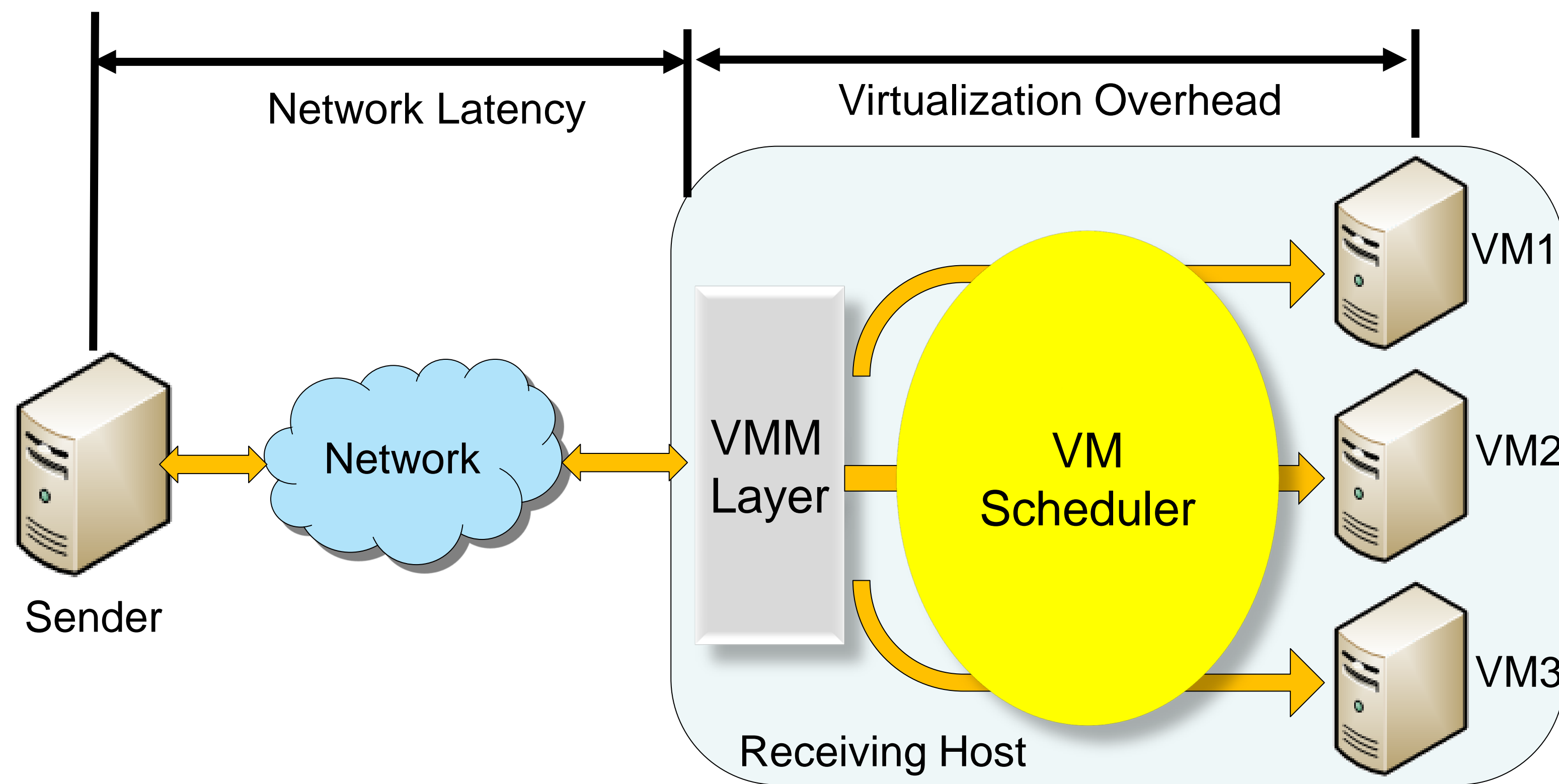
