PAPER • OPEN ACCESS

Numerical simulation of the stochastic Burgers' equation using MLMC and CBC algorithm

To cite this article: Shady A Nagy and Mohamed A El-Beltagy 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **610** 012081

IOP Conf. Series: Materials Science and Engineering **610** (2019) 012081 doi:10.1088/1757-899X/610/1/012081

Numerical simulation of the stochastic Burgers' equation using MLMC and CBC algorithm

Shady A Nagy and Mohamed A El-Beltagy

Engineering Math. & Physics Dept., Engineering Faculty, Cairo University, Giza, Egypt.

E-mails: shady.nagy@guc.edu.eg and zbeltagy@cu.edu.eg

Abstract. Burgers turbulence is a model for developing tools to study the Navier-Stokes turbulence, many hydrodynamical problemsin the theory related to random Lagrangian systems. Many questions that are generally asked in Navier-Stokes turbulence can be answered using Burgers turbulence. The aim of the present paper is to apply Multi-level Monte Carlo (MLMC) on stochastic Burgers' equation by using component-by-component algorithm (CBC). CBC algorithm is developed by the concept of circulant matrix that reduces the cost as a Quasi Monte Carlo technique from *O*(*s n*) to *O*(*s n log n*) where *s* is the dimension of integral with equi-distributed points. In this paper, Burgers' equation is discretized using the finite-volume technique, the MLMC with different random samples is applied and the stability is tested. The results show that MLMC is suitable only for some cases of stochastic differential equations (SDEs) when using pseudo random generator, which is Monte Carlo with high cost than using CBC.

1. Introduction

The Monte Carlo methods are widely used in many stochastic partial differential equations (SPDE) and PDEs with random coefficients [1, 2]. The noise term that added to represent the fluctuations and/or randomness in many engineering and physics applications increases the accuracy of possible events occurs which behave like uncertainty. Consider the following stochastic differential equation in integral form:

$$X_t = \mu(X,t) + \sigma \int dw_t$$

where μ is the drift parameter, w_t is the Brownian motion process that can simulated using different random samples. MC method depends on random numbers generator that can be enhanced using Quasi Monte Carlo (QMC) points which has better convergence rate than ordinary MC points [3]. MLMC is a variance reduction technique that can be used to enhance MC performance. The main advantage of MLMC is reducing the cost of computations compared with MC. In our paper we work on the stability of MLMC, which is related or tailored only for a special kind of stochastic differential equations as for example any linear equation with additive noise term. In this paper, the stochastic Burger's equation which is an example on nonlinear SPDEs is considered. We use MLMC by different number of samples that are generated from MC points as pseudo number generator with normally distributed points and CBC points. CBC is an example for QMC points that requires decreasing the cost more than any other MC points.

The outline of this paper is as follows. In section 2 we present the stochastic Burgers' equation and its physical applications. In section 3 we introduce the MLMC method with discussion of the cost and variance reduction. In section 4 we describe the CBC algorithm which responsible for decreasing the cost depending on vector of numbers that are relatively prime to n which is the base for the generation. It represents in a matrix vector form that has better convergence with $O(s\, n\log n)$. In section 5 we give the simulations using both CBC and pseudo random points and our discussion on the numerical results that related to stochastic Burgers' equation. In section 6 we give our conclusions on MLMC method for nonlinear SPDEs.

2. The Stochastic Burgers' Equation

As an example for a nonlinear SDE, the stochastic Burgers equation is considered. It is a model appears in many applications e.g. superconductors and in studying turbulence [4] and interfaces [5]. A generalized solution for the stochastic Burgers equation in any dimension is recently discussed in [6]. For 1D, the generalized solution will reduce to the classical distributional 1D solution. In the current work, the stochastic Burgers' equation with additive noise is considered. It takes the form:

$$\frac{\partial u}{\partial t}+\frac{1}{2}\frac{\partial}{\partial x}(u^2)=\mu\frac{\partial^2 u}{\partial^2 x}+\sigma(x)W(t),\quad (t,x)\,\epsilon(0,T]\ \times[0,1]$$

Deterministic initial and boundary conditions are assumed as:

$$u(0,x)=u_0(x),\ \ u(t,0)=1\,,\quad u(t,1)=-1$$

It was shown in, e.g. [4], and for finite L^2 norms for u and $\sigma(x)$, the existence of a unique solution with finite second-order moments. This will guarantee the validity of both Wiener chose expansion (WCE) and Wiener-Hermite (WHE) expansions used in the following sections.

