Hybrid OpenMP/MPI Programming and other Models on Multi-Core Architectures

Gabriele Jost, Texas Advanced Computing Center, The University of Texas at Austin
Alice Koniges, NERSC, Lawrence Berkeley National Laboratory
Gerhard Wellein and *Georg Hager, Erlangen Regional Computing Center, University of Erlangen-Nuremberg, Germany
*Rolf Rabenseifner, High Performance Computing Center Stuttgart, Germany
*Ewing (Rusty) Lusk, Argonne National Laboratory
*author only—not speaking

21st International Conference on Parallel Computational Fluid Dynamics

Parallel CFD 2009
May 18-22, Moffett Field, California, USA

Tutorial Outline: Approx Timings

• Architecture Overview (40 min)
• Introduction to Hybrid Programming (30 min)
• Break (15 min)
• Hybrid Programming “How-to” (1 hr. 10 min)
• Break (15 min)
• Case studies (1 hr)
• Other programming models (10 min)
Thanks! - Contributions Include:

- Shekhar Borkar, Intel
- Kirk Jordan, IBM
- John Levesque, CRAY
- Charles Grassl, IBM/Instrumental
- Satoru Shingu, Earth Simulator
- Jim Tomkins, Sandia
- Doug Joseph, IBM
- Hitoshi Sakagami, Himeji Inst.
- Dan Poznanovic, SRC
- John Shalf, Sam Williams, Kathy Yelick LBL
- Ken Koch, Andy White LANL
- Ann Borrett, IBM
- Paul Cook, SGI
- Charles Grassl

AND many others as noted on individual slides!

Parallel Computing Architectures - Defining Terms

- **MPP:** Massively Parallel Processor with distributed memory
  - Historically each RISC-based processor had own separate memory: CRAY T3D/E, original IBM SP series
  - Ambiguous - could use shared address space model “virtual shared memory”
  - Original building blocks of parallel computing based on COTS (commodity of the shelf)
  - For a while MPPs were distinguished from clusters by their tightly-coupled single system image
  - Now it is often one or more of:
    - Proprietary interconnect
    - OS with a light weight kernel (LWK)
  - BlueGene/L and P, XT4’s, Power Architectures all classed as MPP on Top500
- The distinction between “MPP” and “cluster” is diminishing
Parallel Computing Architectures
Defining Terms (cont)

- SMP: Symmetric MultiProcessor, occasionally “Shared-Memory-Parallel”
  - Processors share a single global area of RAM
  - symmetry → all processors have equal access to memory and other parts of the system (e.g., I/O) in strictest definition
- PVP: Parallel Vector Processor
  - Cray J90/C90/T90 Series, mostly 80’s - 90’s
  - NEC SX series, Fujitsu VX series uses them as building blocks
  - Vector instructions included on many current CPU’s (see SSE, AltVec), also Cell
- ccNUMA: “cache-coherent Non-Uniform Memory Access”—allowed SMP nodes to share memory
  - Shared physical address space with automatic hardware replication
  - Sometimes called DSM or Distributed Shared Memory
  - SGI Altix with ccNUMA shared memory interconnect
- Clusters
  - Started with Beowulfs--build your own
  - Graduated to most common architecture on Top500
  - Generally have commodity processor with commodity interconnect
- Now we will see examples of SMP and ccNUMA at the chip level

Simplest Comparison of SMP, MPP or shared memory vs. distributed memory

**SMP Symmetric Multi-Processor**

- Fast Interconnect
- Memory (shared)
- SMP means equal access including I/O
- Sometimes term is generalized to mean Shared Memory Parallel

**MPP Massively Parallel Processor**

- Interconnect (varies)
- Memory physically distributed
Cache Coherence and NUMA introduced to enhance scalability

- SMP’s suffered from bottlenecks based on network contention and differences between processors speed and memory access rate.
- Cache-Coherency led to the introduction of ccNUMA in the early 1990’s

Shared Memory can be NUMA or UMA

Scales more like MPPs than bus-based SMP’s
Hardware tells you that the separate memories are one main memory
One address space over entire machine
Amount of time to access a memory value depends on whether it is local to a node or remote in another node
SGI Origin Series

UMA or SMP
- SMP’s still exist stand alone, e.g., Sun Servers
- Network types, e.g.
  - Crossbar→ independent access from each CPU
  - BUS→ one CPU can block the memory access of the other CPUs
Some of the “top 10” systems
www.top500.org: Nov 2008

