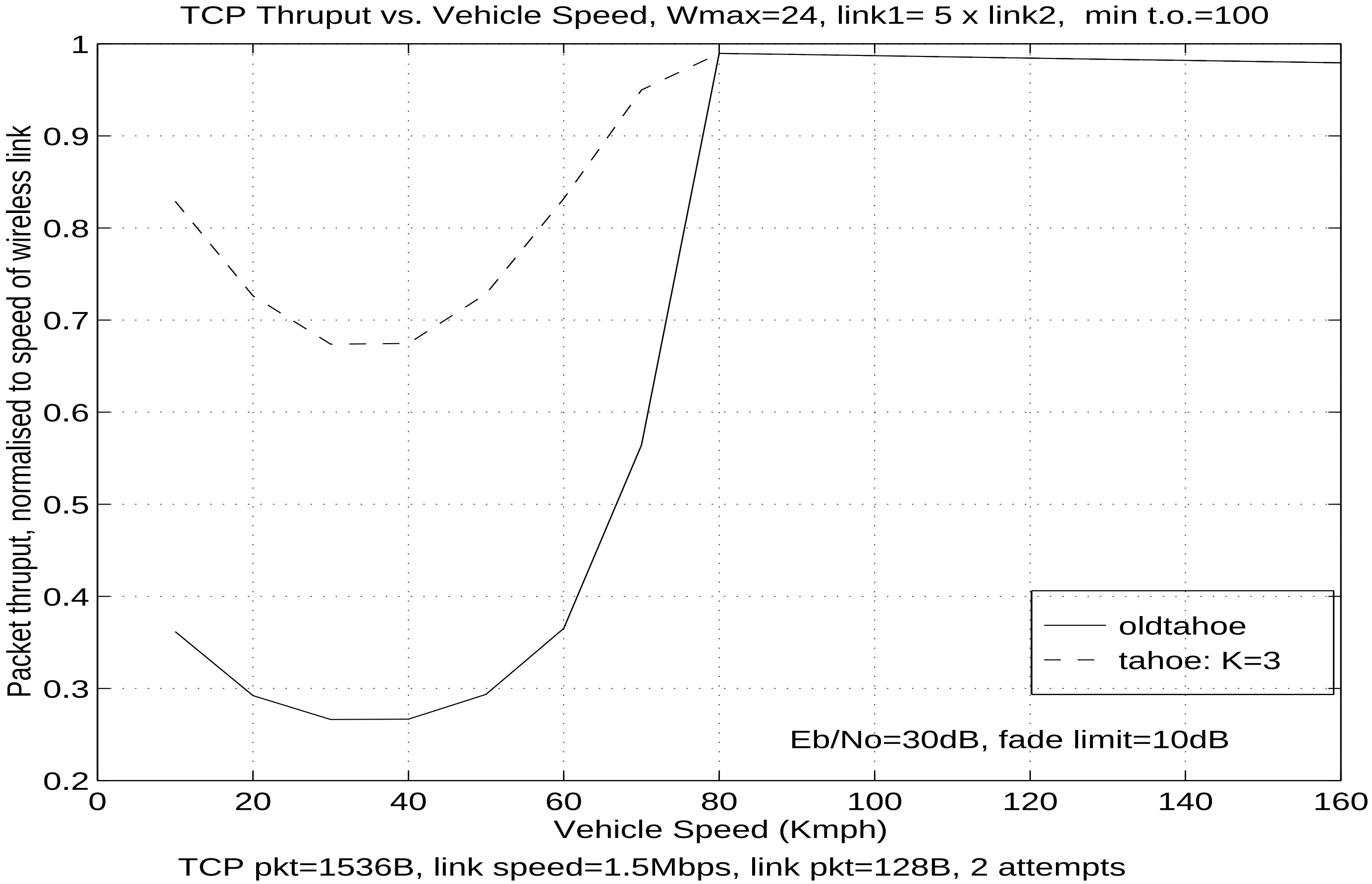
RAYLEIGH FADING: THRUPUT VS. Eb/No

- TCP pkt level loss model, no link protocol, fade limit=10dB



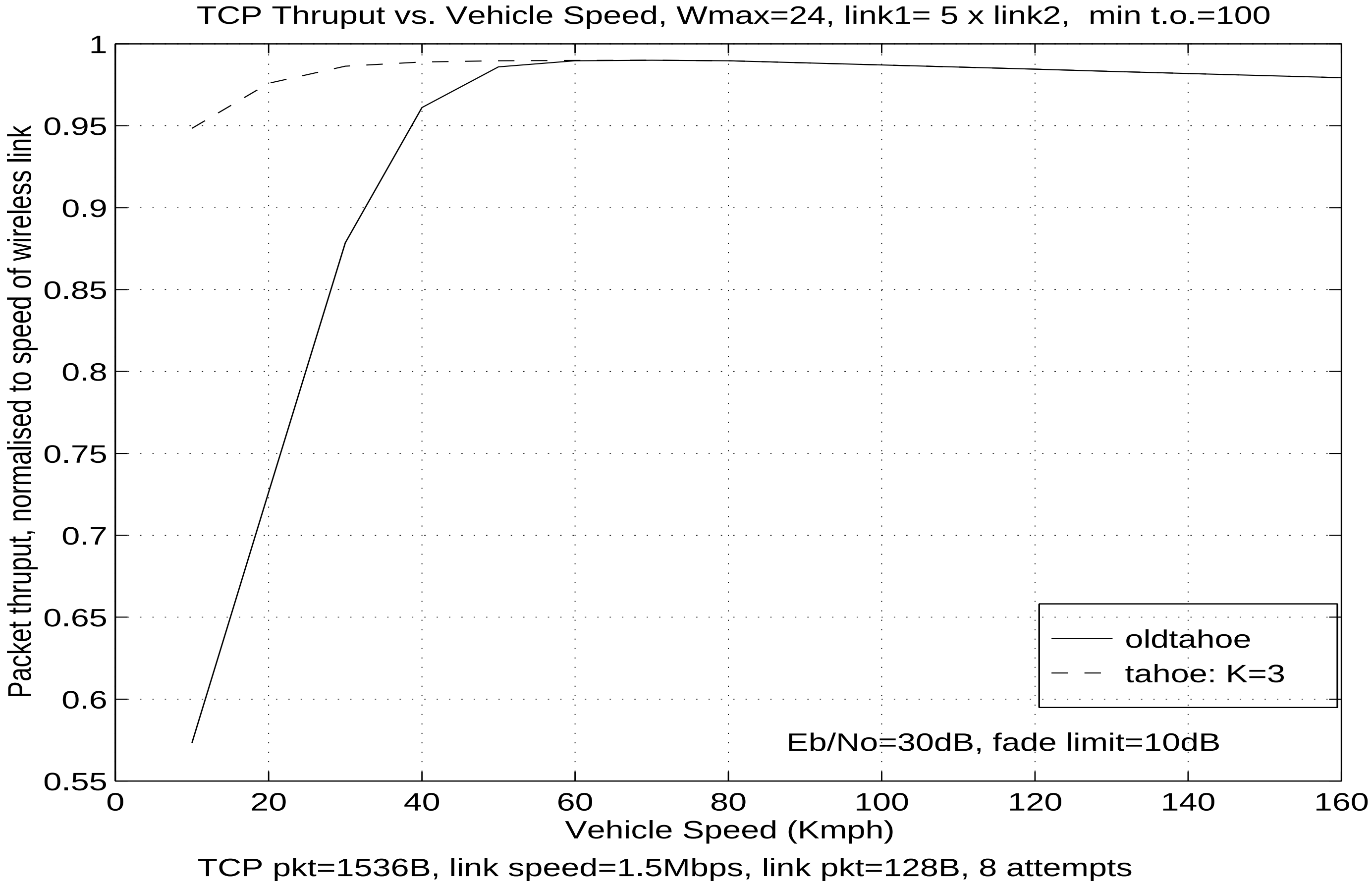
RAYLEIGH FADING: THRUPUT VS. SPEED

- 128 byte link pkts, 2 attempts, Eb/No=30dB, fade limit=10dB



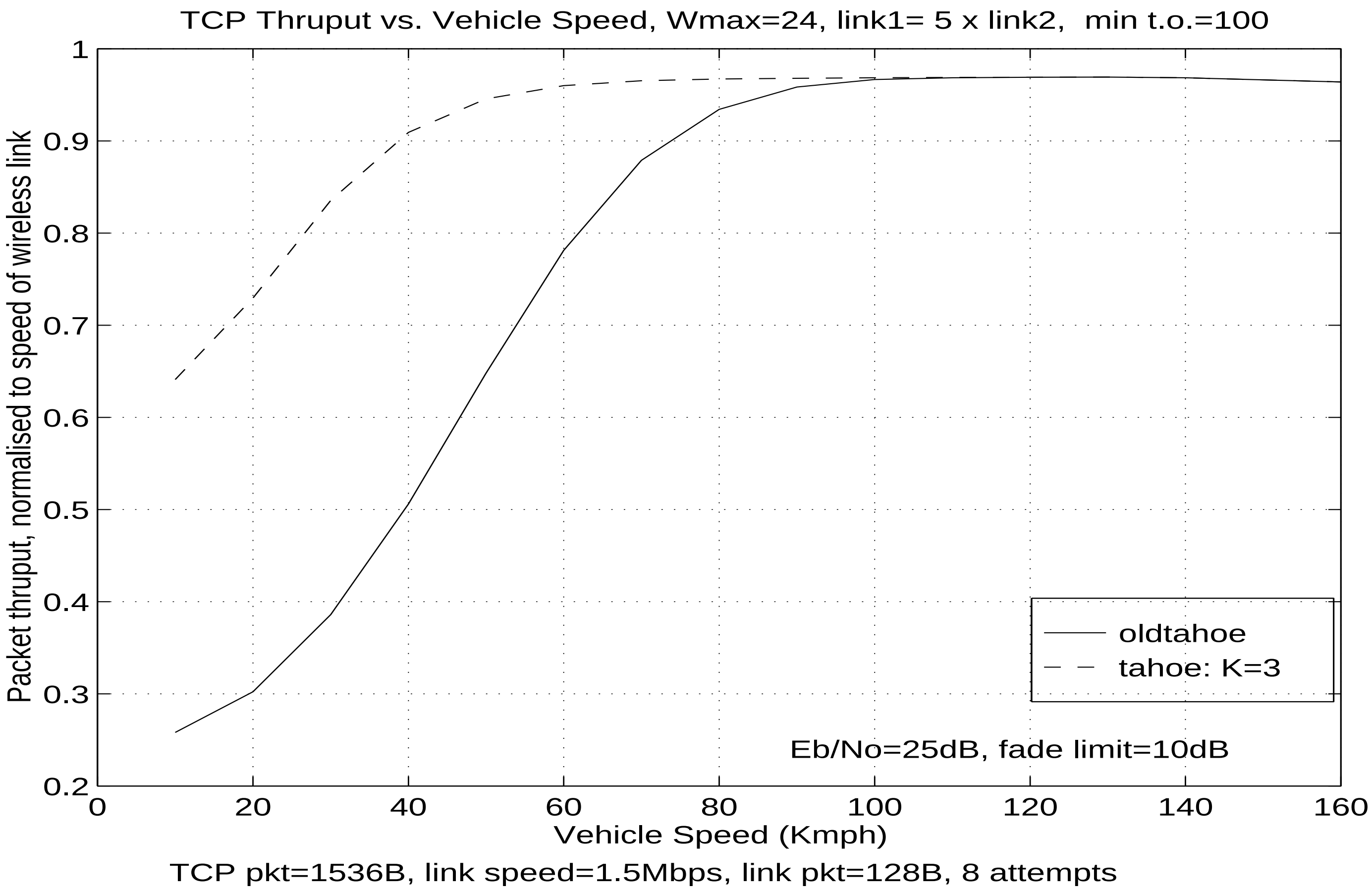
