Congestion Control
Part 2

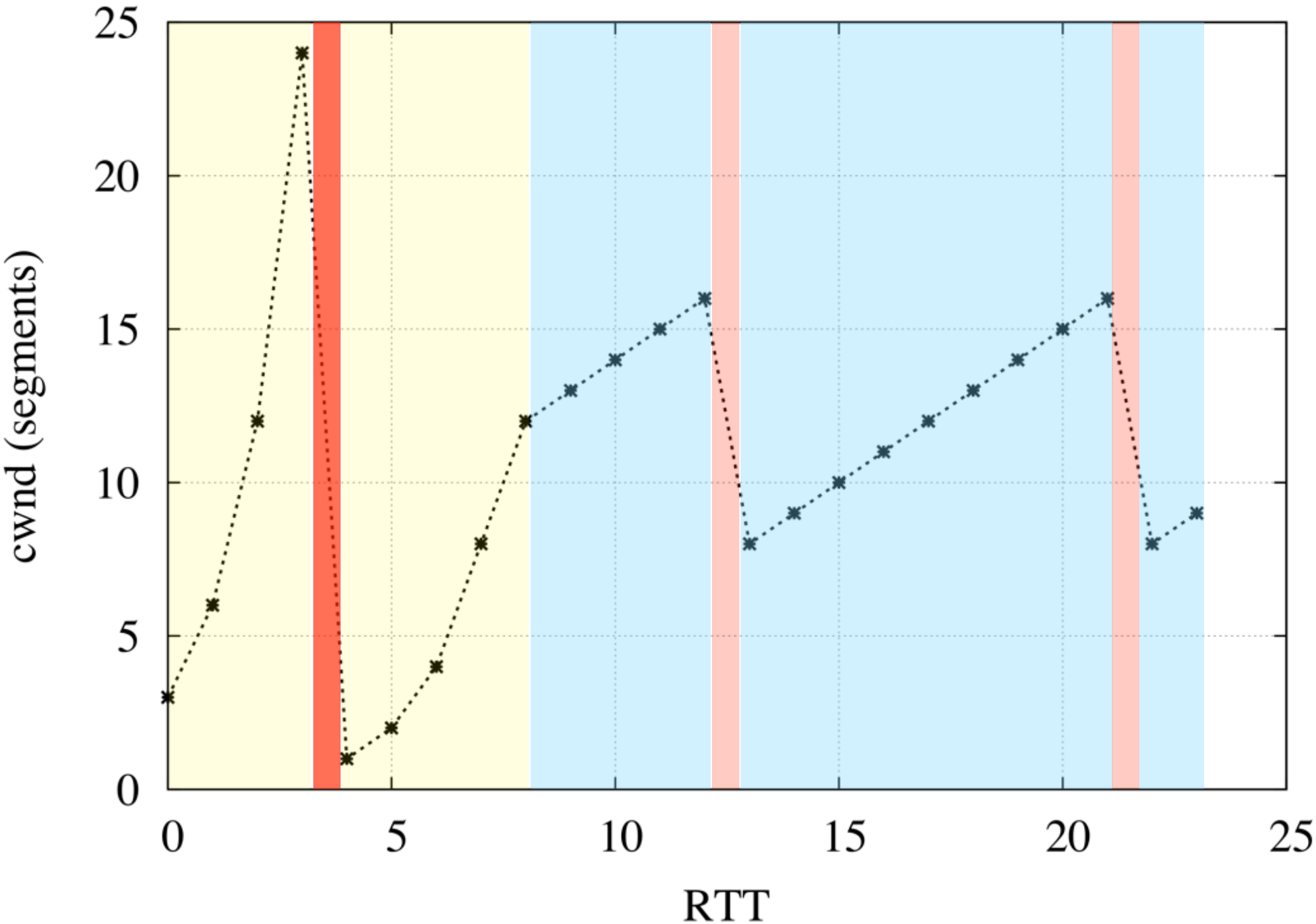
Mark Allman
mallman@case.edu

EECS 325/425
Fall 2018

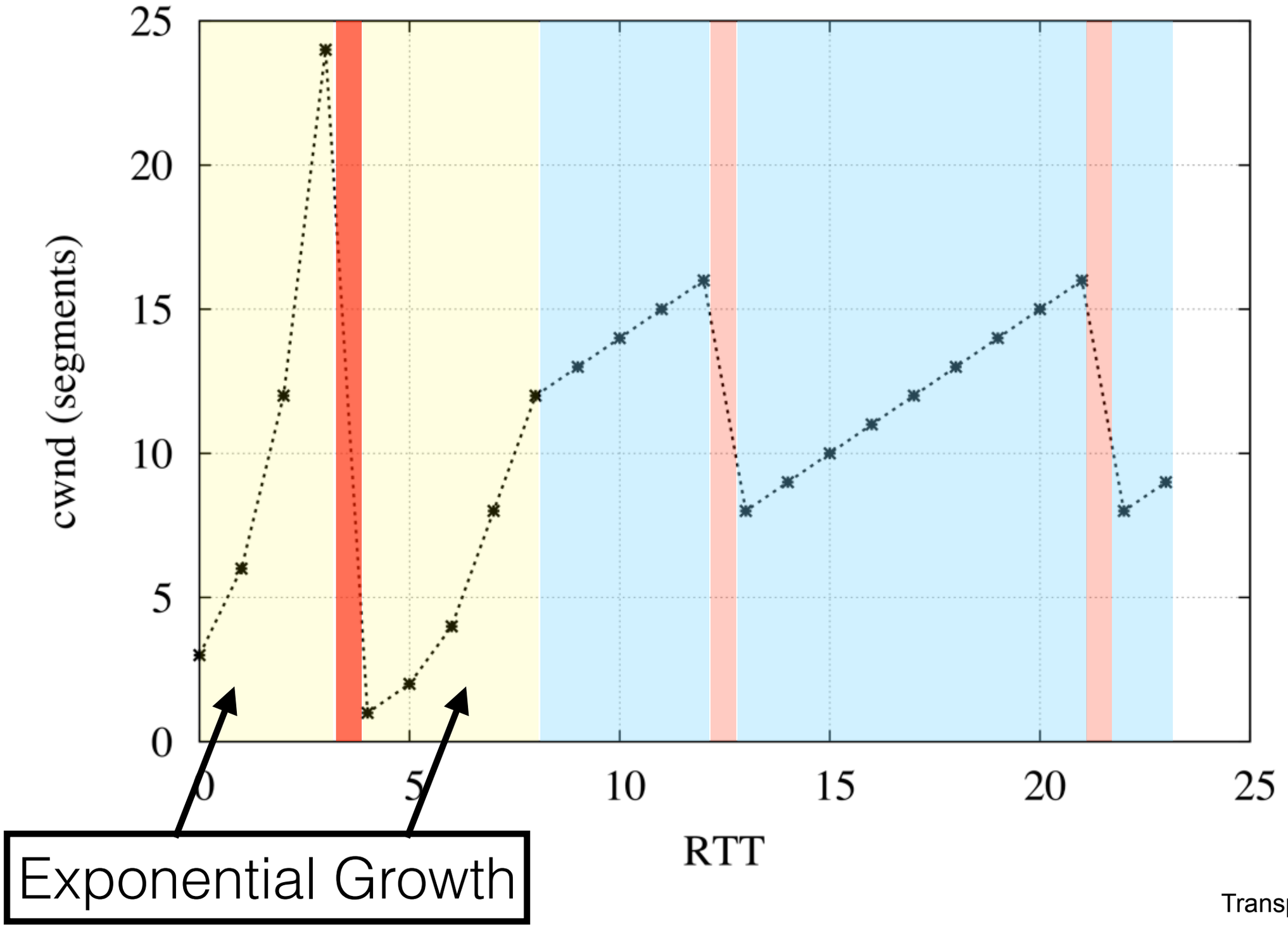
These slides are more-or-less directly from the slide set developed by Jim Kurose and Keith Ross for their book "Computer Networking: A Top Down Approach, 5th edition".

The slides have been lightly adapted for Mark Allman's EECS 325/425 Computer Networks class at Case Western Reserve University.

