BRODA's SobolSeq65536 sequence generator with integrated Intel® Math Kernel Library

Contents

BRODA's SobolSeq65536 sequence generator with integrated Intel® Math Kernel Library 1
Introduction 2
I. Main Aspects of SobolSeq65536 Sequence Generator 2
a. Pricing of at the money call 2
b. Pricing of Asian call with geometric averaging 4
II. SobolSeq65536_MKL sequence generator (SobolSeq65536 sequence generator with integrated Intel® Math Kernel Library) 5
a. SobolSeq65536_MKL Generation Modes 6
b. SobolSeq65536_MKL Performance 7
Conclusion 9
References 9

Introduction

SobolSeq65536 sequence generator is a 65536-dimensional Sobol' low discrepancy sequence generator developed by BRODA company [1]. The Sobol' sequence produced by SobolSeq65536 satisfies Property A in all dimensions and property A' for the adjacent dimensions. BRODA's SobolSeq generators outperform all other known generators both in speed and accuracy as shown in [2].

SobolSeq65536_MKL is the recently developed version of SobolSeq65536 sequence generator with integrated Intel® Math Kernel Library inside. SobolSeq65536_MKL use the same direction numbers as SobolSeq65536 so that it has the same statistical properties. Integrated Intel® Math Kernel Library increases random number generation performance on the Intel® Hardware (Intel® HW).

I. Main Aspects of SobolSeq65536 Sequence Generator

Sobol's low discrepancy sequence produced by SobolSeq65536 generator has better statistical properties than the output sequences of the other known generators as shown in [2]. Results of comparison BRODA's SobolSeq65536 sequence generator and Sobol' sequence generator with direction numbers of S. Joe and F. Y. Kuo [3] (further we call it "Joe&Kuo generator") for two well-known financial benchmarks "European option pricing" and "Asian option pricing" are shown and discussed in this Section.

a. Pricing of at the money call

We consider pricing of the European option call with the following parameters: S_0 = 100, K= 100, r = 0.0, σ = 0.2, T = 1y. The theoretical Black-Scholes price of this at the money call (ATM) is C = 7.965667455. Numerically we price it as a path dependent option with the number of time steps d using (1.1) as a terminal value of the underlying asset.

$$S(T) = S_0\, exp[\,(r - \tfrac{1}{2}\sigma^2)T + \sigma\sqrt{\Delta t}(z_1 + z_2 + \ldots + z_d)] = S_0\, exp[\,(r - \tfrac{1}{2}\sigma^2)T + \sigma\sqrt{T}\bar{z}_d] \quad (1)$$

For the standard algorithm the price of the option can be written as the following d-dimensional integral:

$$C = e^{-rT}\int_{H^d} max[\,0, (S_0\, exp[\,(r - \tfrac{\sigma^2}{2})T + \sigma\sqrt{\tfrac{T}{d}}\textstyle\sum_{j=1}^{d}\Phi^{-1}(u_j)] - K)]du_1 \ldots du_d.$$

Here $\Phi^{-1}(u)$ is an inverse cumulative function of a normal distribution.

Firstly, we compute and present the value of a pricing error versus the number of dimensions at different number of sampled path (simulations) (Figure 1). Comparison of BRODA's and Joe&Kuo generators show unacceptably high values of the error produced by the Joe&Kuo generator for medium and especially high dimensions.

