# Reorder Density Function
# A Metric for Packet Reordering

**Anura P. Jayasumana**

Nischal M Piratla, Abhijit Bare, Tarun Banka,

Rick Whitner* & Jerry McCollom*

Computer Networking Research Laboratory

Colorado State University

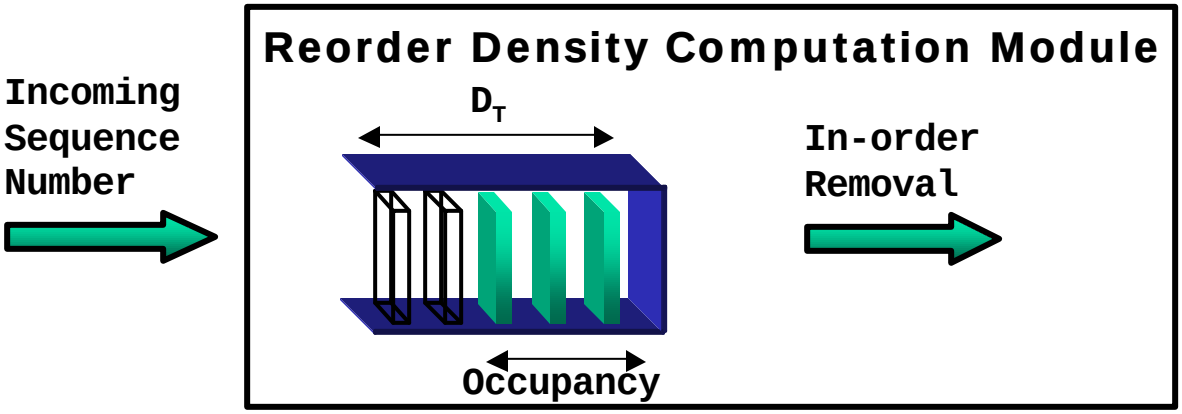
*Agilent Labs

draft-jayasumana-reorder-density-02.txt

# Outline

- RD Review
- Properties
- Improvements (LD, ED)
- Discussion

# Concept




- If a packet with a sequence number higher than the currently expected packet arrives, it is buffered.
- Packets are removed from the buffer, when they become in-order or when the buffer is full.
- Occupancy of the buffer is recorded after each arrival is processed.
- Size of the buffer ($D_T$) determines when a packet is considered lost or useless.

# Terminology

- Buffer Occupancy : Number of packets that arrived out-of-order and are stored temporarily in a hypothetical buffer.
- Buffer Occupancy Threshold ($D_T$) : Maximum size of the hypothetical buffer.
- Reorder Density (RD) : Density function of the buffer occupancy.

$$RD[i] = \frac{F[i]}{\sum_{j} F[j]}$$

where F[i] is the number of arrival instances where i buffers were occupied.

# Examples of RD Computation

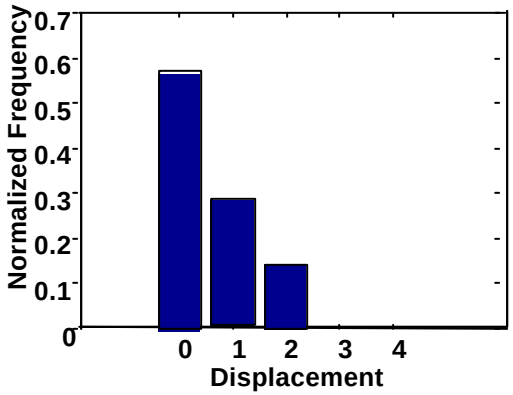
**Case of no packet loss : [1,2,4,5,3,7,6].**

**RD Computation Steps:**

| Expected | 1 | 2 | 3 | 3 | 3 | 6 | 6 |
|---|---|---|---|---|---|---|---|
| Arrival | 1 | 2 | 4 | 5 | 3 | 7 | 6 |
| D | 0 | 0 | 1 | 2 | 0 | 1 | 0 |
| F[D] | 1 | 2 | 1 | 1 | 3 | 2 | 4 |

**RD:**

| Displacement (D) | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| Frequency F[D] | 4 | 2 | 1 | 0 |
| Normalized Frequency RD[D] | 4/7 | 2/7 | 1/7 | 0 |



