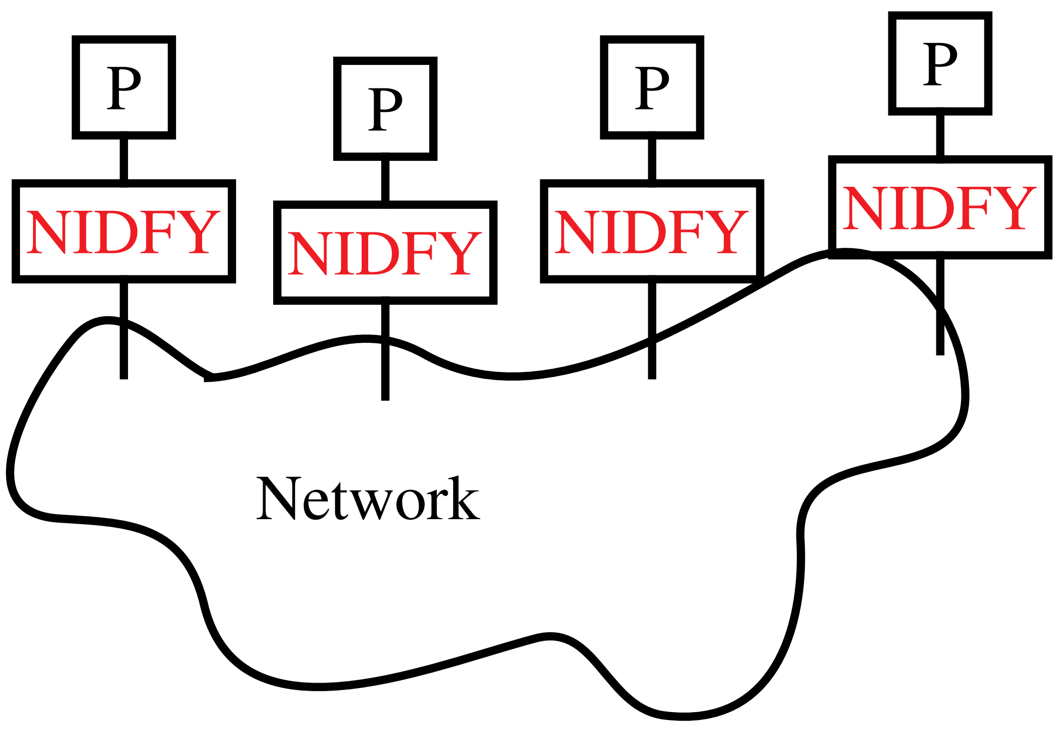
NIFDY

Network Interface for Flow Control and in-order DeliverY



Tim Callahan
Seth Copen Goldstein

Goals

- Congestion Control
  reduce hot-spots
  keep network in operating range
- In Order Delivery
  reduce software overhead
  increase payload

Keep changes to NI, not in network

Reduces resources required

Compare:
Metro
Compresionless Routing
AN2

Method

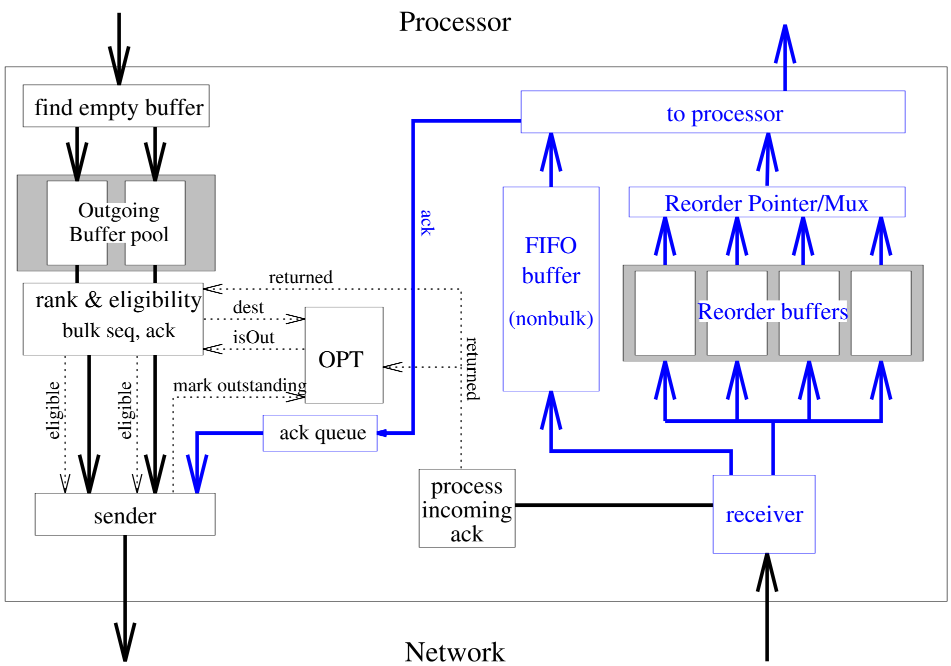
- Scalar Packets
  Allow only one outstanding packet to each destination.