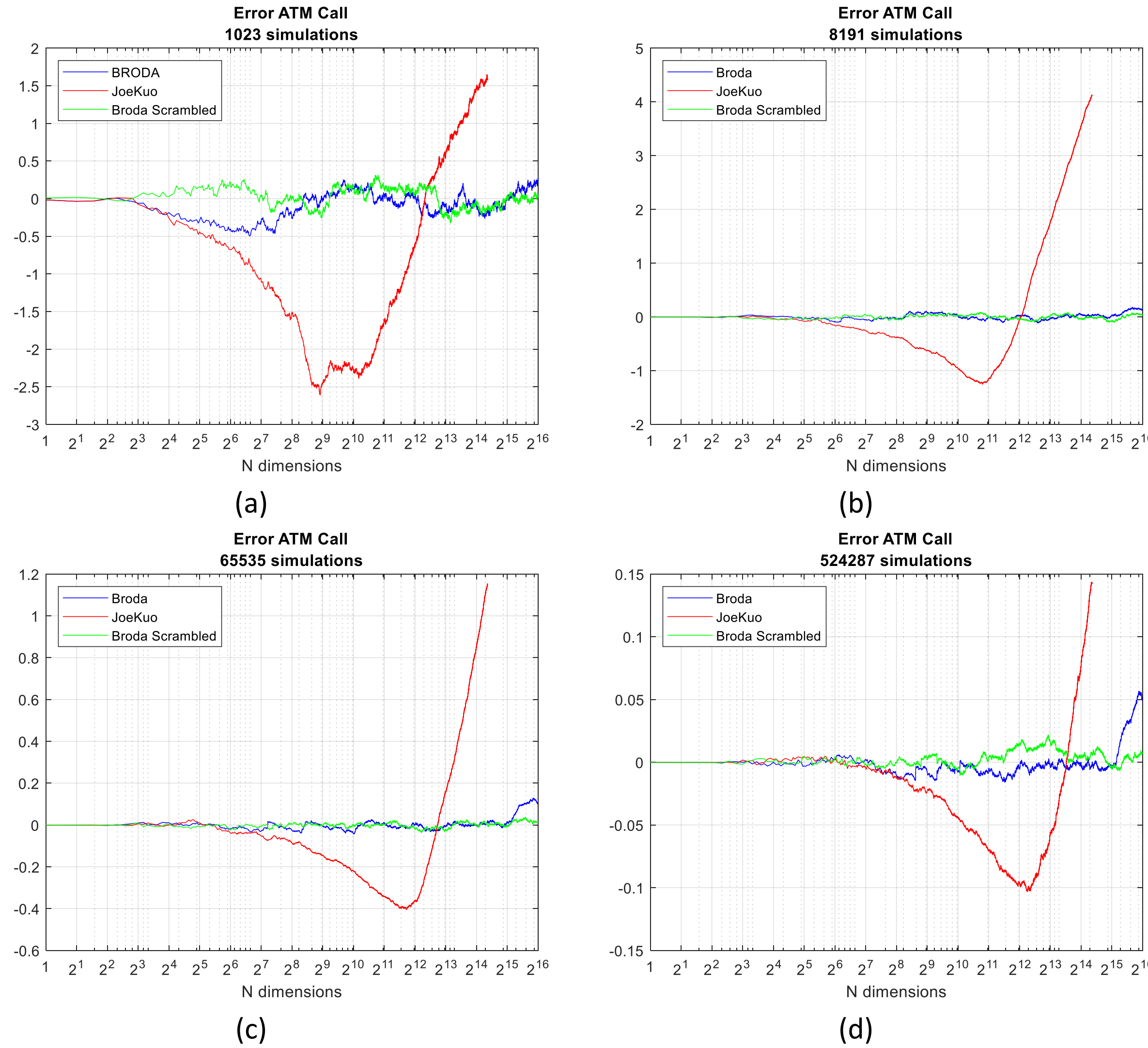


Figure 1. Pricing error of the ATM call versus the number of dimensions at different number N_path of sampled path (simulations: N_path =1023 (a); N_path =8191 (b); N_path =65535 (c); N_path =524287 (d). "BRODA" - BRODA's SobolSeq65536 sequence generator(blue); "BRODA Scrambled" is BRODA's Scrambled SobolSeq65536P sequence generator (green); "JoeKuo" is the Sobol' sequence generator with direction numbers of S. Joe and F. Y. Kuo (red).

Secondly, we compute and present the root mean square error (RMSE) defined as

$$\varepsilon(N) = \left(\frac{1}{L}\sum_{k=1}^{L}(C - C_N^k)^2\right)^{1/2}, \quad (2)$$

where L is a number of independent runs versus number of path N, C_N^k is the numerical value of the option computed at a number of sampled path equal to N at the k-th independent run. For all presented tests L was equal 20. For these tests we also add the results of standard Monte Carlo based on random sampling. For the MC method all runs were statistically independent. For QMC integration for each run a different part of the Sobol' sequence was used. RMSE results are shown in Figure 2.