#1 Petaflop Record
LANL Roadrunner
Rmax: 1.105 Pflops
129600 Cores

CRAY XT4, 5
XT5 Jaguar #2 ORNL/NCCS
150152 Cores
Rmax: 1.059 Pflops
XT4 Franklin # 7 LBNL/NERSC

NASA/Ames
Pleiades #3
SGI Altix ICE 8200
51200 Cores
Rmax: 487 Tflops

Texas “Ranger” #6
U. Texas
26544 Proc
Rmax: 326 Tflops

IBM Blue Gene(L,P) Systems #4, 5
LLNL (L): Rmax: 478 Tflops
Argonne National Lab (P): 450 Tflops

- 499 (498) scalar, 1 (2) vector
- MPPs 88 (98)
- Clusters 410 (400) in majority
- Constellations 2 (2)
  invoked to distinguish
  “clusters of large SMP nodes”
  #Procs/#Nodes > #Nodes

() denotes previous year June list

Original MPPs were single cpus on a chip with local memory

Simplistic View: MPP Node

Processors
Commodity, Proprietary

Memory
DRAM
+ variants for faster access

Communication Mechanism
Shared Memory or Message Passing

Networks
Switch, 3D Torus, Bus, Crossbar
-more than one (BlueGene/L)
-usually proprietary for MPP
The first introduction of multi-mode programming came with the Shared Memory Node Architecture

- Distributed memory across nodes
- Shared memory within a single node
- Example: IBM Power3 with 4, 8 or 16, or ... circa 2000
- Optimal programming model? Didn’t get much speed-up over using MPI everywhere...
- Now, multicore / many-core is here and increasing
- We must consider new models

Single Core Chip Architecture Terminology

- How many levels of cache? Sizes?
- Stream Buffers improve data flow into cache
- Ways to bypass cache?
- Vector units?
- Pseudo vector units?

Generic single core
Multi-core or Multi-processor Cache-based Chip

Typical Layout:

- Each processor:
  - L1 caches
  - Registers
  - Functional units

- Each chip (shared)
  - L2 cache
  - L3 cache
  - Path to memory

On a multi-core chip, get more computational power with (often) same bandwidth to memory, so need to be effective with cache reuse

Note: Different access for L3 cache

Quad Cores and Beyond

Example: 4 core AMD Barcelona

Example: Intel Dunnington
6 cores
16 MB shared
L3 per socket

New many-core chips 64 and up are likely to be appearing in HPC

Quad Core systems now dominate the top500
The x86 multicore evolution

Current Multicore SMP Systems can have different memory access and cache use patterns

Intel Clovertown

AMD Opteron

Uniform Memory Access

Non-uniform Memory Access

Adapted from Sam Williams, John Shalf, LBL/NERSC et al.
Sample Architectures in More Detail

- Power Series, IBM ASC Purple, Upcoming Blue Waters
- Blue Genes
- Cray XT4/5
- SX9
- Cells and Roadrunner
- Typical Top Clusters: Ranger, SGI ICE-Pleiades
- Large Shared Memory Machines

Power Series
Example: RISC Power

- **IBM's Power2**
  - Functional Parallelism: up to 6 instructions simultaneously if they don't conflict
  - Pipelining: long sequences of the same operation e.g., multiply and add
  - Cache optimization
- **IBM's Power3**
  - RISC Superscalar (all units can run in parallel) with out-of-order execution, branch prediction, data pre-fetch, SMP enabled
- **IBM's Power4**
  - Multiprocessor chip with high clock rate
    - Long pipelines, Three cache levels, can use latency hiding
  - Shared Memory
    - Large memory size
- **IBM's Power5**
  - L3 cache moved to processor side of interconnect fabric
- **IBM's Power6 ~2X the frequency of Power5**
  - Change from out-of-order execution to in-order

Simultaneous Multi-Threading can affect hybrid implementations

- Presents SMP programming model to software
- Natural fit with superscalar out-of-order execution core

Charles Grassl, IBM
Conventional Multi-Threading

- Threads alternate
  - Nothing shared

Simultaneous Multi-Threading

- Simultaneous execution
  - Shared registers
  - Shared functional units
A typical IBM Power Series
LLNL’s “Purple”

- Purple System Specs
  - 93.4 TF/s peak from 12,288 Power5 @ 1.9 GHz
  - 50 TB memory
- 1,536 8-way SMP nodes
  - 32 GB memory
  - 8x1.9 GHz Power5 single core ASIC
- Blue Waters POWER7
- Timescale ~2011
- > 200,000 cores
- Private L1 and L2 caches for each core, shared L3
- NSF machine sited at U. Illinois

