Analysis and Simulation of Weighted Random Early Detection (WRED) Queues

Mark Wurtzler

MS Project Defense

Department of Electrical Engineering and Computer Science

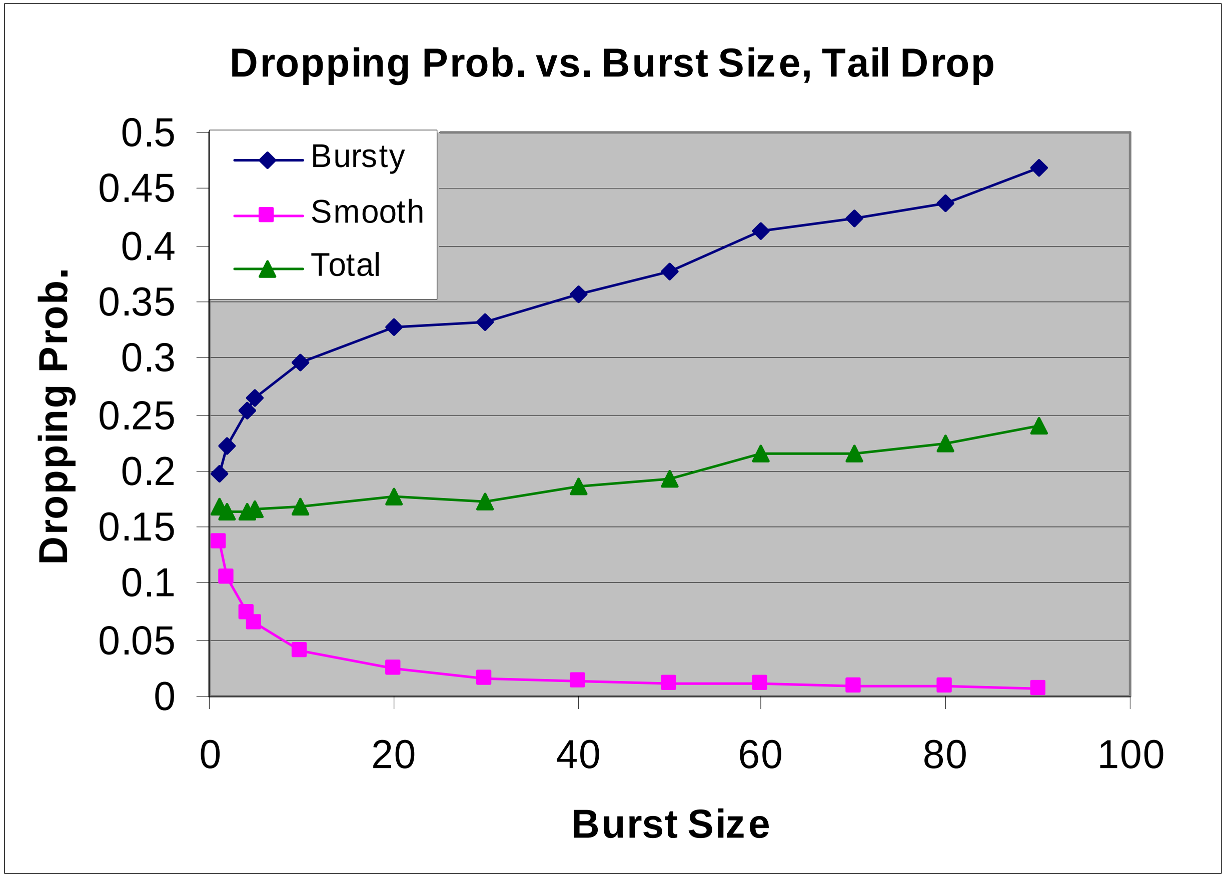
University of Kansas

December 15, 2003

Problem Summary

- Continuous introduction of new applications for existing Internet infrastructure (ex: Voice over IP, Video)
- These applications exhibit very different traffic characteristics, but they may share the same FIFO queue at switching and routing nodes
- If queues are allowed to only drop packets during overflow conditions, then bursty traffic flows (like TCP) will face greater dropping probabilities than smooth traffic

Problem Summary



Simulation Results: Blocking Probability using Tail Drop, ρ=1.2, Queue Size=100, Mean Exponential Packet Length=1000, Link Rate = 10000

Bursty: Batched Poisson Arrivals

Smooth: Constant Interarrivals

Problem Summary

- Traffic on the Internet may become more bursty in some areas as network access speed improves:

It was found that a 33Kbps modem user produces a peak rate that is about 3.3 times the average transfer rate

A similar customer using a 1Mbps broadband connection produces a peak rate that is approximately 100 times the average rate [1].

Problem Summary

- While aggregation of data flows can “smooth” the traffic, this is not always practical for systems with a smaller number of users – bursty traffic may still face high dropping rates
- How can the dropping probabilities be equalized?

Random Early Detection (RED)

- Solution: Random Early Detection w/ Dropping
- Packets are dropped in a probabilistic fashion before the queue reaches an overflow state
- Compared to a tail drop queue, a RED queue will operate with a lower queue fill, especially during peak load conditions
- This will allow bursts of packets to “fit” into the available queue spaces

Random Early Detection (RED)

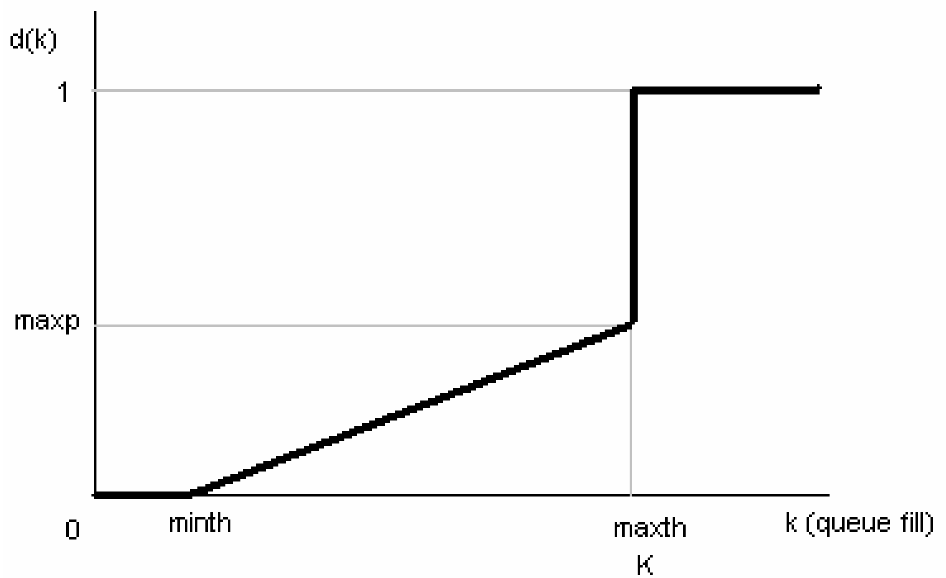
RED Parameters:

min_{th} – Minimum Threshold for Packet Dropping

max_{th} – Maximum Threshold for Packet Dropping

max_p –

Maximum Dropping Probability



RED Dropping Function – Dropping Probability d(k) vs. Queue Fill

$$d(k) = 0 \text{ if } k < min_{th}$$

$$d(k) = 1 \text{ if } k > max_{th}$$

$$d(k) = (\max_p)\left(\frac{k - \min_{th}}{\max_{th} - \min_{th}}\right) \text{ otherwise}$$

Random Early Detection (RED)

- RED aggressiveness d(k) can be increased by decreasing min_{th} or increasing max_p
- For RED using instantaneous queue fill sampling, max_{th} will generally be set to “K”, the queue size
- Aggressive RED will decrease the mean queue fill, but it will also increase overall dropping probability

Goals of RED

Some of the proposed [2] benefits of RED are:

- Congestion Avoidance
- Resistance to TCP synchronization
- Increased fairness to bursty TCP-type traffic

This project focuses on fairness considerations and the enhanced benefits made possible by class-sensitive RED (WRED)

Notes on RED Queue Fill Averaging

- The study of the dynamics of the use of queue fill averaging algorithms with RED is a major current area of research
- Averaging will make RED react more slowly to bursts, but it provides a more accurate sense of queue load
- This project will use instantaneous queue fill sampling in order to make analytic solutions tractable and to emphasize the effects of RED with focus on differentiated service performance

Analysis of RED

- Bonald/May/Bolot proposed the first analytic model for RED (INFOCOM 2000) [3]
- Uses Continuous-Time Markov Chain (CTMC) analysis to find queue state probabilities

Key Assumptions:

- Poisson Arrivals see Time Averages (PASTA) (analysis techniques may not be valid for other arrival distributions [3])
- Assumption #1 [3]: Packets that are part of a burst will be dropped based on the value of d(k) as sampled at the time the burst enters the queue, not at the time the packet enters – analysis will therefore represent a lower bound for the dropping probability

Analysis of RED

- Dropping Probabilities [3]:

Tail Drop:

$$P_{TD} = \pi(K) + \pi(K-1)\frac{(B-1)}{B} + \ldots\ldots + \pi(K-B+1)\frac{1}{B}$$

