Executing OpenMP Programs

Mitesh Meswani
Presentation Outline

- Introduction to OpenMP
- Machine Architectures
  - Shared Memory (SMP)
  - Distributed Memory
- MPI or OpenMP?
- OpenMP hello world program
- OpenMP on
  - Top Gun (AIX/Linux) SMP
  - Itanium2 (Linux) Cluster
  - SGI Origin 2000 (IRIX) SMP
    - Environment Set-up
    - Compilation
    - Execution
- Contacts and Web Sites
Introduction to OpenMP*

- Thread-based Parallelism
- Explicit Parallelism
- Fork-Join Model
- Compiler Directive Based
- Dynamic Threads

*Source: http://www.llnl.gov/computing/tutorials/openMP/#ProgrammingModel
Machine Architectures
Shared Memory

FEATURES:
1) All CPUs share memory
2) CPUs access memory using the interconnection network

Example: Top Gun, Kirk, every node of the Itanium2 cluster is a dual-processor SMP machine
Machine Architectures
Distributed Memory

FEATURES:
1) Each node has its own local memory
2) Nodes share data by passing data over the network

Example: Sun Solaris Workstations
**MPI or OpenMP?**

- **Recommended Use**
  - MPI for Distributed Memory Machines
  - OpenMP for Shared Memory (SMP) Machines

- **Hybrid MPI / OpenMP Use**
  - Use MPI on clusters of SMP nodes to communicate among nodes, and use OpenMP to create multiple threads per SMP node
  - Example: Itanium2 Vampyre Cluster*

*Currently the Vampyre cluster does not have a supporting compiler for hybrid OpenMP and MPI programs.*
OpenMP hello world program

Sample hello world OpenMP program:

```c
#include <omp.h>

main () {
  int nthreads, tid;

  /* Fork a team of threads giving them their own copies of variables */
  #pragma omp parallel private(tid)
  {
    /* Obtain and print thread id */
    tid = omp_get_thread_num();
    printf("Hello World from thread = %d\n", tid);

    /* Only master thread does this */
    if (tid == 0)
      {
        nthreads = omp_get_num_threads();
        printf("Number of threads = %d\n", nthreads);
      }
  } /* All threads join master thread and terminate */
} /* End of main */
```
OpenMP on Top Gun

Top Gun is an IBM eServer pSeries 690 (p690) multiprocessor

<table>
<thead>
<tr>
<th>System Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
</tr>
<tr>
<td>OS</td>
</tr>
<tr>
<td>CPUs</td>
</tr>
<tr>
<td>Peak GFLOPS</td>
</tr>
<tr>
<td>Memory</td>
</tr>
<tr>
<td>Disk Storage</td>
</tr>
</tbody>
</table>
OpenMP on Top Gun
Environment Set-up - 1

- Define the default number of threads in the environment variable OMP_NUM_THREADS

**Example:** for tcsh shell

```
%setenv OMP_NUM_THREADS 4
```
The program can overwrite the default number of threads using the `omp_set_num_threads()` function.

The maximum number of threads that can be created by this function is bounded by the environment variable OMP_NUM_THREADS.

**Example:** Create two threads

```c
omp_set_num_threads(2);
```
OpenMP on Top Gun
Program Compilation

- C Compiler: \texttt{xlc\_r} with compile flag \texttt{-qsmp=omp}

  Usage:
  \begin{verbatim}
  xlc\_r -qsmp=omp [-o outfile] infile
  \end{verbatim}

  Example:
  \begin{verbatim}
  %xlc\_r -qsmp=omp -o hello helloworldOMP.c
  \end{verbatim}

- Other OpenMP compilers:
  - Fortran77: \texttt{xlf\_r}
  - C++: \texttt{xlc\_r}

- Complete list available at:
  \url{http://research.utep.edu/Default.aspx?tabid=20687}
OpenMP on Top Gun
Program Execution

- Execute the program like a normal program executable

Example:

% ./hello
Hello World from thread = 3
Hello World from thread = 0
Number of threads = 4
Hello World from thread = 1
Hello World from thread = 2
OpenMP on Itanium2 Cluster