Possible Chip Configuration:
- 8 cores per chip arranged in dual-chip modules
- 4 hardware threads per core

BlueGene Series
Blue Gene: record performance, low power, high processor count

- **Blue Gene L** gave surprising performance
  - Record Linpack performance of 280.6 TFlop/s
- **Low power, low cost, high performance: 180-360 Tflops**
  - New generation of embedded processors
  - Relatively modest clock rate but low power consumption and low cost
  - Less cooling means pack in tighter
  - 64 cabinets (32x32x64) ~1 MW (1000 Hairdryers) <2500 sq.ft
  - Low latency memory: 3 to L1, 36 to L3, 86 to memory
  - High bandwidth memory 22 GB/s to L3, 5.5 GB/s to memory
- **216 Nodes, 2 P/node, MANY Processors! (> 100,000)**
- Light weight kernel on compute nodes, similar to XT
- Different networks--3D torus, broadcast tree (and others)
- Blue Gene P’s are up and running
- Blue Gene Q’s are finishing design

---

**Blue Gene / L**

*Multiple complementary interconnects support diverse application scaling requirements*
- 3D torus with bi-directional nearest-neighbor links
- 2.1 GB/s combining tree for fast global reductions
- Low-latency global barrier network
- High reliability expected from high level of integration using system-on-a-chip technology

- **Node Card**
  - 32 compute chips
  - 16 compute cards

- **Compute Chip**
  - 2 processors
  - 2.8/5.6 GF/s

- **Midplane**
  - SU (scalable unit) 16 node boards (8x8x8)
  - 1.4/2.9 TF/s
  - 7-10 kW

- **Cabinet**
  - 32 node boards (8x8x16)

- **System**
  - 64 cabinets
  - (32x32x64)
  - 180/360 TF/s ~1 MW
  - *(1000 Hairdryers)*
  - <2500 sq.ft.

---

Was Top500 #1
Before Roadrunner
OpenMP MPI combination was not available on BG/L, but now is with SMP mode on BG/P

**BG/L Mode 1 (Co-processor mode - CPM):**
- CPU0 does all the computations
- CPU1 does the communications
- Communication overlap with computation
- Peak comp perf is 5.6/2 = 2.8 GFlops

**BG/L Mode 2 (Virtual node mode - VNM):**
- CPU0, CPU1 independent "virtual tasks"
- Each does own computation and communication
- The two CPU's talk via memory buffers
- Computation and communication cannot overlap
- Peak compute performance is 5.6 Gflops

**BG/P Virtual Node Mode, SMP Mode, Dual Mode**

BG/P Figure courtesy K. Jordon, IBM
Franklin: NERSC’s Cray XT4

• System structure
  – 9,660 nodes
  – originally 19,320 cores, recently upgraded to 38,640
  – Interconnect: Cray SeaStar2, 3D Torus
    >6 TB/s Bisection Bandwidth; >7 GB/s Link Bandwidth
  – Shared Disk: 400+ TBs

• Performance:
  – Sustained application performance: 38 Tflops
  – Peak performance: 355 Tflops
  – Linpack: 266 Tflops
### Mix-and-match to meet workload requirements

<table>
<thead>
<tr>
<th>Cray XT4</th>
<th>Cray XT5</th>
<th>Cray X2 Blade</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Cray XT4" /></td>
<td><img src="image2" alt="Cray XT5" /></td>
<td><img src="image3" alt="Cray X2 Blade" /></td>
</tr>
<tr>
<td>Optimized for compute/interconnect balance</td>
<td>Optimized for memory-intensive and/or compute-biased workloads</td>
<td>high bandwidth memory intensive &gt;25GFLOPs/single core vector based CPU and global address space scaling to 32K processors</td>
</tr>
</tbody>
</table>

XT5\textsubscript{h} diagrams courtesy John Levesque, CRAY

---

### Cray Introduces XT5 and XT5\textsubscript{h}

**XT5\textsubscript{h} (hybrid) Integrated Scalar and Vector Processing**

- Common Infrastructure
  - Seastar2+ Interconnect
  - Linux Environment
  - Common I/O and File System

- Partitioned Global Address Space (PGAS) Language Support
  - Co-Array Fortran
  - UPC

XT5\textsubscript{h} supports two processor technologies: Opteron and Vector

XT5\textsubscript{h} diagrams courtesy John Levesque, CRAY
Simplified memory hierarchy on the AMD Opteron