RED:

$$P_{RED} = \pi(K) + \pi(K-1)d(K-1) + \ldots + \pi(1)d(1)$$

K = Queue Size

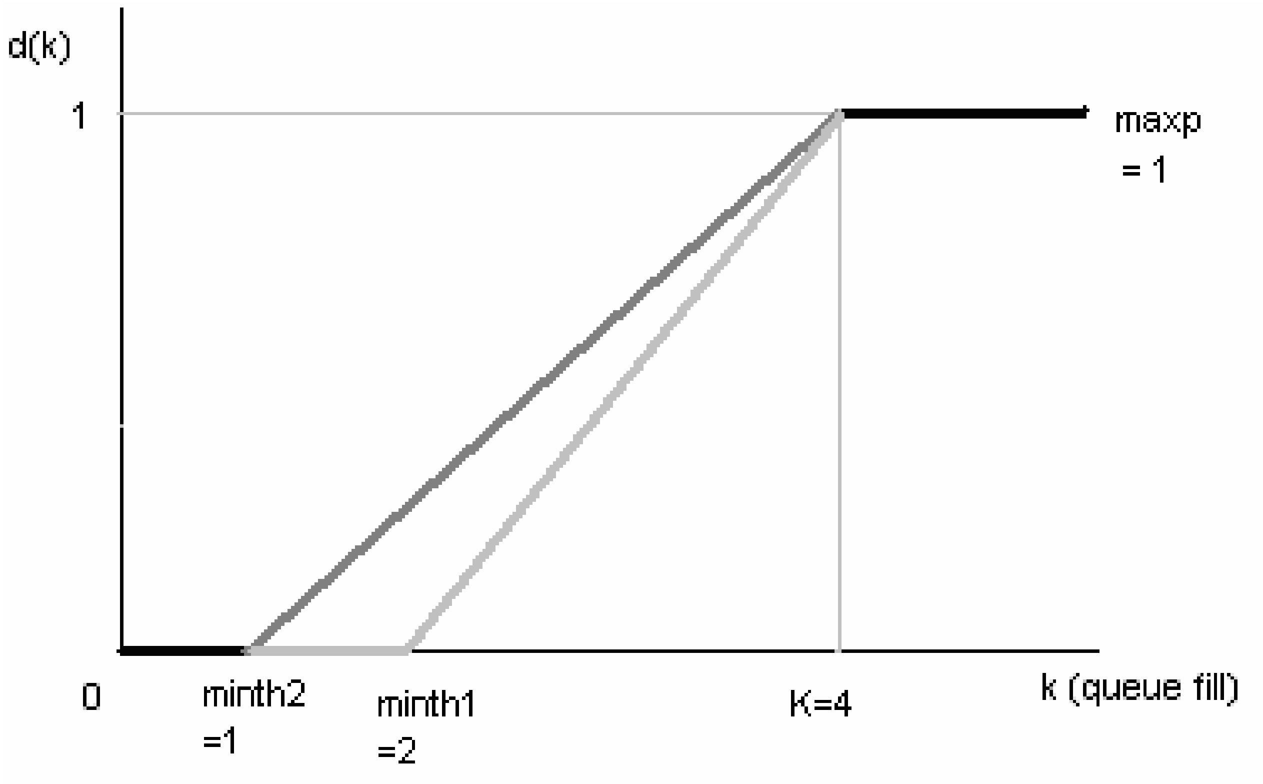
B = Burst Size

Note: Transition probabilities and thus state probabilities will not be the same for Tail Drop And RED CTMCs

Weighted RED (WRED)

- Weighted Random Early Detection performs RED on each class of traffic individually
- It will be shown to be possible to increase the dropping probability fairness to levels beyond what is possible using RED
- WRED allows the network designer to provide differentiated service quality with respect to dropping probability – including allowing bursty traffic to have the higher priority

Weighted RED (WRED)



WRED Dropping Function – Dropping Probability d(k) vs. Queue Fill

Extending Analysis to WRED

- The same type of analysis used for bursty traffic and simple RED queues can be extended to weighted RED
- Transition probabilities will differ
- Dropping can be calculated on a class-by-class basis

WRED Analysis Example

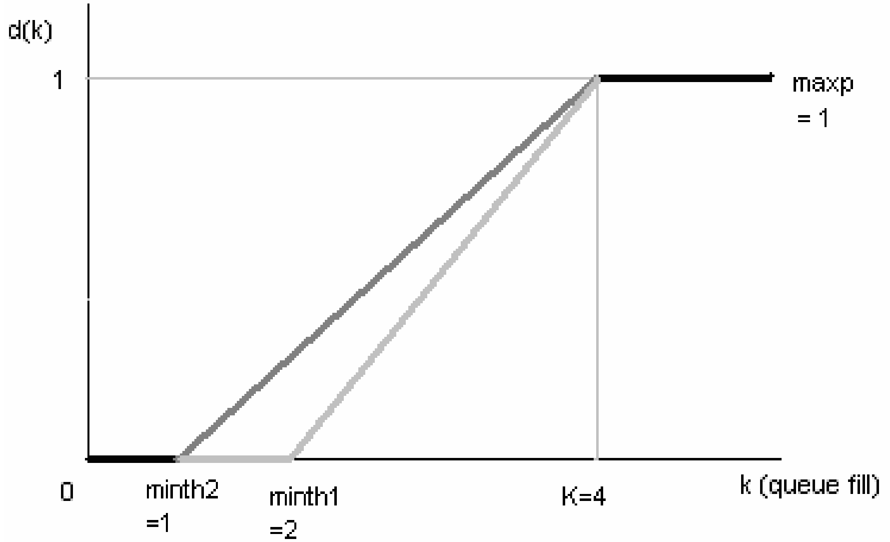
Burst Size (B) = 2

Offered Load (ρ) = 0.6

Note: $\rho_1 = \rho_2 = 0.3$

Queue Size = 4 (System Size = 5)

Arrival Rates (each class) = 0.15 bursts/second, Total Arrival Rate λ=0.3

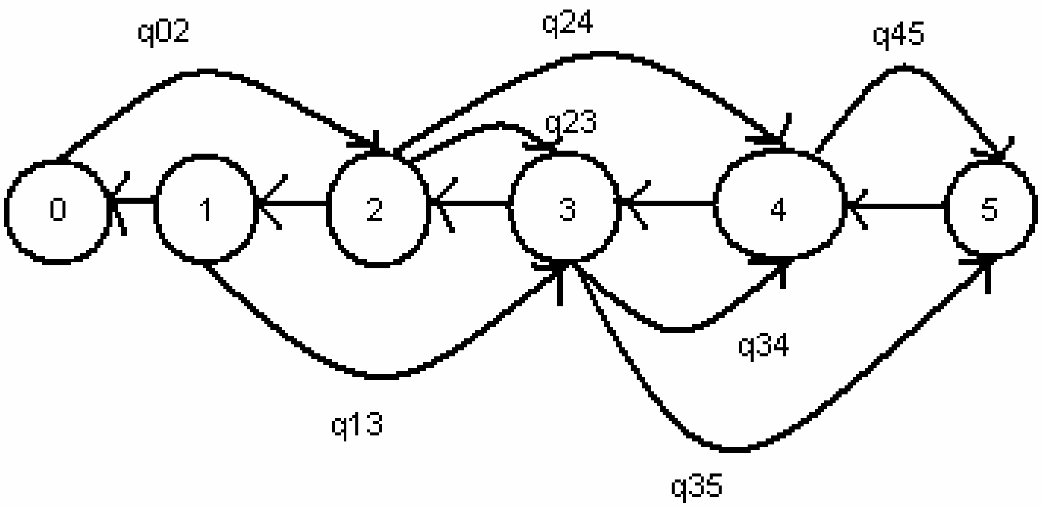


RED Parameters

Number in system (x)	Class 1 Dropping $d_1(x)$	Class 2 Dropping $d_2(x)$
0	0	0
1	0	0
2	0	1/4
3	1/3	1/2
4	2/3	3/4
5	1	1

Dropping Probabilities $d_i(k)$

WRED Analysis Example



All downward transitions at service rate, mu

$$\rho = \frac{B\lambda}{\mu}$$

WRED Continuous-Time Markov Chain

WRED Analysis Example

$$q_{24} = \frac{1}{2}\lambda + \frac{1}{2}\lambda \binom{2}{2}\left(1 - \frac{1}{4}\right)^2 = 0.2344$$

$$q_{23} = \frac{1}{2}\lambda \binom{2}{1}\left(\frac{1}{4}\right)\left(\frac{3}{4}\right) = 0.05625$$

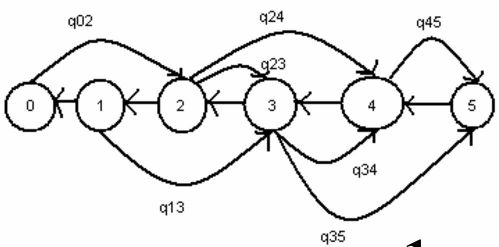
$$q_{35} = \frac{1}{2}\lambda \binom{2}{2}\left(\frac{2}{3}\right)^2 + \frac{1}{2}\lambda \binom{2}{2}\left(\frac{1}{2}\right)^2 = 0.1042$$

$$q_{34} = \frac{1}{2}\lambda \binom{2}{1}\left(\frac{1}{3}\right)\left(\frac{2}{3}\right) + \frac{1}{2}\lambda \binom{2}{1}\left(\frac{1}{2}\right)\left(\frac{1}{2}\right) = 0.1417$$

$$q_{45} = \frac{1}{2}\lambda \left[1 - \left(\frac{2}{3}\right)^2\right] + \frac{1}{2}\lambda \left[1 - \left(\frac{3}{4}\right)^2\right] = 0.1490$$

$$q_{02} = \lambda$$

$$q_{13} = \lambda$$



$$q_{54}, q_{43}, q_{32}, q_{21}, q_{10} = 1$$

Number in system (x)	Class 1 Dropping $d_1(x)$	Class 2 Dropping $d_2(x)$
0	0	0
1	0	0
2	0	1/4
3	1/3	1/2
4	2/3	3/4
5	1	1

Dropping Probabilities $d_i(k)$

WRED Analysis Example

$$\mathbf{Q} = \begin{pmatrix} -0.3 & 0 & 0.3 & 0 & 0 & 0 \\ 1 & -1.3 & 0 & 0.3 & 0 & 0 \\ 0 & 1 & -1.29 & 0.05625 & 0.2344 & 0 \\ 0 & 0 & 1 & -1.2459 & 0.1417 & 0.1042 \\ 0 & 0 & 0 & 1 & -1.1490 & 0.1490 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}$$

Solve for $0 = \boldsymbol{\pi}\mathbf{Q}$ subject to $\sum \pi_i = 1$

$$\boldsymbol{\pi} = \begin{pmatrix} 0.4812 & 0.1445 & 0.1878 & 0.0979 & 0.0681 & 0.0203 \end{pmatrix}$$

Results:

$$P_{RED,1} = \sum_x \pi_x * d_1(x)$$

$$P_{RED,2} = \sum_x \pi_x * d_2(x)$$

$$P_{RED,1} = 0.0983$$

$$P_{RED,2} = 0.1673$$

Class 1 receives preferred service due to its higher minimum threshold

WRED Simulation

Why simulation?