# Examples of RD Computation

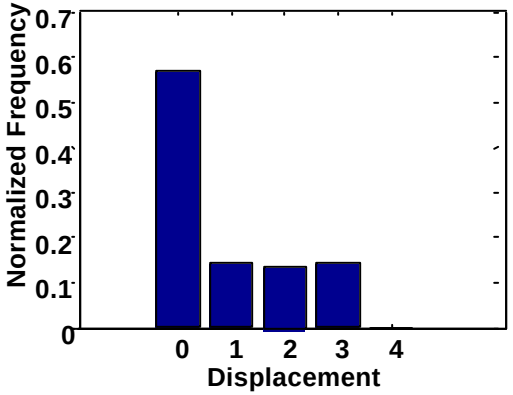
**Case of packet loss : [1,2,4,6,5,7,8] with $D_T$=3.**

**RD Computation Steps:**

| Expected | 1 | 2 | 3 | 3 | 3 | 3 | 8 |
|---|---|---|---|---|---|---|---|
| Arrival | 1 | 2 | 4 | 6 | 5 | 7 | 8 |
| D | 0 | 0 | 1 | 2 | 3 | 0 | 0 |
| F[D] | 1 | 2 | 1 | 1 | 1 | 3 | 4 |

**RD:**

| Displacement (D) | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| Frequency F[D] | 4 | 1 | 1 | 1 |
| Normalized Frequency RD[D] | 4/7 | 1/7 | 1/7 | 1/7 |



# Properties

- On the fly computation
- Computation Complexity
  - Time complexity : $O(N.DT) < O(N^2)$
    N no of packets
  - Space complexity : Constant (DT)
- Shape of RD is related to the nature of reordering.
- $90^{th}$ percentile, mean and standard deviation of RD can be used when a simpler metric is required

# Properties

- Captures both the number of packets that are out of order as well as the amount by which packets are out-of-order
- A packet is considered lost if and only if the buffer overflows (DT)
- Also useful for applications
  Ex. Resource allocation for recovery from reordering
- Given a reorder density function, we can generate packet sequences that satisfies the reorder density function - Not just a measure!

# Modifications

- Late and Early Density:
  - Place Label (PL)
    - – increments for each arrival (subject to DT)
  - Late Packet
  - Early Packet

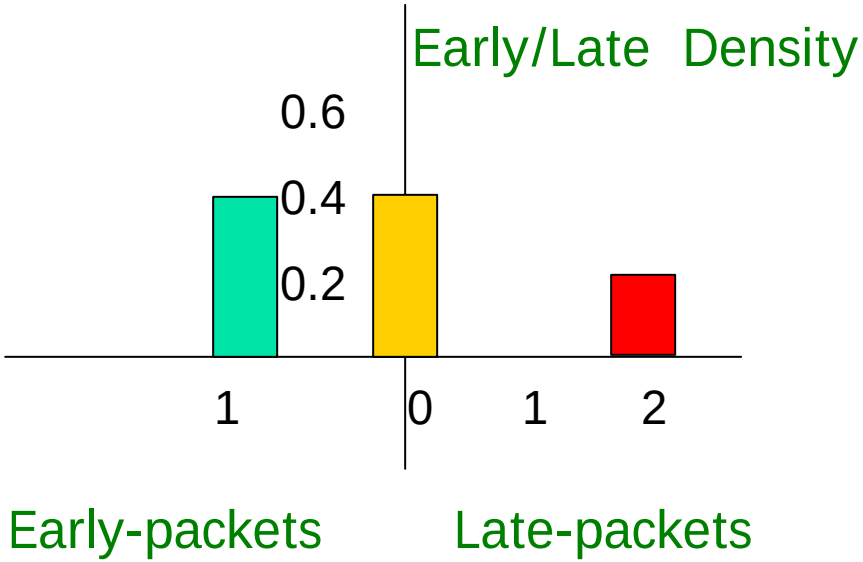
Example 1:

| Arrival: | 1 | 3 | 4 | 2 | 5 |
|---|---|---|---|---|---|
| PL: | 1 | 2 | 3 | 4 | 5 |
| Late | - | - | - | 2 | - |
| Early | - | 1 | 1 | - | - |

# Modifications

Example 1:

| | | | | | |
|---|---|---|---|---|---|
| Arrival: | 1 | 3 | 4 | 2 | 5 |
| PL: | 1 | 2 | 3 | 4 | 5 |
| Late | - | - | - | 2 | - |
| Early | - | 1 | 1 | - | - |