- 36 entry FPU instruction scheduler
- 64-bit/80-bit FP Realized throughput (1 Mul + 1 Add)/cycle: 1.9 FLOPs/cycle
- 32-bit FP Realized throughput (2 Mul + 2 Add)/cycle: 3.4+ FLOPs/cycle

16 instruction bytes fetched per cycle

- Fetch
- Branch Prediction
- Scan/Align
- Fastpath
- Microcode Engine

Instruction Control Unit (72 entries)

- Int Decode & Rename
- FP Decode & Rename

9-way Out-Of-Order execution

- 16 SSE2 128-bit registers
- 16 64-bit registers

- 64 Byte cache line
- Complete data cache lines are loaded from main memory, if not in L2 cache
- If L1 data cache needs to be refilled, then storing back to L2 cache

- 64 Byte cache line
- Write back cache: data offloaded from L1 data cache are stored here first until they are flushed out to main memory

16 Bytes wide data bus => 6.4 GB/s for DDR400

Main memory

Courtesy John Levesque
SSE vectorization is available on AMD

- Function in-lining
- Enable SSE vectorization (when available) - streaming SIMD extensions
  - Fine-grained data parallelism
  - Check compiler output for vectorization of loops
    - C and C++ codes can inhibit vectorization

SIMD is single instruction multiple data

x86 Architecture: SIMD Operations

- Possible data types in an SSE register
  - 16x 8bit
  - 8x 16bit
  - 4x 32bit
  - 2x 64bit
  - 1x 128bit
  - 4x 32 bit
  - 2x 64 bit
x86 Architecture:  
*Floating Point Operations and SIMD*

- Example: Single precision FP packed vector addition

\[
\begin{array}{cccc}
    x_3 & x_2 & x_1 & x_0 \\
    y_3 & y_2 & y_1 & y_0 \\
\end{array}
\]

- Four single precision FP additions are done in one single instruction

- Intel Core2: 3/5-cycle latency & 1/1-cycle throughput for double precision SSE2 ADD/MULT leading to a peak performance of 4 (DP) FLOPs/cycle
  - Single precision: 8 SP FLOPs/cycle
- AMD64/K10 (2008): same characteristics

NEC Vector Line
ParCFD09 Tutorial © Jost, Wellein, Hager, Koniges, Rabenseifner, Lusk, and others

39

Started with Earth Simulator Vector Processor based on SX-6

Based on NEC SX-6,
Held top spot in top500 for 7 Lists
Slight RAM differences, bi-dir btw differences

http://www.es.jamstec.go.jp/

Comparison of SX-8R and SX-9
Note: 102.4 GF on a single core!

NEC will withdraw from Japan's government-backed supercomputer project as part of its efforts to cut costs during the economic slump, May14.
Linux Clusters become Mainstream

Sun Constellation Linux Cluster “Ranger”
Texas Advanced Computing Center

• First of the new NSF Track2 HPC
• Number 3 on the Top 500 list for June 2008
• 3936 Nodes, 62,976 Cores
• Peak Performance 579.4 Tflops
• 15,744 Quad-Core AMD Opteron at 2.3 GHz

Cluster Systems are the top500 majority
SGI Altix Systems

SGI Altix offers shared memory systems and ICE Cluster systems (NASA Pleiades)
### SGI’s Altix Servers

- **Altix 4700 → Ultraviolet**
  - Scalable Shared Memory
- **Altix ICE**
  - Scalable Blade Cluster

SGI Slides Courtesy Paul Cook, SGI
Pleiades Slides Courtesy Rupak Biswas

---

### Pleiades System NASA/Intel/SGI

**Manufacturer - SGI**

**System Architecture**

- 100 Compute Cabinets (64 nodes each; 6,400 nodes total)
- 609 TFin/s cluster
- Total cores: 51,200
- Nodes:
  - 5888 nodes
    - 2 quad-core processors per node
    - Xeon E5472 (Harpertown) processors
    - Processor speed - 3GHz
    - Cache - 6MB per pair of cores
  - 512 nodes (RT Jones domain)
    - 2 quad core processors per node
    - Xeon X5355 (Clovertown) processors
    - Processor speed - 2.66GHz
    - Cache - 4MB per pair of cores

**Diagram:**

- **Greencreek Chipset:**
  - FSB 1333MT/s
  - 10.6 GB/s
  - 21.3 GB/s read

- **Seaburg Chipset:**
  - FSB 1600MT/s
  - 12.8 GB/s
  - 25 GB/s read

- **Clovertown:**
  - Xeon X5355 processors
  - Processor speed - 2.66GHz
  - Cache - 4MB per pair of cores