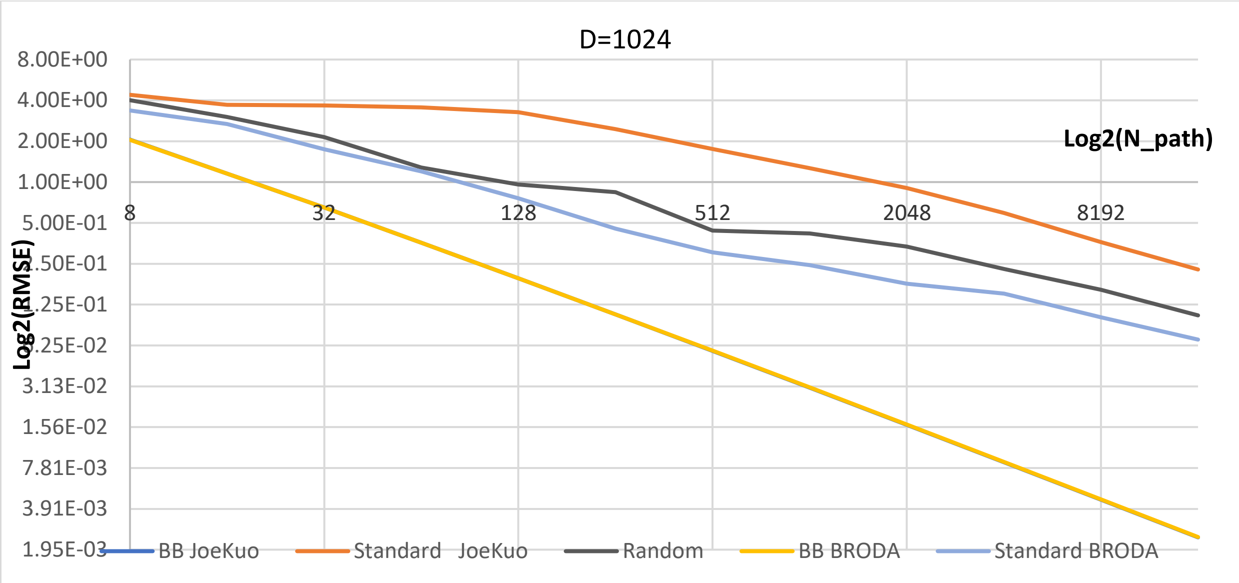


Figure 2. Root mean square integration of the ATM call averaged over 20 trials versus the number of paths for the QMC method computed using BRODA's and Joe&Kuo Sobol' sequence generators for the Standard and Brownian Bridge (BB) sampling schemes. The Standard scheme, Joe&Kuo Sobol' generator (orange); the Standard scheme, BRODA's Sobol' generator (light blue); BB Joe&Kuo Sobol' generator BB (dark blue); BB BRODA's Sobol' generator (yellow); Standard scheme MC sampling (grey). Dimension d =1024.

Comparison of BRODA's and Joe&Kuo generators shows that Joe&Kuo's generator is much less efficient than BRODA's generator in the case of the Standard scheme and it's even less efficient than the random number generator (MC method). In the case of the Brownian bridge (BB) both generators show similar performance. It's explained by the reduction of the effective dimension by the BB [4], hence only low dimensional projections of the Sobol' sequences are important which poses less demand on the quality of the Sobol' sequence.

b. Pricing of Asian call with geometric averaging

We consider pricing of a geometric average Asian call option with the following parameters S_0 = 100, K= 100, r = 0.05, σ = 0.2, T = 0.5y, number of discrete time steps d = 1024. For the standard algorithm the price of a geometric average Asian call option can be written as the following d-dimensional integral:

$$C = e^{-rT} \int_{H^d} max[\,0, (\left[\prod_{i=1}^{d} S_0\, exp[\,(r - \frac{\sigma^2}{2})t_i + \sigma\sqrt{\frac{T}{d}} \sum_{j=1}^{i} \Phi^{-1}(u_j)]\right]^{1/d} - K)]du_1 \dots du_d.$$

The RMSE results are shown in Figure 3.