- More accurate results than analysis with Approximation #1 and/or non-Poisson sources
- Markov Chain analysis becomes cumbersome for large queue sizes and complex mixes of traffic

WRED Simulation Components

- Bursty Source Model:

  Batch Poisson Arrivals representing burst traffic (“TCP”)

  B packets arriving together with independent exponential packet length distribution

- Smooth(er) Source Model:

  Constant interarrival times, but packet lengths are exponential (“UDP”)

  Represents types of traffic with non-identical packets generated at regular intervals

  Similar to some video and speech applications

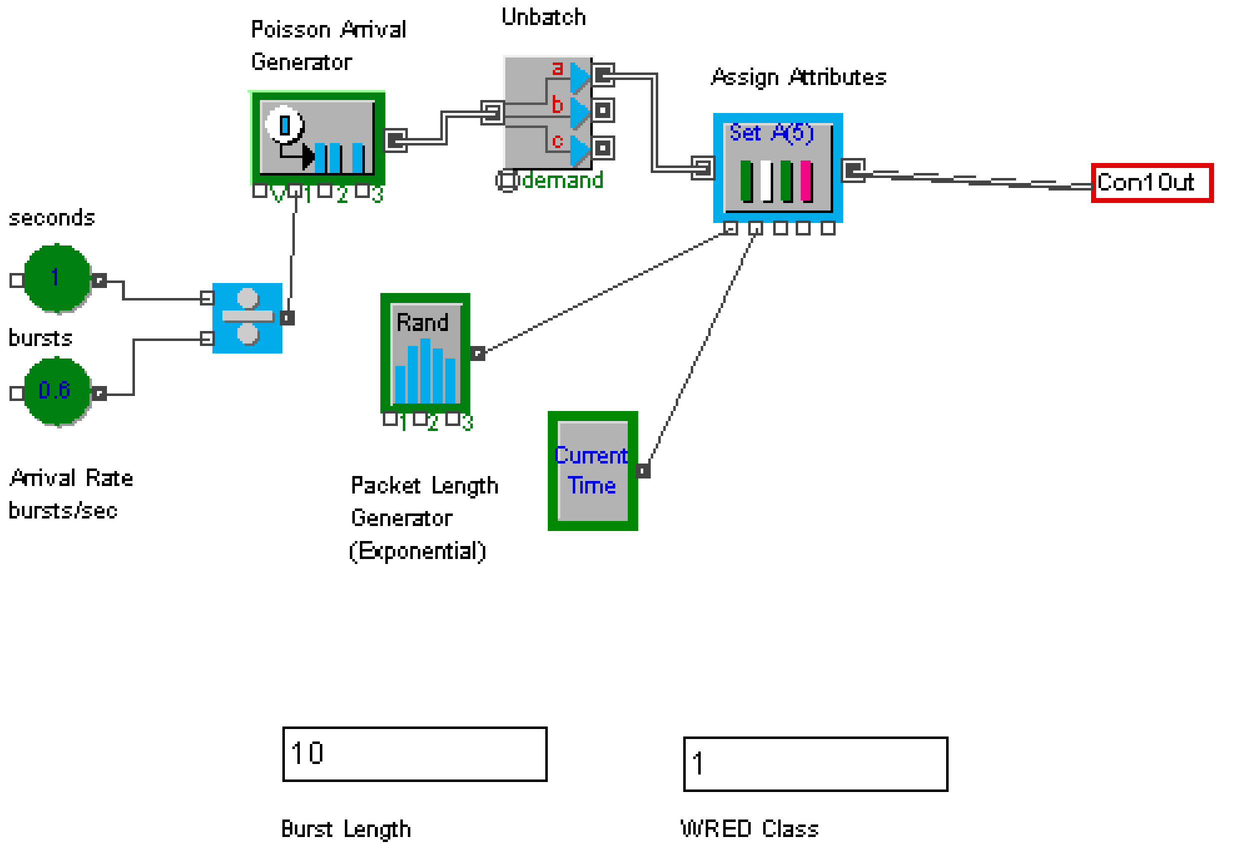
- RED Block

  The WRED device will control which packets are allowed to enter the queue on a class-by-class basis

  One RED block will be assigned to each class of traffic

  Each block can have separate RED parameters (thresholds, max_p, etc.)

WRED Simulation Components: Bursty Source ("TCP") Model



WRED Bursty Source: Verification

- The bursty source can be verified by comparing simulation output to a CTMC-based analytic solution for a tail drop queue

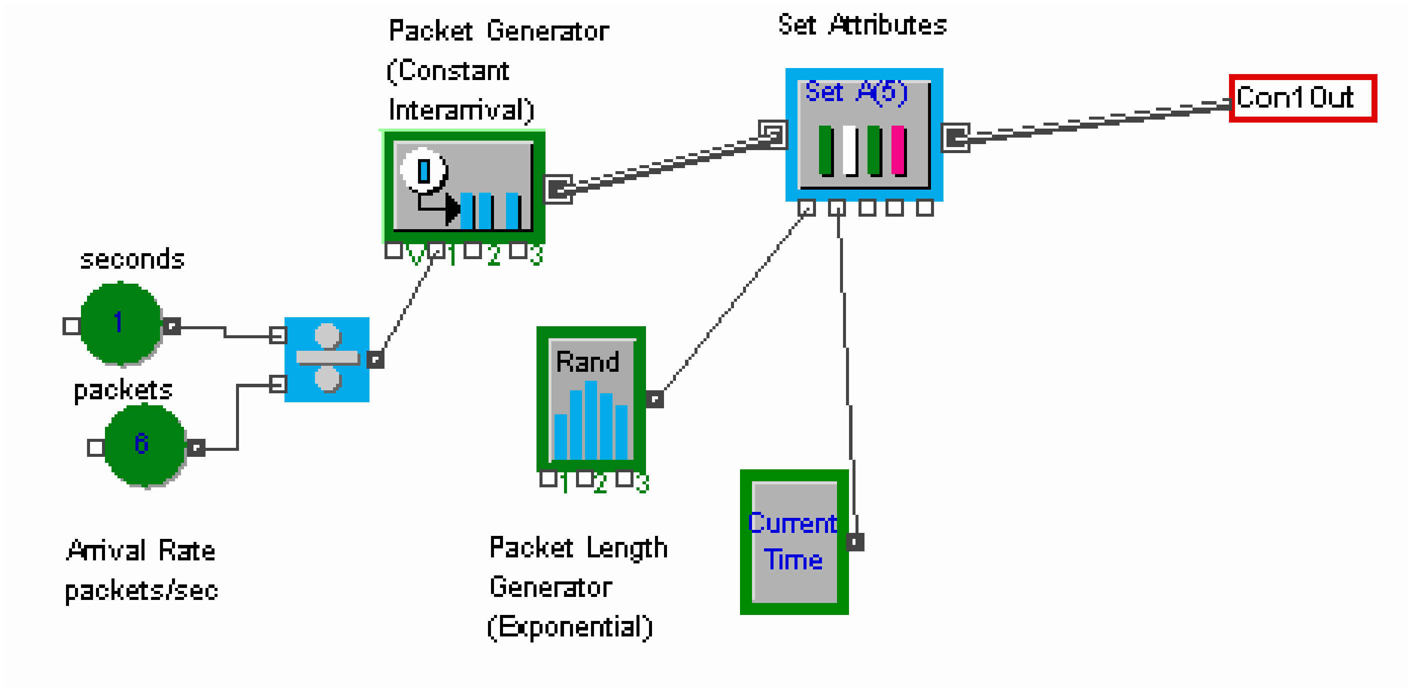
Burst Size	3
System Size	5
Queue Size	4
Burst Arrival Rate	2
Mean Packet Length	1000 bits
Link Capacity	10000 bits
Offered Load	0.6
Total Simulation Time	7000s
Run-in Time	2000s
Number of Runs	5

Simulation Setup

# Of Packets Simulated	149793
# Of Packets Blocked	24395
Blocking Probability/Sim	0.1629
Blocking Probability/Analytic [9]	0.1632
% Error	0.21%

