Adaptive Connection Management for Scalable MPI over InfiniBand

Weikuan Yu, **Qi Gao**, Dhabaleswar K. Panda

Network-Based Computing Laboratory
Department of Computer Science & Engineering
The Ohio State University

Introduction

- Clusters for high performance computing are heading for ***Tens of Thousands*** nodes.

- InfiniBand: an open industrial standard for high speed interconnect.
  - Used by many large clusters in Top 500 list.

- MPI: the *de facto* standard for writing parallel programs

- Challenges and issues in scalability and manageability for MPI over InfiniBand become increasingly critical

InfiniBand Transportation Services

- InfiniBand supports 4 types of transport services

Reliable Connection (RC)	Unreliable Connection (UC)
Reliable Datagram (RD)	Unreliable Datagram (UD)

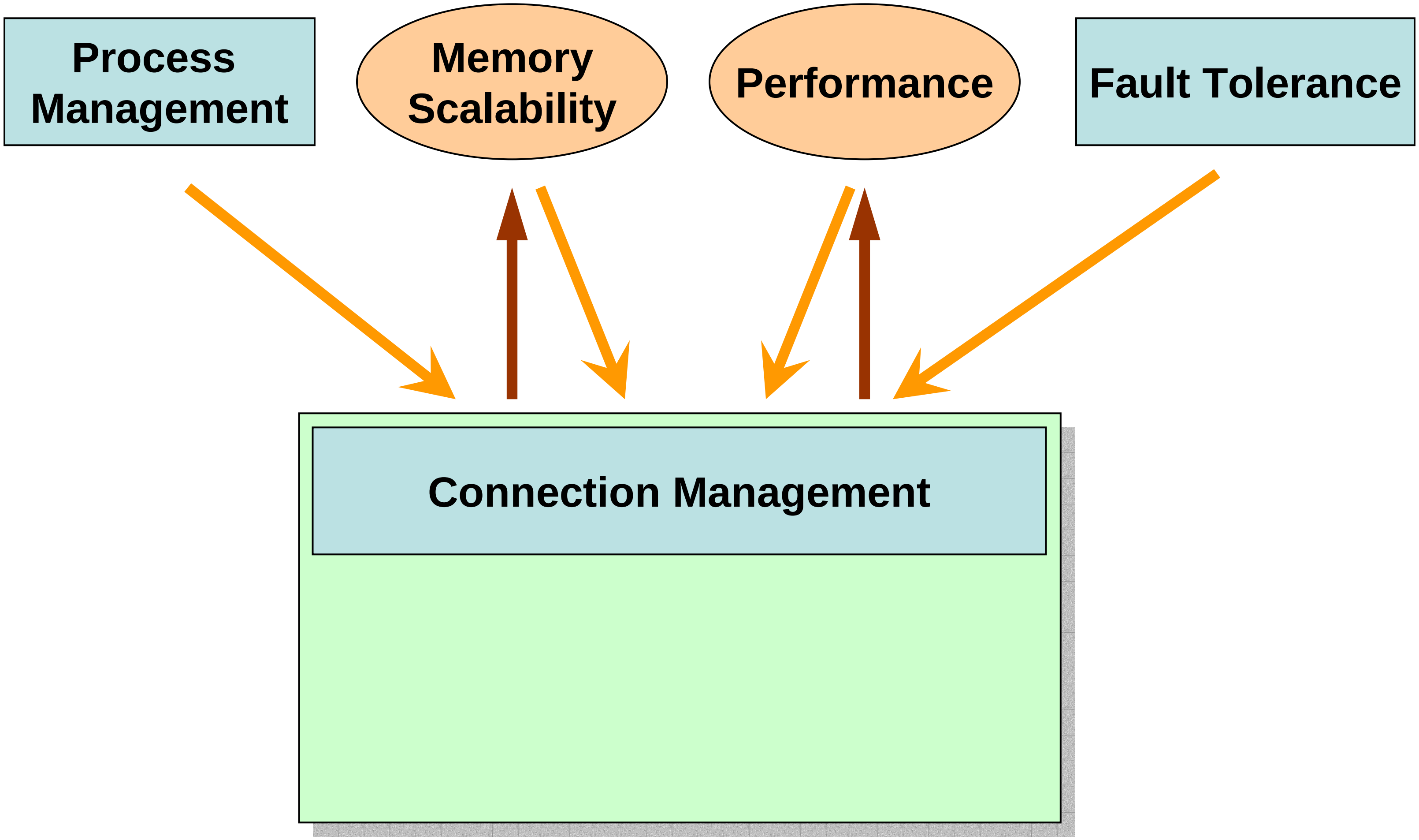
- MPI assumes all processes are logically connected
- To setup RC between each pair of processes:
  - RC connection: **~80KB**; associated buffers : **~200KB**
  - Connection-oriented model: **n-1** connections on each process for fully-connected **n** processes
- For 10,000-node clusters, on each process:
  - 9,999 RC connections: **~780 MB**
  - Buffers for these connections: **~1950 MB**

