CMSC 714
Lecture 8
SGI Origin 2000 and UV

Alan Sussman

Notes

- MPI project due tomorrow at 6PM
  - still having problems with job queues on deepthought2?
- OpenMP project posted by Monday

Shared Memory Multiprocessors

- Cache coherence
  - to keep different copies of same memory location (data block) the same
  - caching causes the problem, but is needed for performance
- Snooping vs. directory-based coherence
  - shared medium (bus or switched network) vs. distributed directory to keep track of shared data blocks
  - either way, all memory accesses are to local copies near a processor, and data blocks change state and move around to where they are needed
  - state of each block kept track of with a finite state machine (shared, exclusive, read-only, etc.)

SGI Origin 2000

- Scalable distributed shared memory (DSM) machine
  - from small building blocks, so scale up and down
- Each node is a dual-processor machine, with access to local memory, interconnection network and I/O system
- Nodes connected via “bristled” fat hypercube network
- Cache coherence maintained via directory that keeps track of each data block (page)
  - both the state of the cache block, and where copies are located
  - protocol appears complicated, but all implemented in hardware, so usually fast – big problem is transitioning to exclusive state for writes, to invalidate copies and TLB entries
  - supports migrating whole pages across nodes, with OS help
- Memory system includes support for fetch-and-op primitives, to speed up some synchronization operations
  - avoid cache coherence activity

SGI UV

- UV is current generation SGI, after Origins
- Scales to 2K cores (2 sockets/node, up to 8 cores/socket), 64TB memory in 1 global shared memory (GSM), in 1 Linux instance
  - limited by physical (46 bit) and virtual (48 bit) address spaces
  - sockets connected via fat tree
- Scales to much larger configurations, connected via NUMAlink through UV ASIC chips
  - globally addressable memory (GAM) across the Linux instances - think PGAS, or put/get, also good for MPI
  - 53 bit physical memory, 60 bit virtual
- System provides fast MPI implementation, fast collective operations, high performance I/O, reliability via error checking and retry, and offloading remote memory accesses to UV ASIC

UV ASIC chip (cont.)

- Provides directory-based cache coherency for GSM and put/get for GAM
- Connects directly to QPI memory interface on Intel Xeon CPUs, not though PCI I/O bus
- Global Register Unit (GRU) for global addressing, TLB for address translation across nodes (directories), fast memory initialization w/o CPU aid, fast block copies (good for message passing too), scatter/gather memory ops
- Active Memory Unit (AMU) – cache coherent atomic memory operations, update multicasting for fast collective operations, message queues in cache coherent memory