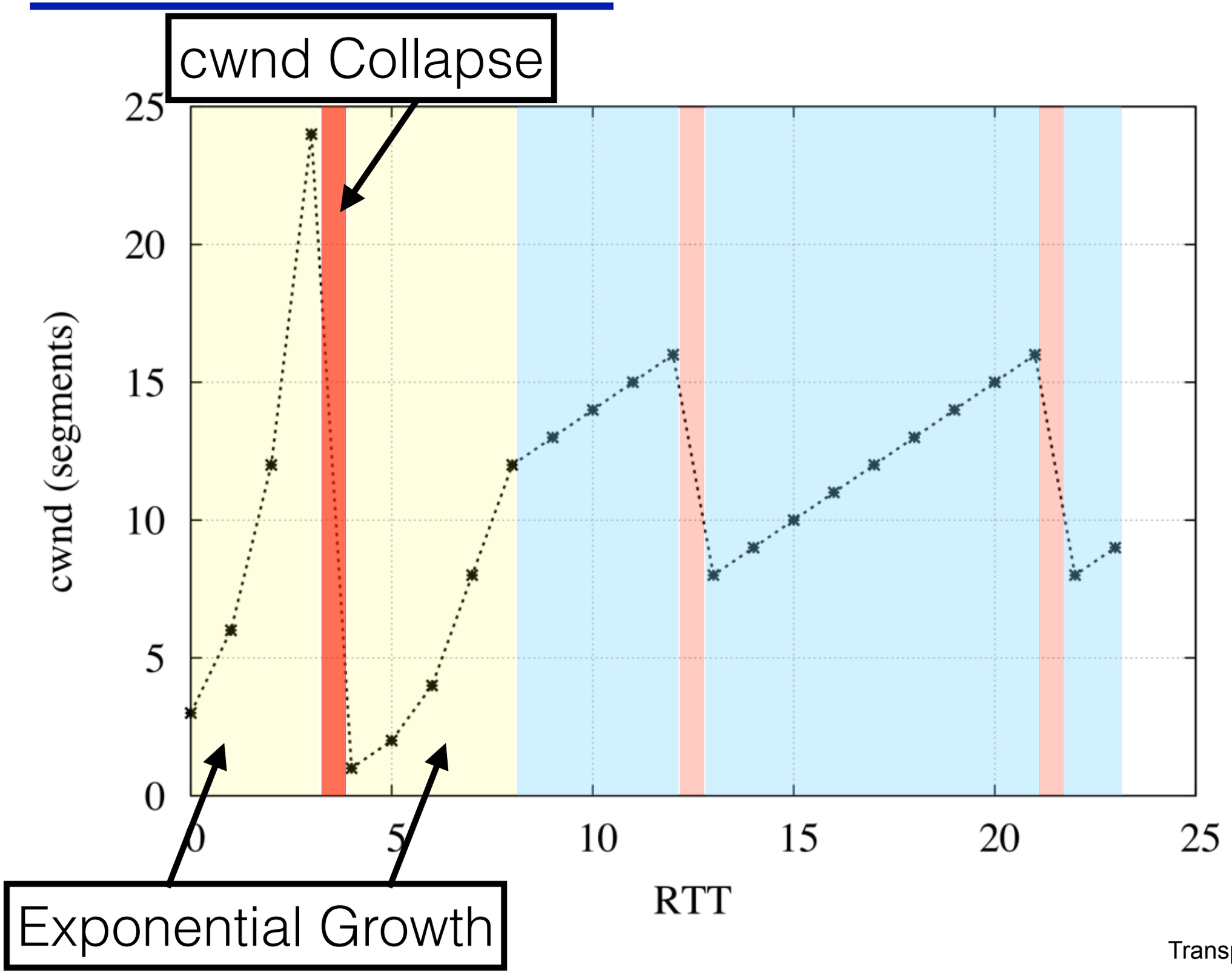
cwnd Evolution



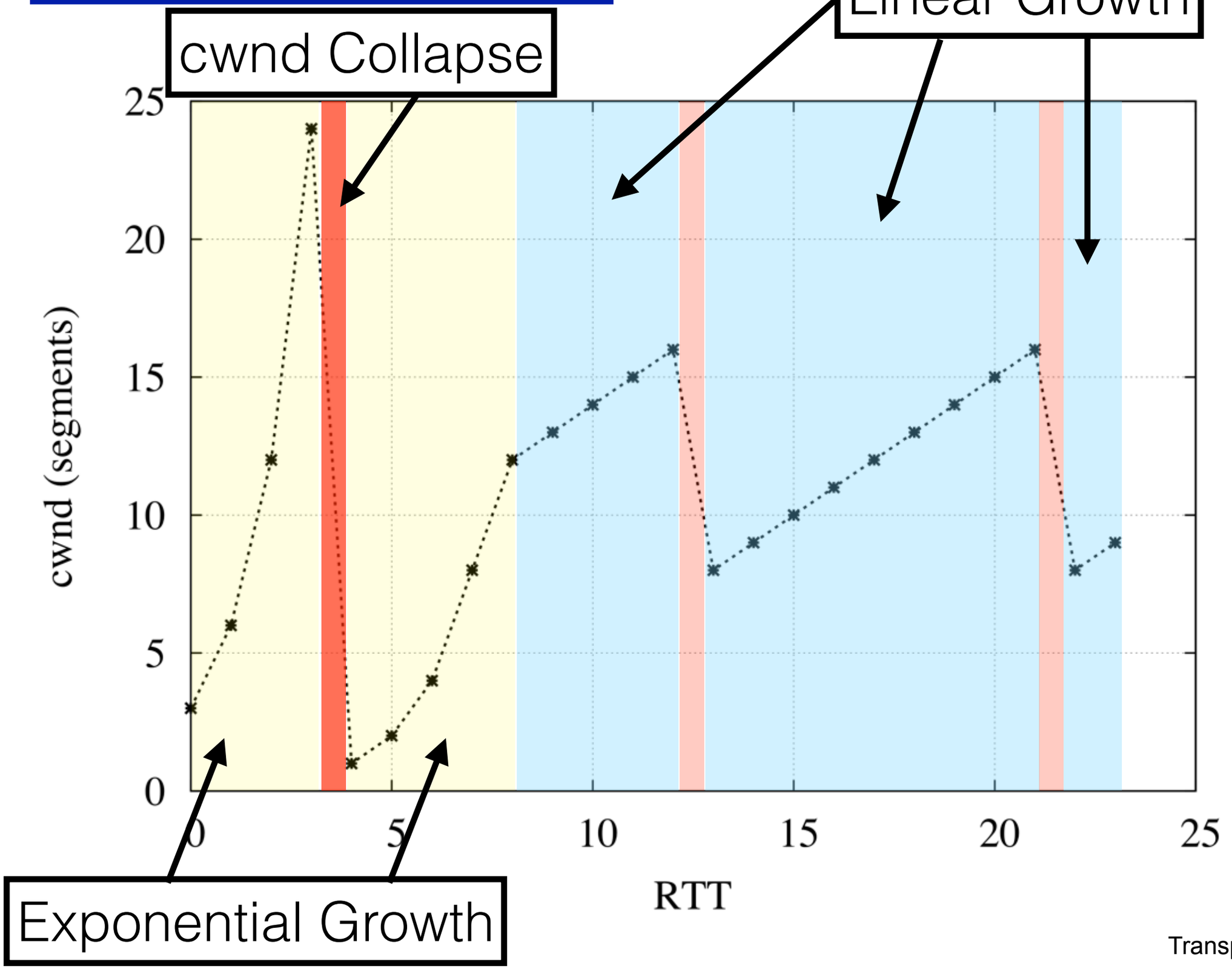
cwnd Evolution



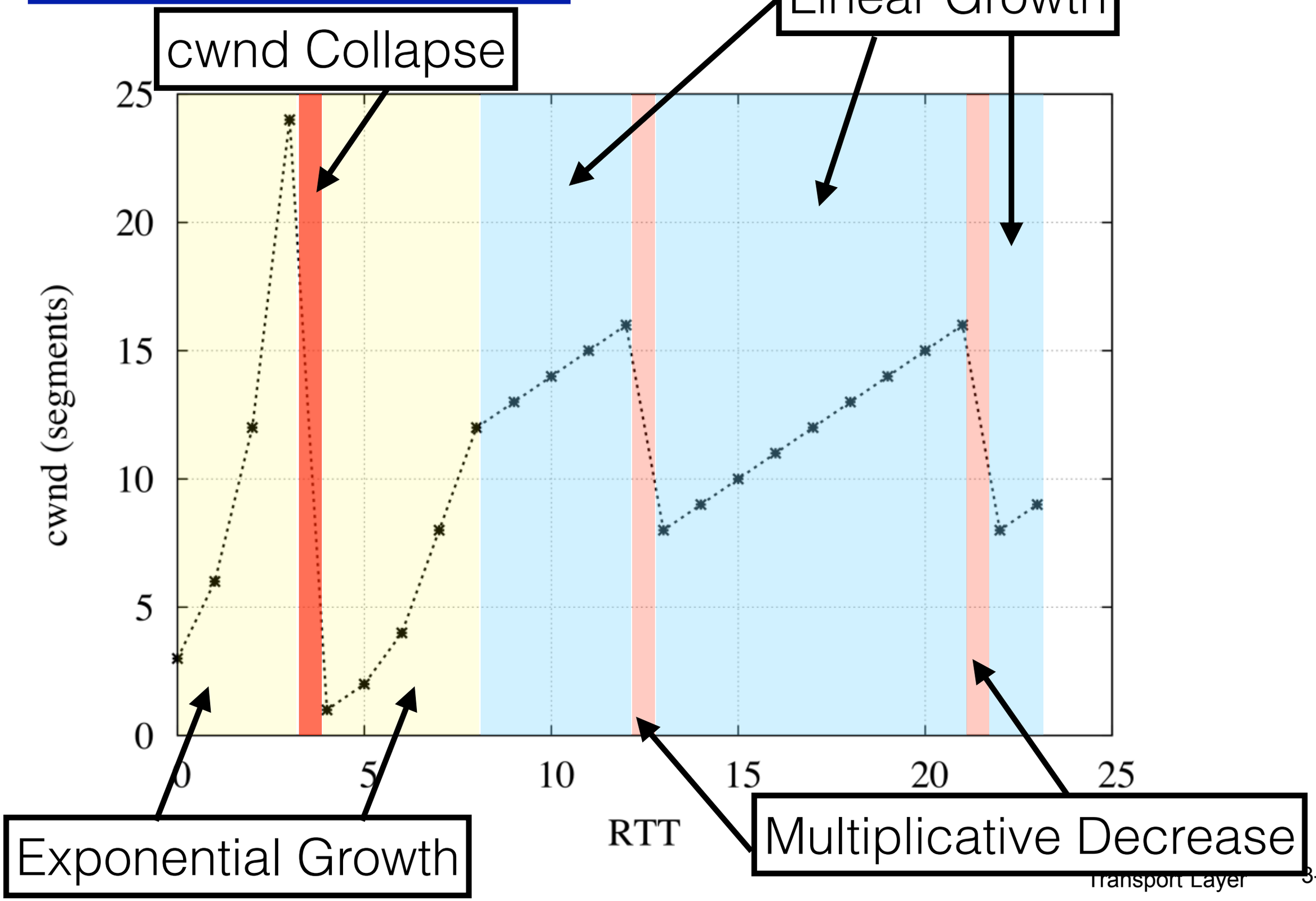
cwnd Evolution



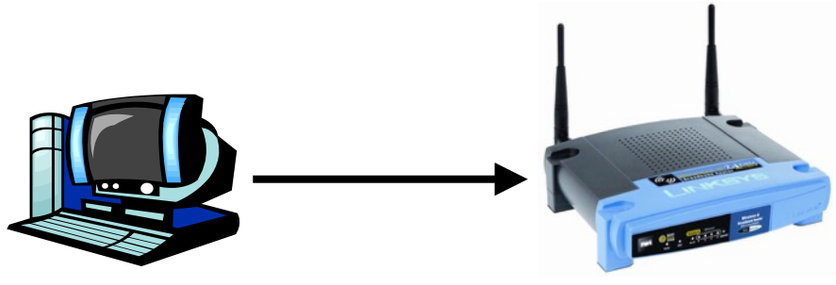
cwnd Evolution



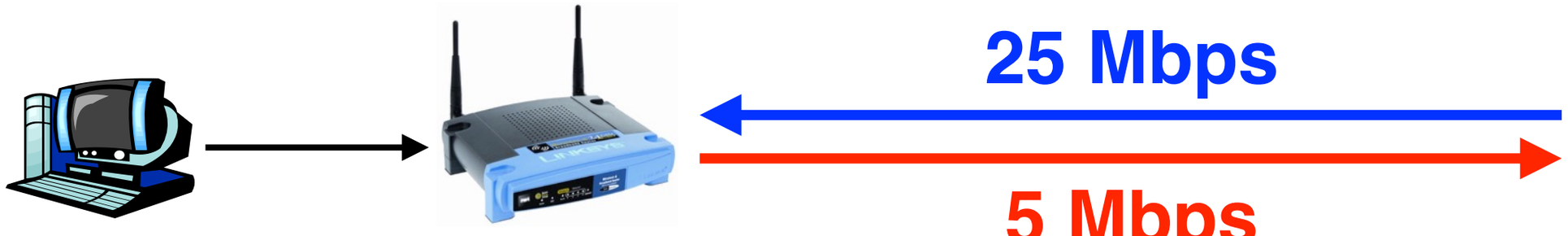
cwnd Evolution



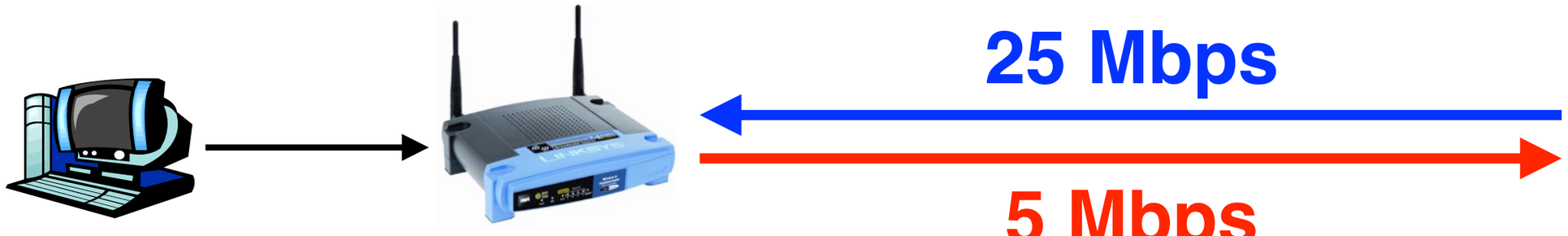
Asymmetry & CC



Asymmetry & CC

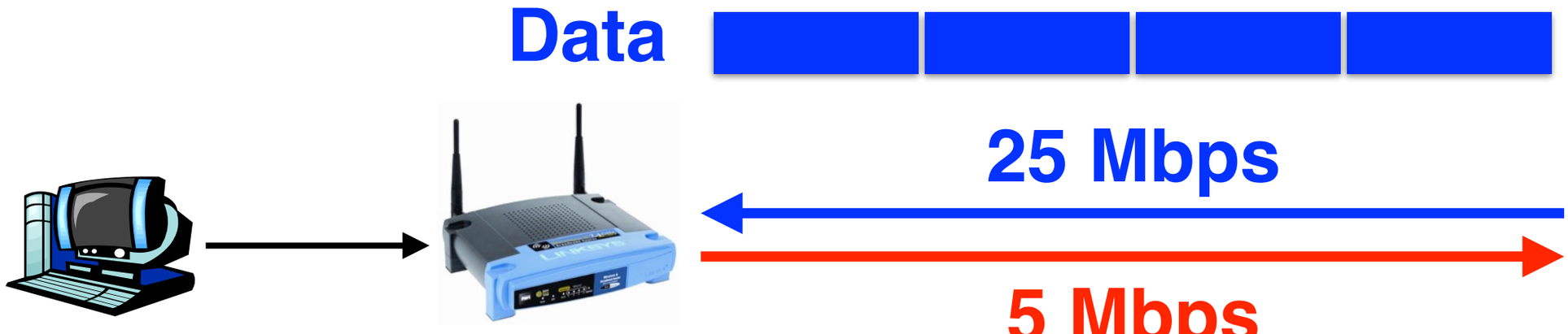


Asymmetry & CC



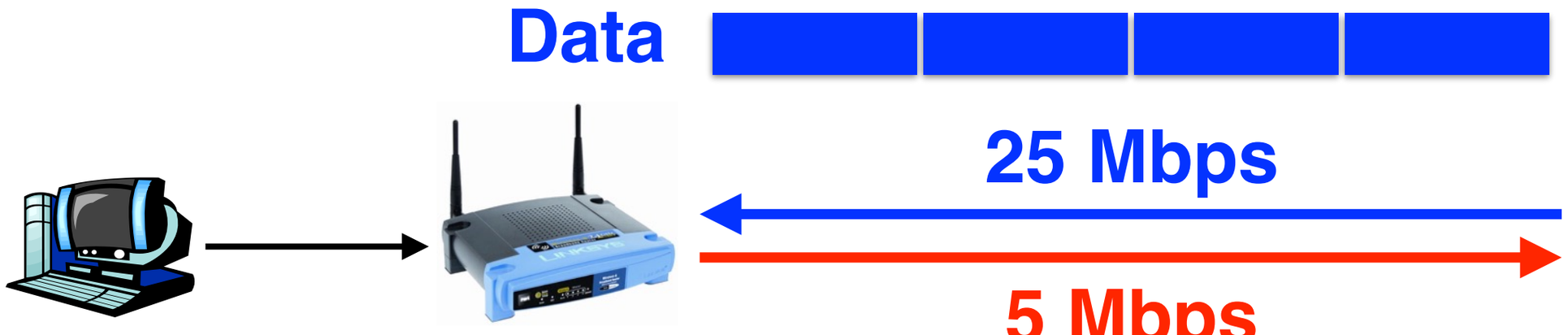

```
Net Asym = 25 / 5 = 5
```

Asymmetry & CC



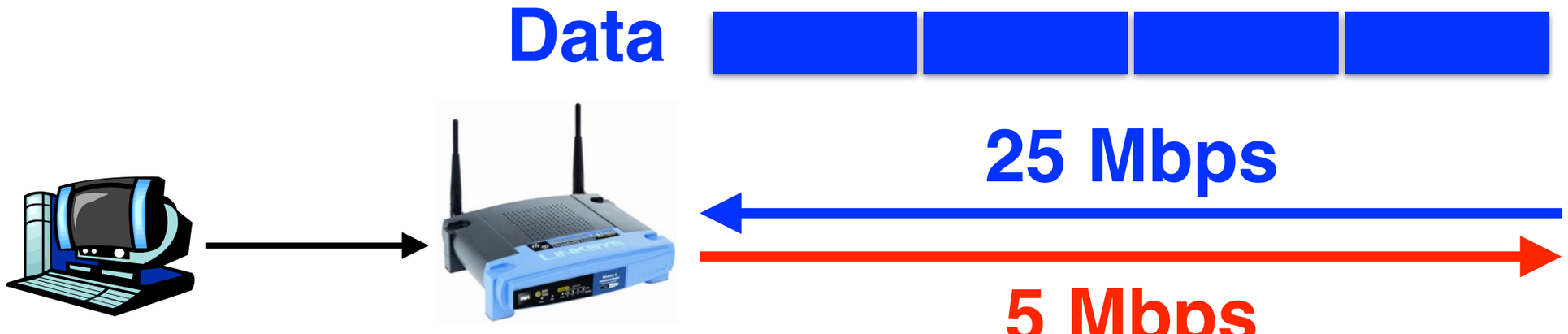