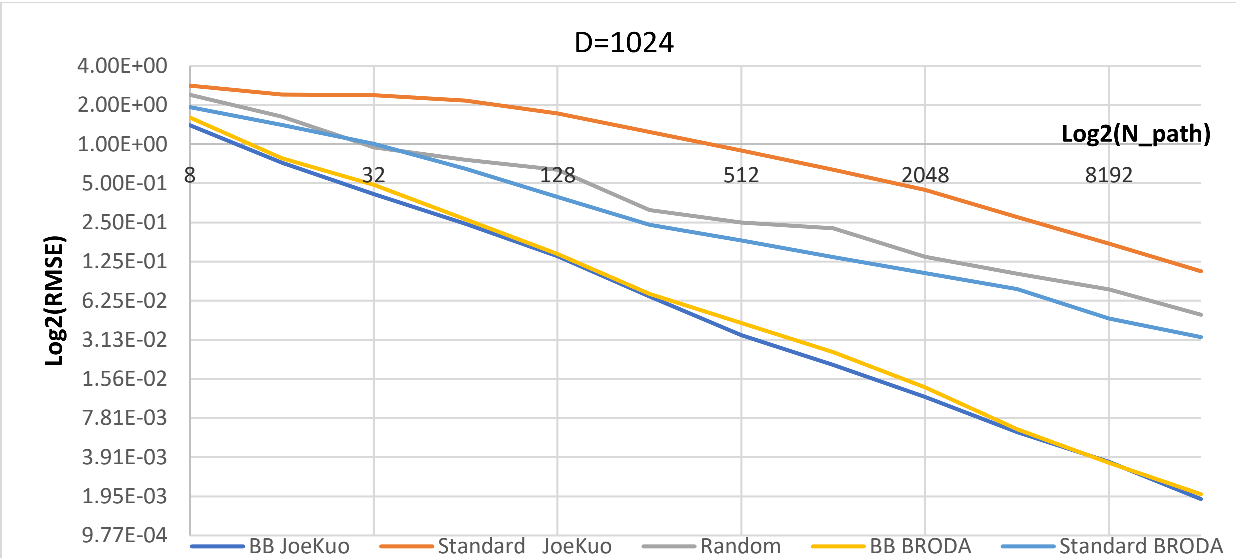


Figure 3. Root mean square integration of the Asian call with geometric averaging averaged over 20 trials versus the number of paths for QMC and MC methods computed using BRODA's and Joe&Kuo Sobol' sequence generators for the Standard and Brownian Bridge (BB) sampling schemes. The Standard scheme, Joe&Kuo Sobol' Sobol' generator (orange); the Standard scheme, BRODA's Sobol' generator (light blue); BB BRODA's Sobol' generator (yellow); Joe&Kuo Sobol' Sobol' generator BB (dark blue); the Standard scheme MC sampling (grey). Dimension D =1024.

Similarly to the previous test case, comparison of BRODA's and Joe&Kuo generators shows that Joe&Kuo's generator is much less efficient than BRODA's generator and even the MC sampling method in the case of the Standard scheme (Figure 3). In the case of the Brownian bridge (BB) both generators show similar performance for the reason outlined above.

II. SobolSeq65536_MKL sequence generator (SobolSeq65536 sequence generator with integrated Intel® Math Kernel Library)

Intel® Math Kernel Library (Intel® MKL) is the fastest and most-used math library for Intel®-based systems. It accelerates math processing routines, increases application performance, and reduces the development time [5]. Intel® MKL Vector Statistics component provides a set of routines implementing commonly used pseudorandom, quasi-random, or non-deterministic random number generators with continuous and discrete distribution. This part of Intel® MKL includes implementation of Sobol' quasi-random number generator which also accepts registration of user-defined parameters (direction numbers and primitive polynomials) during the initialization, which permits obtaining quasi-random vectors of any dimension [6]. This fact formed the basis of Intel® MKL integration into the BRODA's SobolSeq65536 sequence generator.

a. SobolSeq65536_MKL Generation Modes

SobolSeq65536_MKL is SobolSeq65536 sequence generator with integrated Intel® MKL. Direction numbers for the Intel® MKL-based sequence generator are same as for the SobolSeq65536 sequence generator. Both generators have the maximal dimension equal to 65536 and can generate the maximum number of points equal to 2^{32}-1.

