Building Multi-Petaflop Systems with MVAPICH2 and Global Arrays

ABHINAV VISHNU*, JEFFREY DAILY, BRUCE PALMER, HUBERTUS VAN DAM, KAROL KOWALSKI, DARREN KERBYSON, AND ADOLFY HOISIE

PACIFIC NORTHWEST NATIONAL LABORATORY

MVAPICH2 USER GROUP MEETING, AUGUST’ 2013
The Power Barrier

- Power consumption is growing dramatically
  - ~100 MW datacenters are real
  - Facebook, Google, Apple
  - DOE Labs
  - 2% of total US consumption is in datacenters
- 1 M$/ MW-year
  - Significant portion of total cost of ownership
- For the upcoming (2020+) Exascale machine, the power is expected to be between 20MW+
- Many architectural innovations
Workloads and Programming Models

- Program in multiple programming models
  - MPI + X+ Y …
  - Evolve the existing application to “newer architecture”

- Evolution of newer execution models
  - Task models (PLASMA, MAGMA, DaGuE)
  - With possible MPI runtime backend

- Workloads are changing dramatically
  - Examples: Graph500
The Role of MPI Moving Forward: Least Common Denominator

Applications

Solvers

New Execution Models/Runt.

MPI

OFA/RoCE

uGNI/DMAPP

PAMI

+X, Y, Z

August 26, 2013
Examples for X: Global Arrays, UPC, CAF

- A distributed-shared object programming model
  - Shared data view
  - One-sided communication models
  - Used in wide variety of applications
  - Global Arrays - Computational Chemistry
    - NWChem, molcas, molpro …
  - Bioinformatics
    - ScalaBLAST
  - Machine Learning
    - Extreme Scale Data Analysis Library (xDAL)

Physically distributed data

Global Address Space
Global Arrays and MVAPICH2 at PNNL

- PNNL Institutional Computing (PIC) cluster
- PIC:
  - 604 AMD Barcelona, IB QDR systems
  - MVAPICH2 default MPI in many cases
  - Global Arrays uses Verbs for Put/Get and MPI for collectives
  - Continuously updated with stable and beta releases
- Primary Workloads
  - NWChem
  - WRF
  - Climate (CCSM)
HPCS4: Upcoming Multi-Petaflop System@PNNL

- 1440 Dual socket Intel Sandybridge @ 2.6 Ghz
- 128 GB memory/node
  - Requirements from compute intensive/data intensive applications such as NWChem
- 2 Xeon Phi / node, and 8 GB / Xeon Phi
- Mellanox InfiniBand FDR
- Theoretical peak / node: 2.35 TF
- 3.4 PF/s peak performance
- Expected efficiency on LINPACK: > 70%
- Power consumption < 1.5 MW
- Software Ecosystem:
  - MVAPICH2, Global Arrays, Intel-MPI
- Applications
  - NWChem, WRF

August 26, 2013
As a User of MVAPICH2, What I Like (with Feedback from Others at the Lab)!

- **Job startup time**
  - Very important at large scale
  - Consistent improvement over the releases

- **Collective communication performance**
  - Allreduce scales very well
  - Primary collective operation in many applications

- **Low memory footprint**
  - Numerous optimizations over the years

- **A remark on collectives:**
  - Collective performance with OS noise
  - Problems with a non-customized kernel
  - Liu-IPDPS’04, Mamidala-Cluster’05
    - The theme was collectives with process skew
    - OS noise introduces similar issues
As a User of MVAPICH2, What would I like ..

- Scalable Performance
- Reliability
- Power Efficiency
- Efficient Integration with X, Y, Z...

MVAPICH2

2001-
Position Statement 1: Power

Why should software do it?
- Amdahl’s law: Once power optimized hardware is available, the relative power inefficiency of software grows proportionally.

Why should MPI do it (as well)?
- It knows a lot about application indirectly!
- The key is to automatically discover patterns.

Previous work is an important step, albeit with limitations
- Kandalla et al. – ICPP’09
- Efficiency for the size of data movement
  - Most collectives are small size
  - Some applications work on state transitions using pt-to-pt messages.
Example: Neutron Transport @ Los Alamos

Sweep 3D – Neutron transport

The “Slack”
Sedov blast problem
Deposit Force at origin, and see the impact on material
A small subset of processes perform useful work
Difficult to define a static model for deformation
Significant potential for Power efficiency
Very different from Sweep3D problem
+ non-determinism

Hardware
- Lots of dynamism/uncertainty due to near threshold voltage execution
  - The dark silicon effect
- Different chips perform varied levels of bit-correction
  - Difficult to capture on a static performance model

Software
- MPI + X, X is expected to be an adaptive programming model
- Implication is that MPI needs to handle non-temporal patterns in communication
- OS Noise

Possible R&D
- There are clear examples
- The key is to automatically discover communication patterns
- Potential for significant energy savings without one-off solution for a single application
Position 2: Effective Integration with “Other” Programming Models

- MPI + X (+ Y + ..)
- Shared Address Space examples of X
  - OpenMP, Pthreads, Intel TBB ..
- Classical data and Work decomposition
  - MPI_THREAD_MULTIPLE is not required, lower level models would do
- When multiple threads make MPI calls
  - Symmetric communication
- For MPI runtime, the challenge is MPI progress on multiple threads
  - Handling the critical section
Multi-threaded MVAPICH2 Performance

- Dinan et al. – EuroMPI’13
- Provide an endpoint to each thread
- Separate communicators for each thread
  - Prevents Thread Local Storage (TLS) conflicts between threads
- Problem:
  - Space complexity
  - Even harder with over-subscription (Charm++)
- Possible solution partly in specification, and other in runtime
Conclusions and Future Directions

- MPI will continue to be the least common denominator
  - Significant investment in applications
  - Newer execution models are adapting MPI for designing their runtimes
- MVAPICH2 has many features for scalable performance
- Larger scale systems would need MPI to be Power Efficient
  - The algorithms are different from each other
  - The architecture is expected to be radically different
  - Perfect recipe for advanced research and development
- Efficient support in MPI + X model would be critical
  - New execution models are looking forward to using MPI runtime with “revolutionary approaches”
  - The research space is very open
Thanks for Listening

Abhinav Vishnu, abhinav.vishnu@pnnl.gov