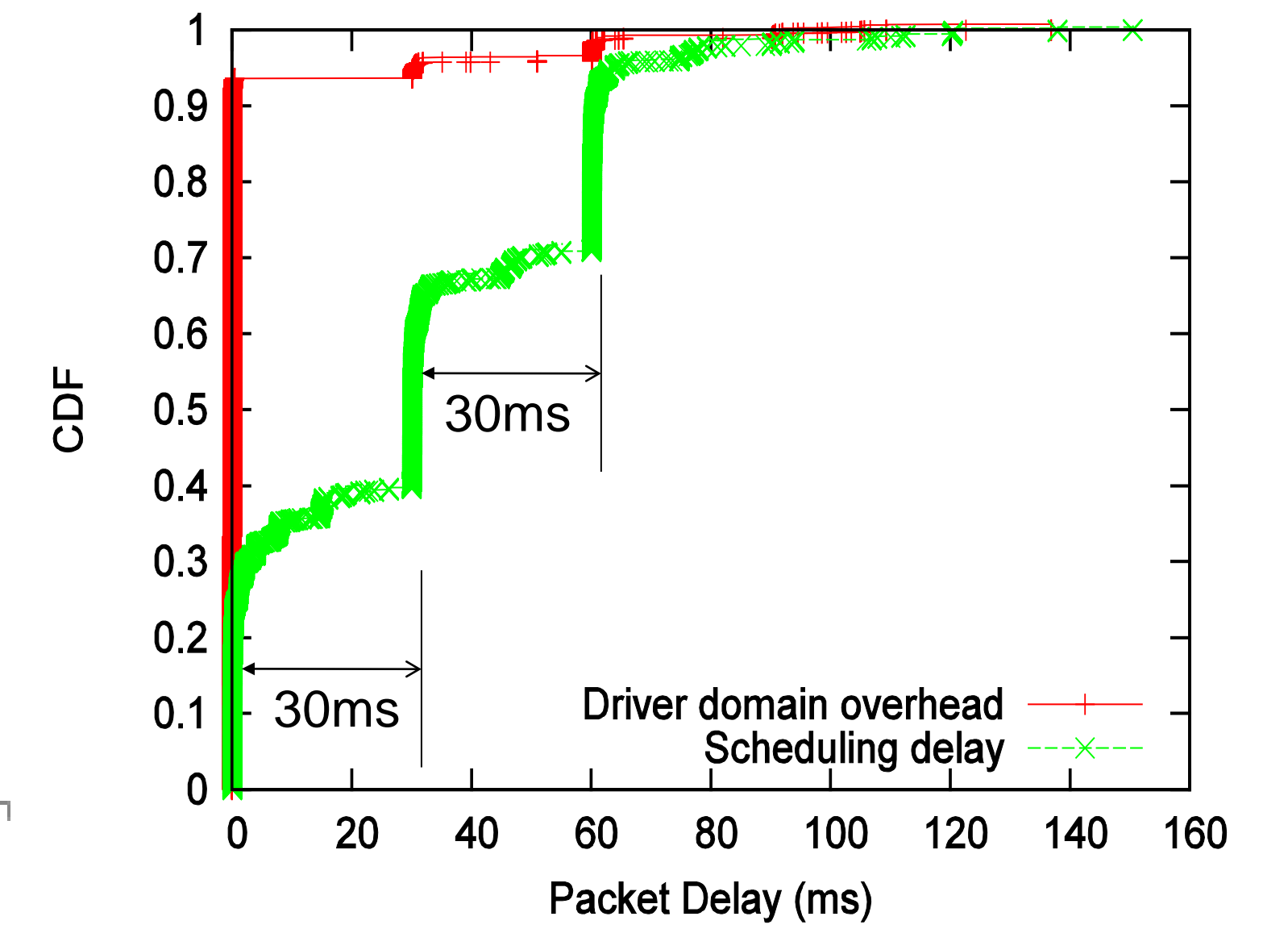