SobolSeq65536_MKL has two variants: the first one is a "single-point" version with a point-by-point access (interfaces are the same as for the SobolSeq65536 generator), the second one is a "block" version, that allows to obtain several Sobol' vectors in a single function call. Example of SobolSeq65536 and SobolSeq65536_MKL calls with random numbers post-processing (high dimensional integral computations) are given below:

Example 1. SobolSeq65536. 2^20 random points generation.

```
std::vector<double> sobol_seq(dim, 0.0);
const double c = 0.01;
std::int64_t n_points = 1ul << 20;
std::int32_t dim = 65536;
double I = 0.0;

for (unsigned int j = 0; j < n_points; ++j)
{
    SobolSeq65536_32(j, dim, sobol_seq.data());
    double x = 1.;
    for (auto pnt : sobol_seq)
        x *= 1. - c * (0.5 - pnt);
    I += x;
}
```

Example 2. SobolSeq65536_MKL. 2^20 random points generation. Generation by the single point.

```
std::vector<double> sobol_seq(dim, 0.0);
SobolSeq65536_MKL_Init(dim);
const double c = 0.01;
const std::int32_t dim = 65536;
double I = 0.0;

for (unsigned int j = 0; j < n_points; ++j)
{
    SobolSeq65536_MKL(j, dim, sobol_seq.data());
    double x = 1.;
    for (auto pnt : sobol_seq)
        x *= 1. - c * (0.5 - pnt);
    I += x;
}

SobolSeq65536_MKL_Free();
```

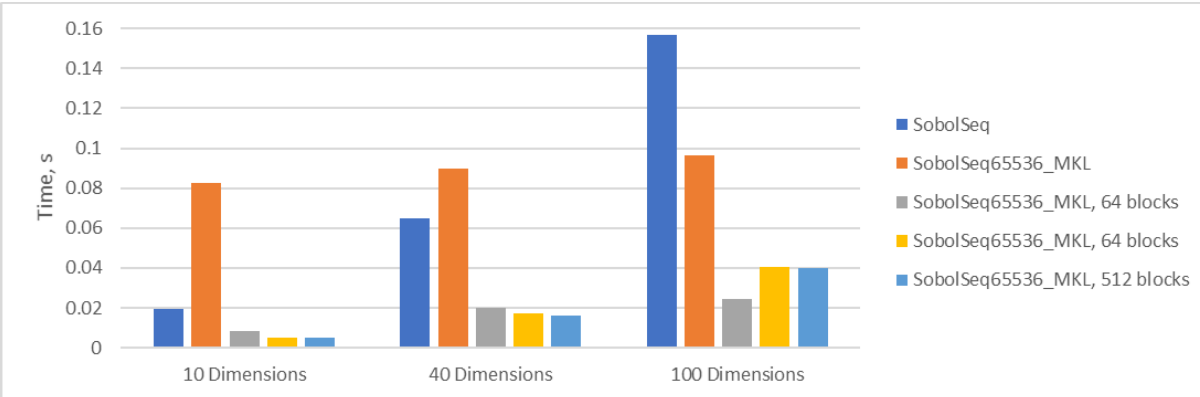
Example 3. SobolSeq65536_MKL. 2^20 random points generation. Generation by the block size of 16 points.

