

# Windows用並列計算 パッケージ Rhpc

中野 純司 モデリング研究系 教授

## はじめに

今日ではCPU単体の性能の向上はほぼ望めず、マルチコアによる並列処理を如何に効率よく行うかが大規模データ処理において重要になってきている。そのため統計解析システムRでも既にいくつかの並列計算のためのパッケージが存在するが、それらの多くはWindowsにおける良い並列環境とは言えない。例えば、よく利用されているパッケージ snow は多くがRで書かれており、コア数の増加に伴って遅延が大きくなる。また高速で知られる multicore はWindowsにおいては、シングルコアしか利用出来ない。

そこでわれわれはこれまでRのパッケージ Rhpc として lapply を並列計算環境 MPI を効率的に利用して実装してきた Rhpc\_worker\_call (計算ノードに対する処理)、Rhpc\_lapply (分散処理単位の処理を行わせる) をWindows環境においても利用できるように改良を加たのでそれを説明する。

## 開発対象としたMPIの実装と制限事項

CRANでは、ソースパッケージとして各パッケージを受け、WindowsやMacOSXのバイナリパッケージはCRANにおいてビルドを行っている。

既にCRANではWindows向けのMPI実装のバイナリが用意されている。RmpiおよびpbdMPI等ではMS-MPIが利用されているので、我々もこれに倣った。

MS-MPIでは現在までのところ、MPIの動的プロセスの生成はサポートしていない。したがって、mpiexecによる起動が必要となる。

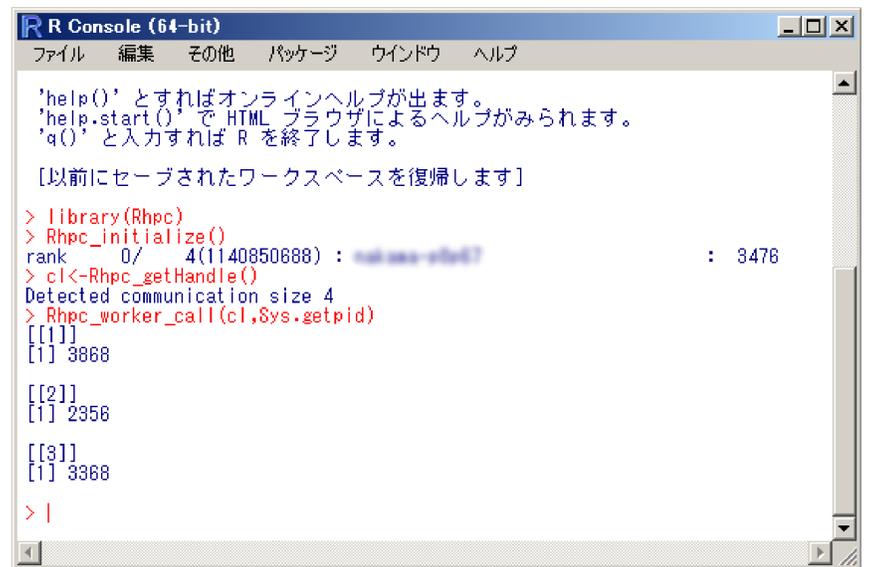
## 起動方法

RhpcのワーカースプロセスはEmbedded Rで構成されており、R.dllを利用しているため、ワーカの起動には、R.dllが検索パスに存在する必要がある。

例えば、RhpcのマスタースプロセスをRguiで起動し、Rhpcのworkerプロセスを適切に配置するには、以下のようにする必要がある。

### mpiexecによるRguiを使ったRhpcの起動例

```
mpiexec.exe \
  -env PATH "C:\Program Files\R\R-3.2.0\bin\x64;%PATH%" \
  -n 1 CMD /C "C:\Program Files\R\R-3.2.0\bin\x64\Rgui.exe" \
  : -env PATH "C:\Program Files\R\R-3.2.0\bin\x64;%PATH%" \
  -n 3 "%USERPROFILE%\Documents\R\win-library\3.2\Rhpc\RhpcWorker64.exe"
```



```
R Console (64-bit)
ファイル 編集 その他 パッケージ ウィンドウ ヘルプ

'help()' とすればオンラインヘルプが出ます。
'help.start()' で HTML ブラウザによるヘルプがみられます。
'q()' と入力すれば R を終了します。

[以前にセーブされたワークスペースを復帰します]

> library(Rhpc)
> Rhpc_initialize()
rank 0/ 4(1140850688) : 3478
> cl<-Rhpc_getHandle()
Detected communication size 4
> Rhpc_worker_call(cl, Sys.getpid)
[[1]]
[1] 3868

[[2]]
[1] 2358

[[3]]
[1] 3368

> |
```

また、単一ノードでは無く、複数ノードで動作させる場合は、更に smpd と呼ばれるデーモンをワーカーノード上に手動で立ち上げ、mpiexec 上でホストを指定する必要がある。

## 性能評価

Windows64bit上で4並列(1 Master 3 Worker)の場合での性能は以下のとおりである。以下の場合、並列数が少ないので、exportにおけるMPIの集団通信における恩恵が若干しかない。一方で、遅延評価の為に\*lapply系はまずまずの性能が発揮されている。

	parallel(SOCK)	Rhpc
Transfer of matrix4000 <sup>2</sup> by *export	1.54sec	1.39sec
10000 times of calc sqrt by *lapply	0.70sec	0.08sec
10000 times of calc sqrt by *lapplyLB	0.91sec	0.11sec

## おわりに

コマンドラインからのmpiexecの起動はいささか使いにくく、パスの指定など使い勝手がよいとは言えないので、現在これらを簡単に使えるようにランチャーアプリケーションの開発を行っている。

32bit版ではあまり性能が出ないが、昨今のパソコンの状況から考えて動作確認程度に止めている。

今回ポータビリティを優先したため、最適化に関してはまだ再考の余地があるのでこの点の更なる改良を行っている。

今回の開発で主要なプラットフォームでのRhpcの動作が出来るようになったので、パソコンからスーパーコンピュータに至までのスケラビリティを持った並列環境になったと言える。