ISIB 2014

High Performance Computing

June 25, 2014

Kate Cowles
374 SH, 335-0727
kate-cowles@uiowa.edu

What to do when a computing problem is too big, or computing would take toolong, for a single computer to hande

- parallel computing
- using graphical processing units for mathematical calculations
- distributed and grid computing

Parallel computing in R: the snow package

- snow is a parallel computing package for R
- snow uses a master/worker model:
  - The user starts an ordinary R session as the master process.
  - This session creates (or connects to) a set of worker processes.
  - Jobs are sent to the worker processes and results are returned.
- The underlying message passing can be based on
  - raw sockets (no additional packages/software needed)
  - PVM (uses rpvm and PVM)
  - MPI (uses Rmpi and LAM-MPI; other MPIs may also work)
- Which communication mechanism is used only matters at startup.

Text on this slide is from a lecture by Luke Tierney in 22S:195 Fall 2007.

Aside: Statistics Dept Beowulf cluster i

- `http://www.divms.uiowa.edu/help/statcluster/`
- Request account through Kate Cowles or Luke Tierney

Simple example of parallel computing in R

```
# R session on node 11 of the statistics cluster

> library(snow)

# establish communication with 10 other cores on 3 nodes

> cl <- makeSOCKcluster( c("node11","node11","node11","node12","node12",
"node12","node12","node13","node13","node13"))

# execute the Sys.info() function on all 10 worker cores, and retrieve
#      just the nodename from it

>  do.call("rbind", clusterCall(cl, function(cl)  Sys.info()["nodename"]))
      nodename
 [1,] "node11.beowulf.stat.uiowa.edu"
 [2,] "node11.beowulf.stat.uiowa.edu"
 [3,] "node11.beowulf.stat.uiowa.edu"
 [4,] "node12.beowulf.stat.uiowa.edu"
 [5,] "node12.beowulf.stat.uiowa.edu"
 [6,] "node12.beowulf.stat.uiowa.edu"
 [7,] "node12.beowulf.stat.uiowa.edu"
 [8,] "node13.beowulf.stat.uiowa.edu"
 [9,] "node13.beowulf.stat.uiowa.edu"
[10,] "node13.beowulf.stat.uiowa.edu"

> x <- 1:100/101

# time a computation on this vector done sequentially on the master node

>  system.time(qtukey(x, 2, df=2))
   user  system elapsed
  4.609   0.000   4.629

# time the same computation done in parallel on the 10 nodes in the cluster;
# the parLapply function sends equal-length segments of the vector to each
#    core; each core does the computation on its segment; then the results
#    are sent back to the master node and assembled there

> system.time(unlist(parLapply(cl, x, qtukey, 2, df=2)))
   user  system elapsed
  0.004   0.000   0.543

> speedup <-  4.629/0.543
> speedup
[1] 8.524862

# shut down the communication among the cluster cores before leaving R
> stopCluster(cl)
```

General programming with graphical processing units

- high end graphics cards such as those used in gaming computers have many cores for rendering graphics (up to several hundred)
- recently languages and programming toolkits have been developed to enable the use of the cores in graphics cards for mathematical computation
- challenges are the same as parallel processing on a cluster
  - has to be possible to divide the whole computing job (or parts of it) into individual small tasks that can be executed independently in parallel
  - slow transfer of data and results between CPU (regular processor) and GPU (graphics processor)

Distributed or grid computing

- distributed system: system of autonomous computers that communicate through a computer network to work together to solve a common computational problem
- distributed computing: use of distributed systems to solve computational problems
  - whole problem split into individual tasks, each carried out by single computer
- organizational grids
- volunteer computing

The TeraGrid: example of a scientific grid

See some slides in separate lecture.

Volunteer computing

- distributed computing in which computer owners donate some of their computer resources to one or more projects
- what kinds of computing problems are suitable
- BOINC software
- World Community Grid
  - an example of a well-run volunteer grid computing project
  - purpose and organization

Blue Waters

`http://www.ncsa.illinois.edu/enabling/bluewaters`