Requirements for Connections for MPI Applications

- How many peers does one MPI process communicate with?
  - J. S. Vetter et. al, in *IPDPS* 02
    - sPPM: average **5.67** for a **96**-process job.
    - Sweep3D: average **3.58** for a **96**-process job.
    - SMG2000: average **64.33** for a **96**-process job.
  - J. Wu et. al, in *Cluster* 02
    - CG: average **5.78** for a **32**-process job.
    - BT: average **9.83** for a **36**-process job.
    - MG: **31** for a **32**-process job.

- *On-demand connection management* had been proposed to reduce the number of connections.

Motivation for More Sophisticated Connection Management for MPI



Outline

- Introduction & Motivation
- Problem Statement
- Adaptive Connection Management
- Evaluation Framework
- Experimental Results
- Conclusion and Future Work

Problem Statement

- What are the issues involved in Connection Management?

- What are the possible schemes to manage connections?

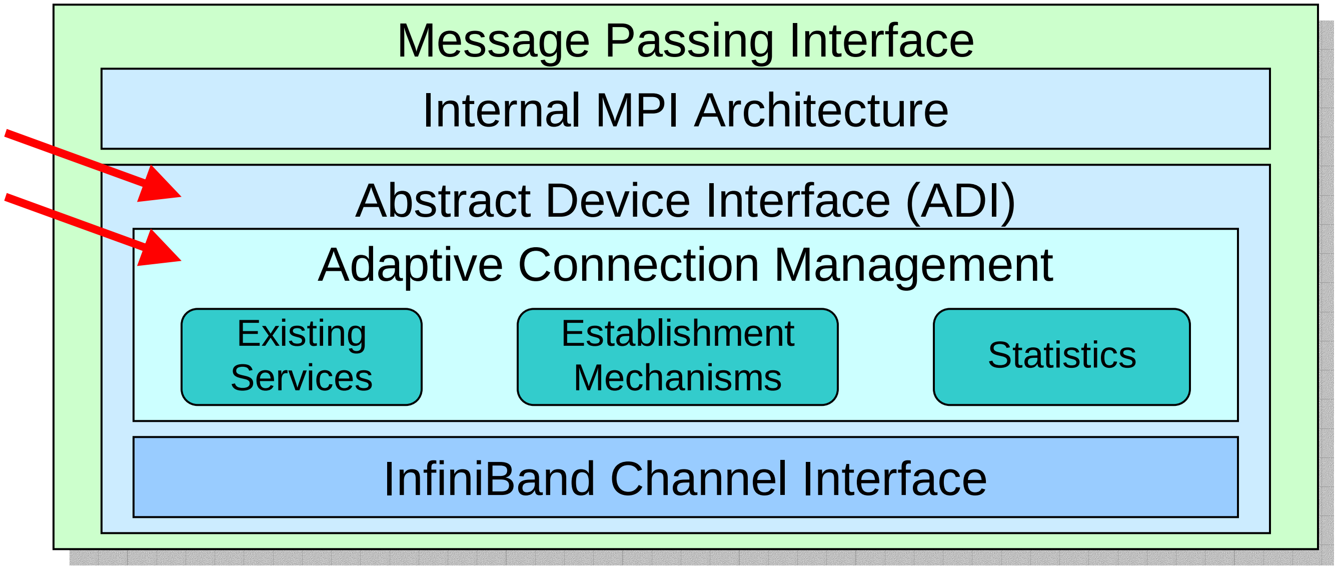
- What are the effects of these schemes on resource usage, performance, etc.?