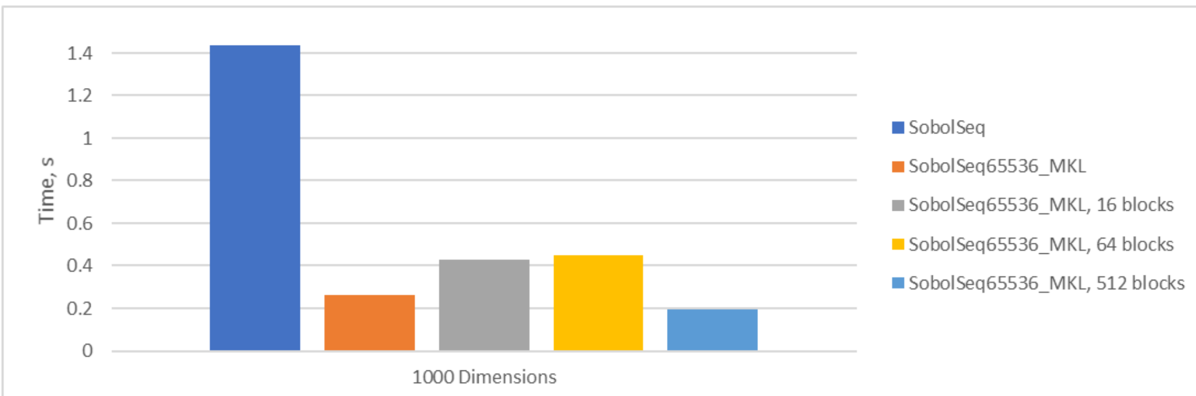
```
std::vector<double> sobol_seq(dim*block_size, 0.0);
SobolSeq65536_MKL_Init(dim);
const double c = 0.01;
const std::int32_t dim = 65536;
const std::int32_t block_size = 16;
double I = 0.0;

for (unsigned int j = 0; j < n_points; j += block_size)
{
    SobolSeq65536_MKL_Block(j, j+block_size-1, dim, sobol_seq.data());
    for (unsigned long p = 0; p < block_size; ++p)
    {
        double x = 1.;
        for (int d = 0; d < dim; ++d)
            x *= 1. - c * (0.5 - sobol_seq[p*dim + d]);
        I += x;
    }
}

SobolSeq65536_MKL_Free();
```

The block generation by SobolSeq65536_MKL can have significant performance acceleration in comparison with the single-point generation mode (performance comparison is represented in chapter II.b SobolSeq65536_MKL Performance).

b. SobolSeq65536_MKL Performance

Performance measurements results (averaged over 10 runs) for 2^20 quasi-random numbers produced by SobolSeq65536 and both generation modes of SobolSeq65536_MKL are presented in Figure 4 (measurement units are seconds). The results were collected on Intel® Xeon® Platinum 8280 CPU 2x28 @ 2.70GHz HW. It follows from the presented charts that a "block" version of SobolSeq65536_MKL always significantly outperforms SobolSeq65536. The best acceleration of SobolSeq65536_MKL versus SobolSeq65536 was achieved in the cases of 10 dimensions – 4X times, 100 dimensions – 4X times, 1000 dimensions – 7.9X times, 16384 – 27.7X times, 65535 dimension – 52.7X times, while a "single" version of SobolSeq65536_MKL outperforms SobolSeq65536 at dimensions above 40.


0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
Time, s
10 Dimensions
40 Dimensions
100 Dimensions
SobolSeq
SobolSeq65536_MKL
SobolSeq65536_MKL, 64 blocks
SobolSeq65536_MKL, 64 blocks
SobolSeq65536_MKL, 512 blocks


1.4
1.2
1
0.8
0.6
0.4
0.2
0
Time, s
1000 Dimensions
SobolSeq
SobolSeq65536_MKL
SobolSeq65536_MKL, 16 blocks
SobolSeq65536_MKL, 64 blocks
SobolSeq65536_MKL, 512 blocks

