Lawrence Berkeley National Laboratory

Recent Work

Title
Poster reception---Optimized collectives for PGAS languages with one-sided communication

Permalink
https://escholarship.org/uc/item/2p00516t

ISBN
9780769527000

Authors
Bonachea, Dan
Hargrove, Paul
Nishtala, Rajesh
et al.

Publication Date
2006

DOI
10.1145/1188455.1188604

Peer reviewed
Optimized Collectives for PGAS Languages with One-Sided Communication

Dan Bonachea, Rajesh Nishtala, Paul Hargrove, Mike Welcome, Kathy Yelick

Partitioned Global Address Space Languages

- Partitioned Global Address Space (PGAS) Languages
  - Global pointers and distributed arrays
  - User-controlled access to data array codes
  - Communicate using implicit reads & writes of remote memory

- Languages: UPC, Titanium, Co-Array Fortran
  - Productivity benefits of shared-memory programming
  - Competitive performance on distributed memory
  - Use Single Program Multiple Data (SPMD) control
  - Fixed number of compute threads
  - Global synchronization, barriers, collectives
  - Exploit fast one-sided communication
  - Individual accesses and bulk copies
  - Berkeley implementations use GASNet

GASNet Portability

- Native network hardware support:
  - Quadrics QM6Ex (Irix,Elan4)
  - Clx X1 - Gray shmem
  - SGI I10 - 3G shmem
  - Clx X3 - Gray Ports (intranet)
  - Dolphin - SCI
  - Infineon - Marvell VAPI
  - Meyrin Myrinet - GM-1 and GM-2
  - IBM Colony and Federation - LAPI
  - Portable network support:
    - Ethernet, UDP works with any TCP/IP
    - MPI 1: portable imt for other HPC systems
    - Berkeley UPC, Titanium & GASNet highly portable
    - Runtime and generated code all ANSI C
      - New platform ports in 2-3 days
      - New network hardware 2-3 weeks
    - CPUs: all, Itanium, Opteron, Athlon, Alpha
      - PowerPC, MIPS, PA-RISC, SPARC, T3E, X-1, 3K-6
    - Other Unix, FreeBSD, NetBSD, Tru64, AIX, IRIX, HPUX, Solaris, MS-Windows/Cygwin
      - Mac OSX, Unicos, BeOS/Catamount, BlueGene

GASNet on the Cray XT3

- GASNet Put/Get operations implemented over Portals Put/Get
  - Remote access region covered by Portals Memory Descriptor
  - Portals Events used for GASNet operation completion
  - Put/Get injection throttled to prevent local event queue overflow
  - No remote thread generation
  - Local Put source and Get destination regions:
    - copied through pre-pinned bounce buffers for small messages
  - Performance of local remote messages = 1 MB

- GASNet Active Message layer currently prototyped over MPI
  - Port to native Portals-based AM is underway
  - UPC LU Application:
    - of remote memory
  - Superset of collective support in UPC and Titanium languages
  - Extensible to variable-contribution and teams-based subset collectives
  - Global and private address space
  - GASNet Communication System
    - Per-thread sync: data has affinity to producer or consumer (MPI style)
    - Global sync: barrier-like data sync (more efficient than full barrier)
    - Extensible infrastructure to allow for future auto-tuning features

- Productivity benefits of shared-memory programming
  - Exploit fast one-sided communication
  - One-sided, lightweight communication semantics
  - Eliminates masking & redundant overheads
  - Better match to hardware capabilities
  - Leverages widespread availability support for remote direct memory access
  - Active Messages support provides extensibility

- Communication micro-benchmarks show GASNet consistently matches or outperforms MPI
  - One-sided, lightweight communication semantics
  - Eliminates masking & redundant overheads
  - Better match to hardware capabilities
  - Leverages widespread availability support for remote direct memory access
  - Active Messages support provides extensibility

- Algorithm selection based on hardware characteristics & network state
  - Eventually enable auto-tuning – find best algorithm & params
  - Leverage hardware collective support (eg. hardware broadcast)

- GASNet put+ack consistently matches or outperforms MPI
  - Better match to hardware capabilities

- Active Messages support provides extensibility
  - Enables network specific tuning & optimization
  - Algorithm selection based on hardware characteristics & network state
  - Eventually enable auto-tuning – find best algorithm & params
  - Leverage hardware collective support (eg. hardware broadcast)

- GASNet put_nb bulk, slab-based and pencil-based implementations - Slab best on XT3
  - New platform ports in 2-3 days
  - New network hardware 2-3 weeks

- Leverage hardware collective support (eg. hardware broadcast)
  - Supports multiple overlapped collectives of any variety

- NAS Fortran/MPI
  - Bandwidth (MB/sec)
  - InfiniBand / Opteron / 64
  - Roundtrip Latency Comparison
  - Time (microseconds)

- NAS Fortran/MPI
  - Message Size (bytes)
  - InfiniBand / Opteron / 64
  - Time (microseconds)

- InfiniBand / G5
  - Bandwidth (MB/sec)
  - Message Size (bytes)

- Cray XT3
  - Performance Comparison
  - Time (microseconds)
  - Transfer Size (Bytes)

- Cray XT3 Porting characterization
  - Data Perpetual Rate (MB/s)
  - Data Perpetual Rate (MB/s)