Outline

- Introduction & Motivation
- Problem Statement
- Adaptive Connection Management
- Evaluation Framework
- Experimental Results
- Conclusion and Future Work

Adaptive Connection Management Model

- MPI should use different InfiniBand transport services according to the different requirements from applications.
  - For infrequent communications, connectionless model is used.
  - pt2pt connections are setup only when the processes communicate very frequently

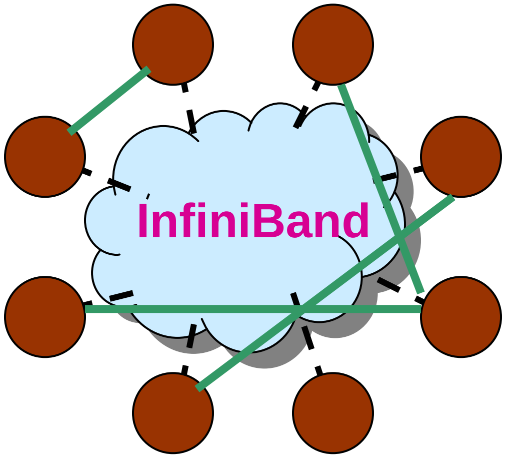

Message Passing Interface
Internal MPI Architecture
Abstract Device Interface (ADI)
Adaptive Connection Management
Existing Services
Establishment Mechanisms
Statistics
InfiniBand Channel Interface

Design Alternatives

- InfiniBand transport services
  - Pt2pt connected - Reliable Connection (RC)
  - Connectionless - Unreliable Datagram (UD)
- Mechanisms for connection establishment
  - UD-based 3-way handshake
  - InfiniBand Communication Management (IBCM)
- Connection management models
  - Any pt2pt connections are setup dynamically
  - Some pt2pt connections are setup in initialization time

Studied Schemes

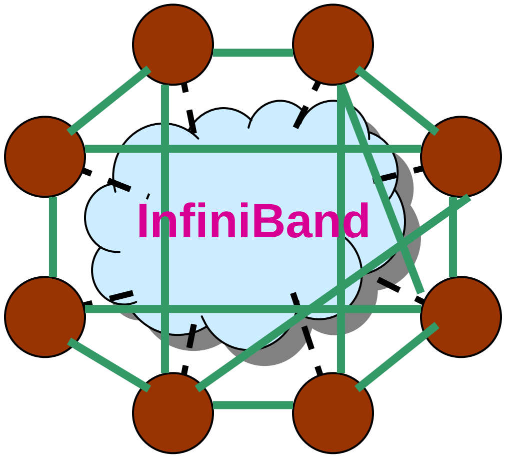
UD-FD



UD-based setup

Fully dynamic

UD-PS

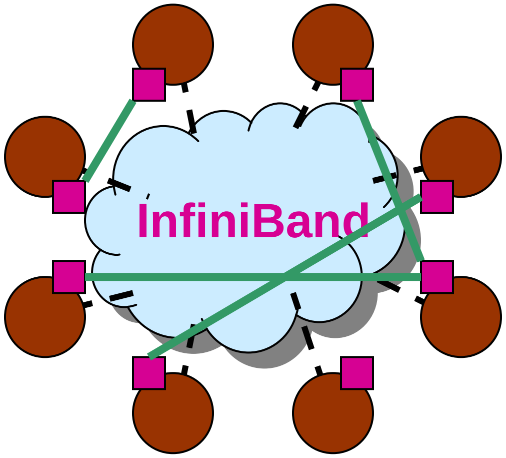


UD-based setup

Partial static

Actually, 2*log N -1 connections per process are setup to cover the need for collective algorithms