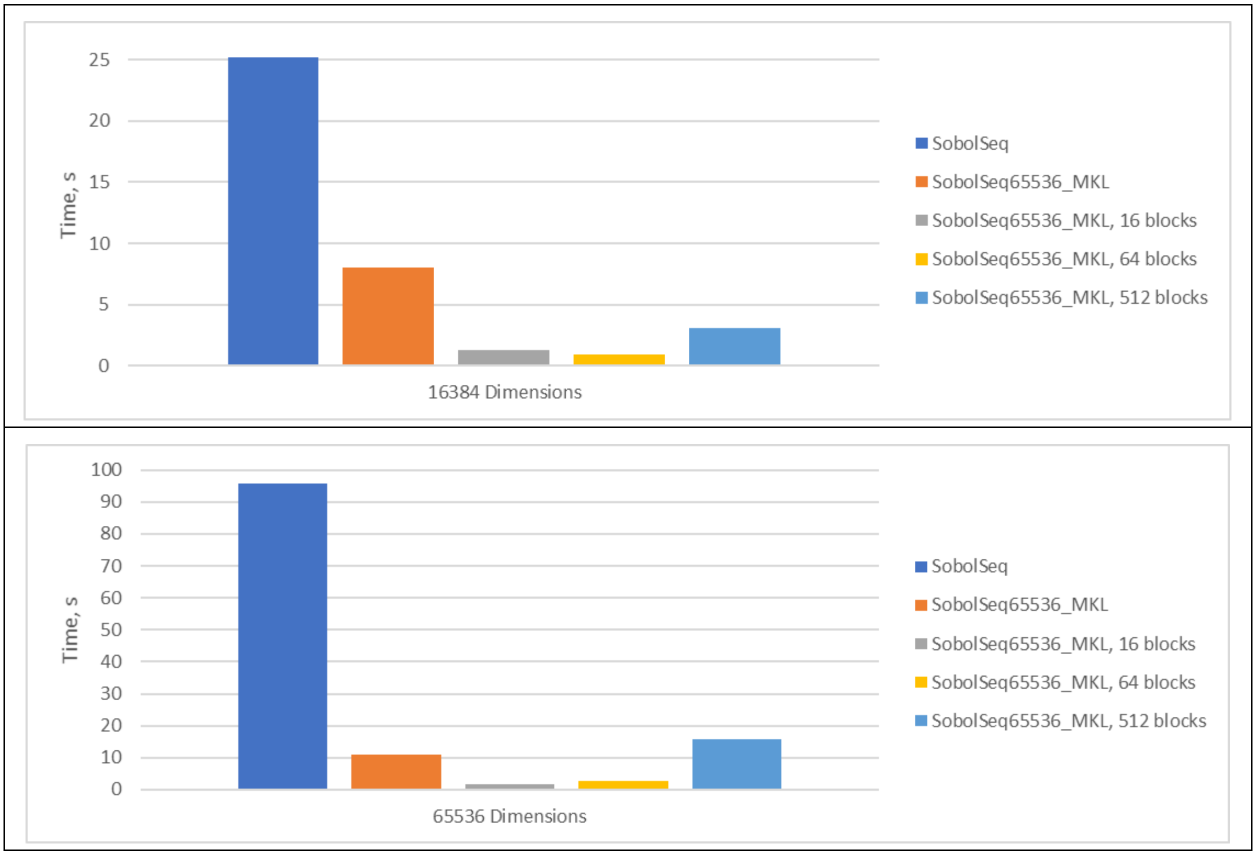


Figure 4. Performance comparison of SobolSeq65536 and SobolSeq65536_MKL generation for the different number of dimensions.

Conclusion

BRODA's SobolSeq65536 sequence generator is a 65536-dimensional Sobol' low discrepancy sequence generator which outperforms all other known generators both in speed and sequence statistical properties. SobolSeq65536_MKL is a recently developed version of the SobolSeq65536 sequence generator with integrated Intel® MKL. SobolSeq65536_MKL accelerates performance of Sobol' sequence generation on Intel® HW. The generator supports two generation modes: a "single point mode": generation is performed by a single point per call, and a "block mode": generation is performed by the set Sobol' vectors per call. Based on the introduced performance results the "block mode" has significant performance benefit in comparison with the "single point" mode because it helps to utilize vectorization opportunities of the Intel® HW.

References

[1] BRODA Ltd. http://www.broda.co.uk (2020).

[2] Sobol' I., Asotsky D., Kreinin A., Kucherenko S. Construction and Comparison of High-Dimensional Sobol' Generators, *Wilmott*, Nov:64-79, 2011, http://www.broda.co.uk/doc/HD_SobolGenerator.pdf

[3] Joe S., Kuo FY. Constructing Sobol sequences with better two-dimensional projections. SIAM Journal on Scientific Computing. 2008;30(5):2635-54.

[4] Bianchetti M., Kucherenko S., Scoleri S., Pricing and Risk Management with High-Dimensional Quasi Monte Carlo and Global Sensitivity Analysis, Wilmott, July, pp. 46-70, 2015, http://www.broda.co.uk/doc/PricingRiskManagement_Sobol.pdf
[5] Intel® MKL - https://software.intel.com/content/www/us/en/develop/tools/math-kernel-library.html
[6] Intel® MKL Vector Statistics – Sobol pseudo random number generator. https://software.intel.com/content/www/us/en/develop/documentation/mkl-vsnotes/top/testing-of-basic-random-number-generators/basic-random-generator-properties-and-testing-results/sobol.html