```
Net Asym = 25 / 5 = 5
```

Asymmetry & CC



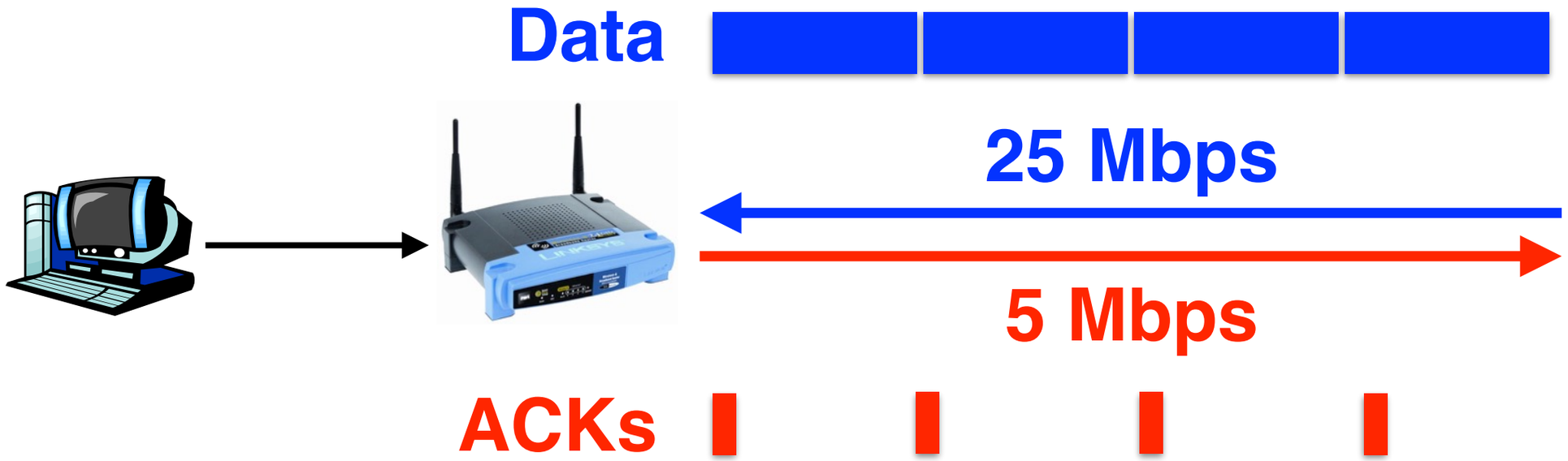

```
Net Asym = 25 / 5 = 5
```

Asymmetry & CC




```
Net Asym = 25 / 5 = 5
TCP Asym = 1518 / 58 ≈ 26
```

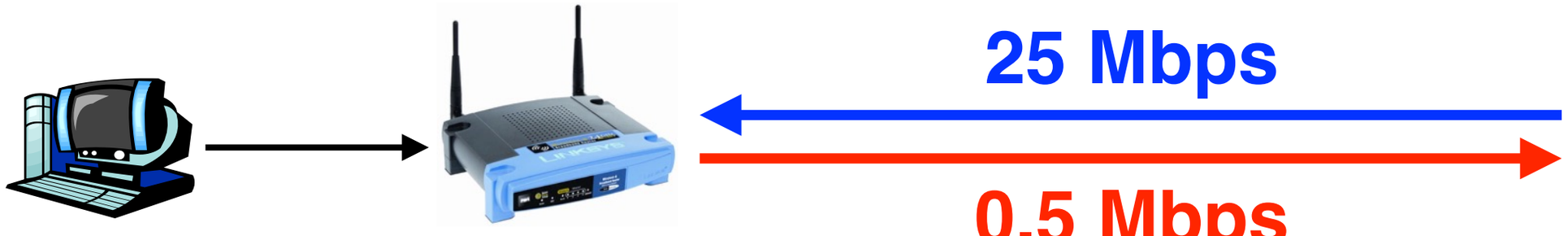
Asymmetry & CC



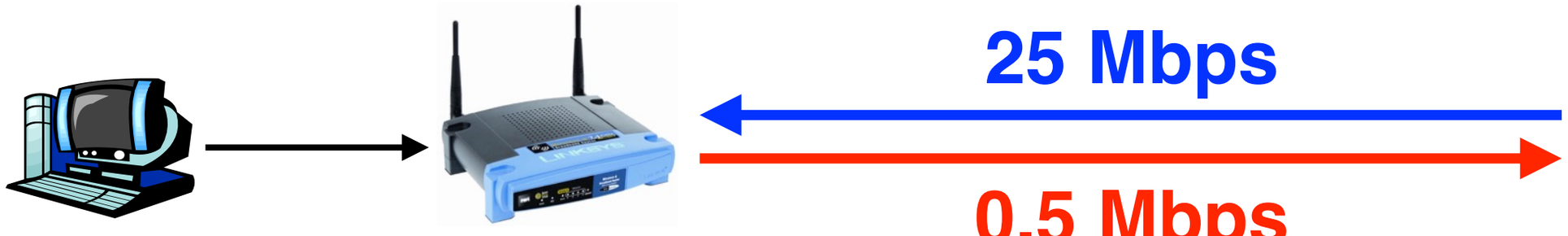

```
Net Asym = 25 / 5 = 5
TCP Asym = 1518 / 58 ≈ 26

OK as long as TCP Asym > Net Asym
```

Asymmetry & CC



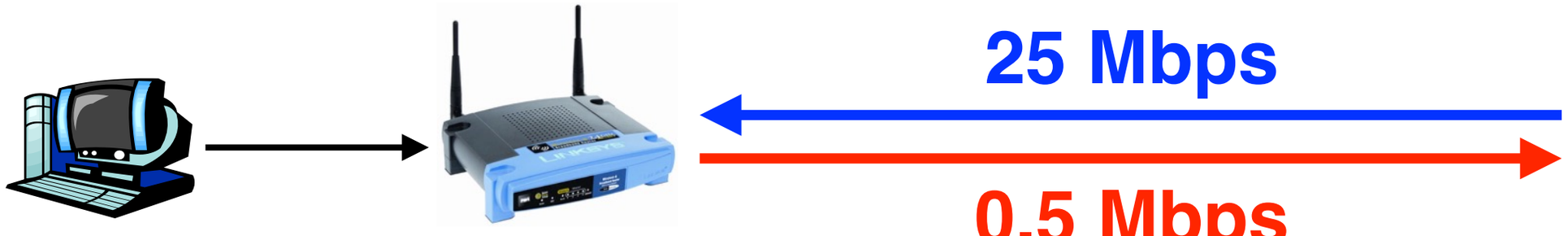
Asymmetry & CC




```
Net Asym = 25 / 0.5 = 50
TCP Asym = 1518 / 58 ≈ 26

OK as long as TCP Asym > Net Asym
```