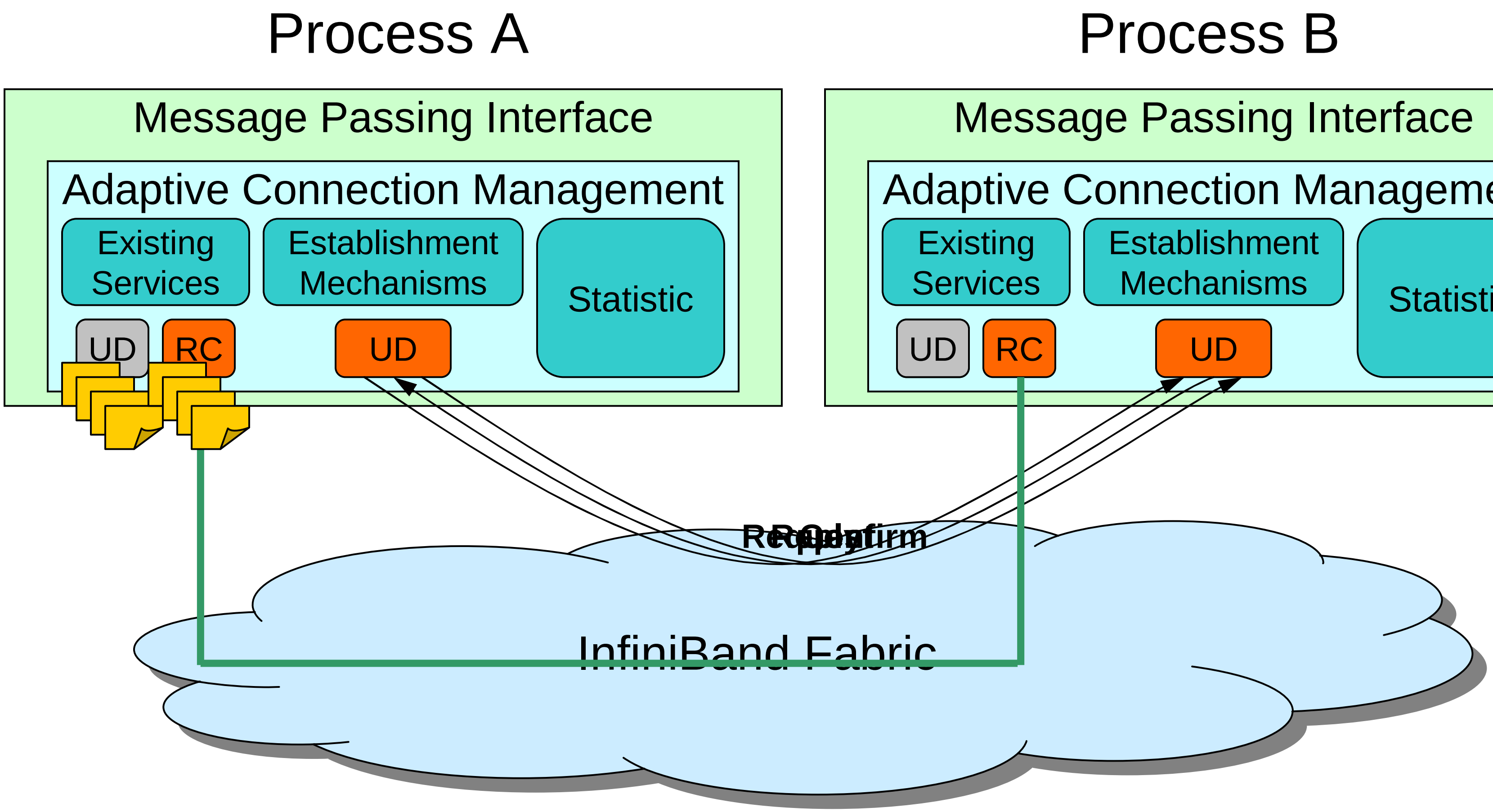
CM-FD



IBCM-based setup

Fully dynamic

Working Scenario



Outline

- Introduction & Motivation
- Problem Statement
- Adaptive Connection Management
- Evaluation Framework
- Experimental Results
- Conclusion and Future Work