# Modifications

- Late and Early Density:
  - Place Label (PL)
    - – increments for each arrival (subject to DT)

| Arrival: | 1 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|
| PL: | 1 | 2 | 3 | 4 | 5 | 6 |

# Modifications

- Late and Early Density:
  - Place Label (PL)
    - – increments for each arrival (subject to DT)

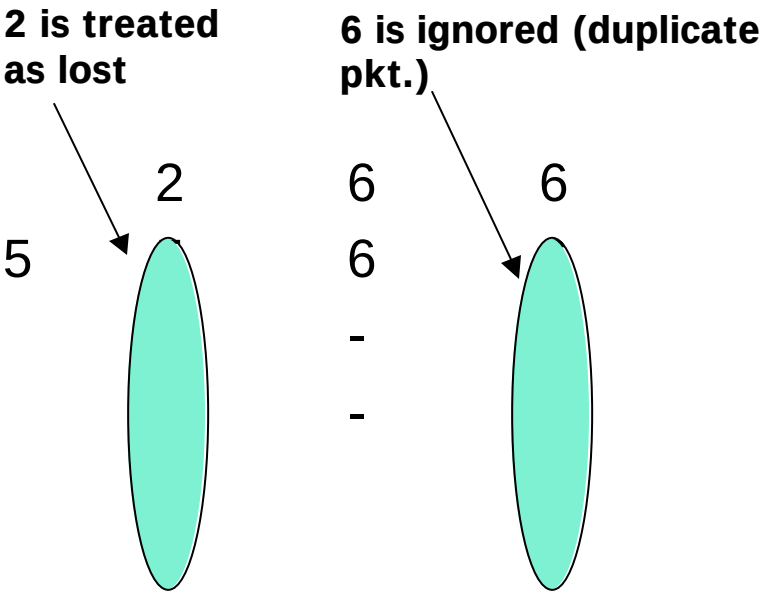
Ex: DT=2

| Arrival: | 1 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|
| PL: | 1 | 2 | 3 | ~~4~~ | ~~5~~ | ~~6~~ |
| | | | | 5 | 6 | 7 |

# Modifications

Example 2: (With DT = 2)

| Arr. | 1 | 3 | 4 | 5 | 2 | 6 | 6 |
|---|---|---|---|---|---|---|---|
| PL | 1 | 2 | 3 | 4 ->5 | | 6 | |
| Late | - | - | - | - | | - | |
| Early | - | 1 | 1 | 0 | | - | |



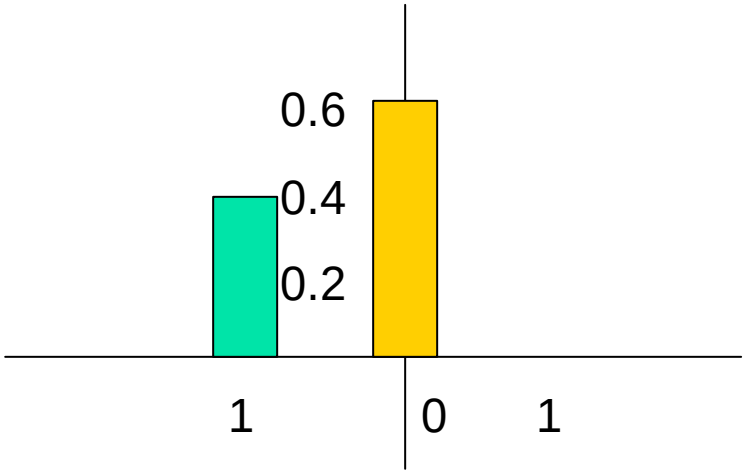
# Modifications

Example 2: (With DT = 2)

| Arr. | 1 | 3 | 4 | 5 | 2 | 6 | 6 |
|---|---|---|---|---|---|---|---|
| PL | 1 | 2 | 3 | 4 ->5 | | 6 | |
| Late | - | - | - | - | | - | |
| Early | - | 1 | 1 | 0 | | - | |

**2 is treated as lost**

**6 is ignored (duplicate pkt.)**




Early-packets Late-packets

# Conclusion

- RD, ED and LD  more completely define reordering
- Order of complexity is still the same
- Extensions – Reordering to satisfy a given RD

# Follow-up options ??

- Merge basic concept with Morton draft
- Pursue as an alternate draft and/or as an informational RFC
- Other ?