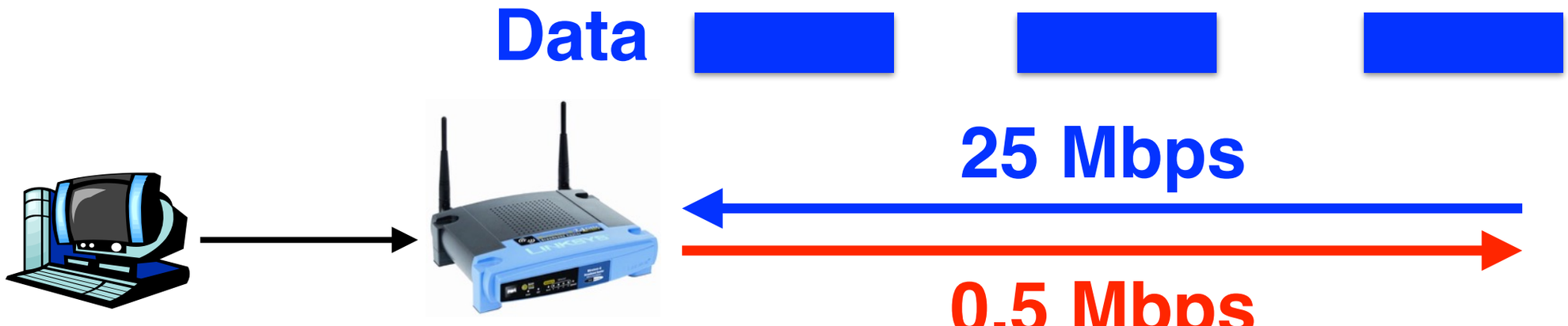
Asymmetry & CC




```
Net Asym = 25 / 0.5 = 50
TCP Asym = 1518 / 58 ≈ 26

OK as long as TCP Asym > Net Asym
```

Asymmetry & CC



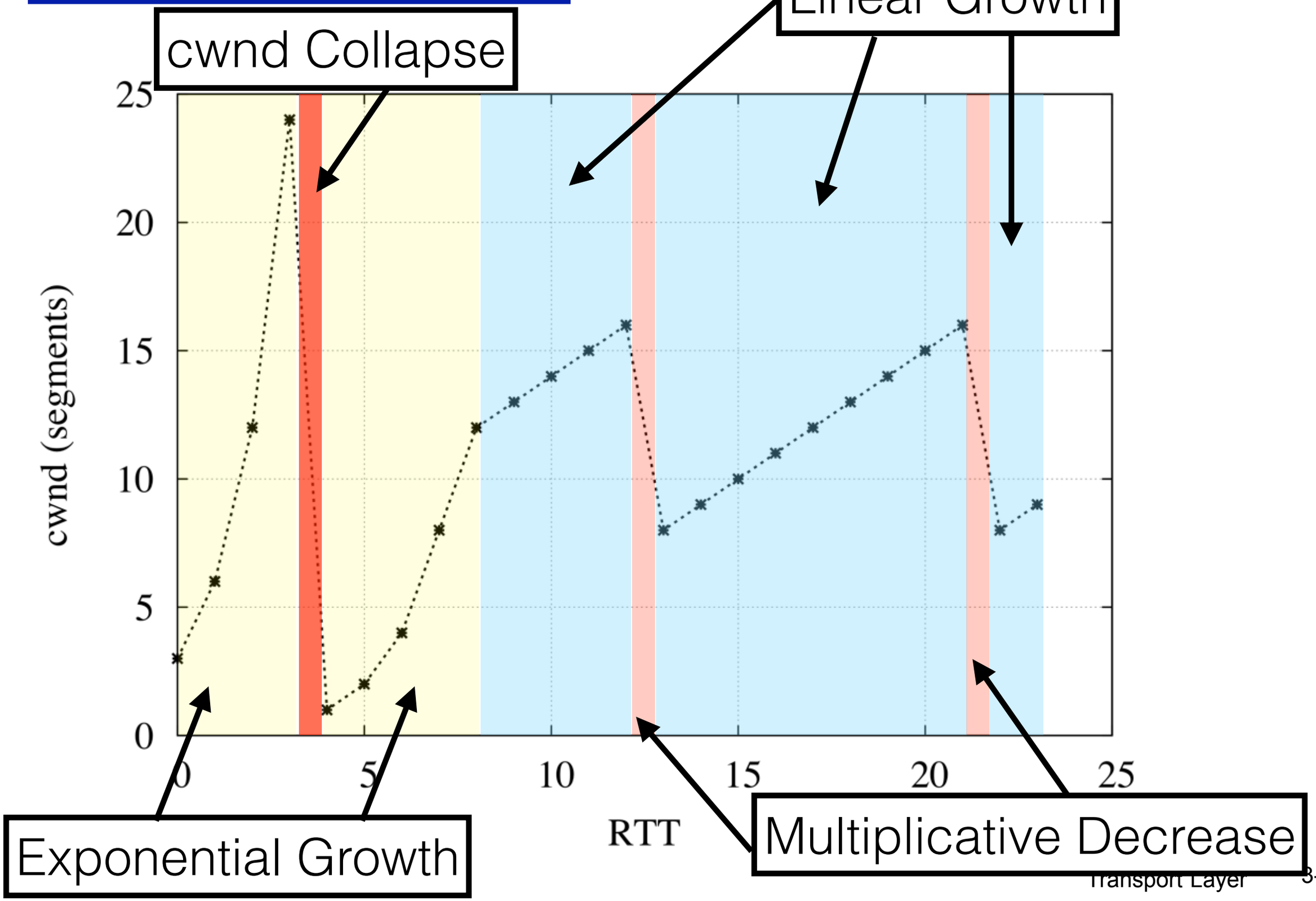

```
Net Asym = 25 / 0.5 = 50
TCP Asym = 1518 / 58 ≈ 26

OK as long as TCP Asym > Net Asym
```

Asymmetry & CC

- What to do in highly asymmetric networks?
  - dynamically set the rate of ACKs

cwnd Evolution



Performance

- We know that network performance is *bounded* by a simple equation:

$$\texttt{window} = \texttt{bw} \times \texttt{delay}$$

$$\texttt{bw} = \texttt{window} \div \texttt{delay}$$

- But, the congestion window is dynamic ….

TCP Performance

- Refine:

  `rate ≈ (1.22 × MSS) ÷`
  `(RTT × sqrt (L))`

  - example: 1500 byte segments, 100ms RTT, want 10 Gbps throughput
  - $L = 2 \cdot 10^{-10}$ Wow – a very small loss rate!

TCP Performance

$$rate \approx \frac{1.22 \times MSS}{RTT \times \sqrt{L}}$$

TCP Performance

$$rate \approx \frac{1.22 \times MSS}{RTT \times \sqrt{L}}$$

- Shorter RTTs get preference

TCP Performance

$$rate \approx \frac{1.22 \times MSS}{RTT \times \sqrt{L}}$$

- Shorter RTTs get preference
  - consider 2 connections: with RTTs of 10msec and 50msec, but all else the same

TCP Performance

$$rate \approx \frac{1.22 \times MSS}{RTT \times \sqrt{L}}$$

- Shorter RTTs get preference
  - consider 2 connections: with RTTs of 10msec and 50msec, but all else the same
    - the connection with a 10msec RTT will get 5x the performance of the other connection

TCP Performance

$$rate \approx \frac{1.22 \times MSS}{RTT \times \sqrt{L}}$$

- Shorter RTTs get preference
  - consider 2 connections: with RTTs of 10msec and 50msec, but all else the same
    - the connection with a 10msec RTT will get 5x the performance of the other connection
- TCP is fair as long as the MSS, RTT and L are the same

Endpoint CC

- Why do we use endpoint CC?
  - What are the pros? Cons?
- Should we involve the network?
  - Why or why not?

Approaches towards congestion control

Two broad approaches towards congestion control:

end-end congestion control:

- no explicit feedback from network
- congestion inferred from end-system observed loss, delay
- approach taken by TCP

network-assisted congestion control:

- routers provide feedback to end systems
  - single bit indicating congestion (SNA, DECbit, TCP/IP ECN, ATM)
  - explicit rate sender should send at

The Network Side of CC

- Active queuing
  - e.g., RED
- Fair queuing
- Traffic shaping

Active Queuing

Active Queuing

- Drop segments *before* the queue fills to invoke endpoint CC

Active Queuing

- Drop segments *before* the queue fills to invoke endpoint CC
- E.g., Random Early Detection (RED)
  - (lots of others)

Active Queuing

- Drop segments *before* the queue fills to invoke endpoint CC
- E.g., Random Early Detection (RED)
  - (lots of others)
- Goal: the network works with endpoints, but *implicitly*
  - (a little explicitness has been added …)

RED

RED

- Track *average* queue occupancy, *q*

RED

- Track *average* queue occupancy, *q*
- Set minimum and maximum thresholds: min^{th} & max^{th}

RED

- Track *average* queue occupancy, *q*
- Set minimum and maximum thresholds: min^{th} & max^{th}
- When $q < min^{th}$, forward all packets

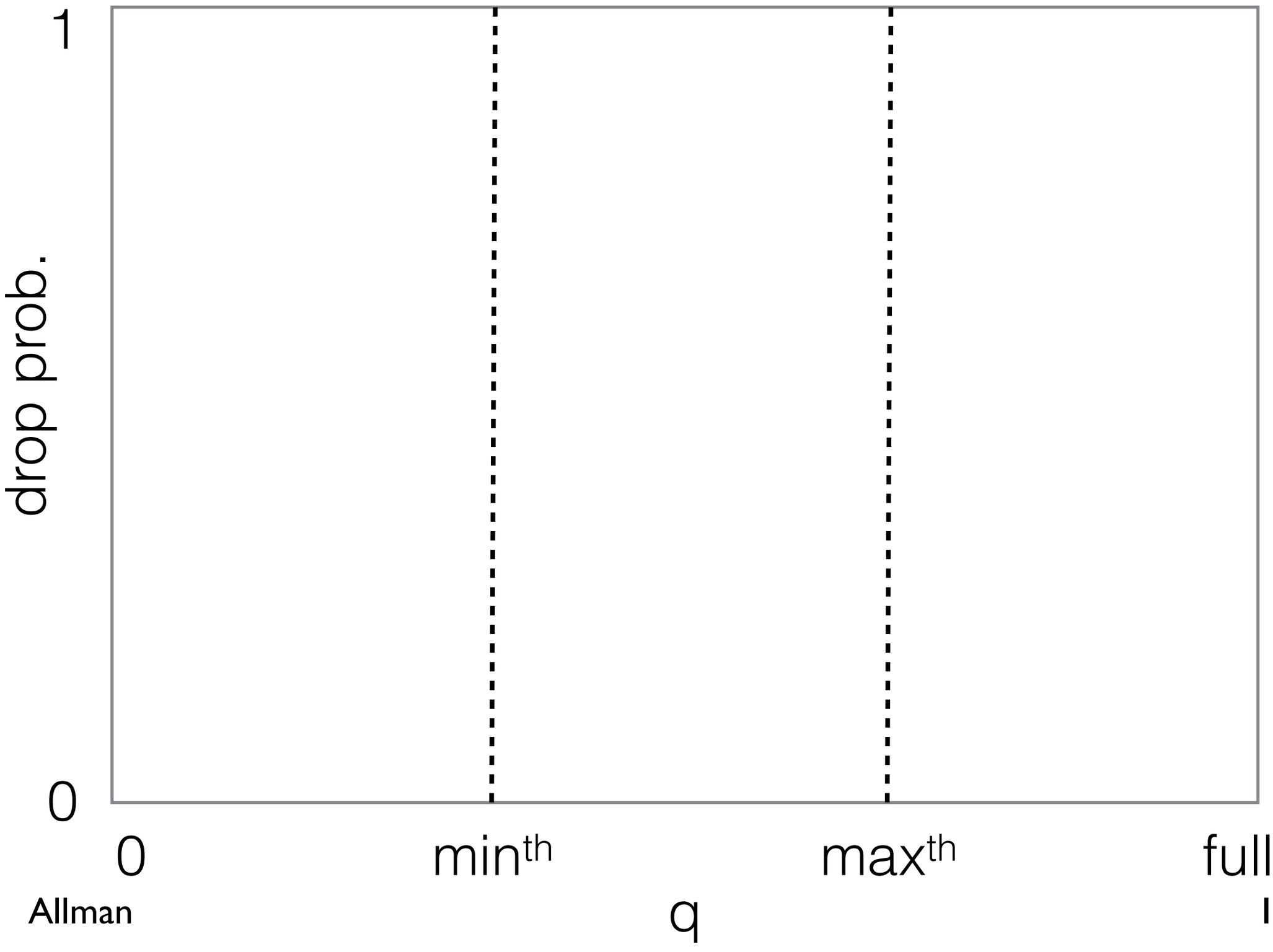
RED

- Track *average* queue occupancy, *q*
- Set minimum and maximum thresholds: min^{th} & max^{th}
  - When $q < min^{th}$, forward all packets
  - When q > max^{th}, drop all packets

