Reliable and Scalable Packet Striping

Hari Adiseshu

Guru Parulkar

George Varghese

Washington University in St. Louis

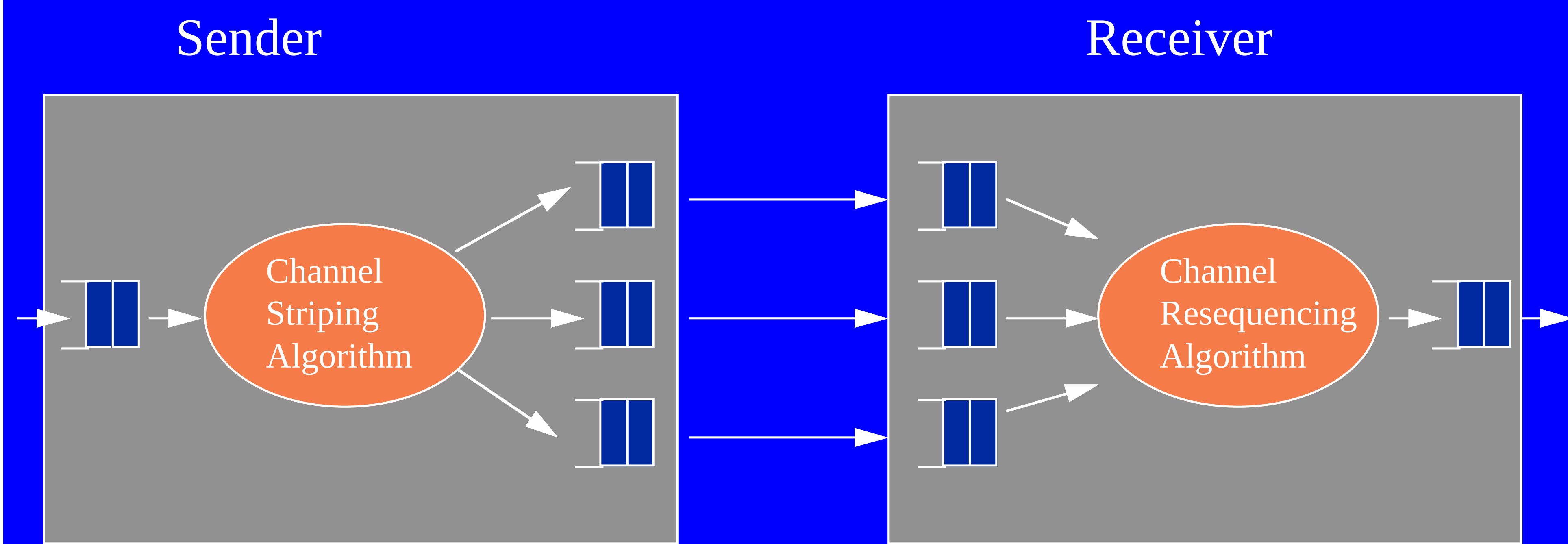
Introduction

- Packet Striping
  - using multiple links as a single logical link
- Used for performance or cost reasons

Current Striping Techniques

- Usually at the physical level (Inverse Multiplexing)
- Physical level striping has drawbacks
- Wanted: packet striping at a logical level
- Multilink PPP provides framing, not load sharing

Packet Striping Reference Model



Assume: All Channels are FIFO

Desirable properties:

- Load Sharing
- FIFO delivery

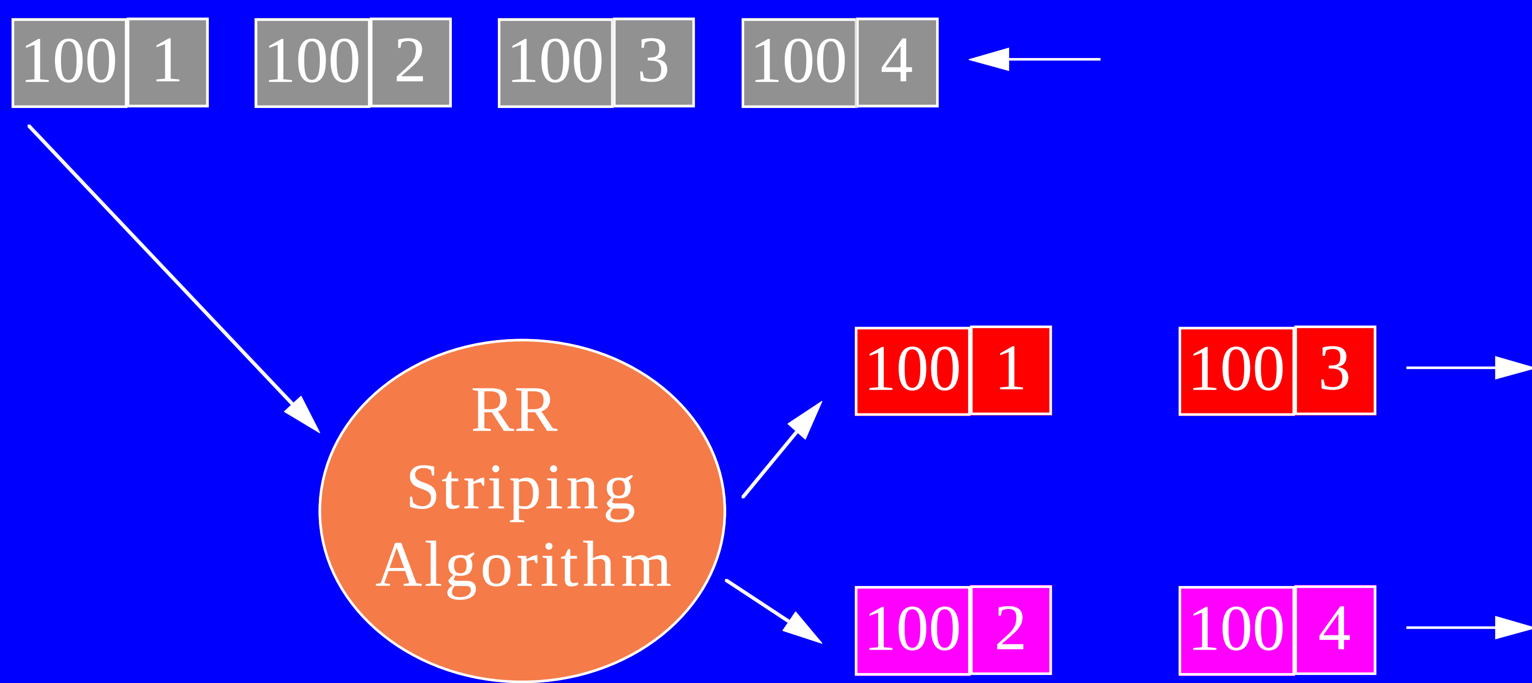
Overview of Talk

- How to achieve load sharing
- How to achieve FIFO delivery
- How to recover from packet loss
- Experimental results

