Running ISA in parallel with the `snow` package

Gábor Csárdi

May 13, 2009

1 Running ISA in parallel

In this document we show a little example on how to speed up ISA analysis by running the ISA iterations in parallel, on a computer cluster, or a multiprocessor machine.

Since a typical ISA analysis consists of using a range of row/column and these runs are independent of each other; it is trivial to parallelize the task by performing the iterations for different threshold parameters on different processors or computers. Here we show and example on how to do this easily with the `snow` and the `Rmpi` packages:

```
> library(isa)
> library(snow)
> library(Rmpi)
```

We generate some simple in-silico data.

```
> pdata <- isa.in.silico()
```

Create the MPI cluster, with eight working nodes. You need to have a working MPI installation for this. See more in the documentation of the `Rmpi` and `snow` packages.

```
> clu <- makeMPIcluster(8)
```

```
        8 slaves are spawned successfully. 0 failed.
```

```
> invisible(clusterEvalQ(clu, library(isa)))
> clusterExport(clu, "pdata")
```

Create a big matrix in which each row is a combination of the threshold parameters.

```
> thr <- seq(1, 3, by = 0.2)
> thr.list <- expand.grid(thr, thr)
```

First we run the ISA on a single processor only, and measure the running time.

```
> system.time(modules <- isa(pdata[[1]], thr.row = thr, thr.col = thr))

   user  system elapsed
 35.998   1.273  37.277
```

Let us now do a parallel run, again, with measuring the running time. If you are really running this on multiple CPUs, then it is much faster.

```
> system.time(modules.par <- parApply(clu, thr.list, 1, function(x) {
+     isa(pdata[[1]], thr.row = x[1], thr.col = x[2])
+ }))

   user  system elapsed
  0.028   0.000  21.893
```

Finally, stop the cluster.

```
> stopCluster(clu)

[1] 1
```

2 Session information

The version number of R and packages loaded for generating this vignette were:

- R version 2.8.1 (2008-12-22), `x86\_64-redhat-linux-gnu`
- `LC_CTYPE=en_US.UTF-8; LC_NUMERIC=C; LC_TIME=en_US.UTF-8; LC_COLLATE=en_US.UTF-8; LC_MONETARY=C; LC_MESSAGES=en_US.UTF-8; LC_PAPER=en_US.UTF-8; LC_NAME=C; LC_ADDRESS=C; LC_TELEPHONE=C; LC_MEASUREMENT=en_US.UTF-8; LC_IDENTIFICATION=C`
- Base packages: base, datasets, graphics, grDevices, methods, stats, tools, utils
- Other packages: isa 0.1, Rmpi 0.5-5, snow 0.3-3

References

[Bergmann et al., 2003] Bergmann, S., Ihmels, J., and Barkai, N. (2003). Iterative signature algorithm for the analysis of large-scale gene expression data. *Phys Rev E Nonlin Soft Matter Phys*, page 031902.

[Csárdi, 2009] Csárdi, G. (Apr 1, 2009). *isa: The Iterative Signature Algorithm*. R package version 0.1.

[Ihmels, 2002] Ihmels, J., Friedlander, G., Bergmann, S., Sarig, O., Ziv, Y., Barkai, N. (2002). Revealing modular organization in the yeast transcriptional network. *Nat Genet* , page 370–7.

[Ihmels, 2004] Ihmels, J., Bergmann, S., Barkai, N. (2004). Defining transcription modules using large-scale gene expression data. *Bioinformatics*, page 1993–2003.

[Kaiser, 2009] Sebastian Kaiser, Rodrigo Santamaria, Roberto Theron, Luis Quintales and Friedrich Leisch. (2009). *biclust: BiCluster Algorithms*. R package version 0.7.2.