Comparative U01 test of MIXMAX and of Standard RNGs

Arsen Yeghiazaryan

in collaboration with

Artyom Hovhannisyan

A. I. Alikhanyan National Science Laboratory, Yerevan, Armenia

Athens, Greece
05.09.2016

Outline

1. TestU01
2. MixMax and MersenneTwister tests for pairs
3. RNGs and Monte Carlo integration
4. Types of Monte Carlo algorithms (Plain, MISER, VEGAS)
5. Results and plots
6. Summary

TestU01

A software library implemented in the ANSI C language, offering a collection of utilities for the empirical statistical testing of uniform random number generators (RNGs).

The library was first introduced in 2007, by Pierre L'Ecuyer and Richard Simard of the Université de Montréal.

TESTU01 offers several batteries of tests including "Small Crush" (which consists of 10 tests), "Crush" (96 tests), and "Big Crush" (106 tests).

MixMax and MT tests for pairs

We took the numbers generated by RNGs not consecutively, but rather took them selectively from the vectors, which represent the dimensionality of the RNGs (256 for MixMax and 624 for MersenneTwister).

MixMax

«Small Crush», «Big Crush» for pairs

Passed for the following pairs - (1,2), (1,3), (1,8), (99,100).

MersenneTwister

«Small Crush» for pairs

Passed for the follwoing pairs - (1,8), (10,15), (70,71), (99,101)

(it took about 3 hours each, so we didn't try «Big Crush» for it).

RNGs and Monte Carlo integration

Monte Carlo integration is a technique for numerical integration using random numbers.

While other algorithms usually evaluate the integrand at a regular grid, Monte Carlo randomly chooses points at which the integrand is evaluated (particularly useful for higher-dimensional integrals).

There are different methods to perform a Monte Carlo integration in GSL library

1. Uniform sampling (Plain Monte Carlo)
2. Stratified sampling (MISER)
3. Importance sampling (VEGAS)

Uniform Monte Carlo sampling

$$\int f\,dV \approx V \cdot \frac{1}{N} \sum_{i=1}^{N} f(x_i),$$

$$\sigma \propto \frac{1}{\sqrt{N}}$$

Random sampling of points is simplest.

But quasi-random sampling is better! – stratified sampling, importnace sampling

$$\int f(x)dx = \int \frac{f(x)}{pdf(x)} \cdot pdf(x)\,dx = E[\frac{f(x)}{pdf(x)}]$$

Uniform Monte Carlo sampling

$$\int f\,dV \approx V \cdot \frac{1}{N} \sum_{i=1}^{N} f(x_i),$$

$$\sigma \propto \frac{1}{\sqrt{N}}$$

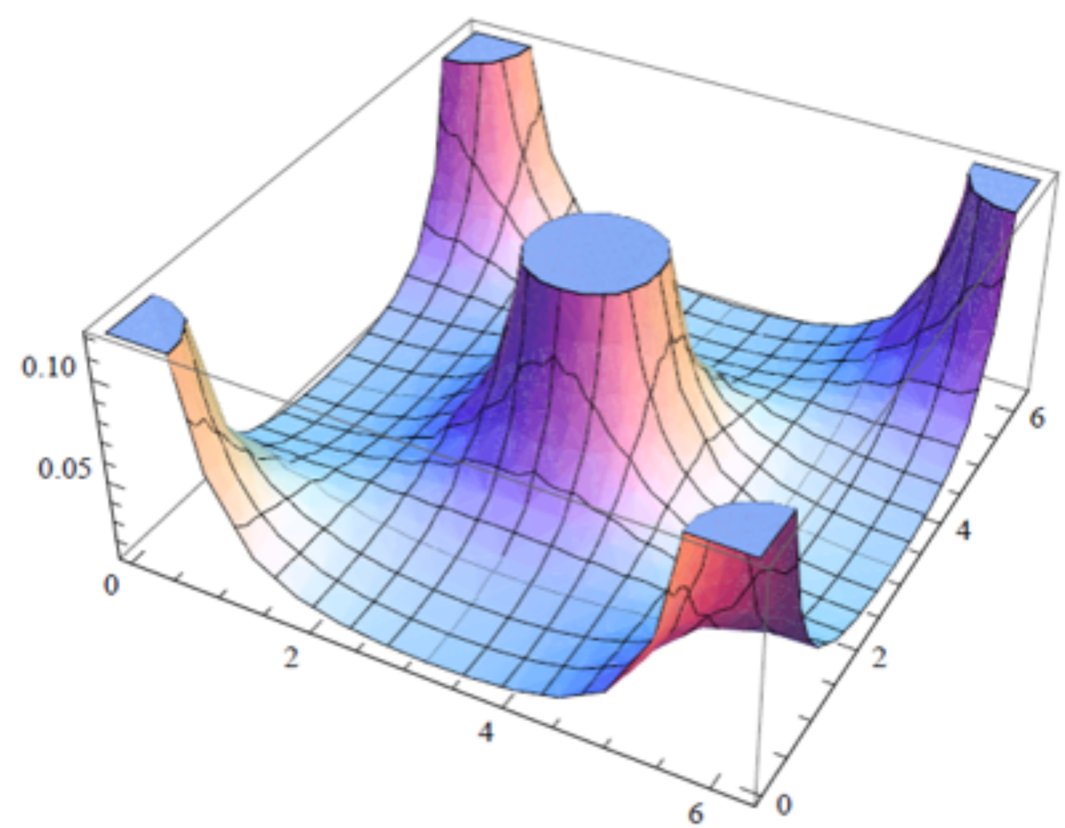
Random sampling of points is simplest.