Contributions

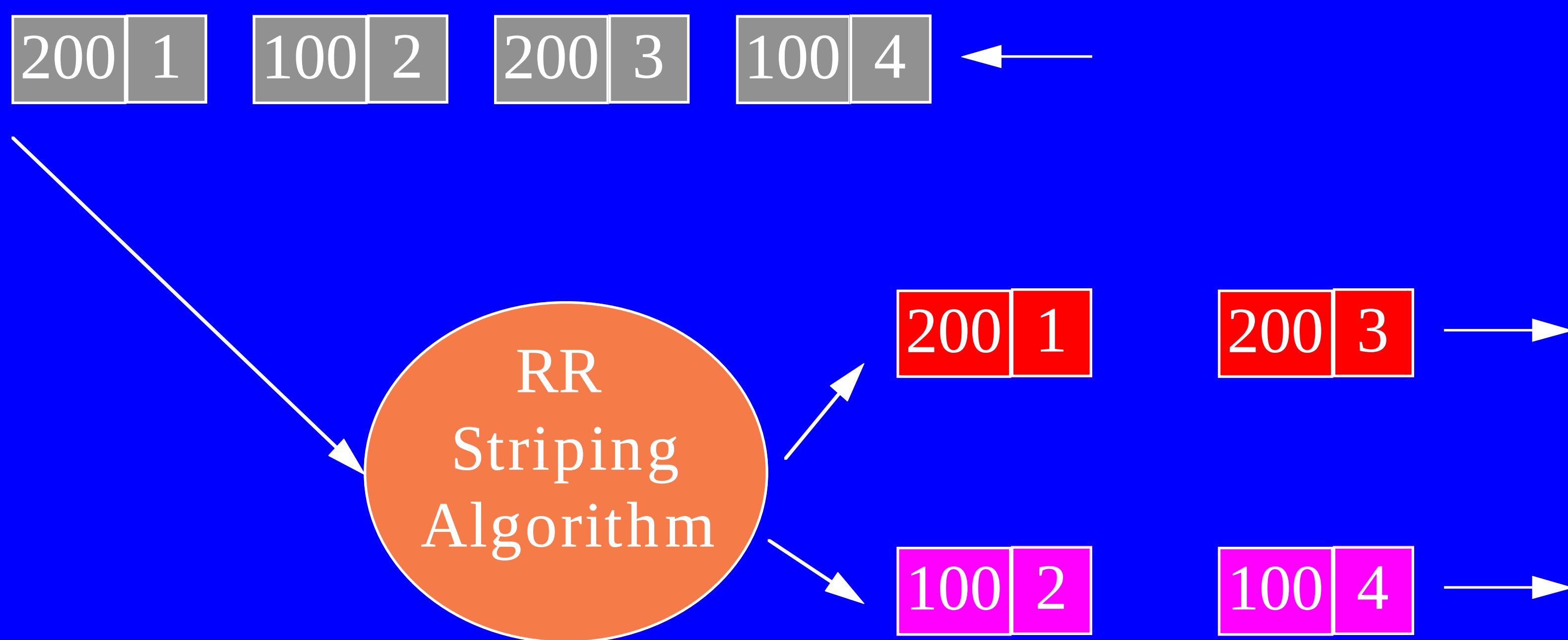
- Provide a transformation of a class of fair queuing algorithms into striping algorithms providing load sharing
- Provide a specific instance of such a fair queuing algorithm
- Show how to achieve Quasi-FIFO delivery at the receiver
- Show how to recover from packet loss through periodic marker packets