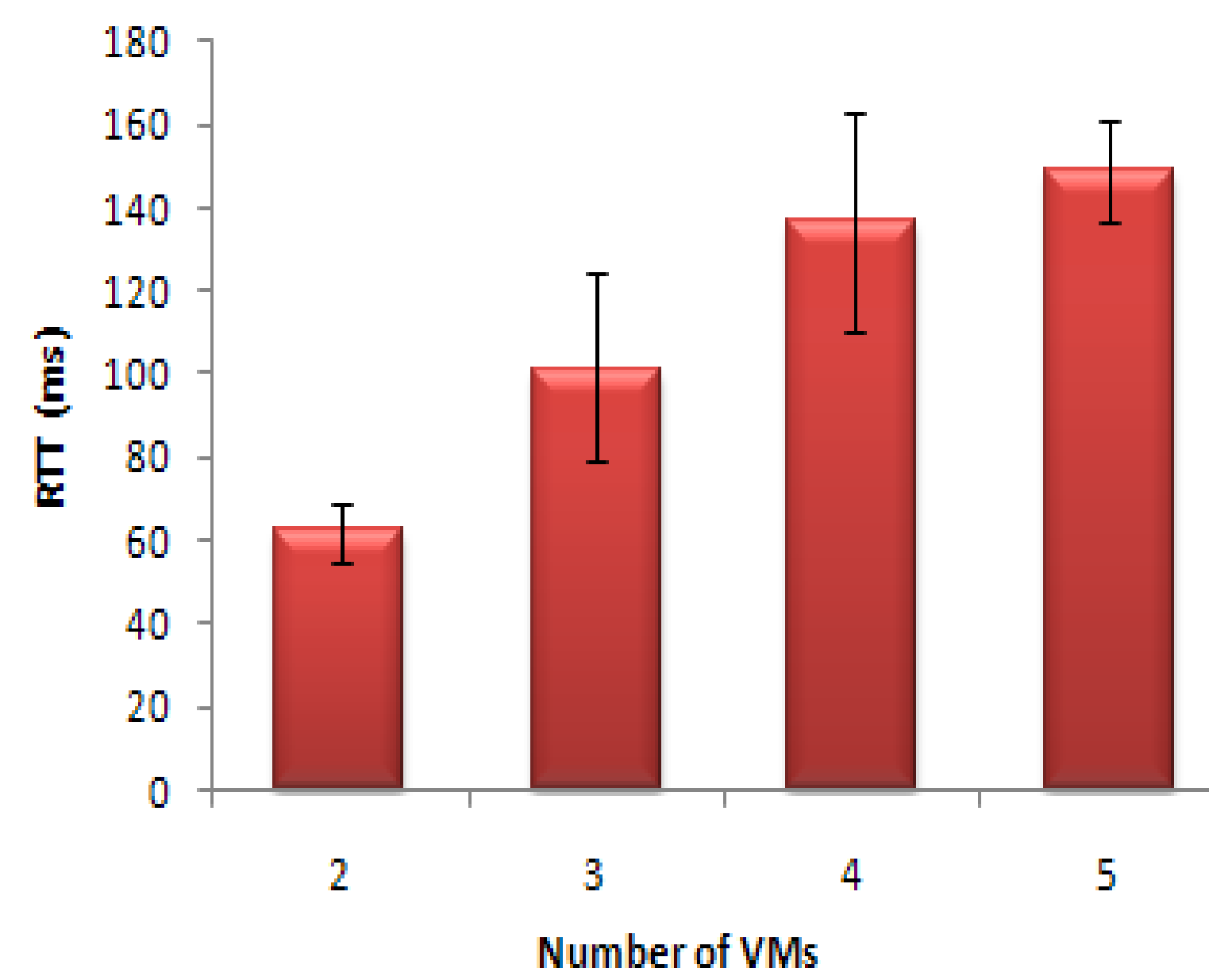