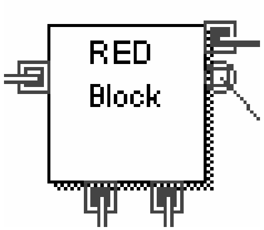
Simulation Results

WRED Simulation Components: Smooth Source (“UDP”) Model

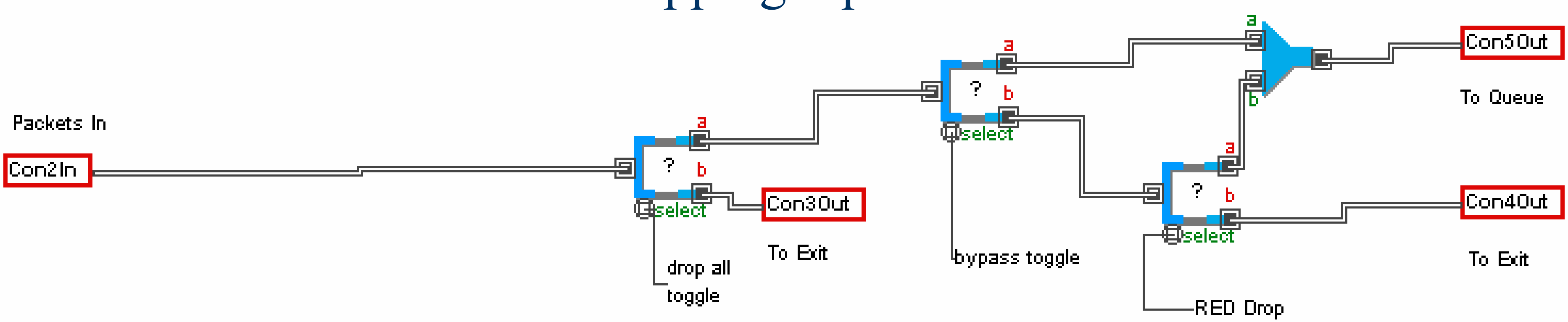


A standard constant interarrival source models the “UDP” traffic

WRED Simulation Components: RED Block Structure

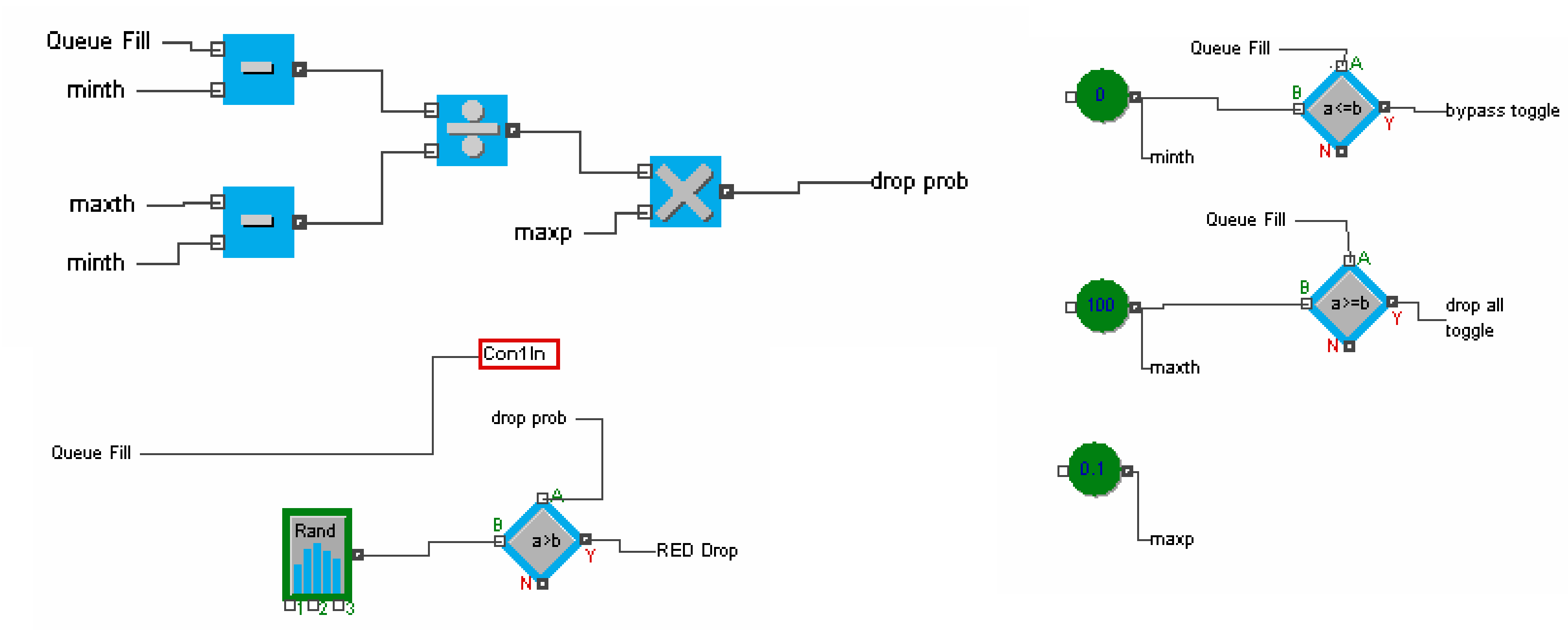


Packets will be routed to either the queue or an exit based on the current number of packets in the queue, threshold settings, and the probabilistic RED dropping input



RED Block Routing

WRED Simulation Components: RED Block Structure



RED Control and Decision Blocks

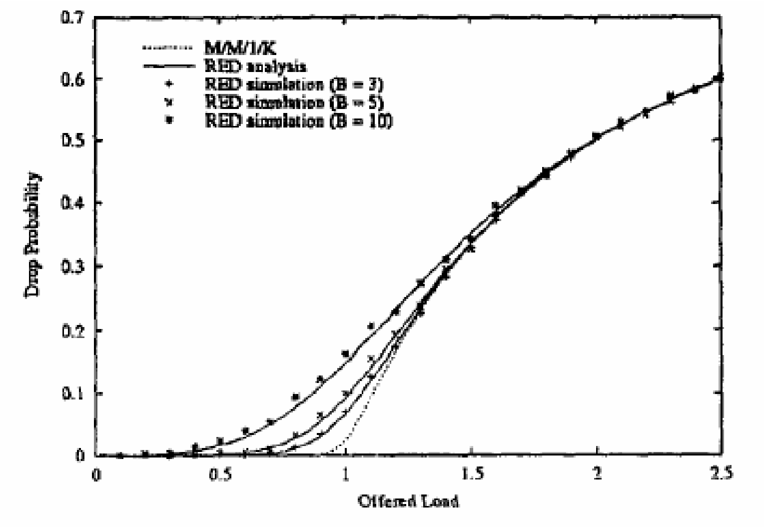
RED Block: Verification

- To verify the performance of the RED block, results from this model are compared to the analytic and simulation results found in the Bonald/May/Bolot paper [3]

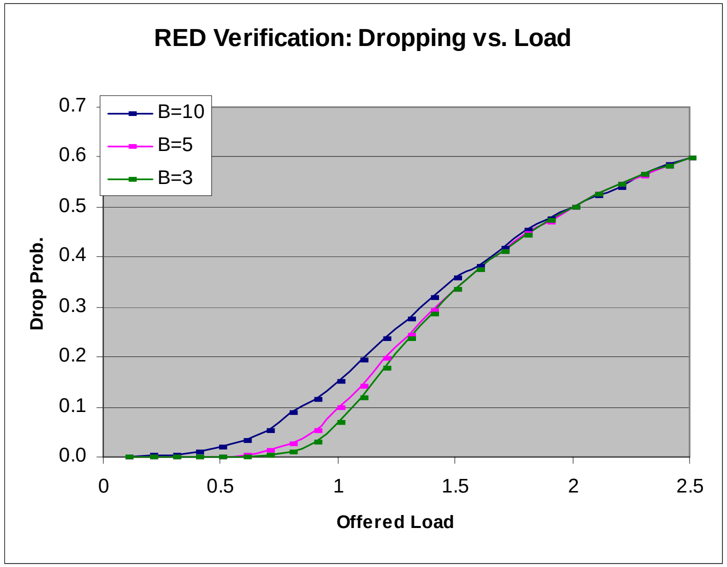
Burst Size	variable
Queue Size	40
Mean Packet Length	1000 bits
Link Capacity	10000 bits
Offered Load	variable
minth	20
maxth	40
maxp	1
Total Simulation Time	7000s
Run-in Time	2000s
Number of Runs	5

RED Block Verification Simulation Parameters

RED Block: Verification



Published Results [3]