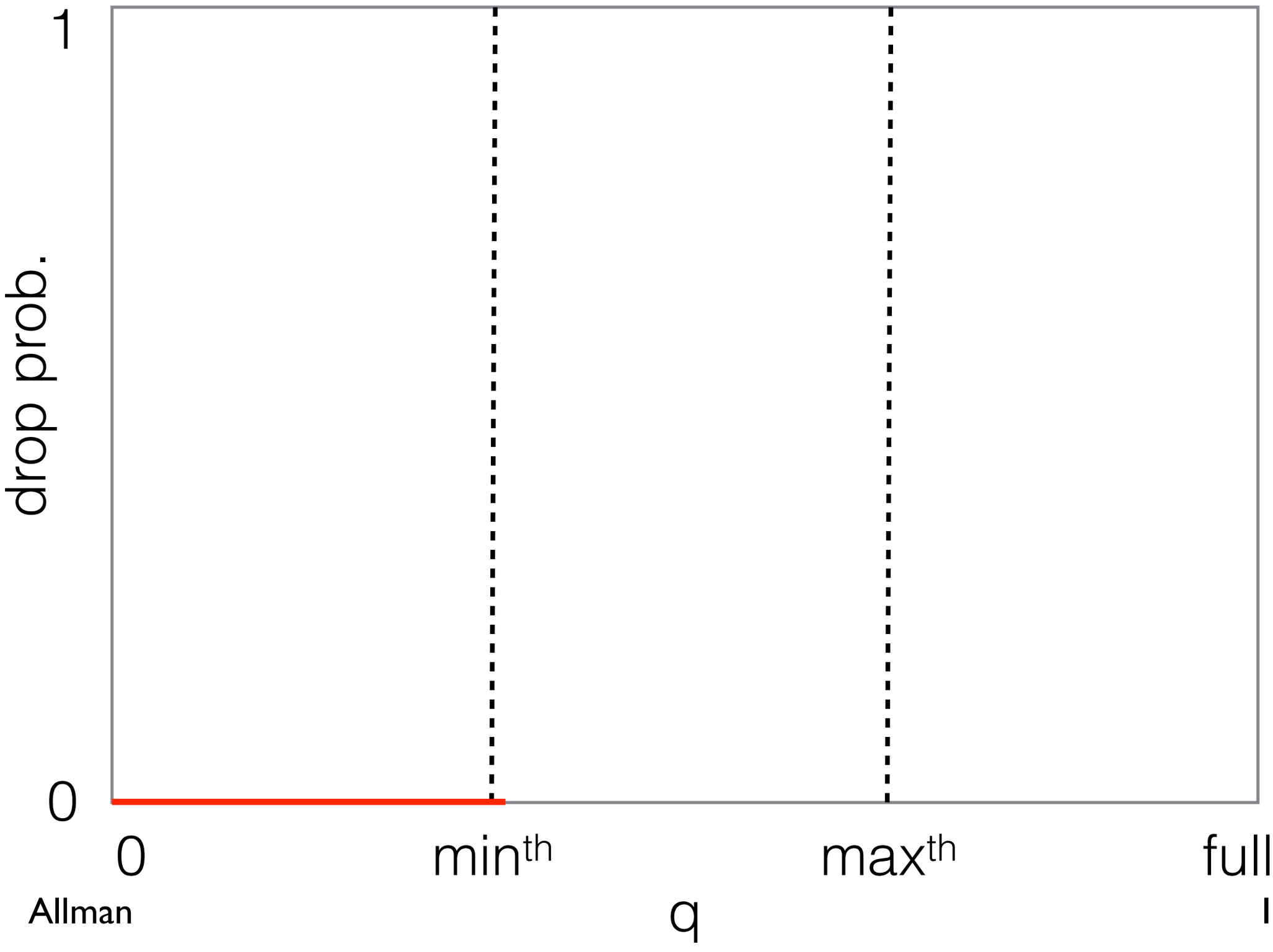
RED

- Track *average* queue occupancy, *q*
- Set minimum and maximum thresholds: min^{th} & max^{th}
  - When $q < min^{th}$, forward all packets
  - When $q > max^{th}$, drop all packets
  - When $min^{th} \leq q \leq max^{th}$, drop probabilistically