## Motivation



- ❑ Server consolidation → Sharing single core among many VMs
- ❑ Increased access latency to CPU due to VM scheduling
  - ❑ ACK generation for incoming TCP packets delayed
- ❑ Negatively impacts progression of TCP connections to the VM
- ❑ Datacenter environment : sub-millisecond network latency
  - ❑ Virtualization overhead dominates round trip time (RTT)

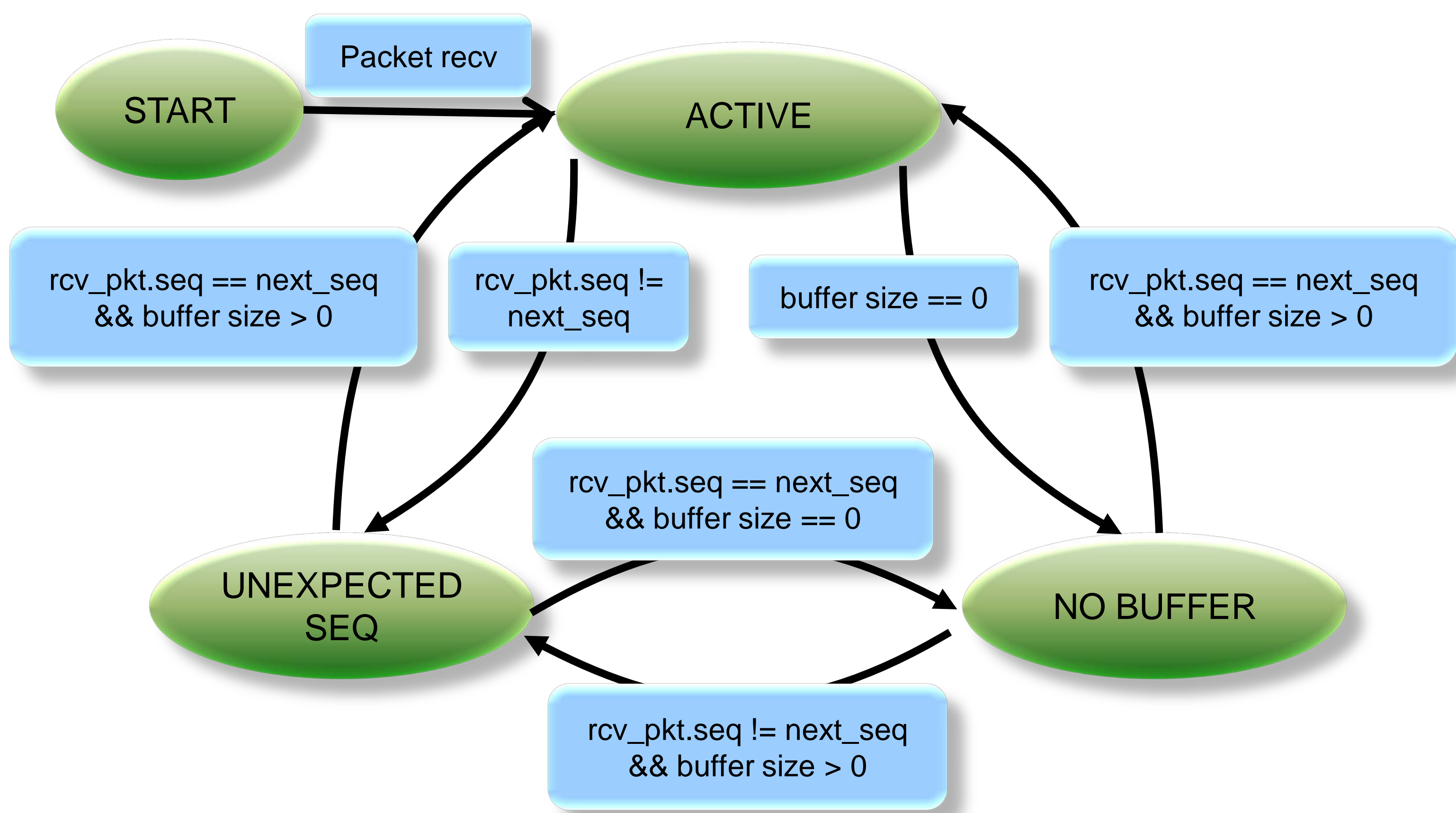
## Illustrative Experiment



- ❑ Average response time
  - ❑ 1000 ping requests to a VM
  - ❑ Each VM running 60% workload
  - ❑ Increase proportional to 30ms Xen's scheduling slice