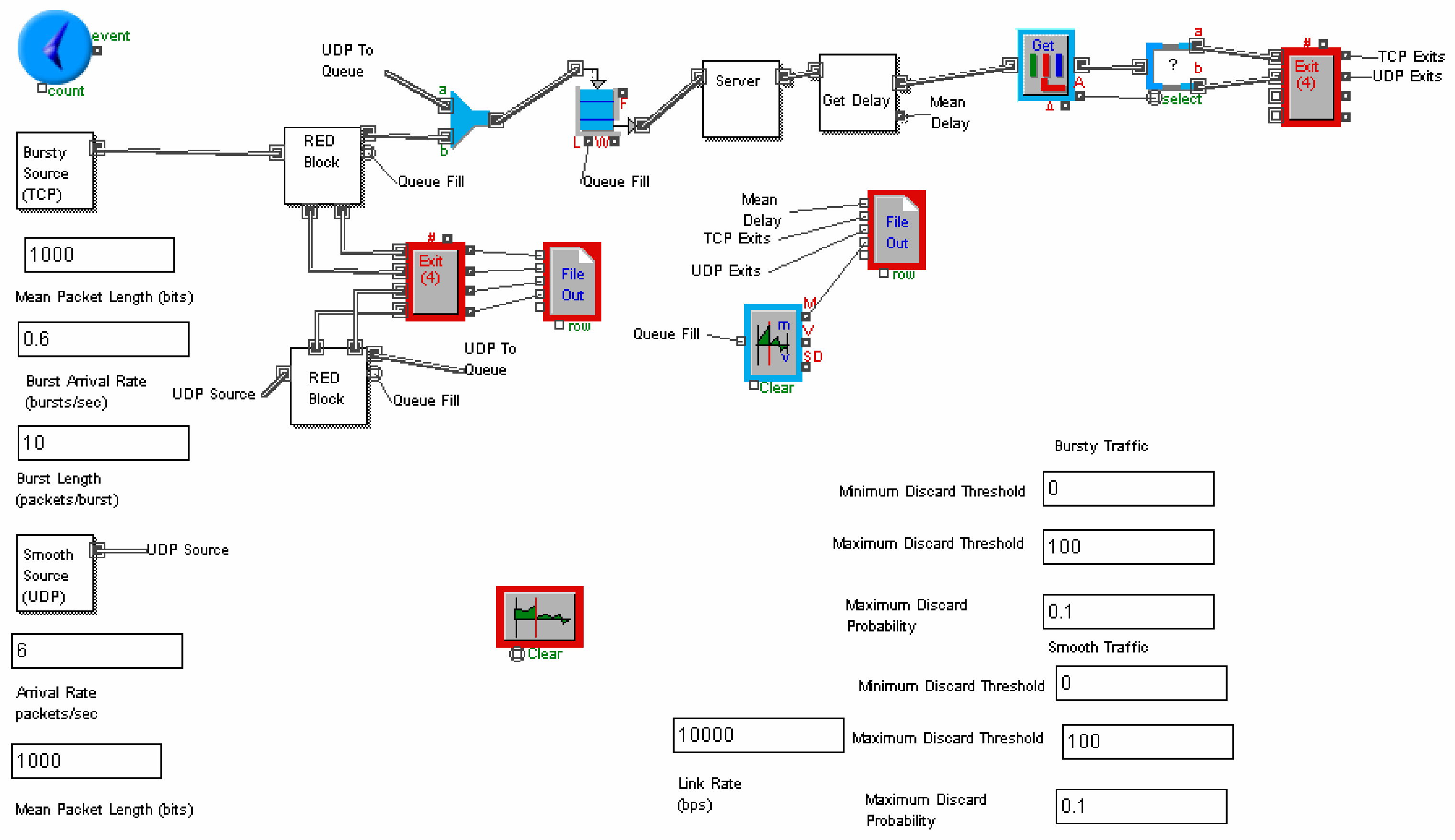


RED Block Results

Observations:

- Despite Approximation #1, the RED analytic solution is accurate when K is large (40+) and B is relatively small
- The RED simulation block closely matches both the analytic solutions and independent published simulation results

WRED Model Overview

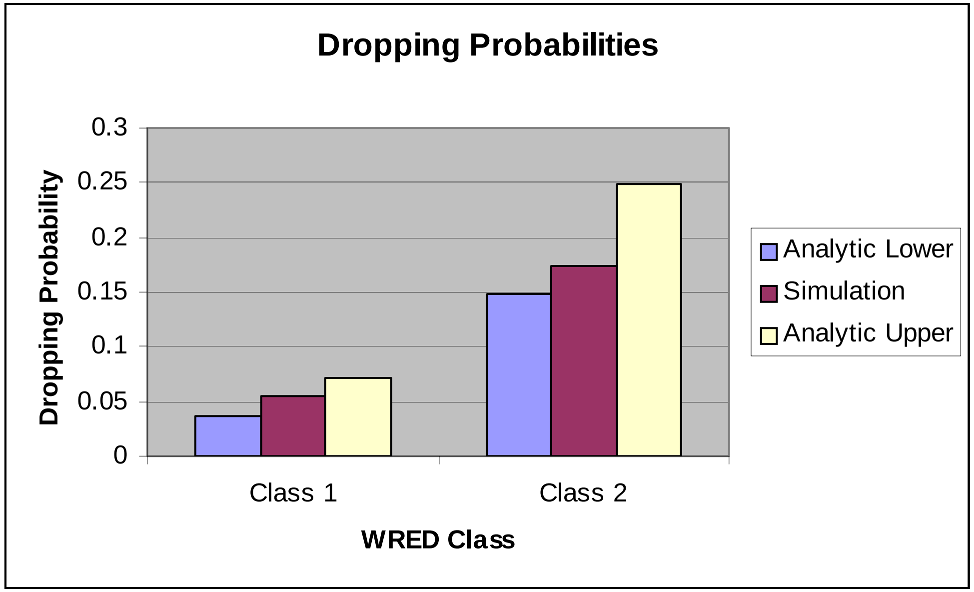


WRED: Simulation vs. Analysis

- To finish the model verification process, the individually tested blocks are combined into the final model and compared to a WRED analytic result similar to the previous example
- A larger system size (10) is used so Approximation #1 will have less effect on the simulation result
- A similar approximation is used to place an upper bound on the analysis – all packets will be dropped using the probability seen by the ***last*** packet in the burst

$$\min_{th1} = 4 \quad \max_{th1} = 9$$
$$\min_{th2} = 1 \quad \max_{th2} = 9$$
$$\max_{p1} = \max_{p2} = 1$$

WRED: Simulation vs. Analysis



Burst Size Class 1	3
Burst Size Class 2	3
System Size	10
Queue Size	9
Burst Arrival Rate	1/class
Mean Packet Length	1000 bits
Link Capacity	10000 bits
Offered Load	0.6
Total Simulation Time	7000s
Run-in Time	2000s
Number of Runs	5

WRED Model Validation Parameters

As expected, the simulation dropping probability is slightly higher than the lower bound found using Approximation #1.

The simulation results are within the range of values expected using analysis

Simulation – RED (No Class Distinction)

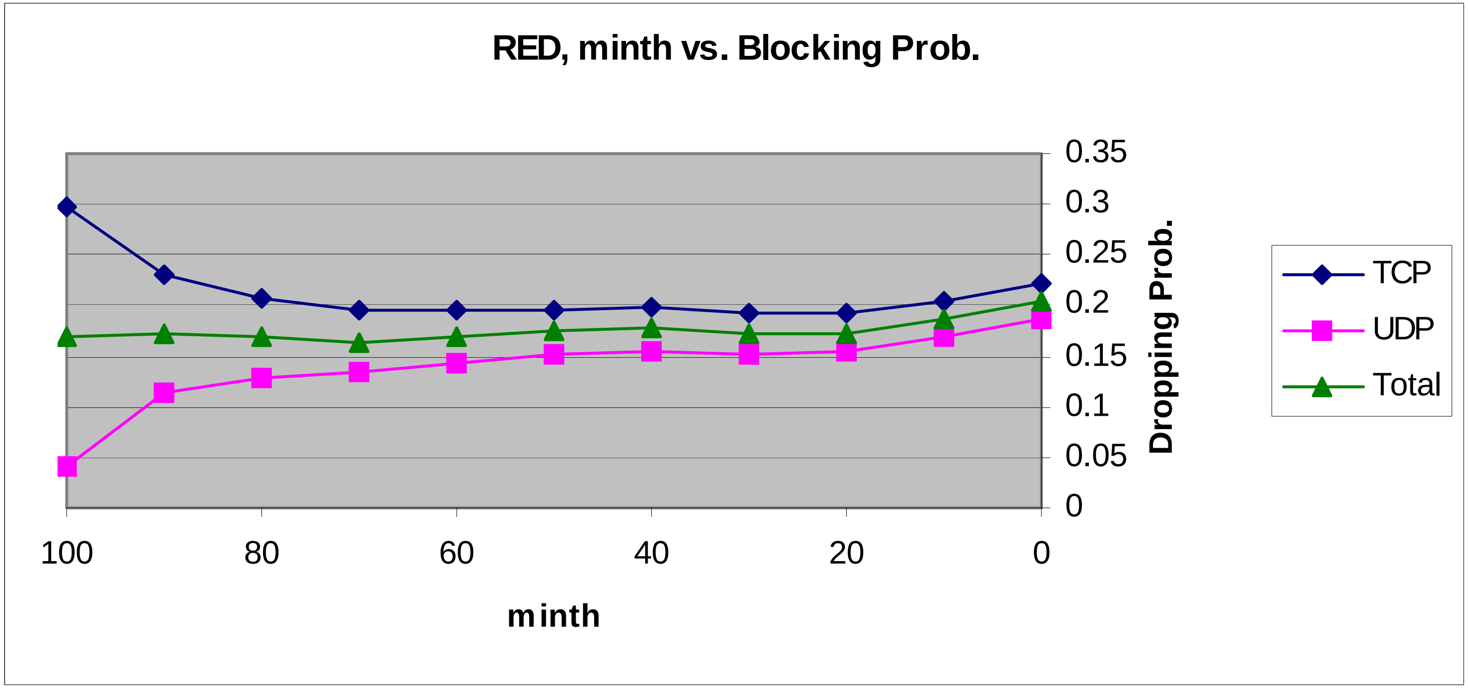
An equal load of smooth and bursty traffic is sent to identical RED blocks

Burst Size Class 1	10 pkts/burst
Burst Arrival Rate	0.6 bursts/s
Packet Arrival Rate Class 2	6 pkt/s
Queue Size	100
maxth1, maxth2	100
Mean Packet Length Class 1 and 2	1000 bits
Link Capacity	10000 bits
Offered Load	1.2
Total Simulation Time	7000s
Run-in Time	2000s
Number of Runs	5

Parameters for RED Simulation

Simulation –
RED (No Class Distinction)

