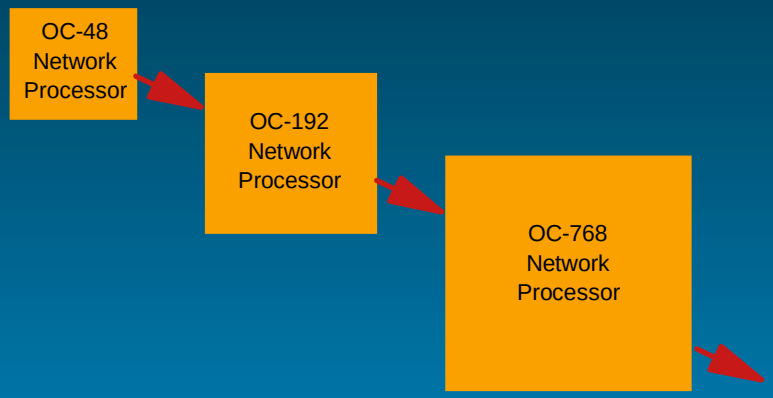
Scaling Network Processor Performance to 40 Gbps

Patricia Sagmeister
IBM Research Zurich

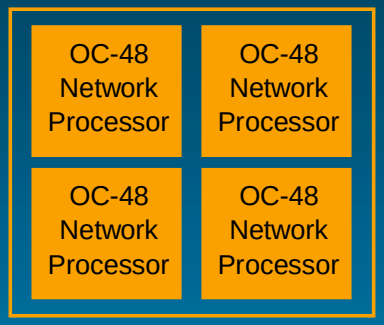
Motivation

- Optical fiber speed ⚡ network processor performance
- Scaling network processor performance

Monolithical Scaling



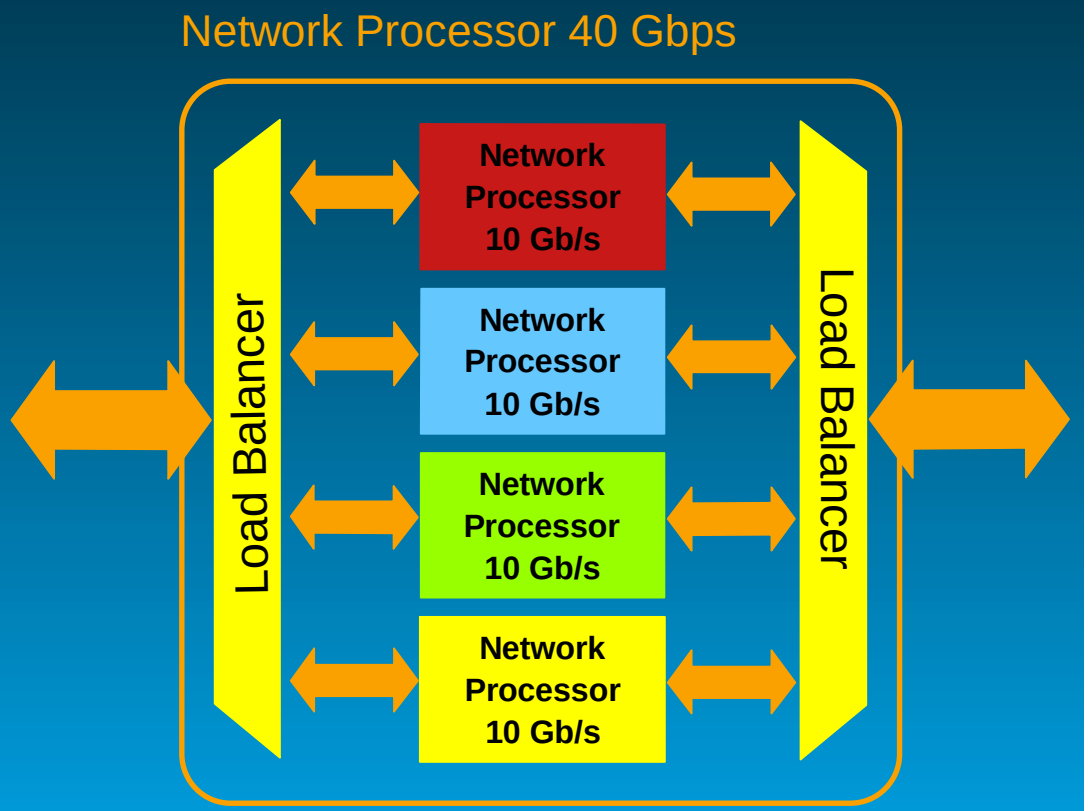
Multiprocessor Scaling



- Requirements
  - Handling up to OC-768 full line speed
  - Short time-to-market
  - Scalable concept for the future