OSU MPI over InfiniBand

- High Performance Implementations
  - MPI-1 (MVAPICH)
  - MPI-2 (MVAPICH2)
- Open Source (BSD licensing)
- Has enabled a large number of production IB clusters all over the world to take advantage of IB
  - Largest being Sandia Thunderbird Cluster (4000 node with 8000 processors)
- Have been directly downloaded and used by more than 345 organizations worldwide (in 30 countries)
  - Time tested and stable code base with novel features
- Available in software stack distributions of many vendors
- Available in the OpenIB/gen2 stack
- More details at

  http://nowlab.cse.ohio-state.edu/projects/mpi-iba/

Evaluation Framework

- Implemented based on MVAPICH version 0.9.5
- Will be released from MVAPICH version 0.9.8 onwards
- Test-bed:
  - Cluster A: 8 nodes, Dual Intel Xeon 2.4GHz processors, 1GB DRAM, PCI-X bus.
  - Cluster B: 8 nodes, Dual Intel Xeon 3.0GHz processors, 2GB DRAM, PCI-X bus.
  - Mellanox InfiniHost MT23108 HCA adapters through Mellanox InfiniScale 24 port switch MTS 2400
- Experiments:
  - Number of pt2pt connections
  - Startup memory usage
  - Initialization time
  - Performance impact on applications

Outline

- Introduction & Motivation
- Problem Statement
- Adaptive Connection Management
- Evaluation Framework
- Experimental Results
- Conclusion and Future Work

Average Number of pt2pt Connections for NAS Benchmarks

	SP	BT	MG	LU	IS	CG
Original	15	15	15	15	15	15
On-Demand*****	8	8	15	/	15	4.75
UD-PS	9.5	9.5	7	7	15	7.8
UD-FD/CM-FD	6	6	5	3.6	15	2.7

16-Process Test

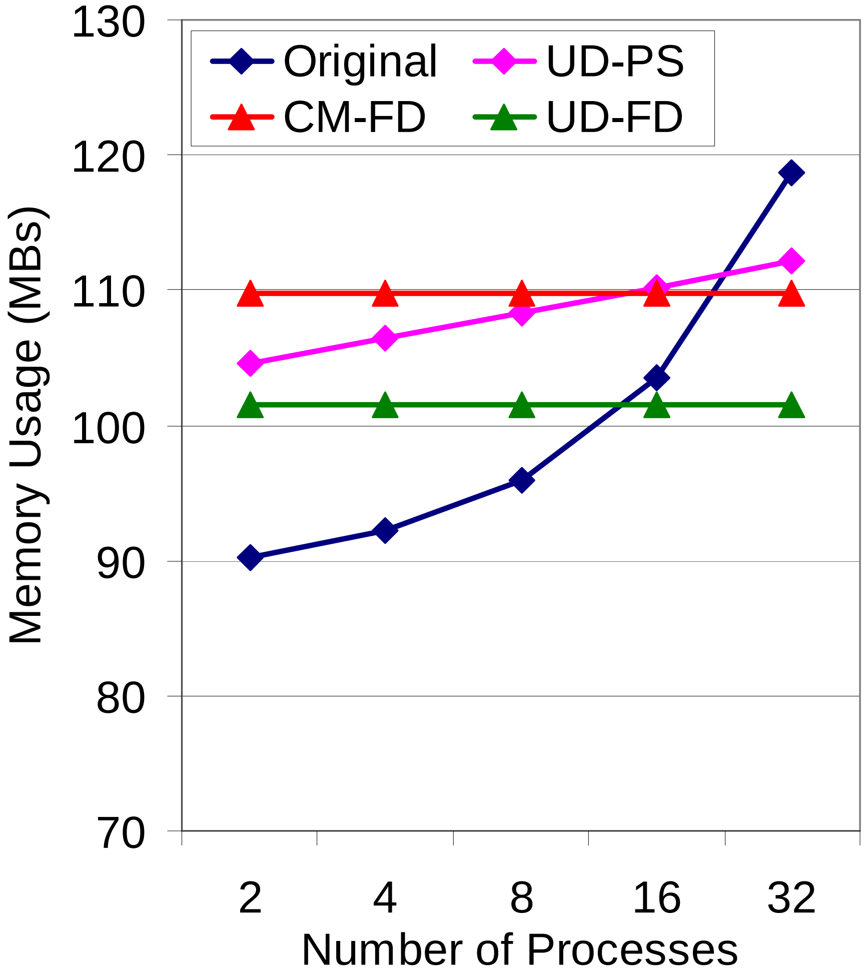
	MG	LU	IS	CG
Original	31	31	31	31
On-Demand*****	31	/	31	5.78
UD-PS	9.5	9	31	9.8
UD-FD/CM-FD	7	4.1	31	3.8