- Bulk Packets
  Use a sliding window protocol

Hardware generates acks; which must be received before additional packets can be injected into the network.

A very simple idea.

The Design



- Single Ported Ram
- Small sized associative memory

Parameters

- Outstanding Packet Table
  # of outstanding packets allowed to different processors.
- Buffer Pool Size
  Used to reduce head-of-line blocking.
- Dialogs per Receiver
  Used for large transfers to overcome round-trip latency of acks.
- Window size per Dialog

Simulation

C++ objects distributed across the CM-5

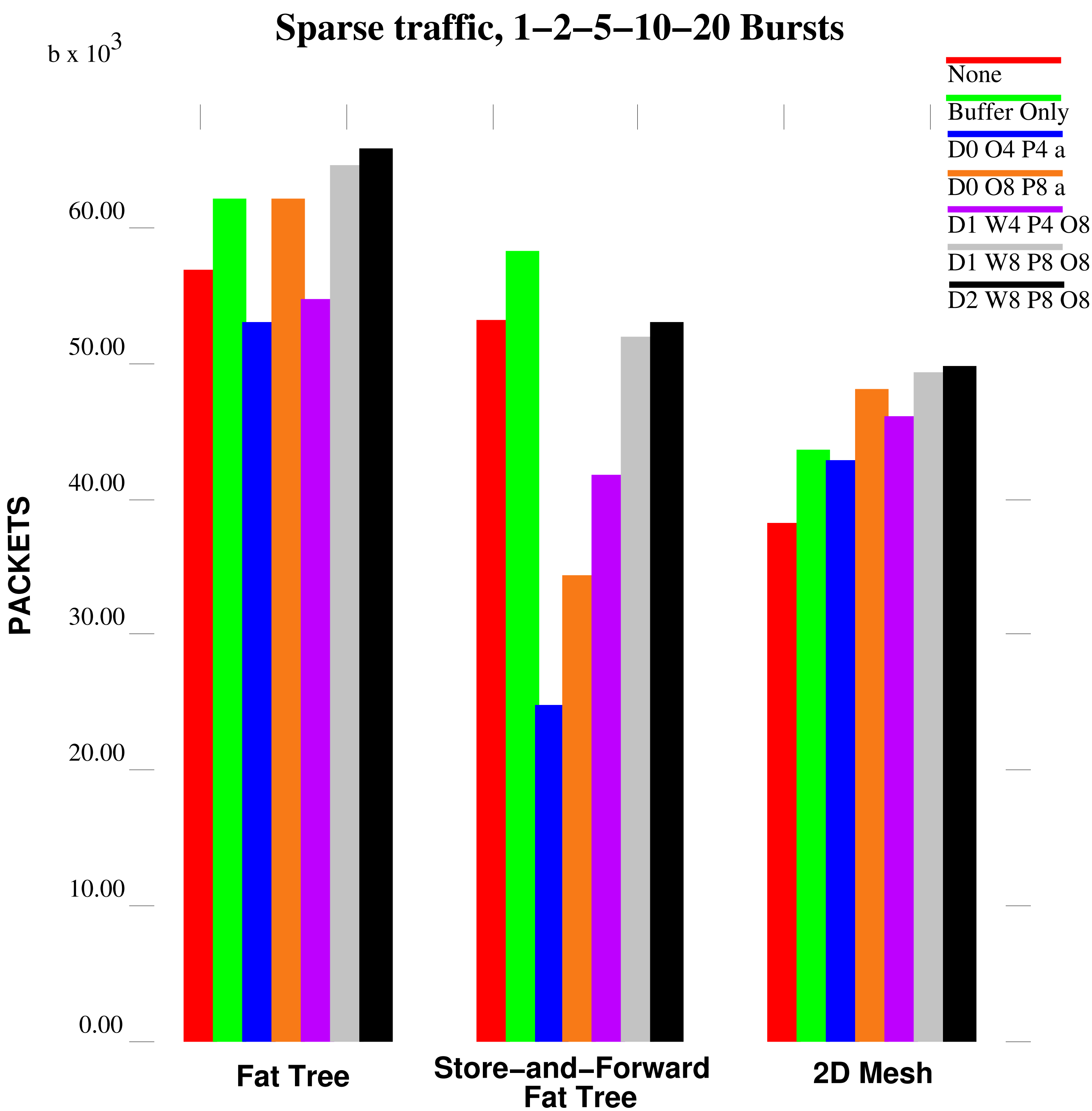
Traffic patterns:

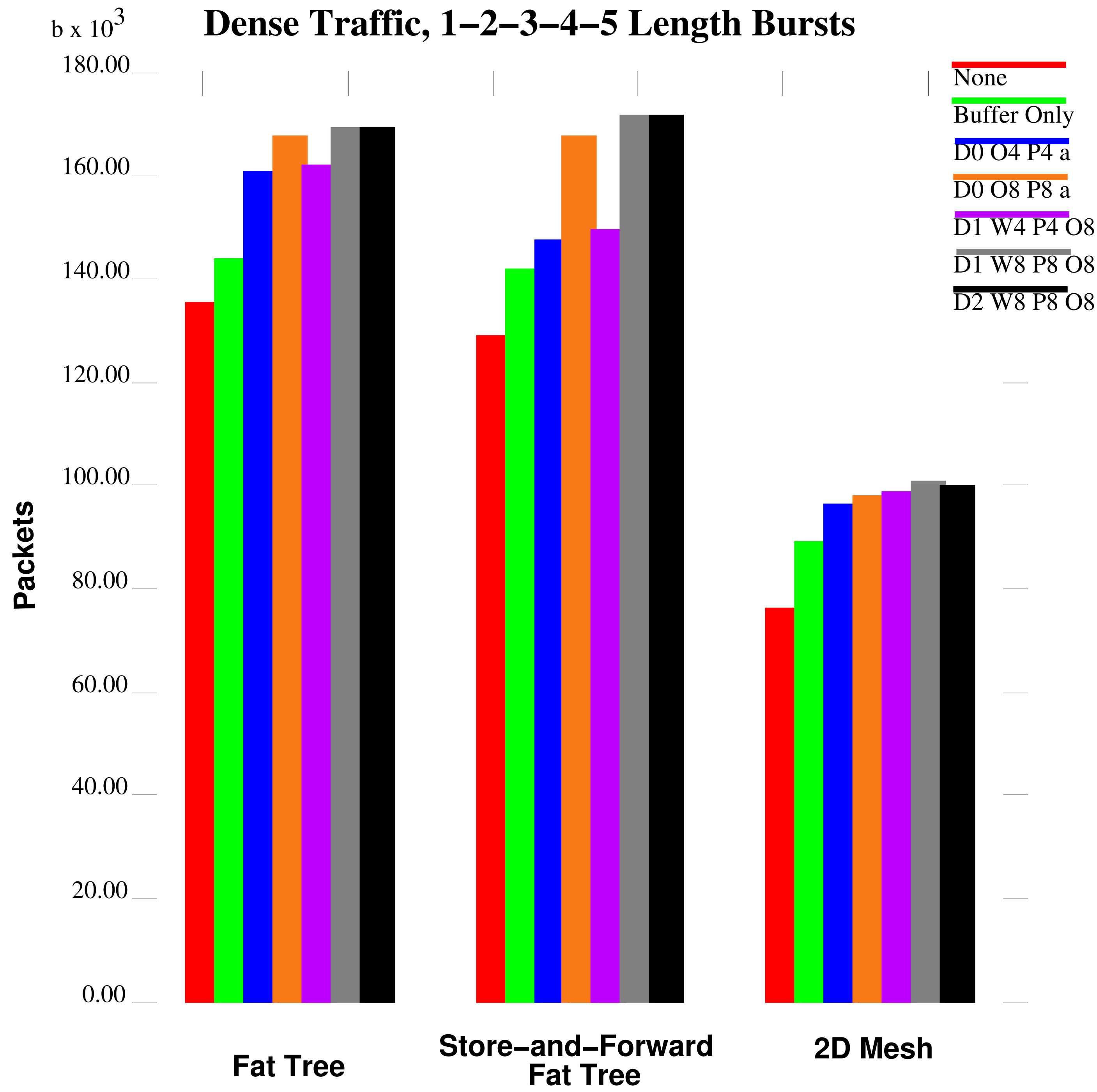
* "Random" - bursty, with occasional barriers
  to ensure all nodes make progress.
* C-shift
* Real applications --
  computation runs directly on CM-5 node,
  calls to network replaced with calls to
  network simulation