Round Robin Striping



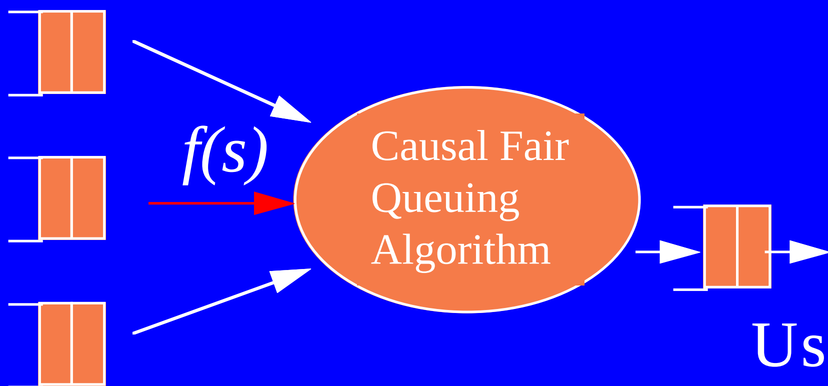
Equal Sized Packets

Round Robin Striping



No Load Sharing with unequal sized packets

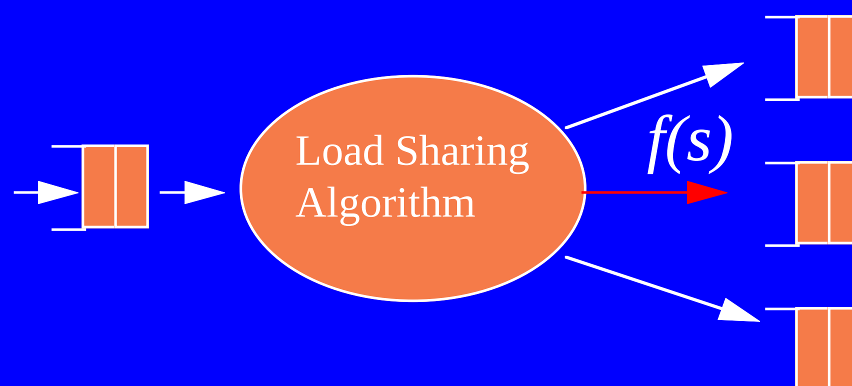
CFQ and Load Sharing



Use function $f(s)$ to select queue

Transmit packet p at head of queue

Update state using function $g(s, p)$



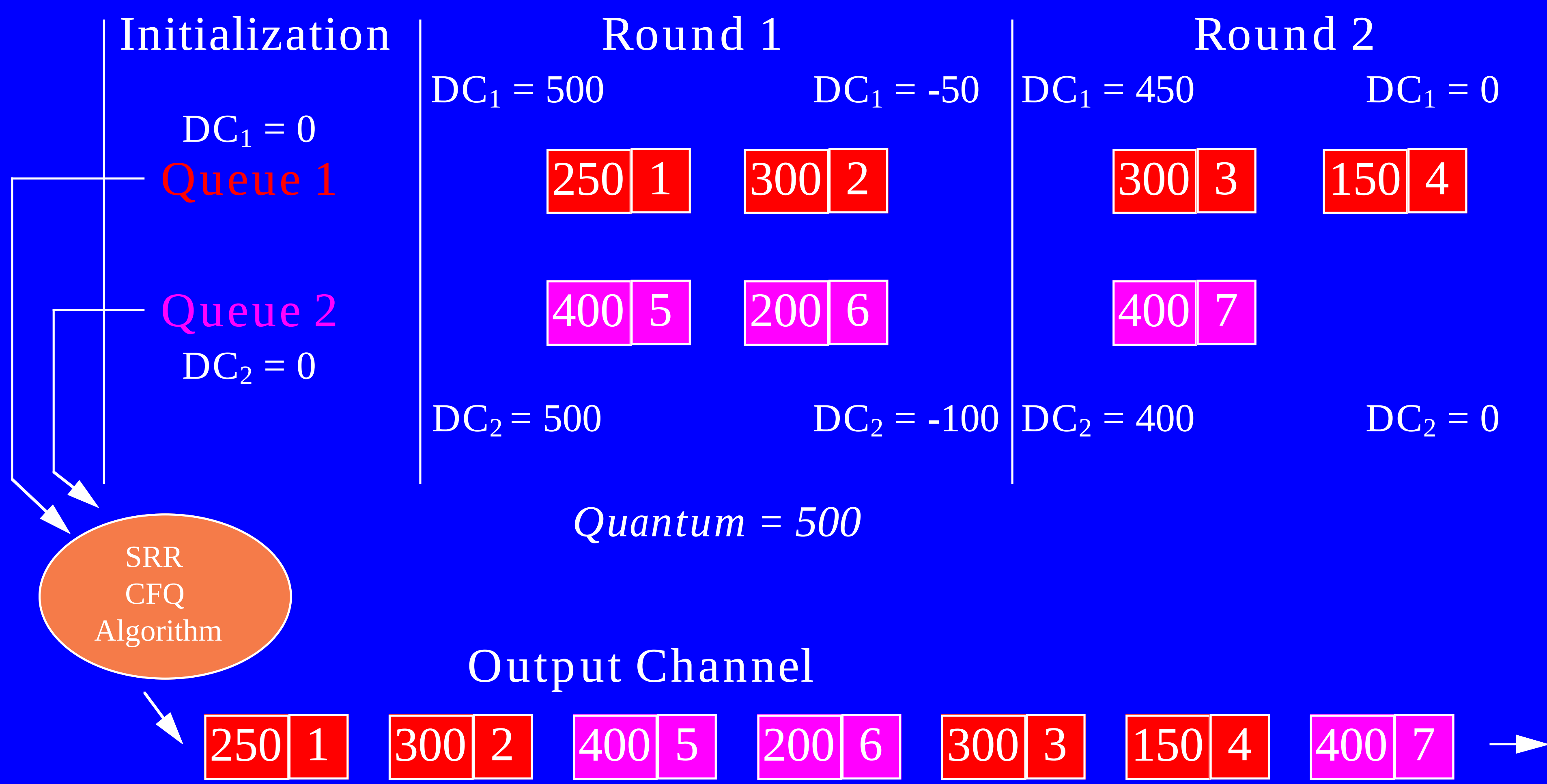
Load Sharing

- CFQ algorithms - FQ algorithm in which queue to be served depends only on packets sent so far
- CFQ algorithms can be modeled as repeated applications of two functions
- Theorem: Causal FQ algorithms can be transformed into load sharing algorithms with the same fairness properties