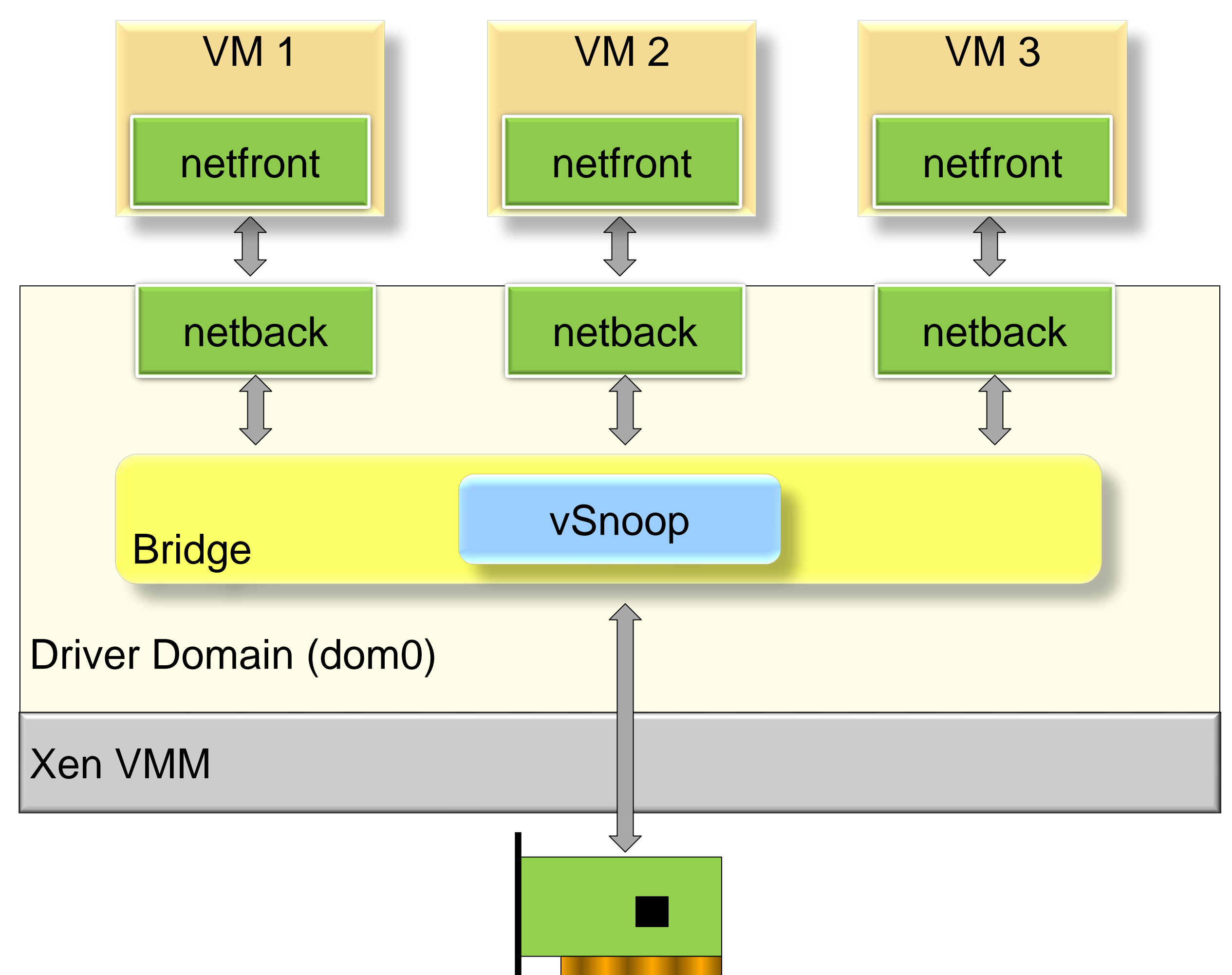
- ❑ Packet transfer delay: dom0 → VM
- ❑ CDF of the packet delay
  - ❑ Scheduling delay
  - ❑ Virtual device driver overhead
  - ❑ 30ms "jumps" in CDF