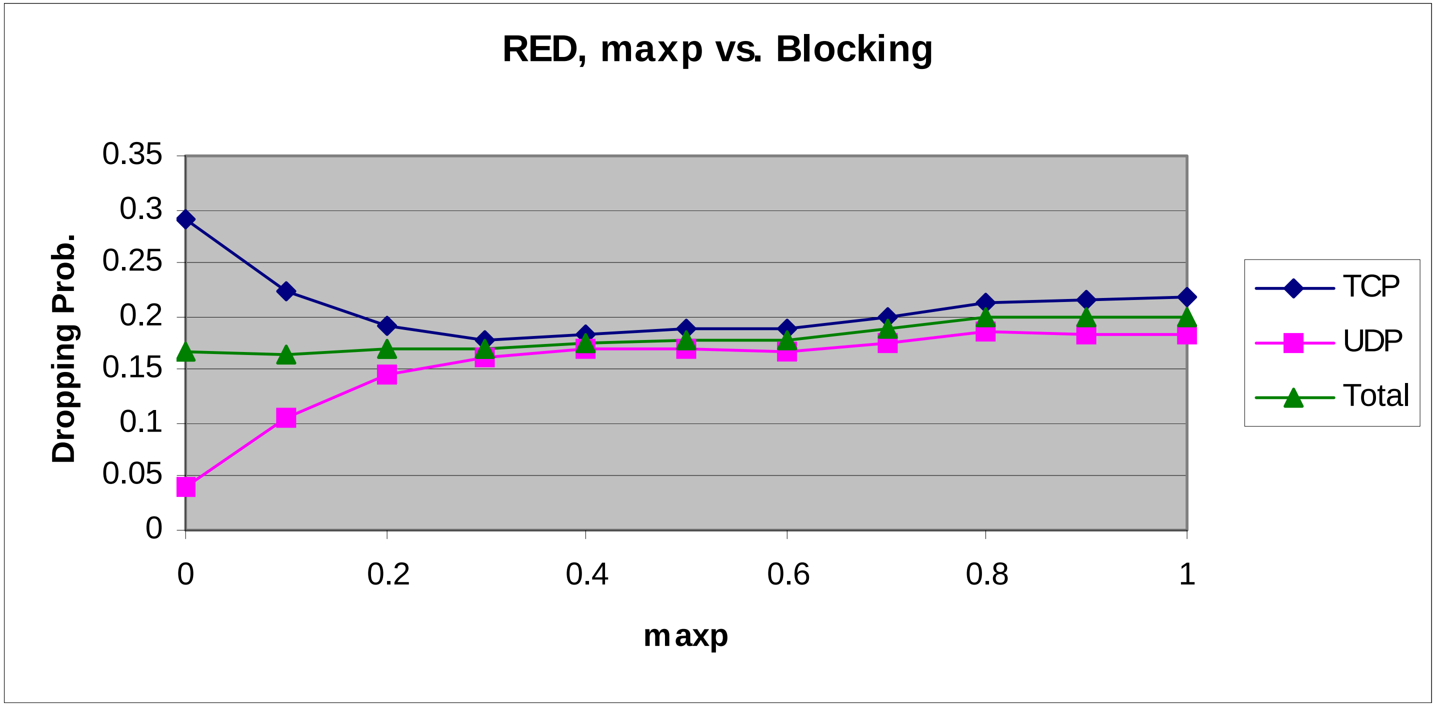
RED aggressiveness is varied by decreasing the minimum threshold for packet discard



Dropping Probability vs. Minimum Drop Threshold, ρ=1.2, max_p=1

Simulation – RED (No Class Distinction)

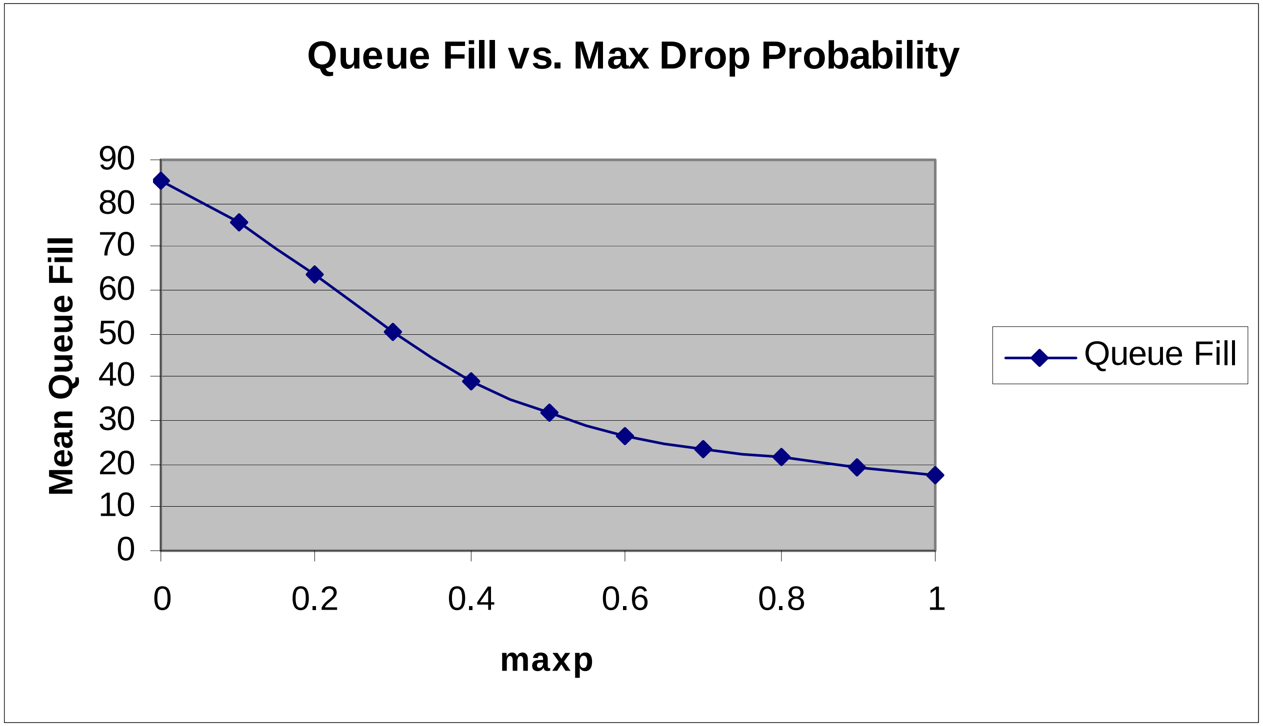
RED aggressiveness is varied by increasing the maximum probability of packet discard



Dropping Probability vs. Maximum Dropping Probability, ρ=1.2, min_{th}=0

Simulation – RED (No Class Distinction)

RED balances packet dropping by lowering the average queue fill



Queue Fill vs. Maximum Dropping Probability, ρ=1.2, min_{th}=0

Simulation – RED (No Class Distinction)

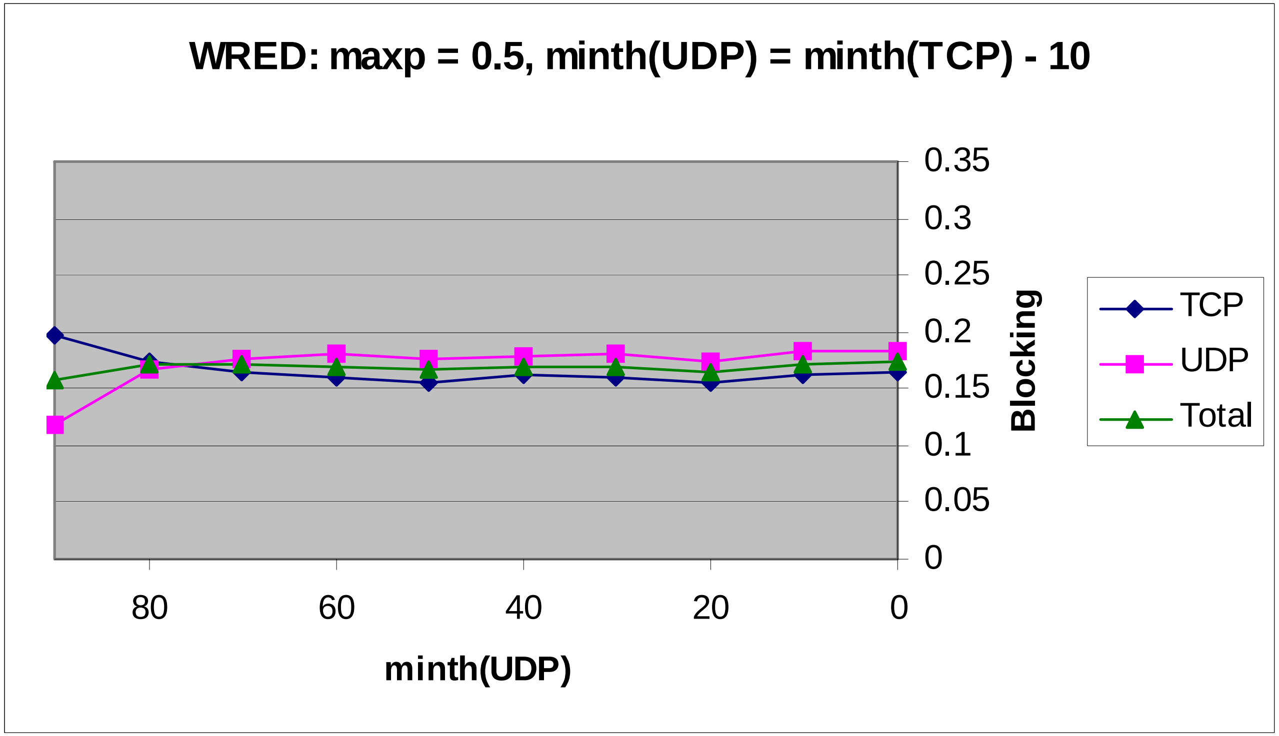
- RED does improve fairness, but further refinement could be performed by using WRED to provide class-specific RED dropping
- Simulations show that fairness increases rapidly with the introduction of RED
- Extreme RED parameters do not provide greater equality, but they do cause an undesirable overall increase in dropping probability
- With these considerations, a maximum dropping probability of 0.5 is selected for use

Simulation – Using WRED to Eliminate Bias

- Differentiated dropping parameters will be introduced by lowering the minimum discard threshold for UDP traffic relative to TCP
- Adjustment of these parameters will show that near-equal dropping probabilities are achievable through the use of WRED