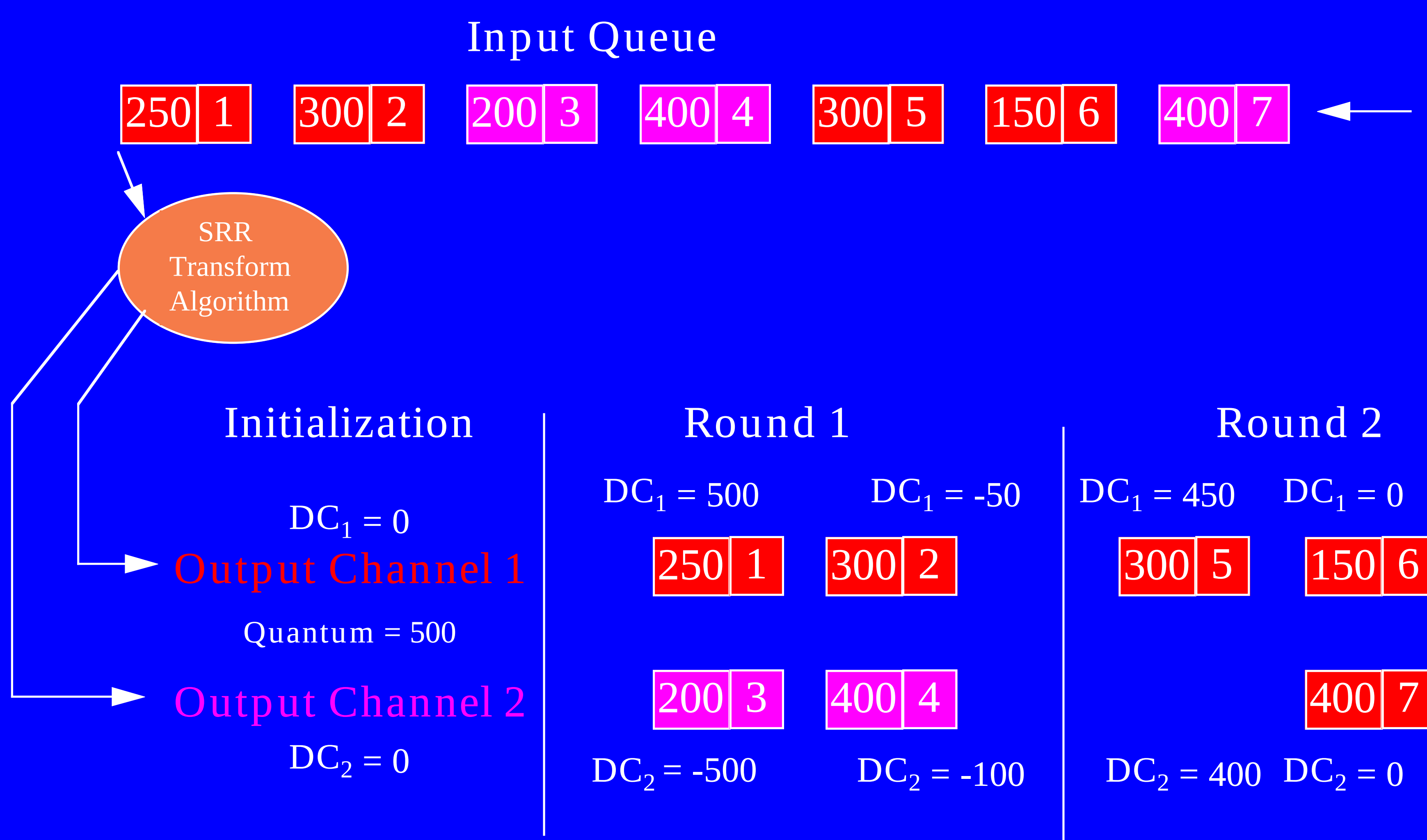
Surplus Round Robin (SRR)

- Example of a CFQ algorithm
- Each input queue associated with a *Quantum* and a *Deficit Counter* (DC)
- Queues served in round robin order
- At each visit to a queue, DC is incremented by *Quantum*
- Each queue served as long as DC is positive

SRR Causal Fair Queuing



SRR Transform for Load Sharing



SRR Results

- SRR transform provides load sharing if the *Quantum* associated with each channel is proportion to its capacity, and is greater than the maximum packet size
- *K* rounds after initialization, the difference between data actually sent over a channel *i* and the amount that actually should have been sent is bounded by $Quantum_i$

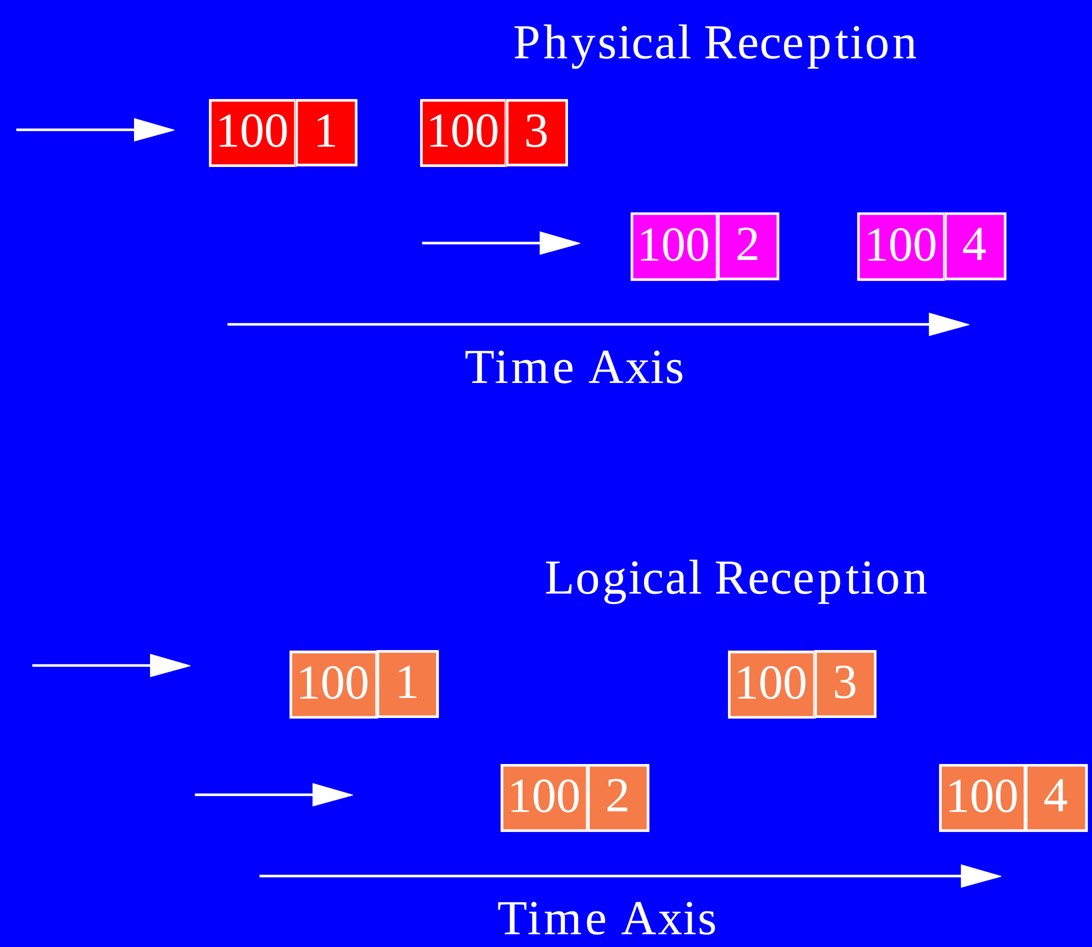
Contributions

- Provide a transformation of a class of fair queuing algorithms into striping algorithms providing load sharing
- Provide a specific instance of such a fair queuing algorithm
- Show how to achieve Quasi-FIFO delivery at the receiver