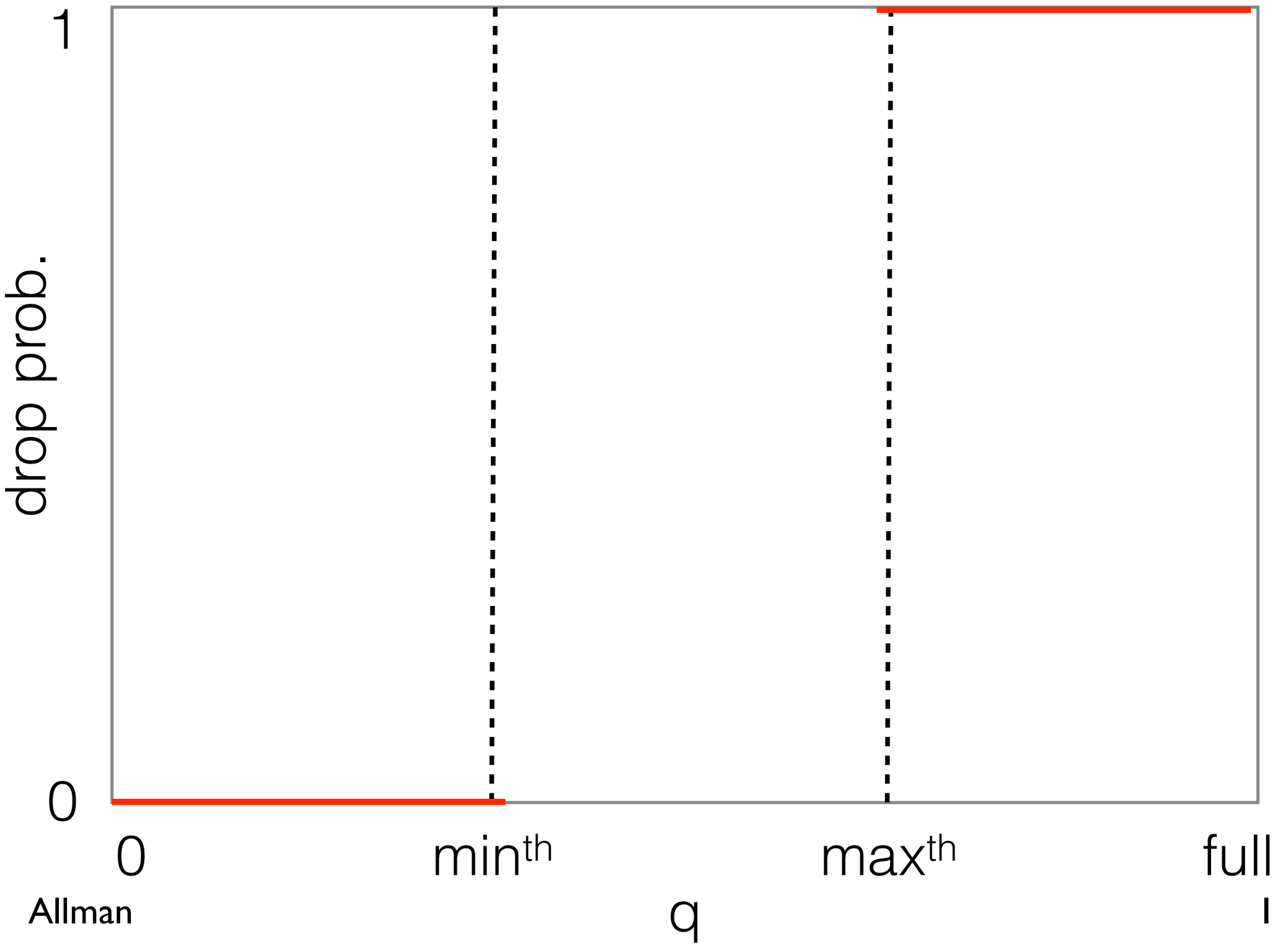
RED



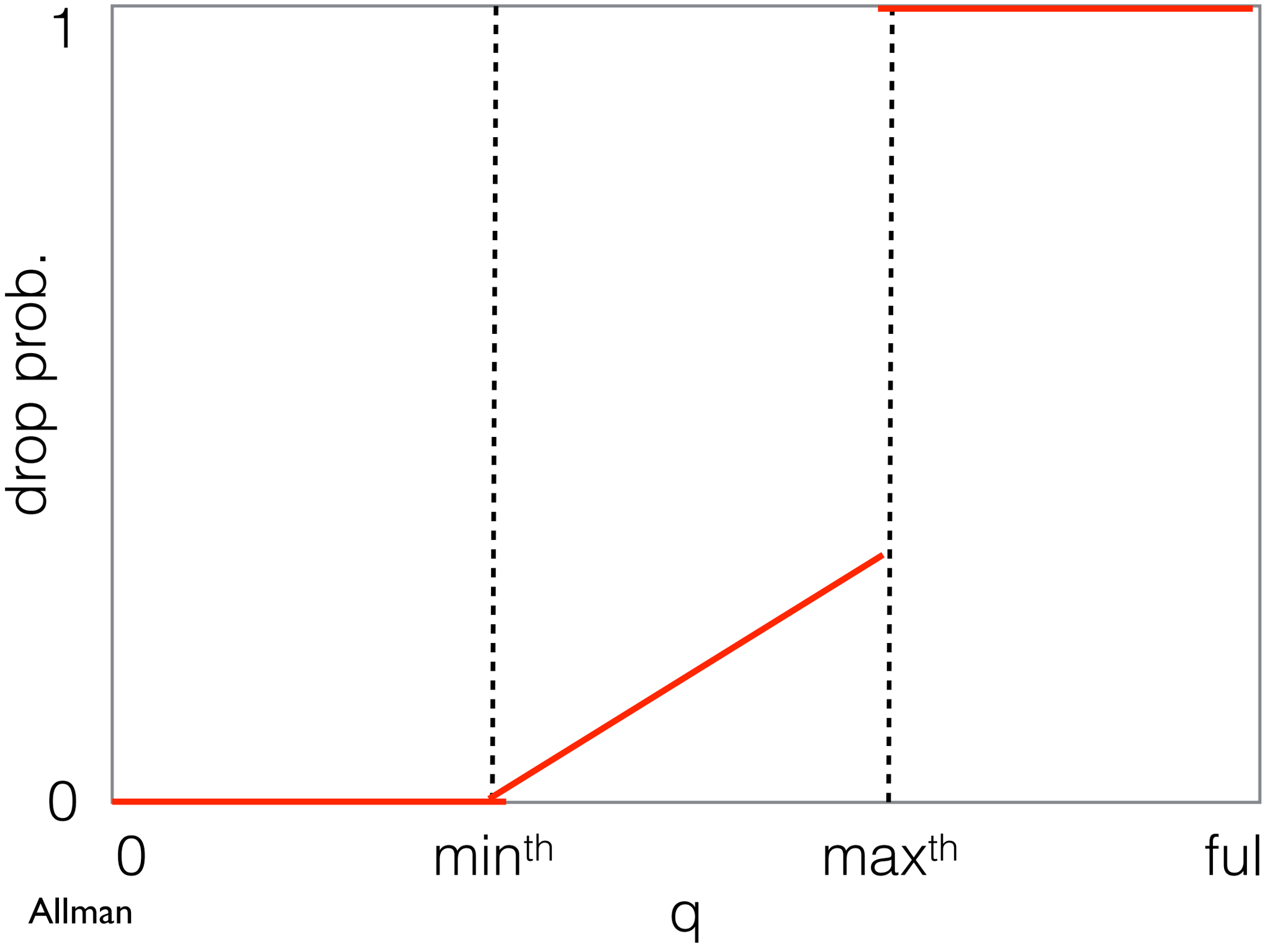
RED



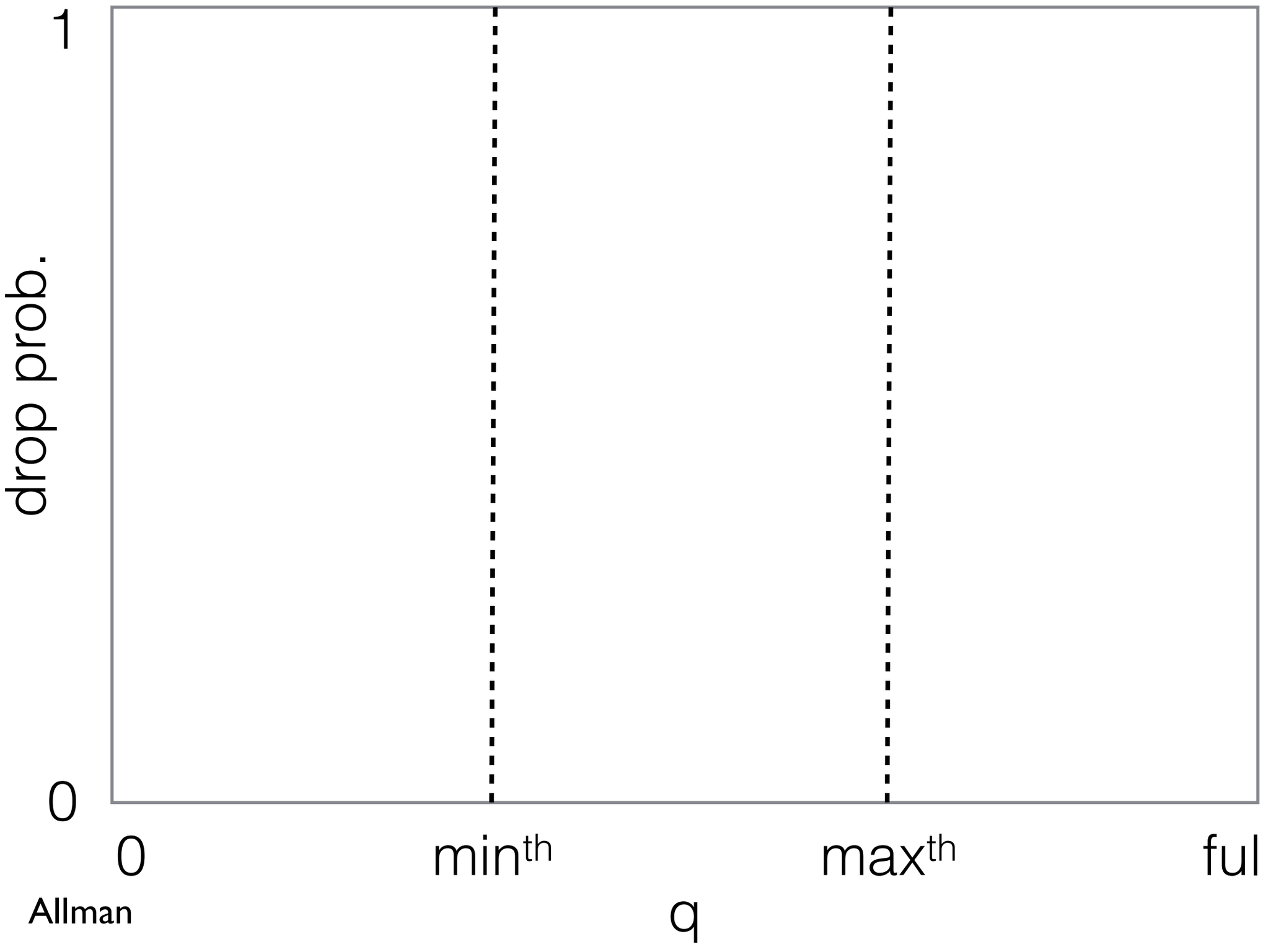
RED



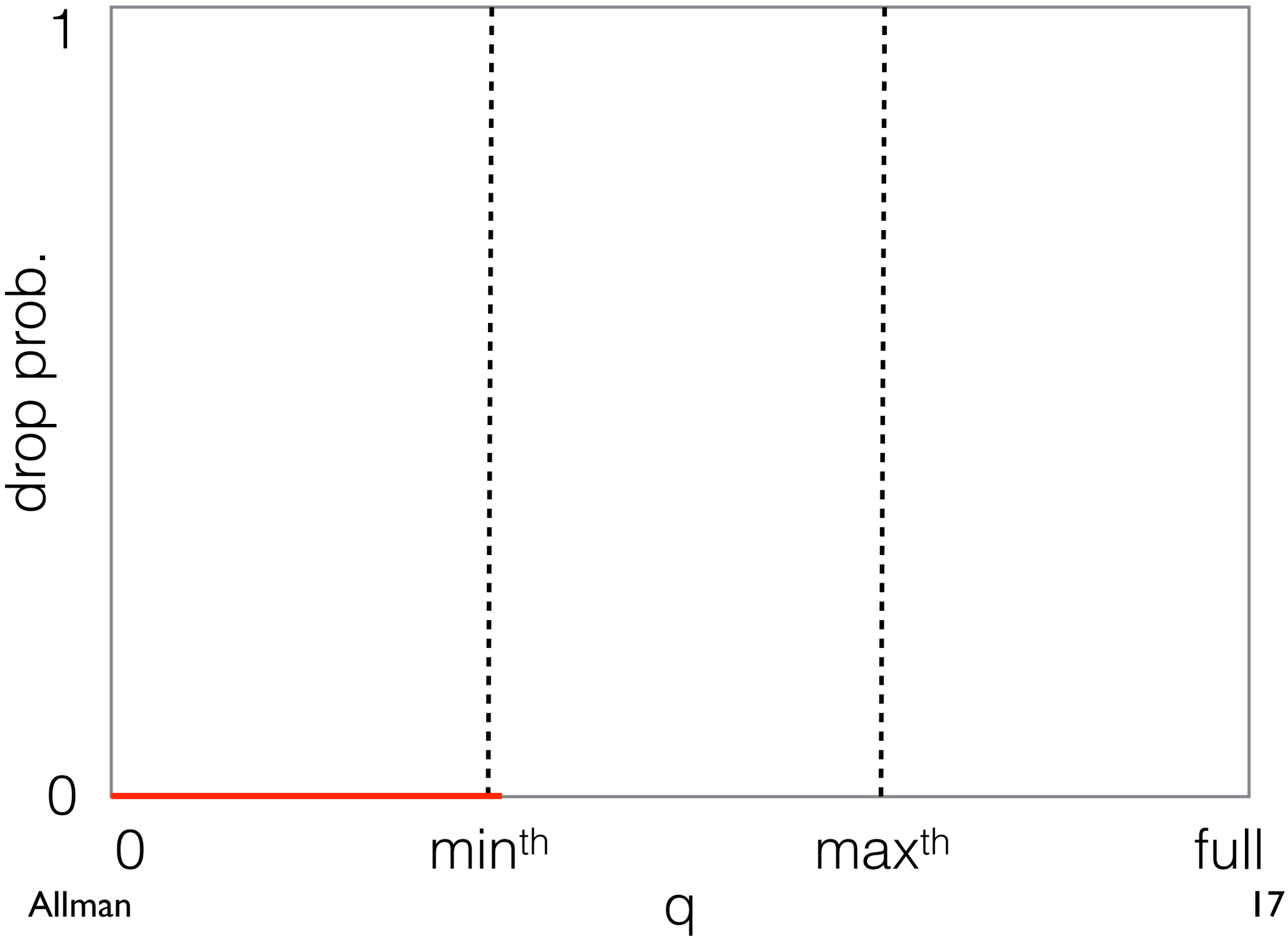
RED



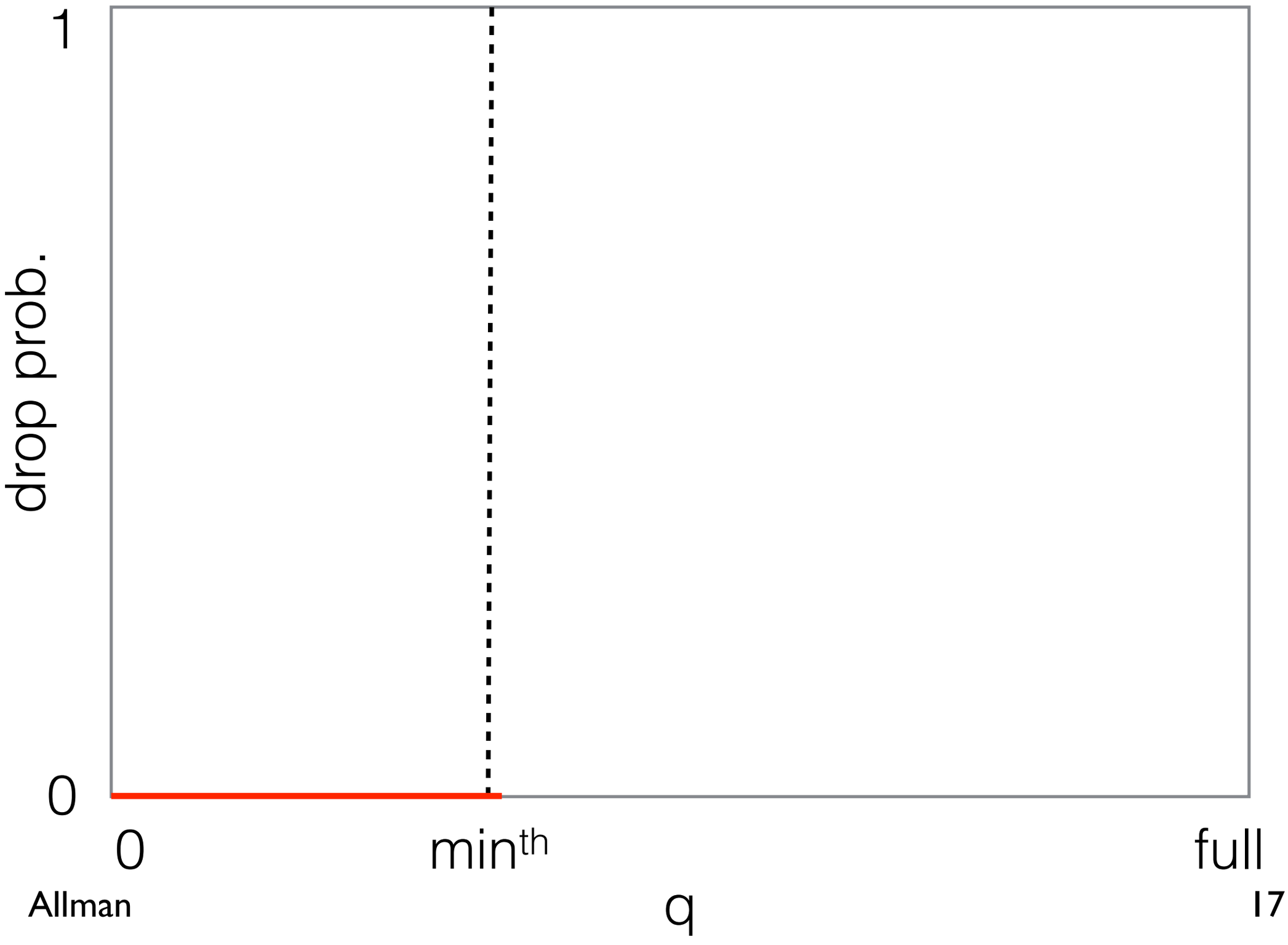
Gentle RED