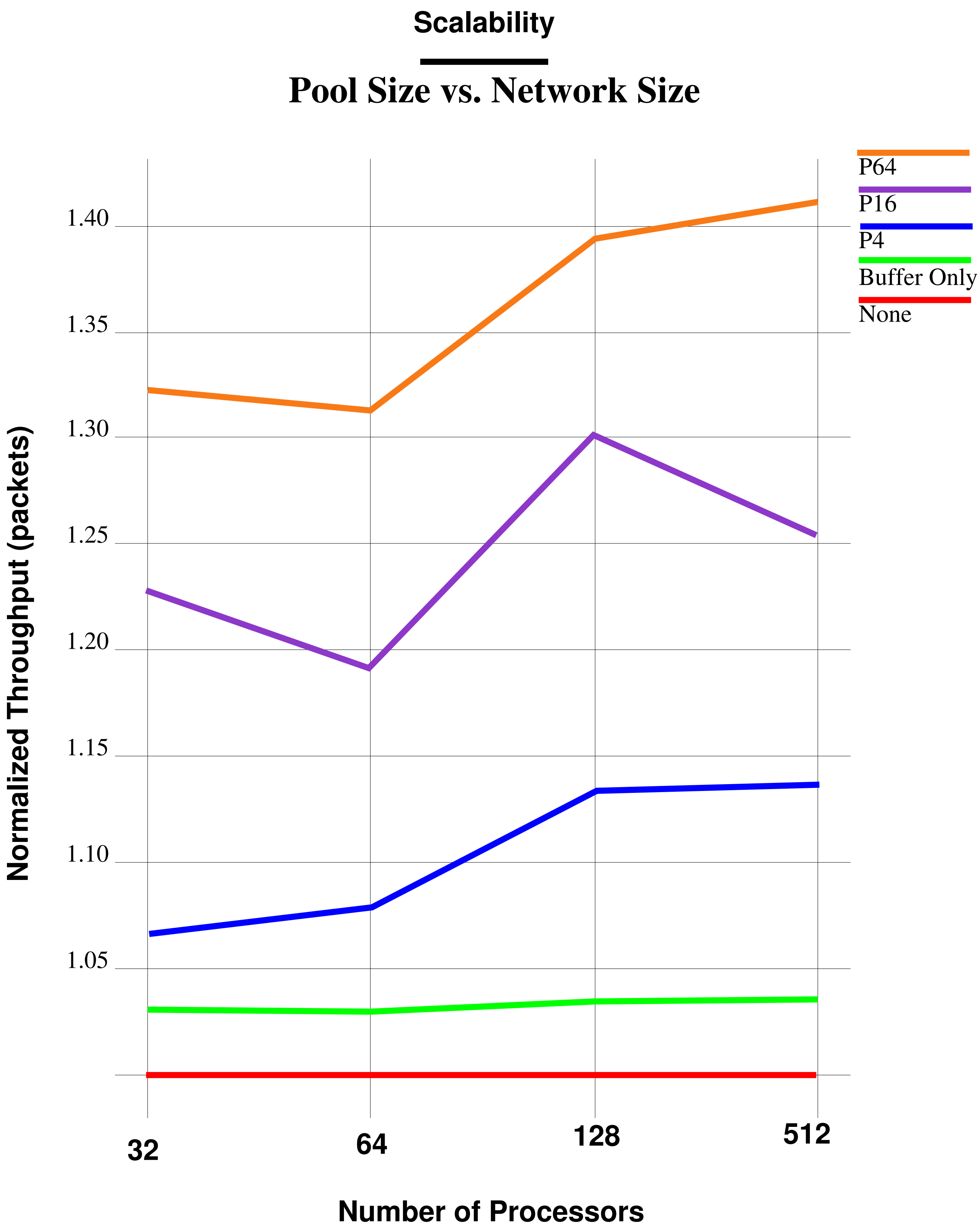
NIFDY Parameters : O, P, D, W

Networks:

* Full 4-ary fat tree
* CM-5 fat tree
* 2D, 3D meshes and tori w/ wormhole, VC routing
* All types of butterflies


Sparse traffic, 1−2−5−10−20 Bursts
b x 10^3
PACKETS
0.00
10.00
20.00
30.00
40.00
50.00
60.00
None
Buffer Only
D0 O4 P4 a
D0 O8 P8 a
D1 W4 P4 O8
D1 W8 P8 O8
D2 W8 P8 O8
Fat Tree
Store−and−Forward Fat Tree
2D Mesh


Dense Traffic, 1–2–3–4–5 Length Bursts
b x 10^3
180.00
160.00
140.00
120.00
100.00
80.00
60.00
40.00
20.00
0.00
Packets
None
Buffer Only
D0 O4 P4 a
D0 O8 P8 a
D1 W4 P4 O8
D1 W8 P8 O8
D2 W8 P8 O8
Fat Tree
Store–and–Forward Fat Tree
2D Mesh


Scalability
Pool Size vs. Network Size
P64
P16
P4
Buffer Only
None
1.40
1.35
1.30
1.25
1.20
1.15
1.10
1.05
Normalized Throughput (packets)
32
64
128
512
Number of Processors

NIFDY Summary

* NIFDY provides in-order delivery to the processor, reducing software overhead and increasing packet payload.

* NIFDY provides end-to-end flow control, reducing network congestion and increasing throughput at high loads.

* NIFDY is especially useful in irregular communication when software techniques such as those in STRATA cannot be applied.