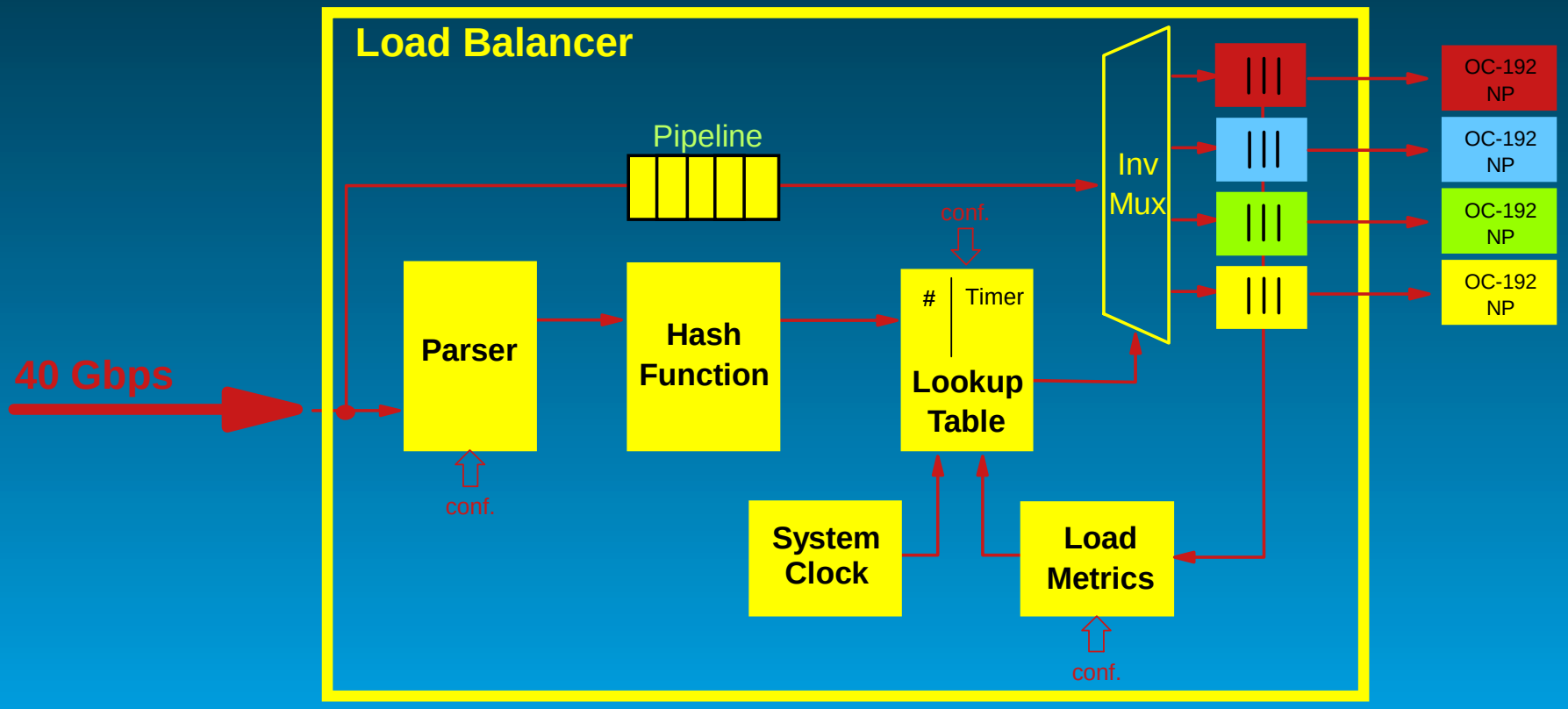
Load Balancer Concept

- Distribution of high-speed link load onto several network processors
  - Concurrently operating NPs
  - No inter-processor communication
- Characteristics
  - Real-time applicability for high speeds
  - Flow preservation
  - Support of heterogeneous NP structures
  - Low complexity
  - Concept flexibility
  - Fault tolerance

