Parallel random number generators in R

Asked 7 years, 5 months ago Modified 7 years, 1 month ago Viewed 1k times



Is there an implementation of the Mersenne-Twister and Wichmann-Hill parallel random number generators in R? I'm using the parallel package, but it seems that there is just the implementation of the L'Ecuyer parallel random number generator.



r parallel-processing statistics probability montecarlo



Share Improve this question Follow

asked Jan 3, 2015 at 20:47



Sorted by:

1 Answer



The default random number generator in R is Mersenne-Twister. You can change between them using

1



setRNG('Wichmann-Hill')
setRNG('default')#or setRNG('Mersenne-Twister')



If you want to generate numbers in parallel, you can use the foreach package.



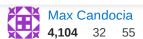
```
require(foreach)
require(doParallel)
c1 <- makeCluster(2)
registerDoParallel(c1)
generateRandom <- function(rng='default',n) {
   setRNG(rng)
   runif(n)
}
result = foreach(i = 1:2,rng = c('default','Wichmann-Hill'), .combine = 'c')
%dopar%
   generateRandom(rng,1e7)
stopCluster(c1)</pre>
```

This example generates 20 million random numbers from a U(0,1) distribution, with each half

Join Stack Overflow to find the best answer to your technical question, help others answer theirs.

Sign up





Join Stack Overflow to find the best answer to your technical question, help others answer theirs.

Sign up