But quasi-random sampling is better! – stratified sampling, importnace sampling

$$\int f(x)dx = \int \frac{f(x)}{pdf(x)} \cdot pdf(x)\,dx = E[\frac{f(x)}{pdf(x)}]$$

Results and plots

$$I_n = \frac{1}{(2\pi)^n} \int_0^{2\pi} ... \int_0^{2\pi} dx_1 ... dx_n \frac{1}{(1 - \prod_{i=1}^{n} \cos x_i)}$$



$$I_4 = \frac{\Gamma(1/4)^4}{4\pi^3} \approx 1.39320$$

$$\lim_{n \to \infty} I_n = 1$$

$$I_n = \frac{1}{(2\pi)^n}\int_0^{2\pi}\dots\int_0^{2\pi} dx_1 \dots dx_n \frac{1}{(1-\prod_{i=1}^{n}\cos x_i)}$$

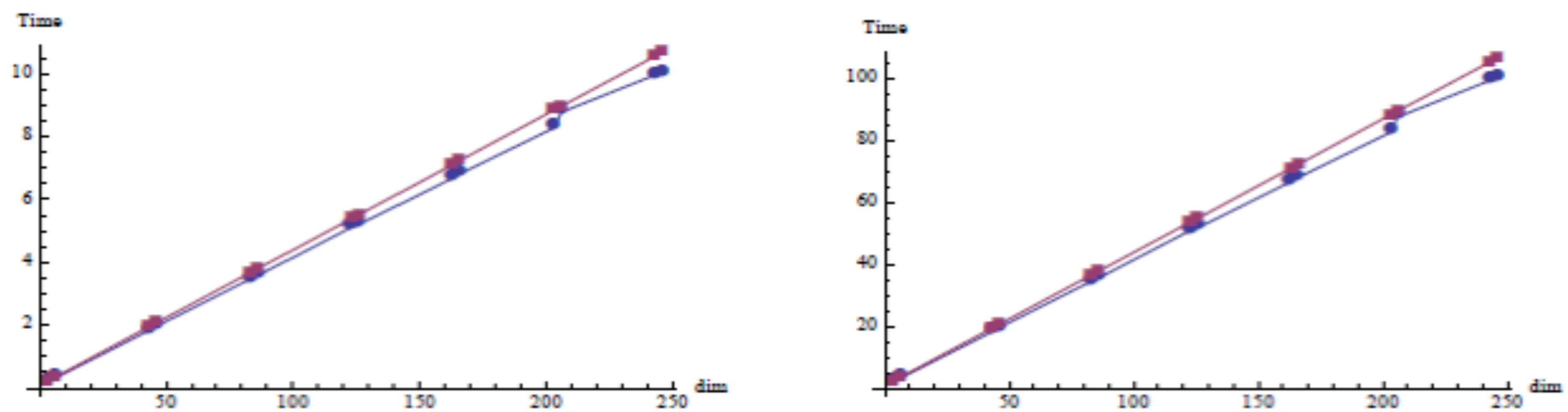


Figure 2: The dependence of calculation time from the dimension of integral I_n Left: Number of points - 10^6. Right: Number of points - 10^7. The blue and purple lines represent the above-mentioned dependence for MixMax and MersenneTwister, respectively. The plots show, that MixMax takes less time to calculate the integrals.

$$I_n = \frac{1}{(2\pi)^n} \int_0^{2\pi} \dots \int_0^{2\pi} dx_1 \dots dx_n \frac{1}{(1 - \prod_{i=1}^{n} \sin^2 x_i)}$$

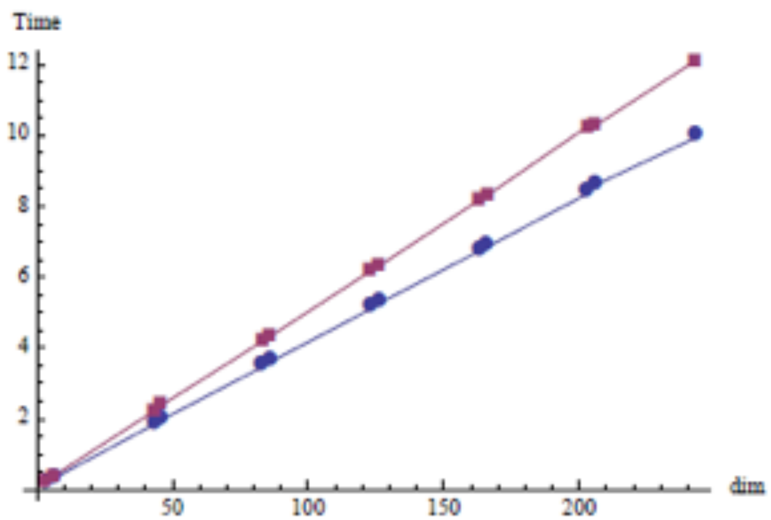


Figure 3: The dependence of calculation time from the dimension of integral $\hat{I}_n$ Left: Number of points - 10^6. Right: Number of points - 10^7. The blue and purple lines represent that dependence for MixMax and MersenneTwister, respectively. Again, the plots show, that MixMax takes less time to calculate the integrals.

For this integral the difference of calculation time between MixMax and MersenneTwister is even wider, about 20-25%!.

It is important to note that these percentage estimates are dependent on the CPU characteristics of the computer and may vary for different machines, in favour of MixMax!

Summary

1. MIXMAX passes testU01 for all tested pairs
2. MIXMAX is faster than Mersenne Twister for all our applications