FIFO delivery

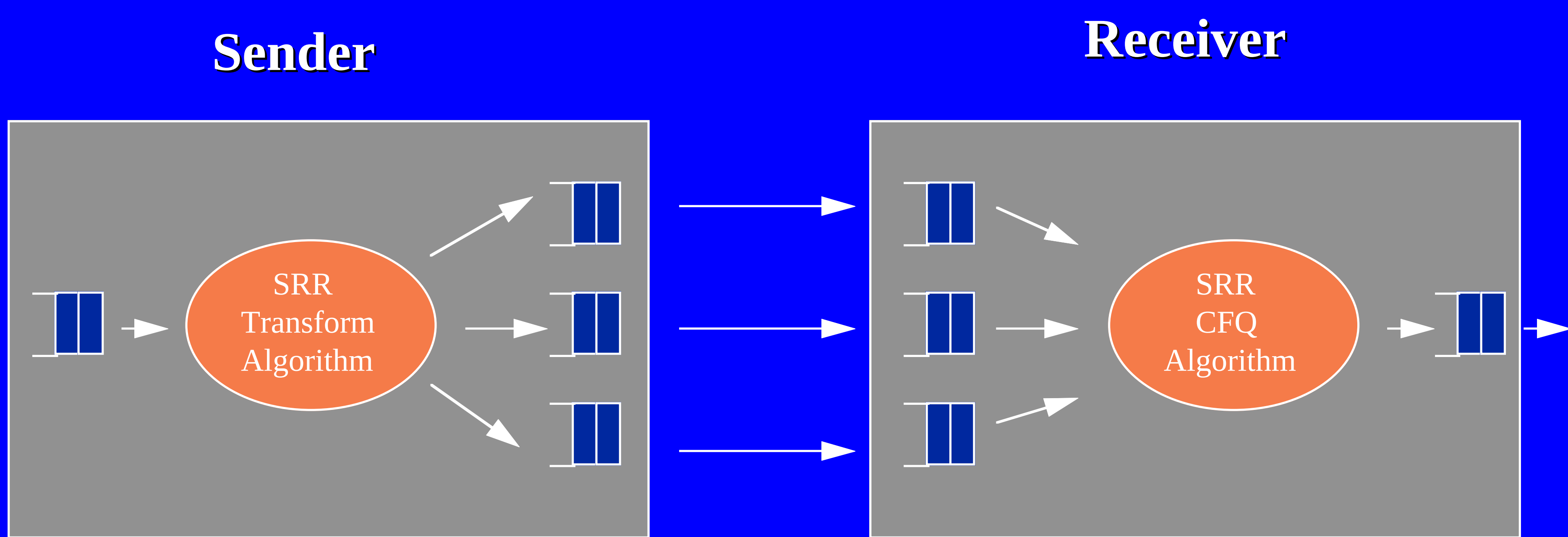
- Guaranteed FIFO - add sequence number to packets
  - not always possible, e.g.. ATM cells, transparent Ethernet striping
- Quasi-FIFO - FIFO in the absence of packet loss