Simulation – Using WRED to Eliminate Bias

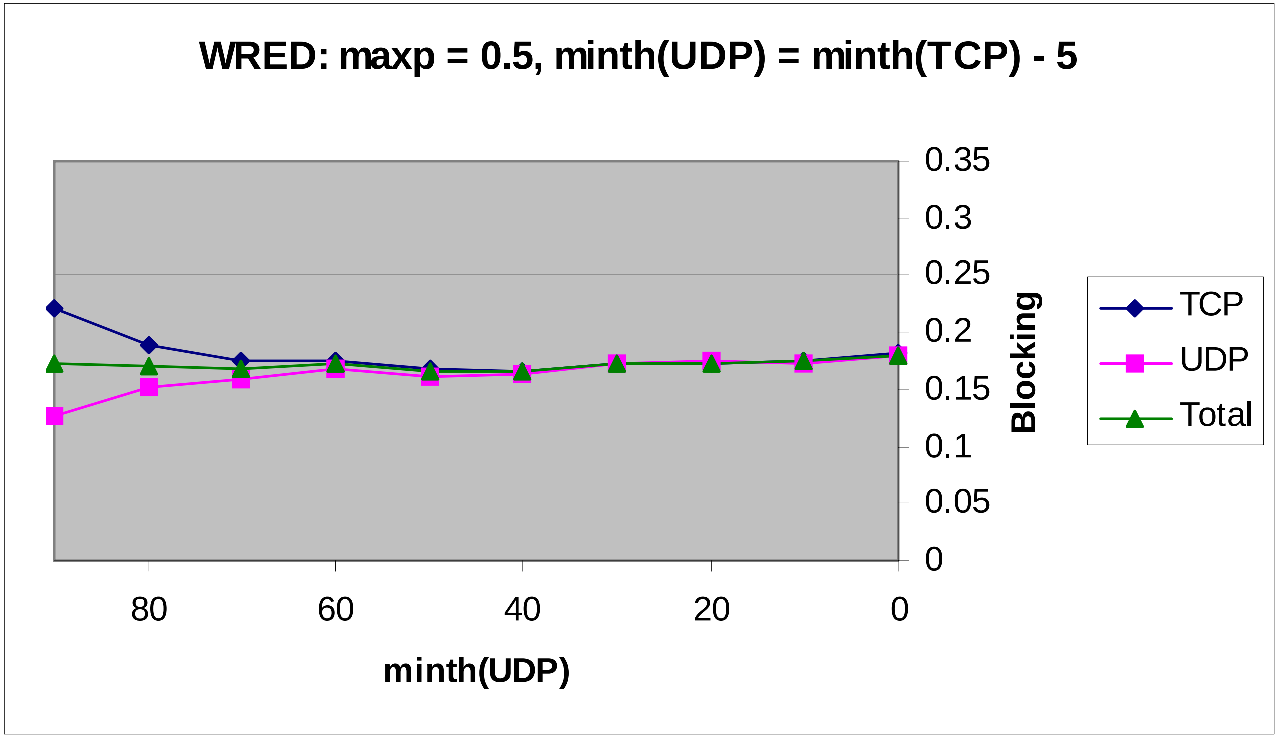
A threshold difference of ten packets is found to favor TCP traffic; the reduction of the preference is needed



WRED Dropping Probability, Five Packet Minimum Threshold Preference for TCP Traffic, maxp=0.5, ρ=1.2

Simulation – Using WRED to Eliminate Bias

A threshold preference of five packets is found to provide near-equal dropping probabilities for both classes of traffi



WRED Dropping Probability, Five Packet Minimum Threshold Preference for TCP Traffic, maxp=0.5, ρ=1.2

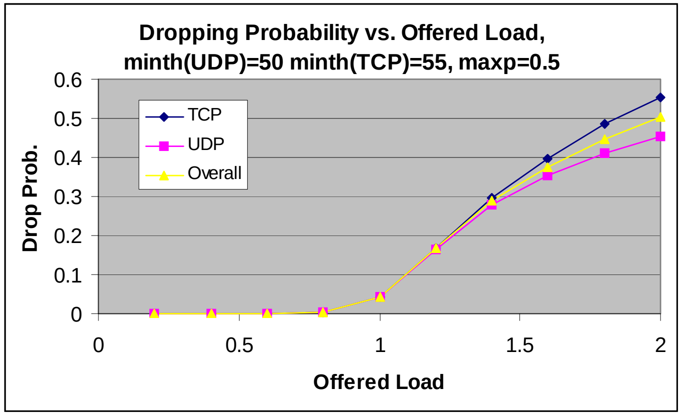
Simulation – Using WRED to Eliminate Bias

But are these parameters valid across all traffic loads?

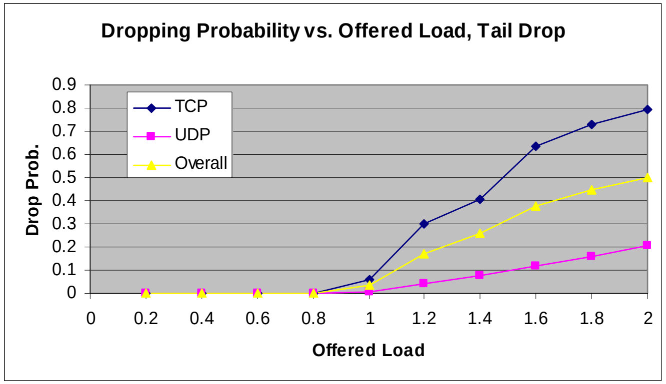
Parameters provide equality across a wide range of realistic loads

At extreme loads, WRED is unable to compensate – but dropping is severe for all traffic at $\rho > 1.5$

These parameters will be effective at equalizing dropping for most realistic loads



WRED Fairness vs. Offered Load



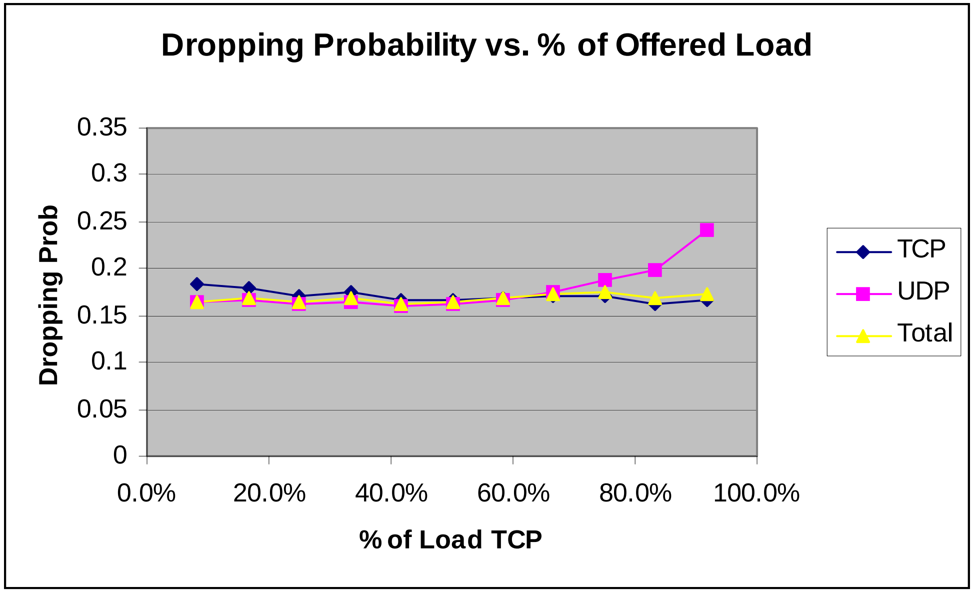
Tail Drop Fairness vs. Offered Load

Simulation – Using WRED to Eliminate Bias

How will the system perform as the traffic composition changes?

The predominant traffic class will receive slightly better dropping performance at the extremes, especially when bursty traffic is more than 70% of total traffic

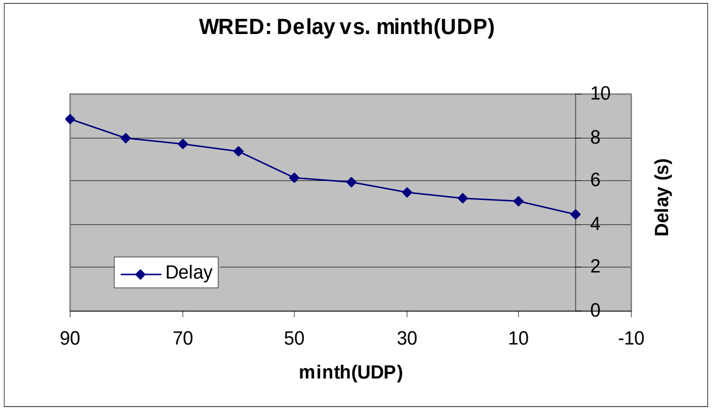
However, the traffic-weighted mean remains stable and the system is not overly sensitive to the traffic makeup



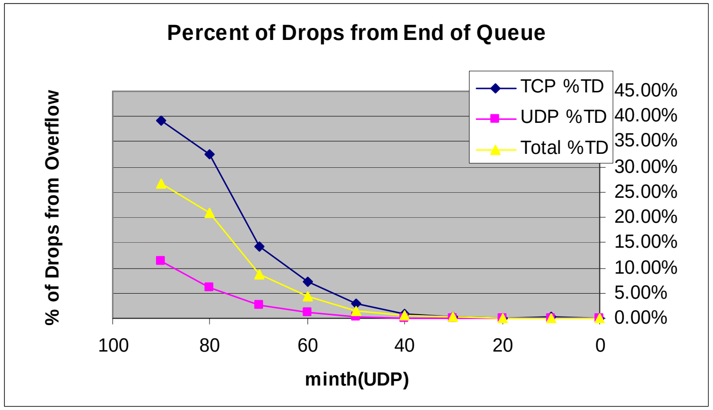
Class Dropping Probabilities vs. % Of Traffic that is TCP, minth=55/50, maxp=0.5, ρ=1.2