## vSnoop's State Machine



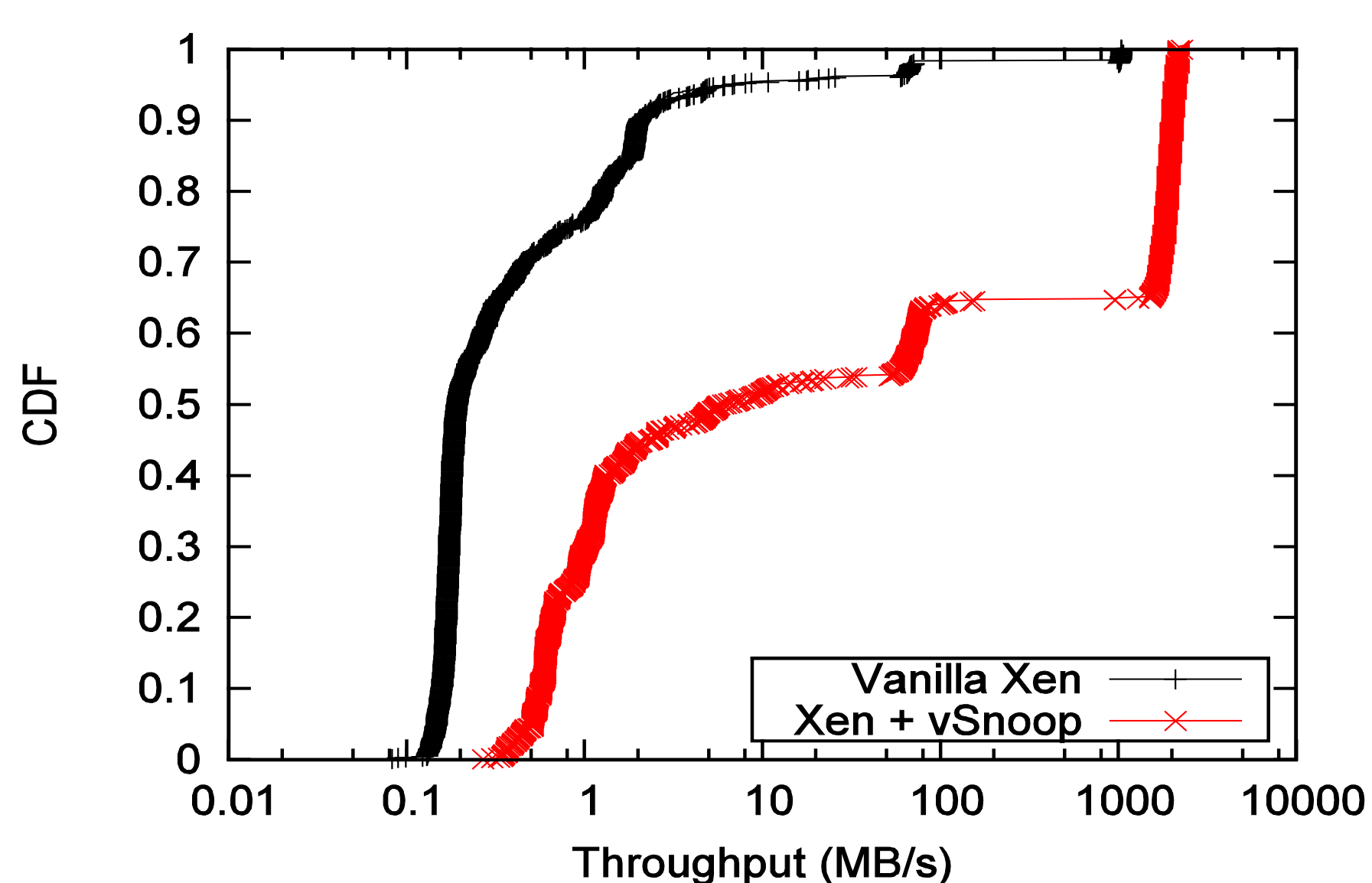
- ❑ Mask virtualization induced latencies by acknowledging at dom0
- ❑ Helps mainly receive performance
- ❑ Acknowledge only when it is safe to do so
  - ❑ Early acknowledgement should not violate TCP semantics
  - ❑ Enough buffer space should be guaranteed between dom0 and VM
- ❑ State machine maintained per flow basis
  - ❑ Next sequence number, VM's window size, Current ACK number, Current state of connection