- Vampyre Cluster: 8-processor Intel Itanium2 Cluster, with dual (SMP) 900MHz Itanium2 processors per node
- Network Features: Externally accessible by 100 Mbps Ethernet; internal network runs at 1Gbps
- OS: Linux kernel 2.4.18-e.25
- OpenMP Compilers: Omni and Intel* Compilers

*Intel Compiler is available only for teaching purposes.
OpenMP on Itanium2 Cluster
Environment Set-up

- Confirm that Omni and the Intel Compilers are visible in your path using the `which` command
  
  **Example:** for the Omni C compiler and Intel C Compiler
  
  ```
  % which omcc
  /usr/local/Omni/bin/omcc
  
  % which icc
  /usr/local/intel/cc/bin/icc
  ```

- Environment Variables: define OMP_NUM_THREADS
  
  **Example:** for tcsh shell
  
  ```
  % setenv OMP_NUM_THREADS 4
  ```

- Program can overwrite the default number of threads using the `omp_set_num_threads()` function
OpenMP on Itanium2 Cluster Program Compilation

- Omni C Compiler: `omcc`
  Usage:
  ```bash
  omcc [-o outfile] infile
  ```
  Example:
  ```bash
  % omcc -o hello1 helloworldOMP.c
  ```

- Intel C Compiler: `icc` with compile flag `-openmp`
  Usage:
  ```bash
  icc -openmp [-o outfile] infile
  ```
  Example:
  ```bash
  % icc -openmp -o hello2 helloworldOMP.c
  ```

- Other OpenMP Compilers:
  - Fortran: `omf77`
  - C++: `icc`
OpenMP on Itanium2 Cluster

Program Execution

- Execute the program like a normal program executable

**Example:** executing on node Sabina

```
sabina > ./hello1
Hello World from thread = 3
Number of threads = 4
Hello World from thread = 0
Hello World from thread = 1
Hello World from thread = 2
```
OpenMP on SGI Origin 2000

- Kirk: SGI Origin 2000 SMP machine with 8 180MHz MIPS R10000 processors
- 2.9 GB main memory, 32 KB L1 data cache, 32 KB L1 instruction cache and 1 MB unified L2 cache
- Operating System: IRIX64 Release 6.5
- Machine IP: kirk.sseal.cs.utep.edu
OpenMP on SGI Origin 2000
Environment Set-up

- Confirm that Omni Compilers are in your path using the *which* command
  
  **Example:**
  
  % which omcc
  
  /usr/local/Omni/bin/omcc

- Environment Variables: define OMP_NUM_THREADS
  
  **Example:** for tcsh shell
  
  % setenv OMP_NUM_THREADS 4

- Program can overwrite the default number of threads using the *omp_set_num_threads()* function
OpenMP on SGI Origin 2000
Program Compilation

- Omni C Compiler: omcc

Usage:

    omcc [-o outfile] infile

Example:

    % omcc -o hello helloworldOMP.c
OpenMP on SGI Origin 2000

Program Execution

- Execute the program like a normal program executable

Example:

% ./hello
Hello World from thread = 1
Hello World from thread = 3
Hello World from thread = 0
Hello World from thread = 2
Number of threads = 4
Contacts and Websites

- System Administrators:
  - Jose Hernandez (jose@cs.utep.edu) for Top Gun
  - Leopoldo Hernandez (leo@cs.utep.edu) for Itanium2 Cluster (Vampyre) and SGI Origin 2000 (Kirk)

- System Web Sites:
  - Top Gun: http://research.utep.edu/topgun
  - Itanium2 Cluster: http://www.cs.utep.edu/~bdauriol/courses/ParallelAndConcurrentProgramming/vampyre.html

- OpenMP Links:
  - http://www.openmp.org
  - Omni OpenMP: http://phase.hpcc.jp/Omni/home.html
  - Tutorials:
    - http://www.llnl.gov/computing/tutorials/openMP/
    - http://www.msi.umn.edu/tutorial/scicomp/general/openMP/
Questions?