Logical Reception



- Receiver Buffering
- Receiver simulation of the sending algorithm

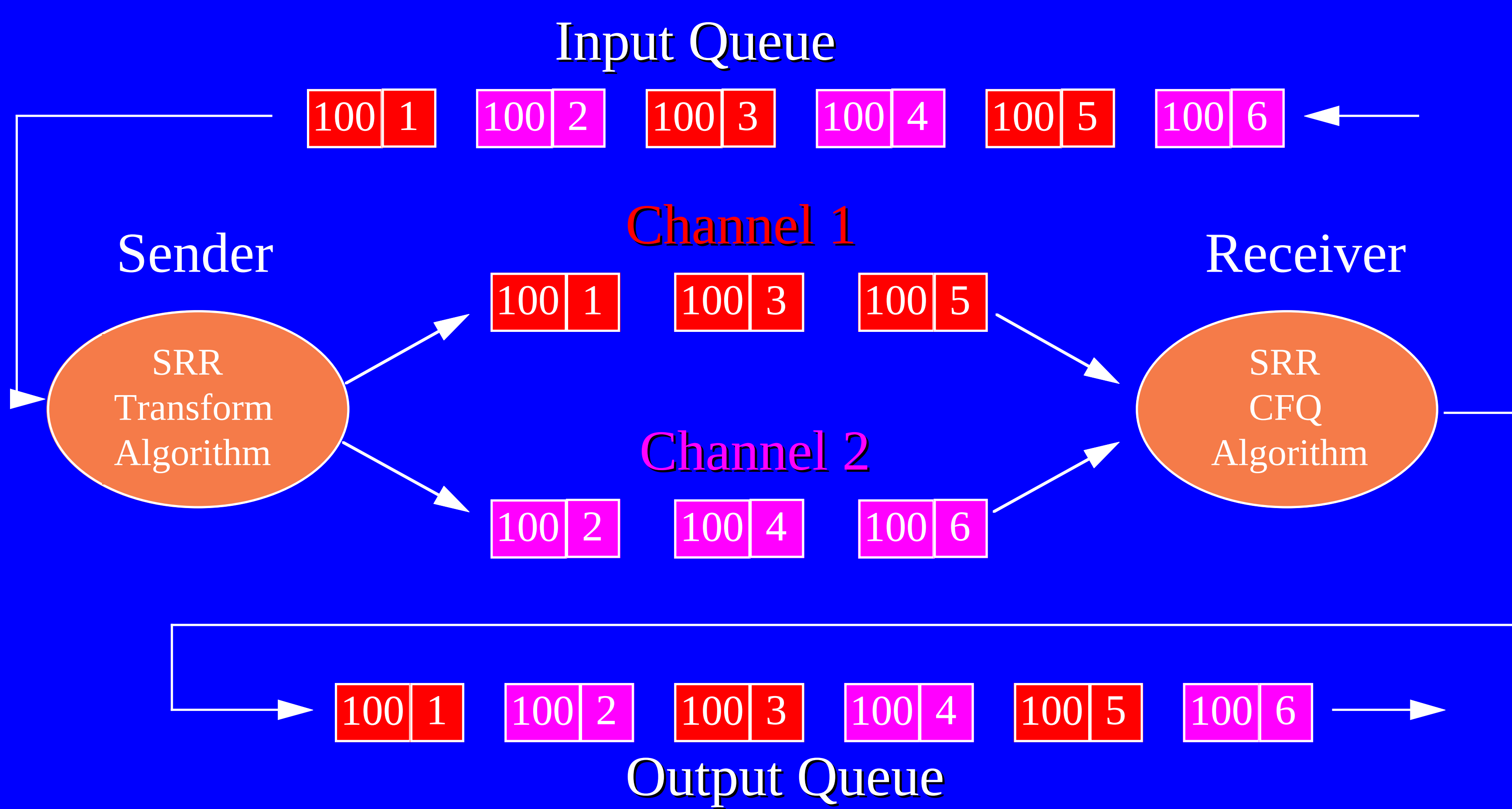
Quasi-FIFO reception using SRR



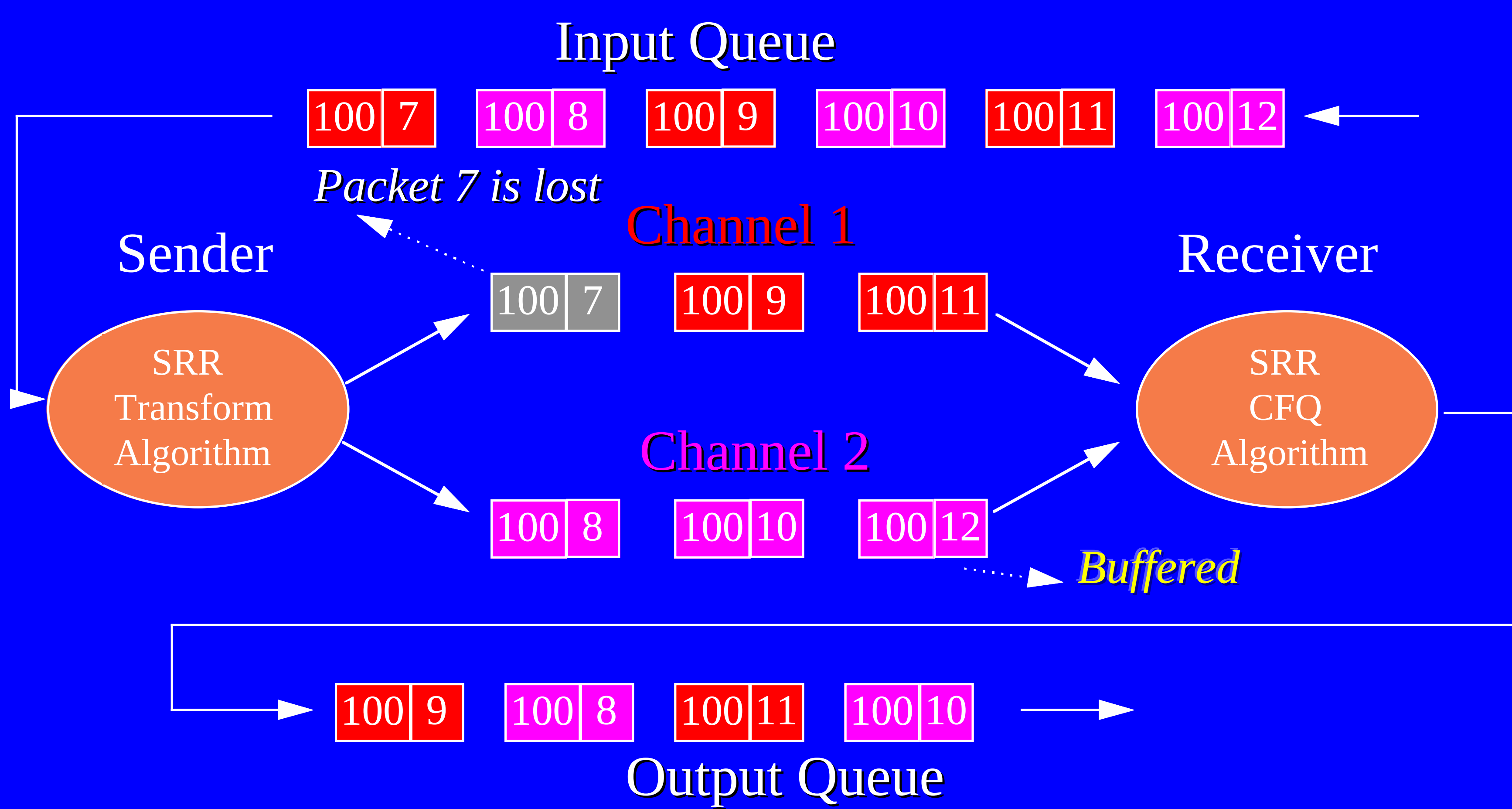
Quasi-FIFO result

- The combination of a transformed version of a CFQ algorithm at the sender, and the CFQ algorithm at the receiver provides Quasi-FIFO delivery