## Xen-Based Prototype Implementation

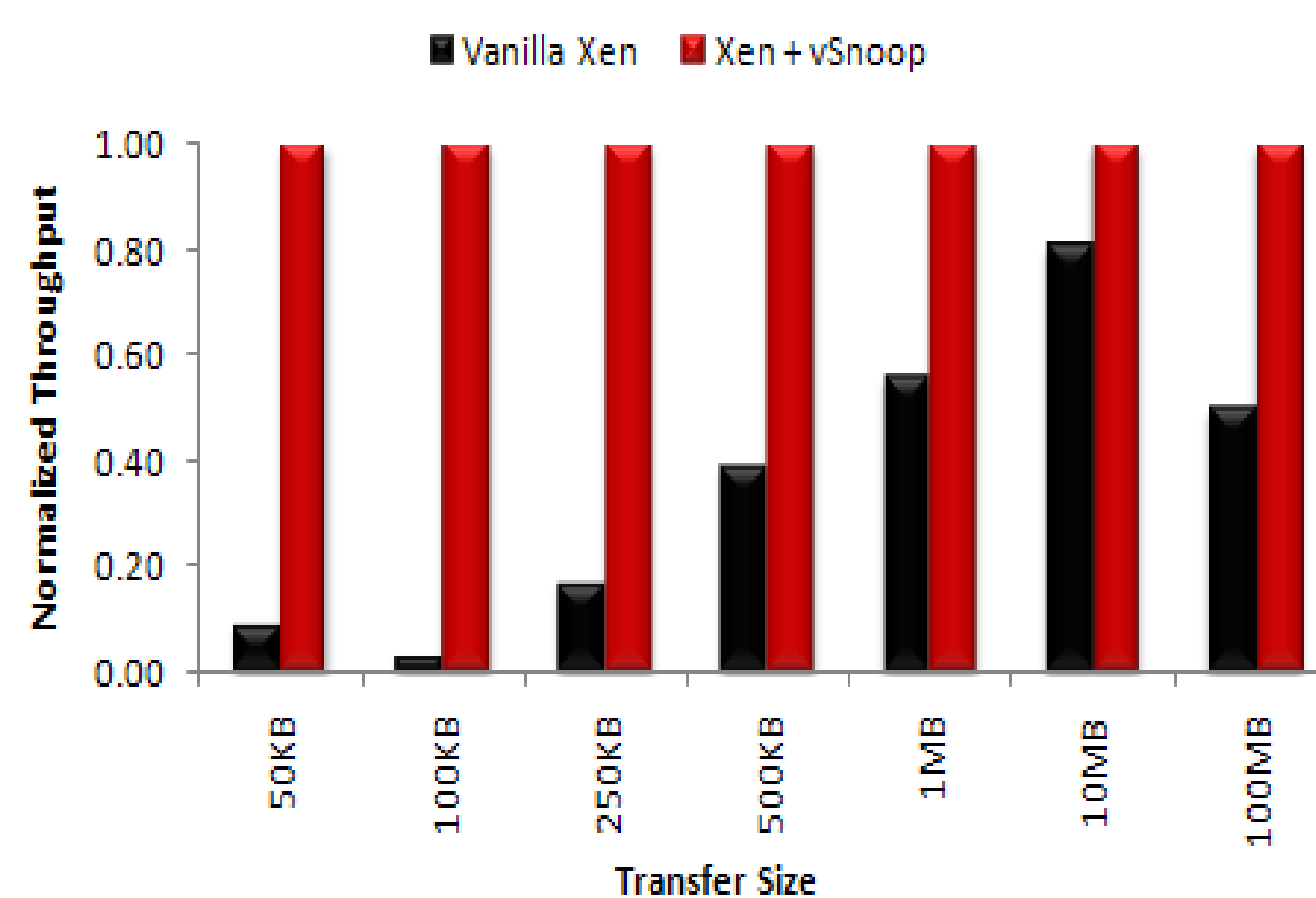


- ❑ Prototype implementation on Linux 2.6 on Xen 3.3
  - ❑ Two hook functions attached to Linux bridge
- ❑ Capable of handling Xen live migration
  - ❑ Target VM vSnoop enabled or not enabled
- ❑ Tuned Xen netfront driver to use up to 75% available slots

