

Generating Sobol Numbers

There are several open-source libraries that provide quasi random number generators. A good (and platform-independent) C++ library is *QuantLib*, a library for quantitative finance that can be downloaded on www.quantlib.org. The installation is quite easy (follow the instructions on www.quantlib.org/reference/install.html). You can unzip a tarball using `tar xvfz tarballname`.

To generate Sobol numbers, include the class *SobolRsg*. The constructor takes the dimension and a seed as parameters. Note that the seed 0 yields a random initialization for high dimensions ($d > 360$), i.e. not every run of the program produces the same Sobol sequence. After calling the constructor, subsequent points of the Sobol sequence can be obtained by calling the function `nextSequence()`. The return type has a member named `value` that is a d -dimensional vector containing the Sobol number. One can skip to any point in the sequence using the function `skipTo()`. Note that one usually skips the first few points of a Sobol sequence as it is known that they are not very well distributed. A sample program that generates and prints one Sobol number might then look like the following:

```
...
#include <ql/math/randomnumbers/sobolrsg.hpp>

using namespace std;
using namespace QuantLib;

typedef std::vector<QuantLib::Real> QMCPPoint;

int main(int argc, char* argv[]){
    ...
    SobolRsg sobol(dim, 1); // Constructor with seed for deterministic sequence
    sobol.skipTo(4096); // Initial Skip

    QMCPPoint point = sobol.nextSequence().value; // Generate next point
    for(int i = 0; i < point.size(); i++){
        cout << point[i] << " ";
    }
    cout << endl;
    ...
    return 0;
}
```

The program is compiled using a C++ compiler and a linker option to the Quantlib library. Additionally, it might be necessary to specify the location of the Boost library. On a Linux system, this might look like

```
g++ -o SobolGen SobolGen.cpp -lQuantLib -I/usr/local/include/boost_1_38_0
```

Recently (since version 6.2 R2008a), MATLAB has included a few quasi random number generators in its Statistics Toolbox, including one to generate Sobol numbers. A Sobol sequence is initialized by calling for example

```
sobol = sobolset(dim, 'Skip', 4096);
```

and points can afterwards be accessed by simple indexing. Hence

```
point = sobol(10);  
poin = sobol(1:20);
```

generate for example the 10th point and the first 20 points of the sequence (relative to the initial skip).