Effect of Packet Loss



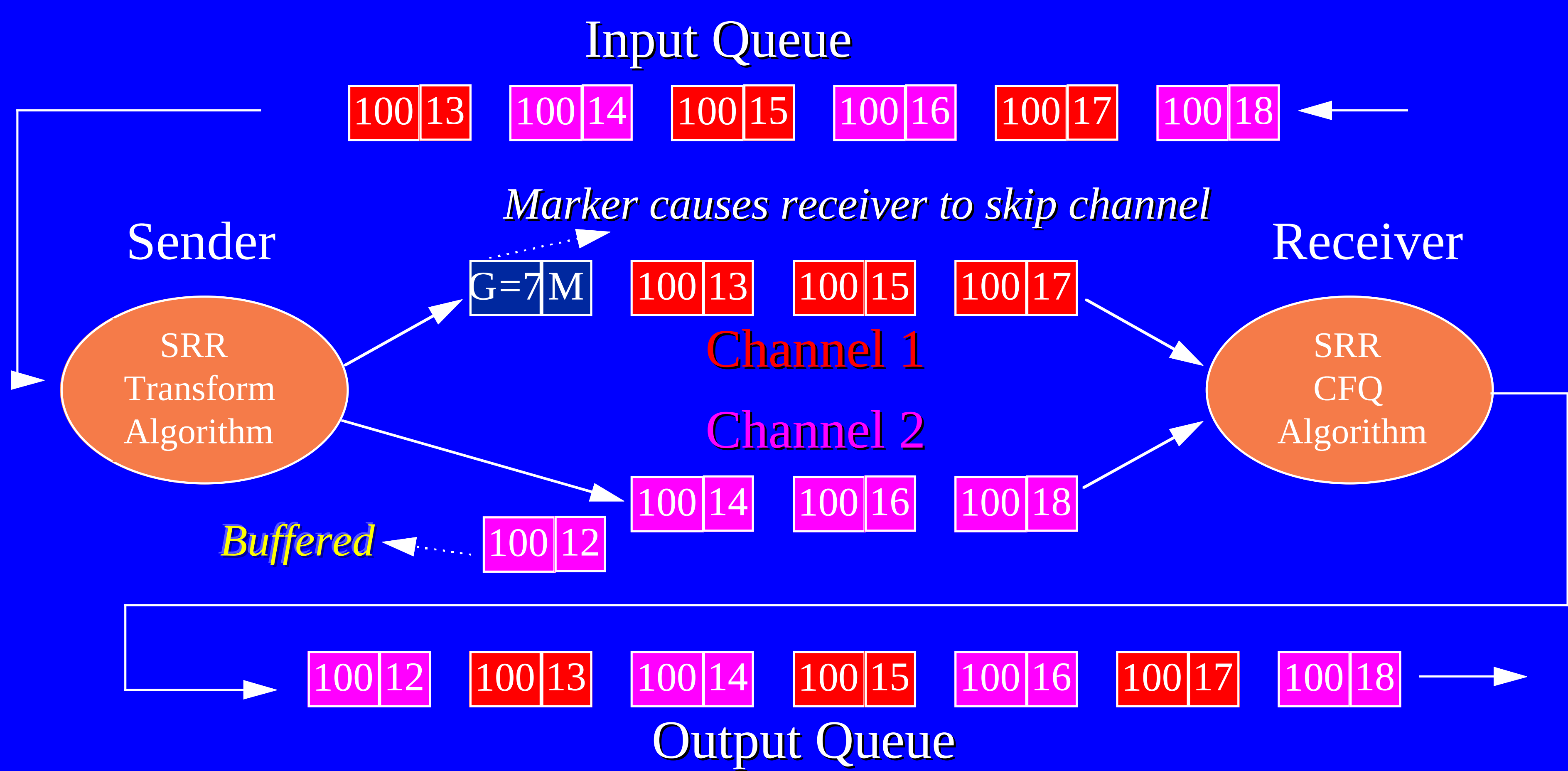
Effect of Packet Loss



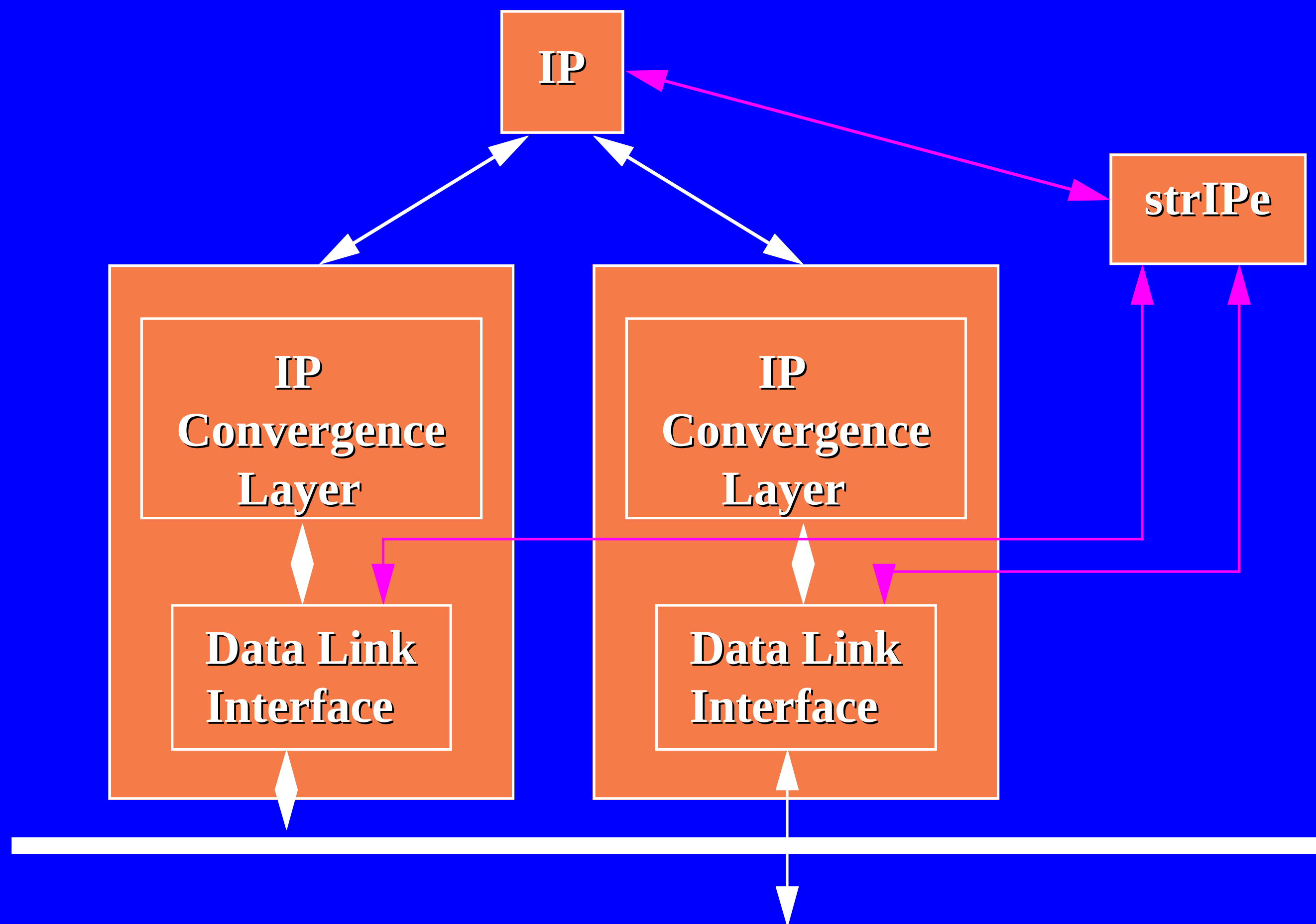
Recovery from Packet Loss

- Synchronization depends on shared state between sender and receiver
- On packet loss, the shared state is lost
- Solution: propagate state periodically from the sender to the receiver
- For the specific case of SRR algorithm, state consists of Round Number, and Deficit Counter values for each link
- Periodically propagate these values to the receiver using marker packets