- **Harpertown:**
  - Xeon E5472 processors
  - Processor speed - 3GHz
  - Cache - 6MB per pair of cores

---

ParCFD09 Tutorial © Jost, Wellein, Hager, Koniges, Rabenseifner, Lusk, and others
SGI® Altix® ICE®
The Blade Cluster Designed for Scalable HPC

Altix® ICE Blade
8-Core, 32GB, 2-IB

Altix ICE Rack:
• 42U rack (30" W x 40" D)
• (4) Cable-free blade enclosures with (16) 2-Socket nodes
• (128) DP Intel® Xeon® sockets
• (4) Single or Dual IB plane(s)
• Minimal switch topology simplifies scaling to 1000s of nodes

SGI® Altix® 4700 Shared Memory Platform

• NUMAflex™ shared-memory and cross-partition scalability with industry-standard HPC processor, COTS Linux Distros
• Moving to Intel Nehalem EX with next release.
Systems with GPUs, FPGAs etc.

**Cell Processor**

--was enhanced for HPC

Think of the typical cell processor designed for the PlayStation PS3:

- CPU Calculation power is ~220 - 230 Gflops (Cell single precision)*
- GPU Calculation power is ~1.8 TFlops (Nvidia graphics chip)

Total System Calculation power is **2 TFlops**

```
Mem. Contr.
Config. IO
```

```
64b Power Processor
Synergistic Processor

Synergistic Processor
```

Cell Processor courtesy
Doug Joseph, IBM

*Cell has nominally 8 SPE's, this is 4GHz estimate, PS3 designed to use 7 of these. Each SPE is capable of sustaining 4 FMADD per cycle
IBM’s Cell Broadband Engine™ Blade
Cell systems and Roadrunner

- Cell double precision floating point enhancement
  - Enables each SPE to perform two DP FPMULADD ops per cycle
  - Gives a peak of 102.4 DP FP Gflop/s@3.2 Ghz (half peak SP FP rate)
- Cell Blade memory capacity increase to 16GB
- 25.6 GB/S memory bandwidth

Applications can perform very near peak

EIB is 4 ring buses Element Interconnect Bus
256KB in LS
SMF = DMA, MMU + bus connect
MIC = Synergistic Memory Flow Controller
PPE has 32KB L1 512KB L2
SPE: Synergistic Processing Element
SUX: Synergistic eXecution Unit
PPE: Power Processing Element
MIC/BIC: Memory/Bus Interface Controller

Cell Processor
courtesy Doug Joseph, IBM

Note: dual XDR replaced by dual DDR
in HPC enhanced Cell (the PowerXCell 8i)

Roadrunner is First Petaflop Machine!
A New top500 #1 Record

Connected Unit cluster
180 compute nodes w/ Cells
12 x3655 I/O nodes

6,120 dual-core Opterons ⇒ 44 TF
12,240 Cell eDP chips ⇒ 1.3 PF

I/O nodes not counted

12 links per CU to each of 8 switches

Eight 2nd-stage 288-port IB 4X DDR switches

Roadrunner slides courtesy Ken Koch, LANL
and IBM, and RR Team
Roadrunner is Cell-accelerated, not a cell cluster

Roadrunner uses Cells to make nodes ~30x faster
400+ GFlop/s performance per hybrid node

One Cell chip per Opteron core
Adapted from LA-UR-07-6213
From Andrew White and Ken Koch, LANL

The next generation: low power, high concurrency, many-core
New Designs for Power Efficiency, High Parallelism/Concurrency

- Power5 (Server)
  - 389 mm²
  - 120 W @ 1900 MHz
- Intel Core2 sc (Laptop)
  - 130 mm²
  - 15 W @ 1000 MHz
- PowerPC450 (BlueGene/P)
  - 8 mm²
  - 3 W @ 850 MHz
- Tensilica DP (cell phones –and Green Flash energy-efficient architectures)
  - 0.8 mm²
  - 0.09 W @ 650 MHz

Even if each core operates at 1/3 to 1/10th efficiency of largest chip, you can pack 100s more cores onto a chip and consume 1/20 the power!

Green Flash: Wehner, Oliker, and Shalf (LBL Rowen (Tensilica))

An 80-tile 1.28 TFLOPS INTEL CPU Prototype

Tiles arranged in 10 X 8 2D mesh