Simulation – Using WRED to Eliminate Bias

- Other considerations: Delay & Congestion



Packet Delay



Proportion of Drops Due to Overflow

The use of WRED decreases delay due to the lower average queue fill value

Also, fewer drops occur from the end of the queue during overflow conditions

Simulation – Using WRED for Differentiated QoS

- Another approach to WRED implementation is to allow bursty class dropping probabilities that would not be possible using simple RED

Problem Scenario:

A system must be designed that assigns high-priority bursty traffic a dropping probability that is approximately ½ of the dropping probability for the low-priority smooth traffic

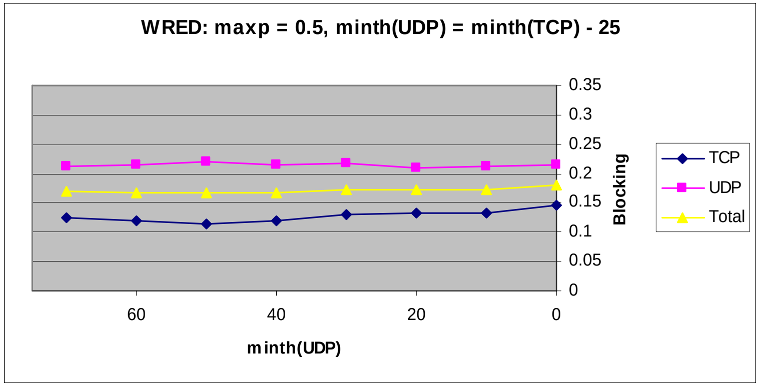
Can WRED solve this problem?

Simulation – Using WRED for Differentiated QoS

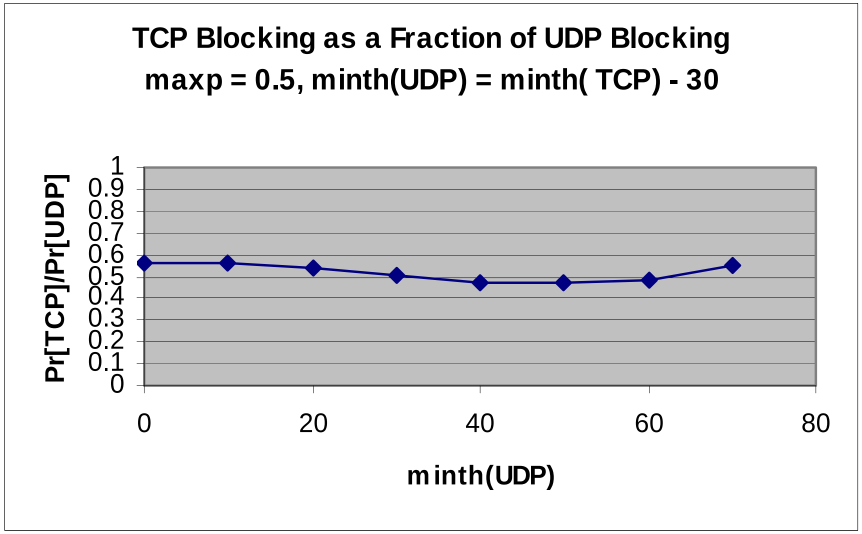
Answer: Yes*

$Min_{th}(UDP)=30$, $max_p=0.5$

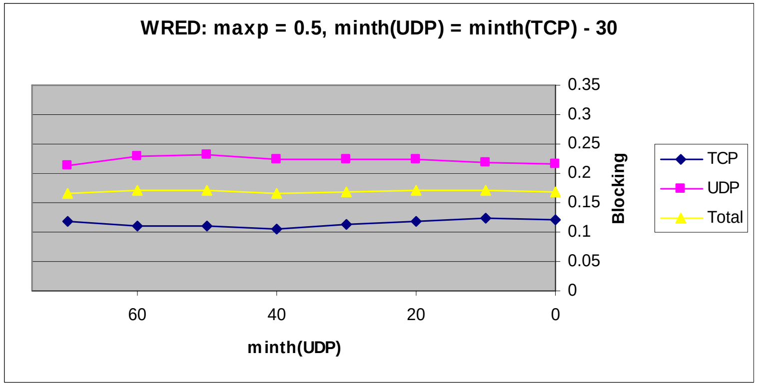
$Min_{th}(TCP)=60$



25 Packet Minimum Threshold Preference for TCP Traffic, maxp=0.5, ρ=1.2



Relative Dropping Probability maxp=0.5, ρ=1.2



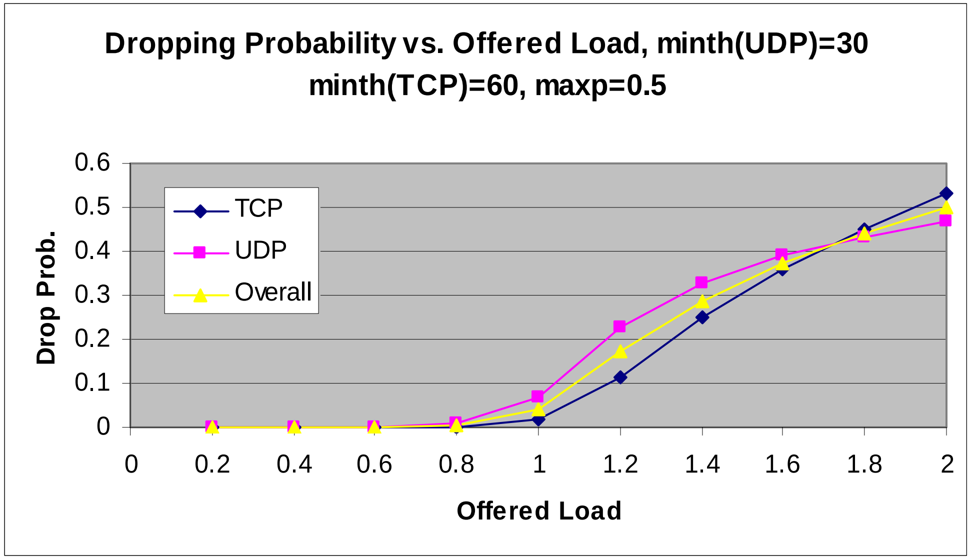
30 Packet Minimum Threshold Preference for TCP Traffic, maxp=0.5, ρ=1.2

Simulation – Using WRED for Differentiated QoS

*But, the WRED parameters only provide the desired relationship at a single load point

Once again, extreme loads will exceed WRED's ability (with these design parameters) to overcome the bias against bursty traffic

However, the system still provides priority service to bursty traffic over a wide range of loads (< 1.6)



WRED Dropping vs. Offered Load: Differentiated Dropping Probabilities

Conclusions

- WRED offers greater flexibility for network designers by allowing multiple classes of dissimilar traffic to have specified relative dropping probabilities
- However, care must be taken when considering the operating load: Aggressive differentiation of WRED parameters may make the system more sensitive to load
- Slight discrimination against a less dominant traffic stream occurs when a system is designed for equal class loads

Possible Future Research

- Analysis/Simulation of networks of WRED nodes
- WRED marking as a trigger for TCP control mechanisms
- Dynamic adjustment of WRED parameters

Acknowledgements & Questions

Thanks to:

- Dr. David Petr for his guidance, ideas, and time throughout this project and my classwork
- Dr. Victor Frost and Dr. Gary Minden for their service on the defense committee

- Questions?

References

References cited in this presentation:

[1] D. Clark, W. Lehr, et al., “Provisioning for Bursty Internet Traffic: Implications for Industry and Internet Structure.” MIT ITC Workshop on Internet Quality of Service. Nov, 1999. http://www.ana.lcs.mit.edu/papers/PDF/ISQE_112399_web.pdf

[2] Floyd, S. & Jacobson, V. (1993). Random Early Detection Gateways for Congestion Avoidance. *IEEE/ACM Transactions on Networking*. Vol. 3. 397-413.

[3] T. Bonald, M. May, et. al., “Analytic Evaluation of RED Performance.” *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*. IEEE , Volume: 3 , 26-30 March 2000
Page(s): 1415 -1424 vol.3

References

Other references used for this project:

- J. Cao, W. Cleveland, et al., “Internet Traffic Tends Toward Poisson and Independent as the Load Increases.” *Nonlinear Estimation and Classification*. http://cm.bell-labs.com/cm/ms/departments/sia/doc/lrd2poisson.pdf
- P. Kuusela and J.T. Virtamo, “Modeling RED with Two Traffic Classes.” http://keskus.tct.hut.fi/tutkimus/com2/publ/ntsRED2cl.pdf
- Feng, W.-C.; Kandlur, D.D.; Saha, D.; Shin, K.G. A self-configuring RED gateway *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings.* IEEE , Volume: 3 , 21-25 March 1999 Page(s): 1320 -1328 vol.3
- W. Leland, M. Taqqu, W. Willinger, and D. Wilson, “On the Self-Similar Nature of Ethernet Traffic”, *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp 1-15, February 1994.
- Extend Model Software & Documentation, http://www.imaginethatinc.com, 2002.
- D. Petr, University of Kansas EECS 963 Class Notes. (Summer 2003)
- Firoiu, V.; Borden, M**. “**A study of active queue management for congestion control.” *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings.* IEEE , Volume: 3 , 26-30 March 2000 Page(s): 1435 -1444 vol.3