## TCP Throughput Improvement

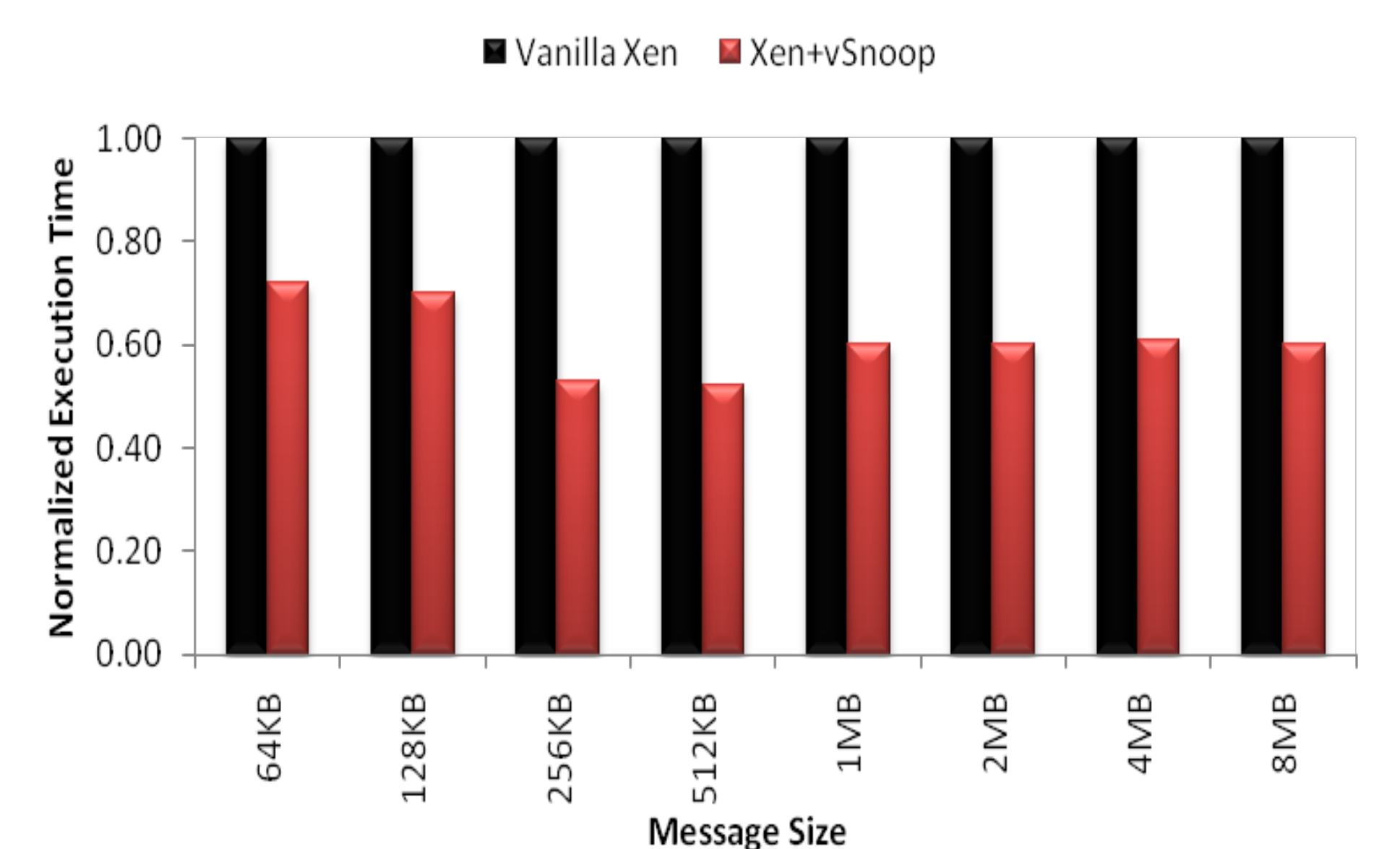


CDF of achieved throughput for 100KB transfer



Median improvement for each transfer sizes

## HPC Application Performance



Normalized execution time for Intel MPI benchmark

- ❑ TCP throughput measurements
  - ❑ 3 VMS are sharing the same core, 60% load each
  - ❑ 1000 transfers of each size
- ❑ Throughput may be higher than link rate due to buffering at VMM layer
- ❑ Up to 650% improvement for 100KB transfers

- ❑ Intel MPI Benchmark running All-to-All pattern
- ❑ 4 machines each having a 2 VMs
  - ❑ One VM running the benchmark
  - ❑ 1 other non-idle VM
- ❑ vSnoop : Up to 50% reduction in running time