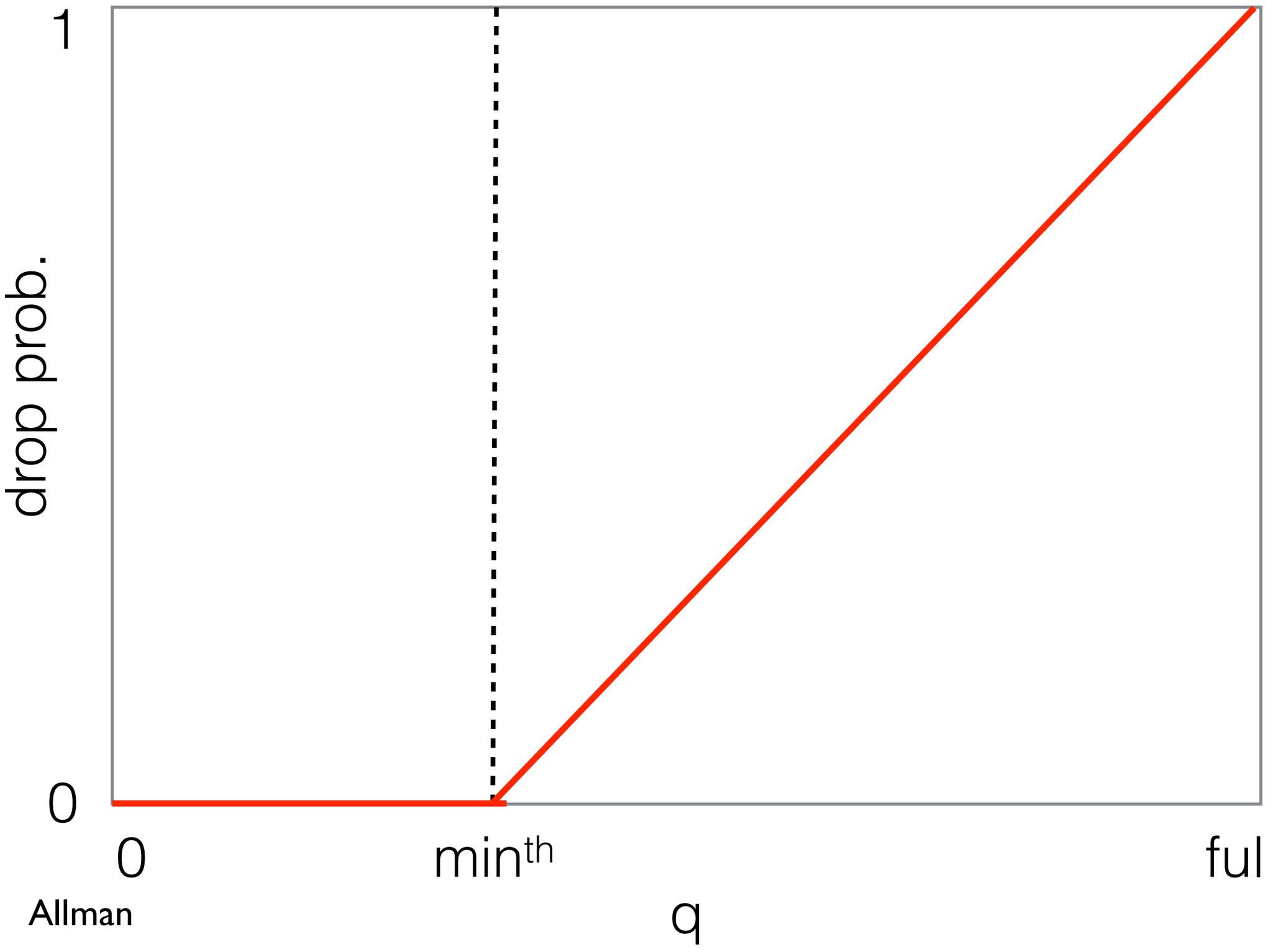
Gentle RED



Gentle RED



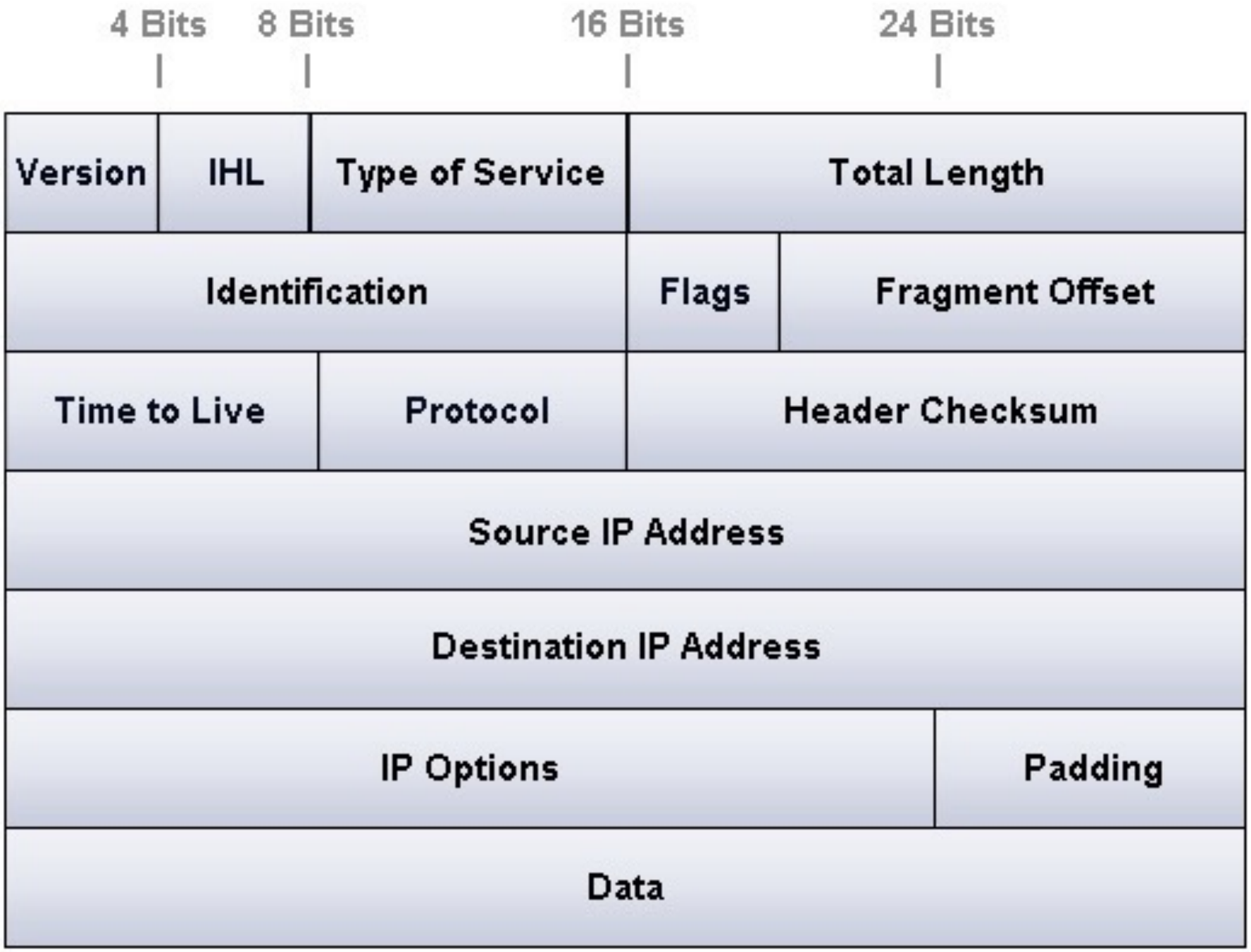
Gentle RED



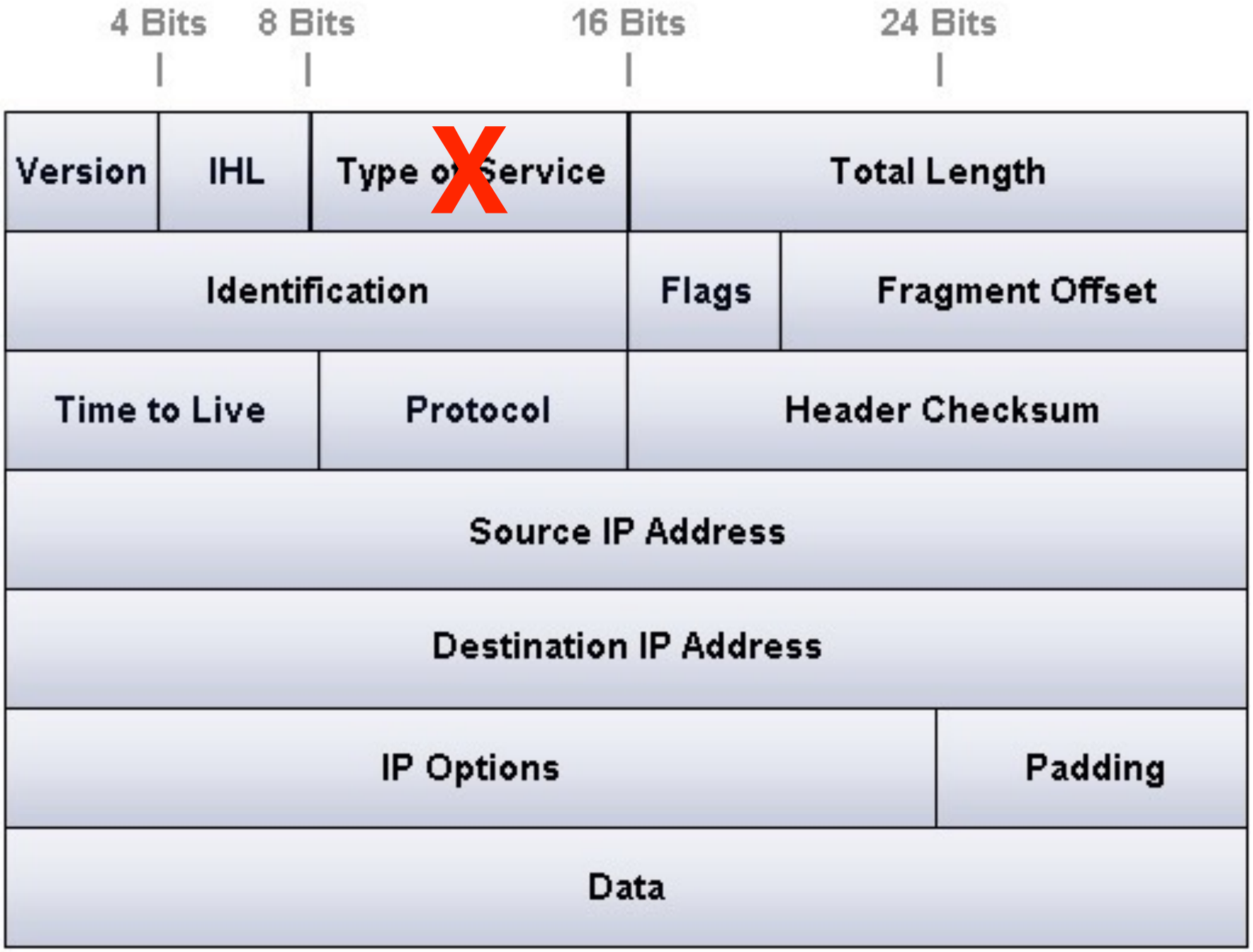
Active Queuing

- Operators are somewhat hesitant to use these schemes .... why?!