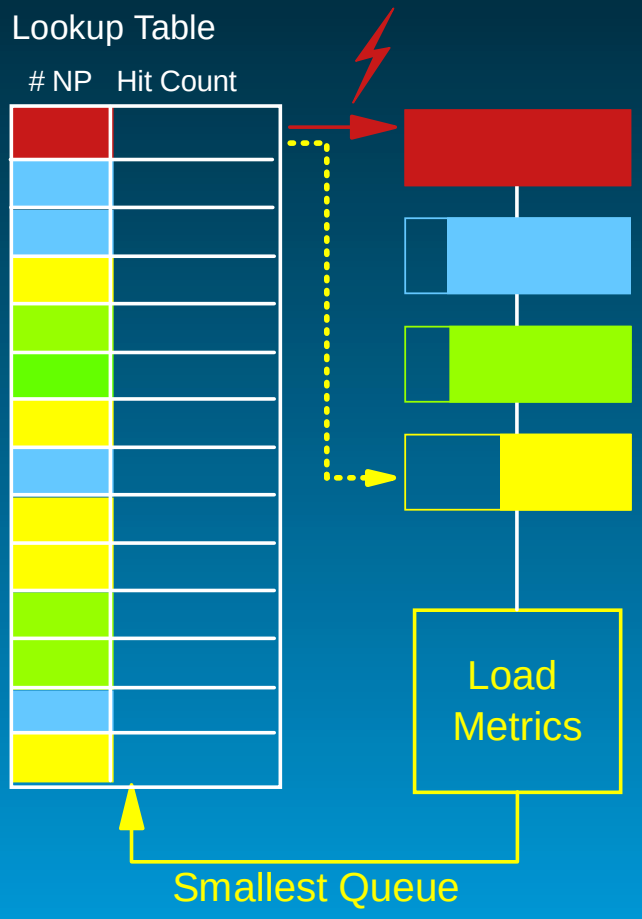


Load Balancer Architecture

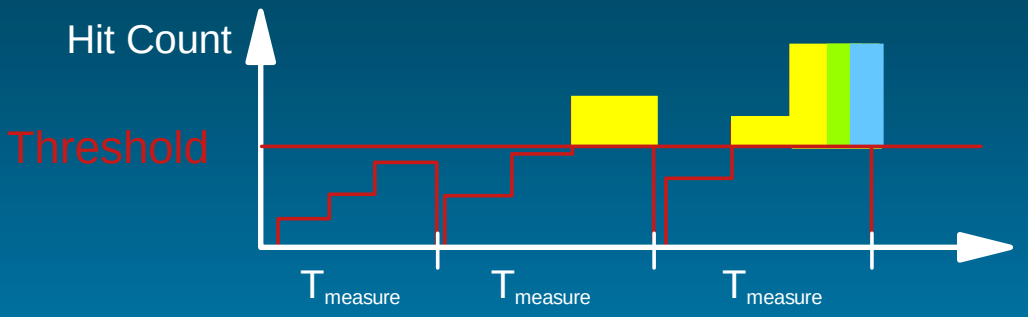
- Association of flows to network processors
  - Based on packet header
  - Two-stage approach