32-Process Test

In fully dynamic scheme, the number of pt2pt connections is further reduced from the On-demand scheme

* On-Demand numbers are from paper written by J. Wu et. al. for Cluster’02

Startup Memory Usage

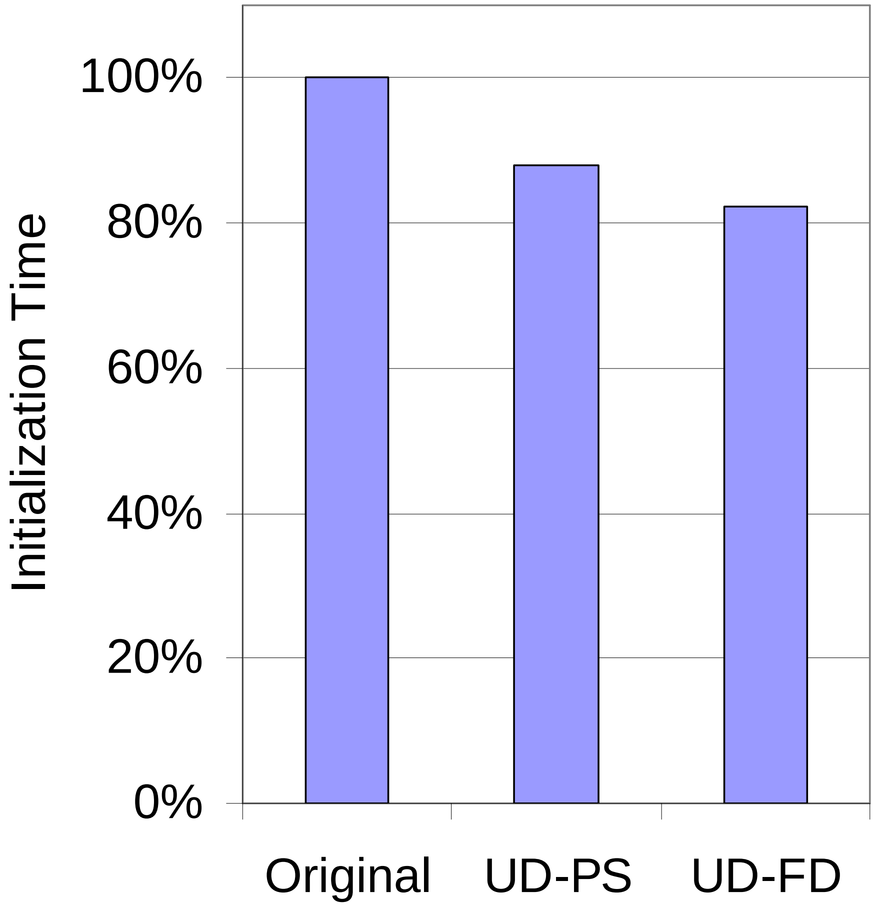


- Total memory usage of each MPI process
- Measured by *pmap* after MPI_Init()

For UD-PS, the startup memory usage increases logarithmically.

For UD-FD and CM-FD, the startup memory usage does not increase.