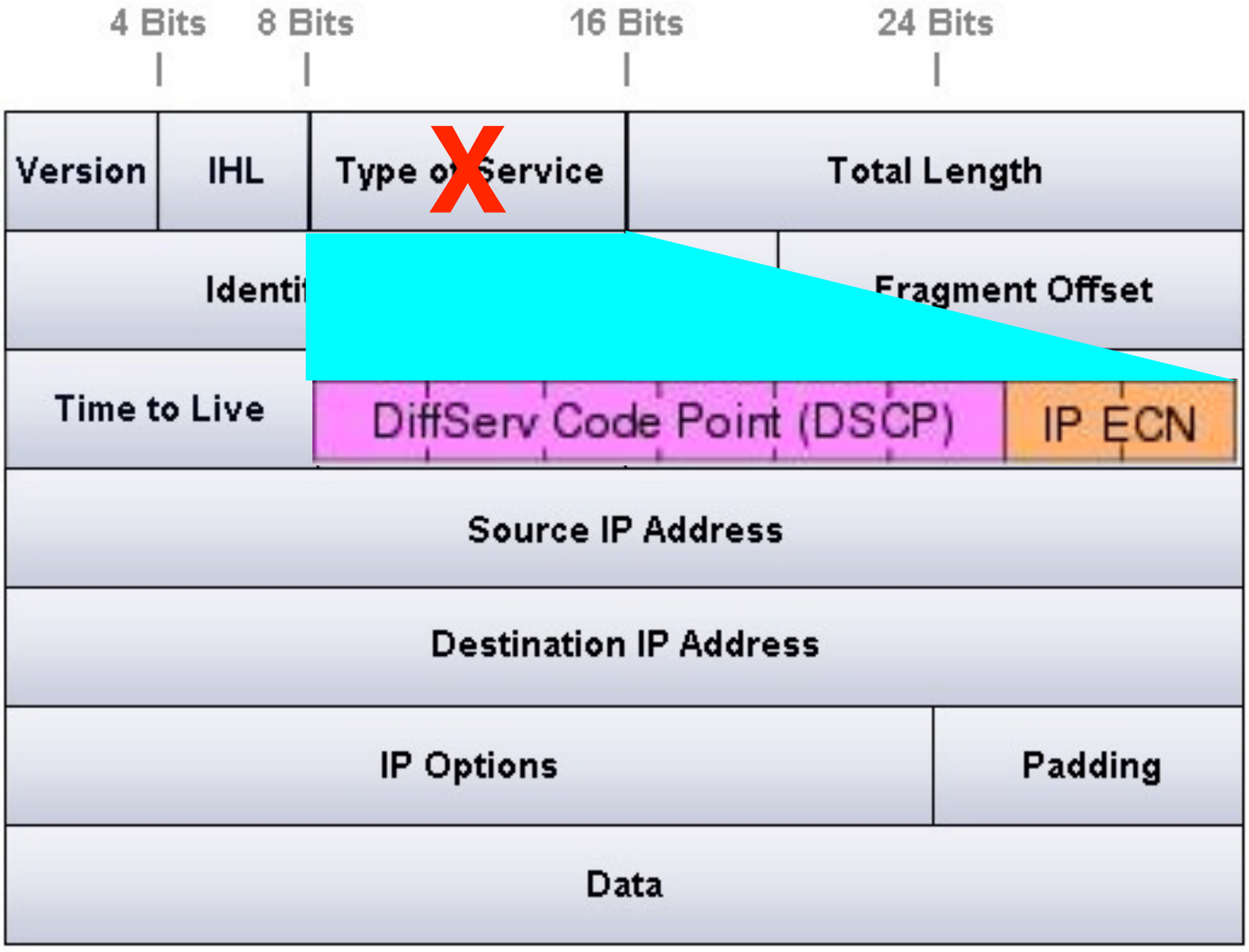
Explicit Congestion Notification



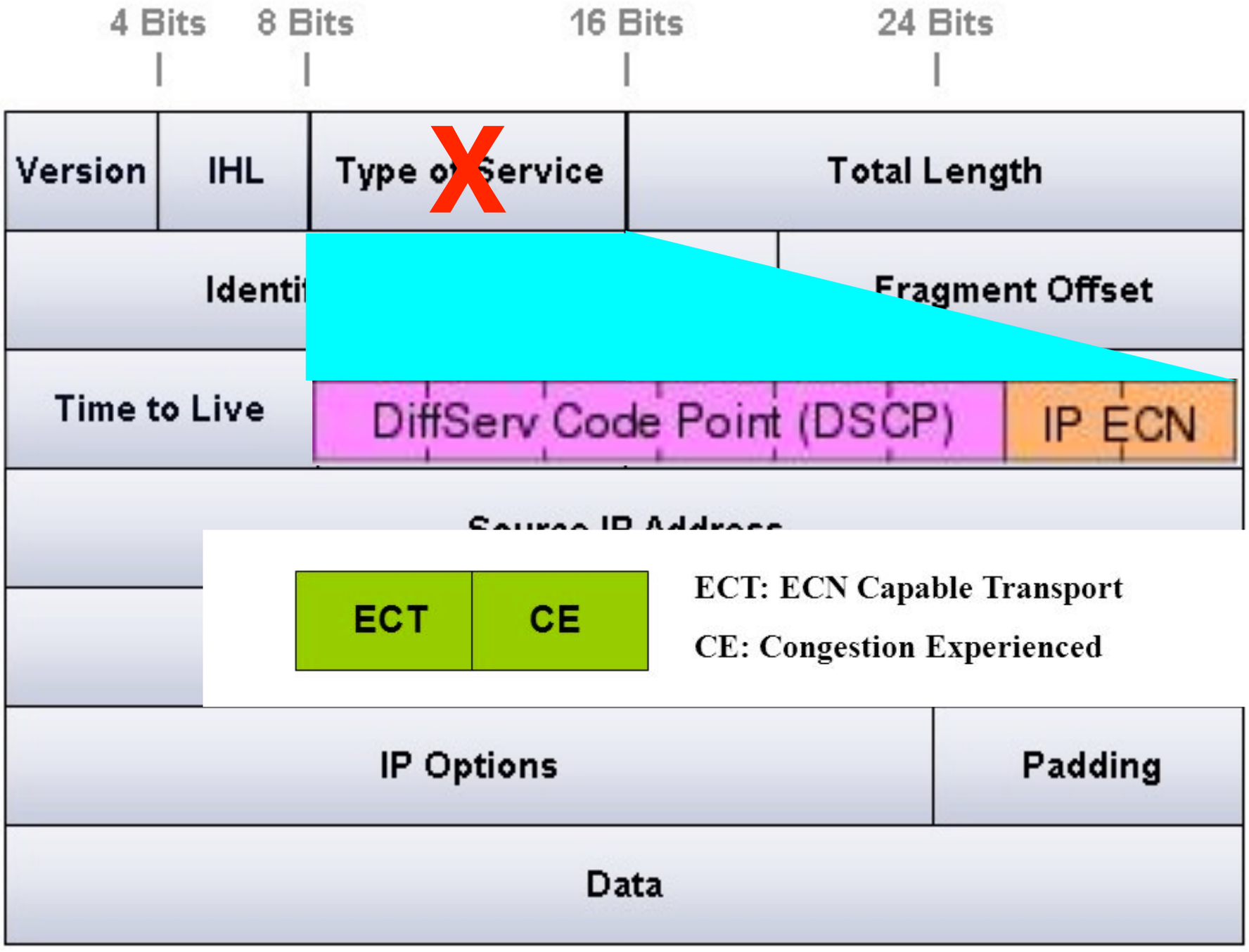
Explicit Congestion Notification



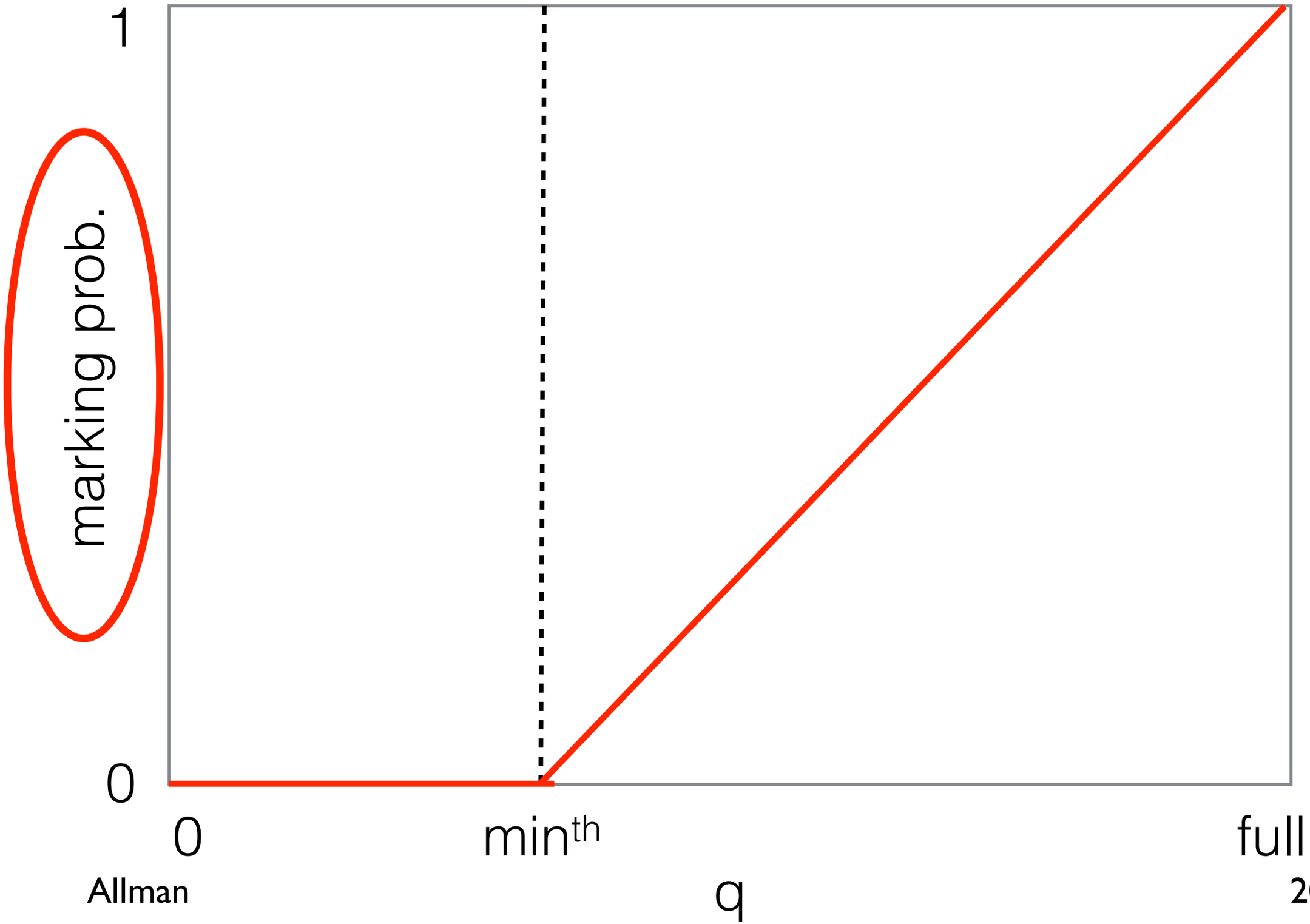
Explicit Congestion Notification



Explicit Congestion Notification



Gentle RED + ECN



Fair Queuing

Fair Queuing

- Independent logical queues for different traffic
  - e.g., for different source IPs
  - e.g., for different TCP connections
  - e.g., for different transport protocols

Fair Queuing

- Independent logical queues for different traffic
  - e.g., for different source IPs
  - e.g., for different TCP connections
  - e.g., for different transport protocols
- I.e., protect traffic from competition

Fair Queuing

- Independent logical queues for different traffic
  - e.g., for different source IPs
  - e.g., for different TCP connections
  - e.g., for different transport protocols
- I.e., protect traffic from competition
- We have figured out the hardware to do this at line rate

Traffic Shaping

- Lots of techniques to manage traffic at various points in the network according to policy
  - expensive!
- Does this obviate the need for endpoint CC?