Emergency Mechanisms

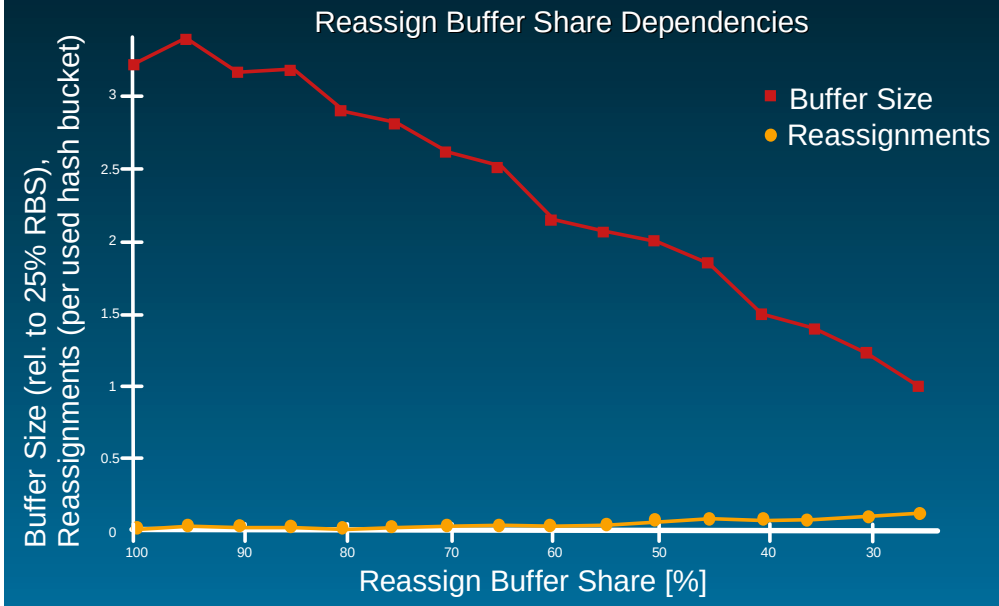


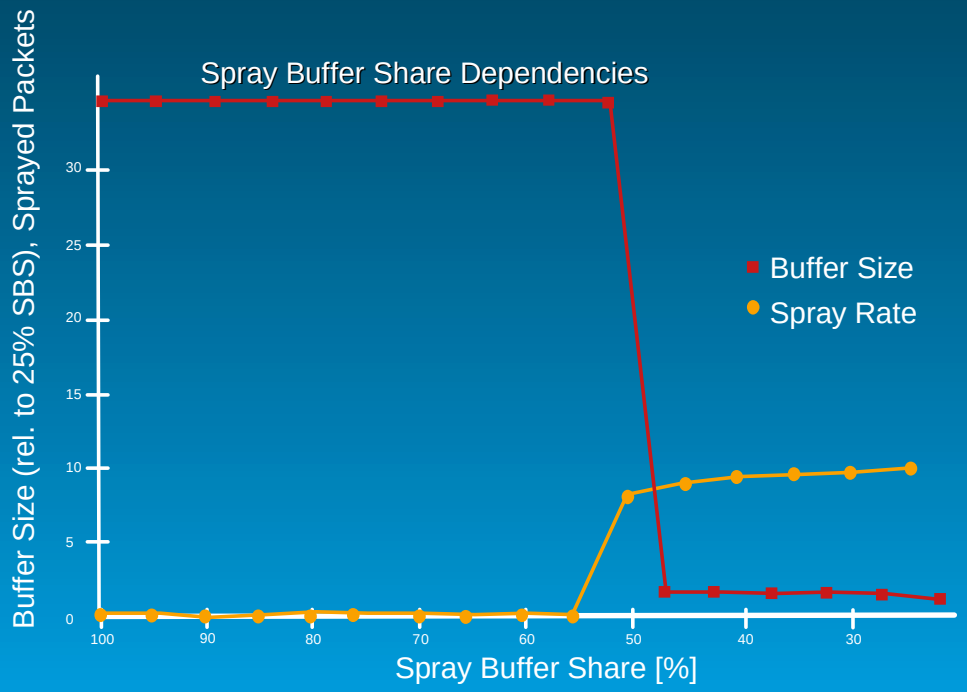
- **Spraying**
  - **Flow exceeds current capacity of NP**
  - **Send packets to shortest queue**



- **Reassignment**
  - **Unbalanced flow allocation**
  - **Send flow to shortest queue**

Analysis Results





Conclusion

- The Load Balancer is a flexible and fault tolerant solution for scaling network processor performance
- The scaling is achieved by distributing the load of a high-speed link in real-time onto concurrently operating, independent network processors in a flow-preserving manner.
- An association between each flow and a network processor is established through hashed packet header information proceeded by a table lookup operation. This enables flexible, traffic dependent reassociations.
- Emergency mechanisms, e.g. reassignment & spraying, which handle flows exceeding the capacity of the associated network processor or imbalances in flow allocation ensure the implementability with on-chip memory.
- The Load Balancer approach is currently in its implementation phase for full OC-768 line speed in a standard 0.18 μm CMOS process