Chapter 15 Getting started with OpenMP

OpenMP is concerned with a single cluster node or motherboard

A node has up to four sockets; each socket has up to 60 cores; each core is an independent unit, with access to all the memory on the node.

OpenMP is based on two concepts: the use of threads and the fork/join model of parallelism.

The threads that are forked are all copies of the master thread, they have access to all that was computed so far, which is their shared data. They also have private data and they can identify themselves, they know their thread number.

Get a good speedup you would typically let your number of threads be equal to the number of cores.

Compiling and running an OpenMP program

#include “omp.h” in C

use omp_lib or #include “omp_lib.h” in Fortran

what you type on the command line

gcc

$ gcc -o fname fname.c -fopenmp

Intel compiler

$ icc -o fname fname.c -openmp

Export OMP_NUM_THREADS = 8, OMP_NUM_THREADS is an important environment variable.

Directives

In C/C++ the prama mechanism is used.

#pragma omp somedirective clause (value, othervalue)

Parallel statement

#pragma omp somedirective clause (value, othervalue)

{Parallel statement 1;

Parallel statement 2;}

In Fortran looks like a comment

!$ omp directive clause (value)

Statement

!$ omp end directive

A block preceded by the omp parallel pragma is called a parallel region.

Important definitions: structured block, construct, region of code.

Chapter 16 Parallel Regions

A block preceded by the omp parallel pragma is called a parallel region which is executed by a newly created team of threads.

Nested parallelism, by default, the nested parallel region will have only one thread. To allow nested thread creation, set

OMP_NESTED = true

Chapter 17 Loop parallelism

```
#pragma omp parallel
{
   Code1 ();
#pragma omp for
   for (i=1; i<=4*N; i++) {
   code2 ();
   }
   code3();
}
```

Loop schedules

#pragma omp for schedule (….)

Static schedules, the iterations are assigned purely based on the number of iterations and the number of threads (and the chunk parameter)

Dynamic schedules, iterations are assigned to threads that are unoccupied. They are good for iterations take an unpredictable amount of time, so the load balancing is needed.

The default static schedule is to assign one consecutive block of iterations to each thread.

#pragma omp for schedule (static[,chunk])

Various schedules can be set with the schedule clause:

affinity, auto, dynamic, guided, runtime, static.

Reductions are a common type of loop with dependencies.

Collapsing nested loops

#pragma omp for collapse (2)

```
for ()
   for ()
    code ();
```

It is only possible to collapse perfectly nested loops.

Ordered iterations

Nowait

OpenMP can only handle ‘for’ loops: while loops can not be parallelized.

Questions

1. Why is a chunk size of 1 typically a bad idea?