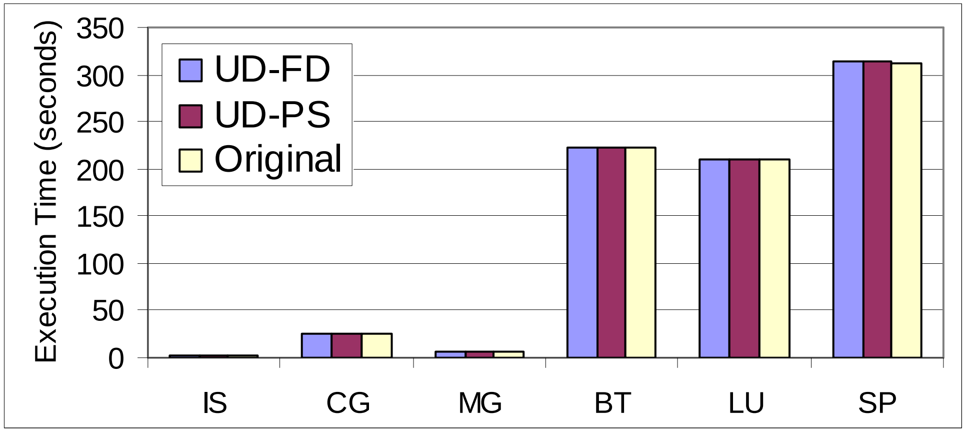
Initialization Time



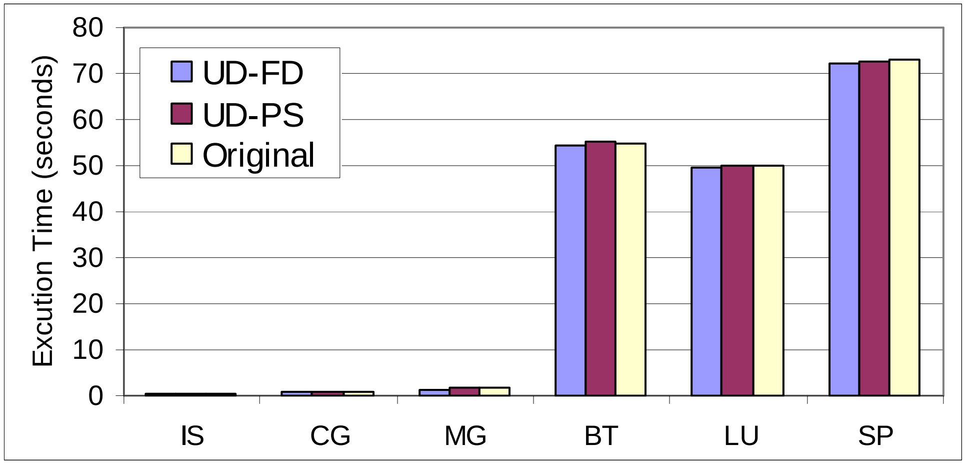
- Time for MPI_Init() of a 32-Process Job

New schemes reduce the initialization time for MPI jobs

Performance of NAS Benchmarks



Class B



Class A

- Execution Time for NAS Benchmarks.
- BT, SP on 16 processes
- IS, CG, MG, LU on 32 processes.

New schemes have almost same performance with much less resources.

Outline

- Introduction & Motivation
- Problem Statement
- Adaptive Connection Management
- Evaluation Framework
- Experimental Results
- Conclusion and Future Work

Conclusion and Future Work

- Studied the issues and design alternatives of connection management for MPI over InfiniBand
- Proposed an *Adaptive Connection Management* model with multiple schemes
- Experimental results show
  - Number of pt2pt connections is further reduced
  - Deliver almost same performance with much less resource usage

- Future work
  - Incorporate to MVAPICH release from version 0.9.8 onwards
  - Study more applications on larger clusters
  - Develop more sophisticated schemes
  - Support dynamic process management and fault tolerance

Acknowledgements

Our research is supported by the following organizations

- Current Funding support by

- Current Equipment support by

Web Pointers

http://nowlab.cse.ohio-state.edu/

MVAPICH Web Page

http://nowlab.cse.ohio-state.edu/projects/mpi-iba/