RAYLEIGH FADING: THRUPUT VS. SPEED

- 128 byte link pkts, 8 attempts, Eb/No=30dB, fade limit=10dB



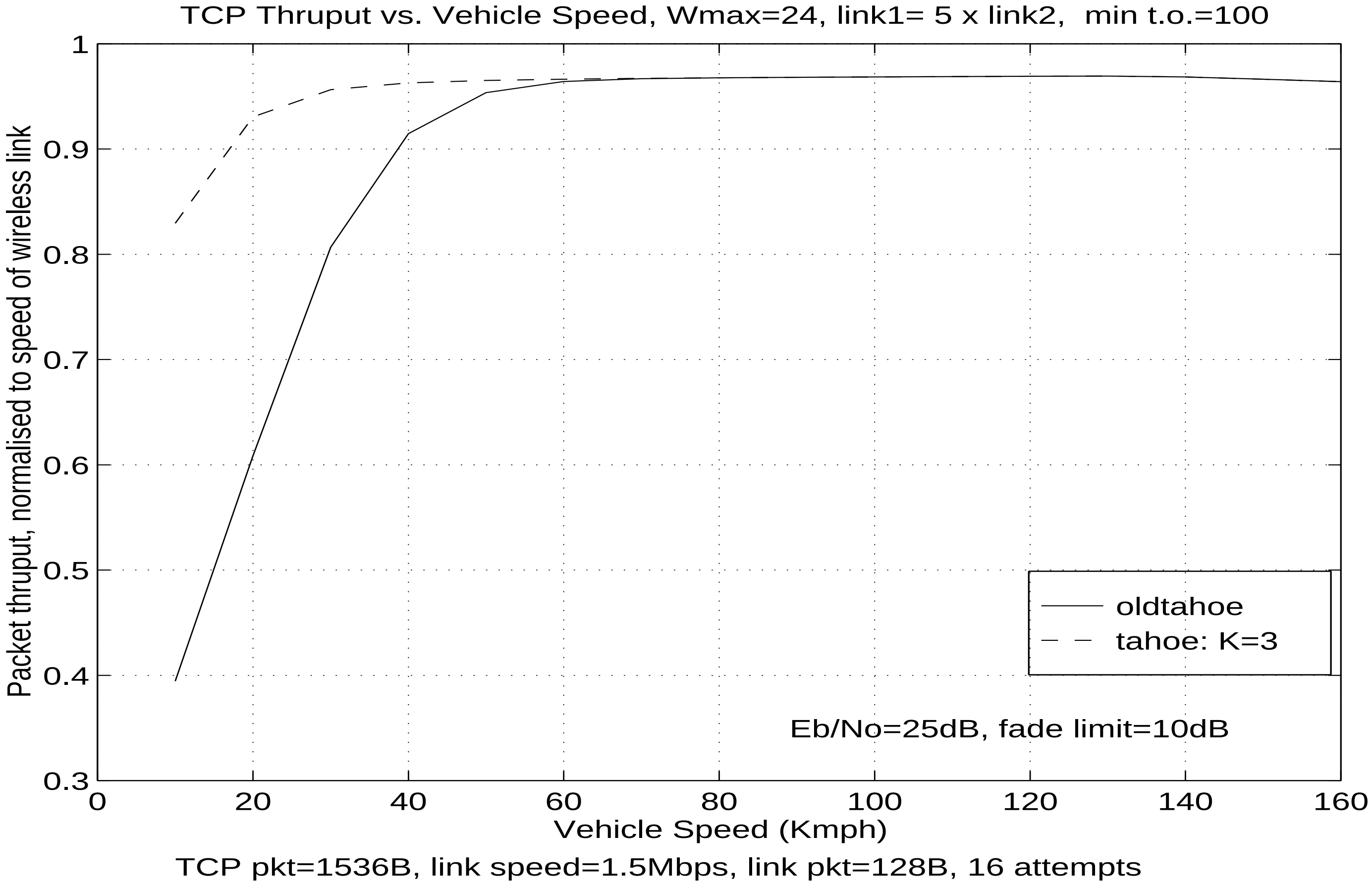
RAYLEIGH FADING: THRUPUT VS. SPEED

- 128 byte link pkts, 8 attempts, Eb/No=25dB, fade limit=10dB



RAYLEIGH FADING: THRUPUT VS. SPEED

- 128B link pkts, 16 attempts, Eb/No=25dB, fade limit=10dB



OBSERVATIONS: FADING MODEL

- 15dB to 20dB fade margin needed to get reasonable thruput
  - * without requiring a very large number of reattempts
- Performance improves at low speeds, and very high speeds
  - * dip in throughput at intermediate speeds
- Choice of link protocol parameters important
  - * link packet length
  - * number of reattempts
  - * for fading channels, need to outlast the fade

FUTURE WORK

- Analysis with cellular system parameters
  - GSM, and IS99
  - channel coding and interleaving
- Analysis for wide area networks
  - additional issue of keeping round-trip "pipe" filled
- Validate analysis against controlled experiments