deterministic analysis (cont'd)

- To calculate <u>average throughput</u>, consider one cycle, from the end of 1 congestion avoidance phase to the next
  - Long run average throughput $= \frac{\#\text{pkts transmitted in cycle}}{\text{duration of cycle}}$

- <u>TCP-tahoe</u>
  - every cycle starts w/ slow-start
  - in steady state, threshold $W_{th1} = \frac{\Delta + B + 1}{2}$
  - <u>Case 1</u>: $W_B > W_{th1}$
    - only 1 slow-start per cycle, which ends when window reaches W_{th1}
    - $n_{ss} \approx W_{th1}$

      $t_{ss} \approx T \log_2 W_{th1}$
  - <u>Case 2</u>: $W_B \leq W_{th1}$
    - 2 slow-start phases
    - 1^{st} slow-start phase ends when window reaches

      $W^* \approx \min\{2W_B - 1, W_{th1}\}$

      $n_{ss1} \approx W^*$

      $t_{ss1} \approx T \log_2 W^*$
    - 2^{nd} slow-start phase has threshold

      $W_{th2} = \frac{W^*}{2}$

      $n_{ss2} \approx W_{th2}$

      $t_{ss2} \approx T \log_2 W_{th2}$
    - $n_{ss} = n_{ss1} + n_{ss2}$

      $t_{ss} = t_{ss1} + t_{ss2}$
  - congestion avoidance phase starts with window

    $$W_0 = \begin{cases} W_{th1} & \text{if } W_B > W_{th1} \\ W_{th2} & \text{if } W_B \leq W_{th1} \end{cases}$$

    which we can substitute into previously derived expressions for $n_{ca\text{-}pipe\text{-}fill}$, $t_{ca\text{-}pipe\text{-}fill}$, $n_{ca\text{-}buf\text{-}fill}$ & $t_{ca\text{-}buf\text{-}fill}$
  - long term average throughput

    $$\approx \frac{n_{ss} + n_{ca\text{-}pipe\text{-}fill} + n_{ca\text{-}buf\text{-}fill}}{t_{ss} + t_{ca\text{-}pipe\text{-}fill} + t_{ca\text{-}buf\text{-}fill}}$$

    *Note: the period from buffer overflow to detection of loss at sender is neglected

- <u>TCP-reno</u>
  - every cycle ends with detection (by triple duplicate ACK) of loss, at window size $\Delta + B + 1$
  - every cycle consists of 1 congestion avoidance phase where window is initially $W_0 = \frac{\Delta + B + 1}{2}$
  - long term avg throughput

    $$\approx \frac{n_{ca\text{-}pipe\text{-}fill} + n_{ca\text{-}buf\text{-}fill}}{t_{ca\text{-}pipe\text{-}fill} + t_{ca\text{-}buf\text{-}fill}}$$

    * Note above applies

- link utilization as a function of $\frac{\text{buffer size } B}{\Delta+1} \triangleq \beta$

$\left(\frac{L}{C} = 0.01, \quad \Delta = 100\right)$

β	Link utiliz^n (Anal./Simulat^n) Tahoe	Reno
0.1	0.604/0.604	0.818/0.818
0.2	0.660/0.664	0.871/0.870
0.31	0.708/0.718	0.915/0.911
0.32	0.856/0.858	0.919/0.916
0.8	0.953/0.954	0.996/0.994

- at $\beta = 0.31$, threshold where buffer is at the size where Tahoe goes from 2 slow starts to 1 slow start
- simulation implements more complicated fast recovery algorithm used in practical implementations of TCP-Reno:
  - when loss is detected by a triple duplicate ACK, set window to $\frac{1}{2}$ its current value W, +3 (inflating the window by the no. of packets successfully removed fr the network) & retransmit lost pkt
  - each time another duplicate ACK arrives, inflate the window by 1 additional packet, & transmit a new packet if allowed by the window
  - when the next non-duplicate ACK arrives, set the window back to $\frac{W}{2}$

Stochastic model for a connex^n in a WAN (Padhye et al 98, simplif'd)

