Hybrid Computing LAB

TACC OpenMP Tutorial

PRESENTED BY:
Lars Koesterke
Kent Milfeld
Eval hybrid code, compile, launch, MPI inside OpenMP

- Compile code with MPI wrapper and OpenMP options.
- Slurm request for 2 nodes with 1 rank per node.
- Set run environment (68 threads/task, use default affinity)
  - export OMP_NUM_THREADS=68
- Evaluate MPI mask with amask.
- Launch with TACC’s ibrun launcher (=mpirun)
  - ibrun a.out  #ask usual
- Insert MPI code within an OpenMP parallel region
  - Change MPI_Init to MPI_Init_thread
  - Use single directive to for a single MPI_Bcast call.
Create your first hybrid code, combining OpenMP/MPI

• Sum up a sequence of numbers in a parallel OpenMP region
• Sum up the partial values from the OpenMP calculations.

```c
MPI_init(NULL,NULL);
MPI_Comm_rank(MPI_COMM_WORLD,&rank);
nthreads=omp_get_max_threads();

#pragma omp parallel for reduction(+:sum) num_threads(nthreads)
for(i=1;i<=nthreads;i++) sum=sum+omp_get_thread_num() + (rank*nthreads)

MPI_Allreduce(&sum,&tot_sum,1,MPI_INT, MPI_SUM,MPI_COMM_WORLD);
```

• Compile with MPI “compiler” and –qopenmp option.
HYBRID_PI

Perform MPI_Allreduction in OpenMP parallel region

• A parallel integration of $4/(1+x^2)$ calculates PI.
• OpenMP parallel regions perform integration over intervals, and MPI sums the result (partials) from each rank in a MPI_Allreduce.
• The MPI partial summation is moved inside an OpenMP parallel region. (Sounds familiar?)
• The appropriate MPI_Init is performed.
• Results are checked.
HYBRID_NUMA

Launching HYBRID CODE with AFFINITY

• Hop on a SNC-4 numa node (idev, and select SNC4 reservation)
• Set run environment:
  • # of ranks (launcher) -np #
  • # of cores for each rank’s mask I_MPI_PIN_DOMAIN=##
  • # of threads for each rank OMP_NUM_THREADS=####
• export …
  ibrun -np # a.out
• env … mpiexec.hydra -np # a.out
• env … mpiexec.hydra -np # myscript

• For SNC-4 numa nodes:
  When DOMAIN=numa, 4 masks are set …