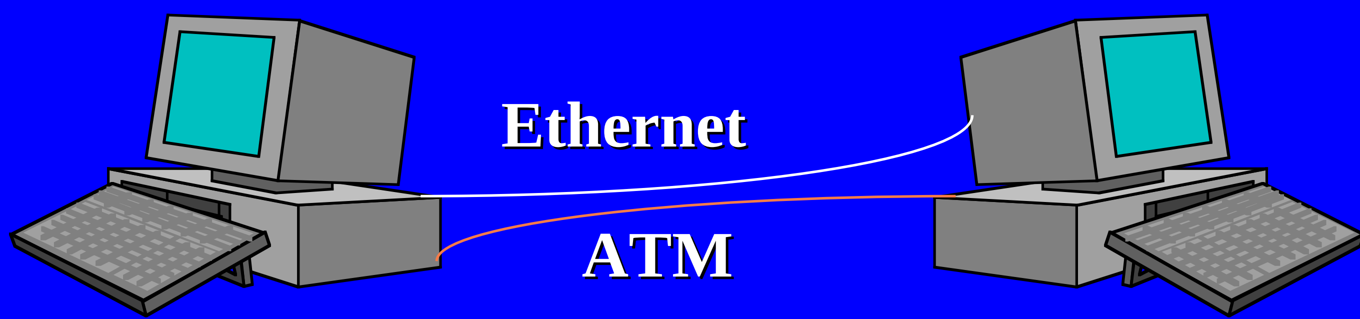
Sync Recovery



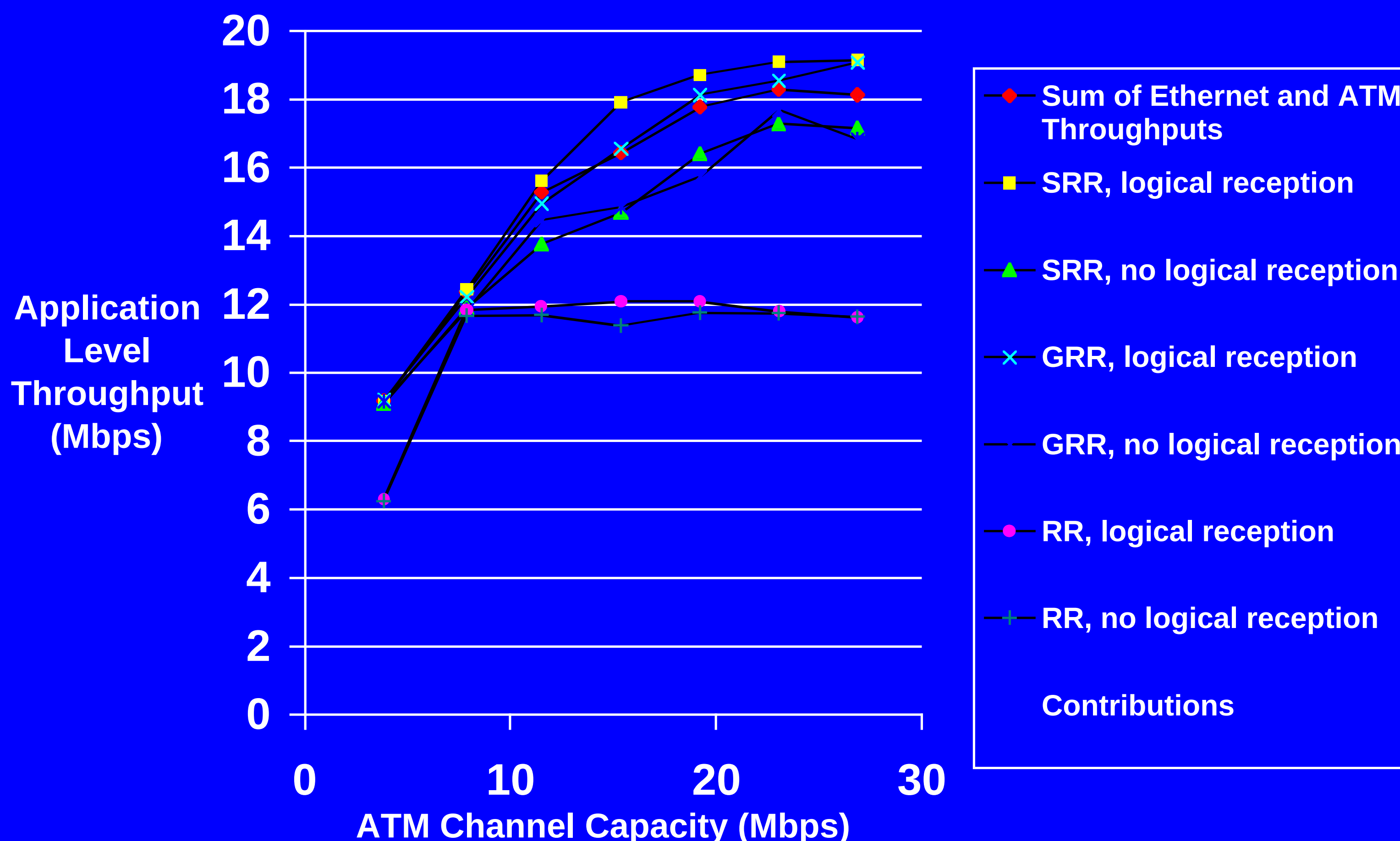
strIPe layer



Experimental Setup



NetBSD Implementation



Experimental Results

- Throughput with SRR matched aggregate throughput
- Throughput more with resequencing
- Throughput of SRR greater than other variants

Conclusions

- Simple transformation of a class of FQ algorithms can be used for scalable striping algorithms, providing FIFO packet delivery with variable sized packets and variable speed channels
- Quasi-FIFO provides transparent striping, with the marker algorithm restoring synchronization in the event of packet loss