- Assume Bernoulli packet losses — each packet is lost independently with probability p
- Let d be the average RTT (including queueing delay)
- Assume p & d are extraneous parameters not affected by the transfer being analysed
  - reasonable model for a WAN with many other connections
- Consider a TCP-reno connection that is evolving in congestion avoidance
  - examine evolution of window process $w(t)$ at multiples of the RTT d
- We assume d, c are large enough that an entire window of packets can be sent in each RTT
- Consider cycles (of random duration) each consisting of 1 congestion avoidance phase ending in loss & fast recovery; index cycles by $k = 0, 1, 2, \ldots$
- Let W_k be the window at the start of cycle k,
  - T_k the time at which cycle k starts,
  - N_k the no. of RTTs that elapse without loss in cycle k,
  - M_k the index of the packet in the (N_k+1)th RTT that is lost
- Note that W_k determines the distributions of N_k, M_k & $(T_{k+1} - T_k)$; also (W_k, N_k) determines W_{k+1}

  → W_k is a Markov chain & (W_k, T_k) is a Markov renewal process (MRP)
- Let R_k be the no. of packets transmitted successfully in cycle k
  - R_k is independent of anything else given $(W_{k-1}, W_k, T_k - T_{k-1} \triangleq U_k)$

  → By the renewal reward thm,

  long term average throughput

  $$= \frac{E_\pi(R)}{E_\pi(U)} \qquad — (1)$$

  where E_π denotes expect^n wrt the stationary distribution π of W_k

- Consider a cycle k

- In the first N_k RTTs, the window grows by 1 each RTT & the no. of pkts transmitted in the 1st N_k RTTs is

$$W_k + (W_k + 1) + \dots + (W_k + N_k - 1)$$

$$= N_k \left(\frac{2W_k + N_k - 1}{2} \right)$$

- Note that

$$W_{k+1} = \frac{1}{2} \times \text{Window at which loss is detected}$$

$$= \frac{W_k + N_k}{2} \quad \text{—— (2)}$$

(loss occurs in $(N_k+1)^{st}$ RTT when window is $W_k + N_k$, detected before window can grow further)

- Consider no. of pkts transmitted up to & including the 1st lost pkt in cycle k

$$Y_k = \underbrace{N_k \left(\frac{2W_k + N_k - 1}{2} \right)}_{\text{\# pkts sent in 1st } N_k \text{ RTTs}} + M_k \quad \text{— (3)}$$

- $E(Y_k) = \frac{1}{p}$

However, taking the expect^n of the RHS is complicated

- We make the following simplifying assumptions:
  - assume variations in W_k are small, so take $W_k \approx W_{k+1} \approx w$, a constant

→ from (2), $N_k = 2W_{k+1} - W_k$
$\approx w$

- also assume $E(M_k) \approx \frac{w}{2}$ (expected position of lost packet is in the middle of the packets sent in that RTT)

- from (3)

$$\frac{1}{p} \approx w \left(\frac{3w-1}{2} \right) + w$$

$$\Rightarrow 3w^2 + w - \frac{2}{p} \approx 0$$

$$\Rightarrow w \approx \frac{-1 + \sqrt{1 + \frac{24}{p}}}{6}$$

$$\approx \sqrt{\frac{2}{3p}} \quad \text{for small } p$$

- roughly, in each cycle, the window grows linearly from w to $2w$ over w RTTs, transmitting about $w\left(\frac{w+2w}{2}\right) = \frac{3w^2}{2}$ packets which we set equal to $E(Y_k) = \frac{1}{p}$

- loss occurs in $(N_k+1)^{st}$ RTT & is detected in $(N_k+2)^{nd}$ RTT, so $U_k \approx d(N_k + 2)$

$$E(U) \approx d(w+2)$$

- From (1), long term average throughput $\approx \dfrac{\frac{1}{p}}{d\left(\sqrt{\frac{2}{3p}} + 2\right)}$

$$\approx \frac{1}{d}\sqrt{\frac{3}{2p}} \quad \text{packets per unit time for small } p$$

<u>Note</u>: this represents the throughput limitation due to loss, throughput also limited by bottleneck link capacity & buffers