Burgers turbulence is a model for developing tools to study the Navier-Stokes turbulence, many hydro dynamical problems and, more recently, in the theory related to random Lagrangian systems. The propagation of random nonlinear waves in non-dispersive media is also studied with the stochastic Burgers' equation. Many questions that are generally asked in Navier-Stokes turbulence can be answered using Burgers turbulence. This is because Burgers equation has the same quadratic nonlinearity as in Navier-Stokes equation [7]. The results obtained by the 1D stochastic Burgers' equation are surprisingly similar to the experimental investigations of the real-life 3D turbulence [8].

After the Hopf-Cole transformation in the fifties, it was realized for a few decades that the stochastic Burgers' equation could not be used to study the flow turbulence. Many trials were done to revisit the Burgers' turbulence successfully, see e.g.[7]. Also, in the current work, we aim to assure that, without digging into the details, the stochastic Burgers equation can be used successfully to study the flow turbulence using the MLMC. The chaotic dynamics and the sensitivity to perturbations can be realized using the stochastic Burgers' equation.

The review article [7] contains detailed applications of the stochastic Burgers' equation and the basic principles of the forced Burgers turbulence. Also, the decay of the solutions of the 1D unforced Burgers' equation with random initial data or random force are studied. More details about the mathematical aspects regarding Burgers turbulence and theoretical background of using WHE can be found in [9]. It was shown in [10] and later in [11] when solving the homogeneous isotropic turbulence for moderate Reynolds numbers that the non-Gaussian part of the solution contributes only a small portion of the energy spectrum except at the very end of the inertial sub-range. It was estimated in [12], using renormalized (time-dependent basis) WHE, to be around 1.5% only of the Gaussian part. So, we shall neglect the non-Gaussianity in the current work for the sake of simplicity and comparison. This assumption is consistent with the well-known fact about the near (and sometimes exact) Gaussian property of the turbulence.

Monte Carlo (MC) sampling is suffering from low accuracy for relatively large time intervals when solving some SDEs, especially of partial differential ones. This problem is more severe in WCE as it uses a limited number of random variables K that will add additional error to the solution as interpretation. Many trials were done to overcome this issue and make the two expansions usable for larger times and relatively higher Reynolds numbers. In case of WHE, the renormalization (time-dependent) basis is suggested, see [8] and the references therein. For the WCE, some authors suggest combining the WCE with Monte-Carlo sampling as in [10] or to use the multistage recursive WCE as in [13]. In the current work, and for the sake of comparison, the basic formulations for MLMC on the random noise using CBC and pseudo random generator which by default is MC with high cost than CBC.

Assuming the turbulence is nearly Gaussian, we can use random generators successfully with moderate times that are normally distributed. For large times, the non-Gaussian part of the energy may become larger than the Gaussian part and hence violates the Gaussian approximation. This means for larger times and/or larger Reynolds numbers, WHE will be poorly convergent and higher-order terms will be needed [14]. In the current work, we shall use relatively shorter time intervals.

3. Multi-level Monte-Carlo (MLMC)

Theorem [15, 16]: Let Q denote a random variable, and let Q_l denote the corresponding level ℓ numerical approximation. If there exist independent estimators $\hat{Y}_l$ based on N_l Monte Carlo samples each costing C_l and a positive constants $\alpha, \beta, \gamma, c_1, c_2, c_3$ such that $\alpha \geq \min(\beta, \gamma)$ and

1) $\left|E\left[\widehat{Q_l} - Q\right]\right| \leq c_1 * 2^{-\alpha l}$

2) $E\left[\hat{Y}_l\right] = \begin{cases} E\left[\widehat{Q_0}\right], & l = 0 \\ E\left[\widehat{Q_l} - \widehat{Q_{l-1}}\right], & l > 0 \end{cases}$

3) $V\left[\hat{Y}_l\right] \leq c_2 * N_l^{-1} * 2^{-\beta l}$

4) $E[C_l] \leq c_3 * 2^{\gamma l}$

Then there exists a positive constant c_4 such that for any $\varepsilon < e^{-1}$ there are values L and N_l for which the multilevel estimator

$$\hat{Y} = \sum_{l=0}^{L} Y_l$$

has mean square error (MSE) with bound

$$E\left[\left(\hat{Y} - E[Q]\right)^2\right] < \varepsilon^2$$

with computational complexity C with bound

$$E[C] \leq \begin{cases} c_4 \varepsilon^{-2} & , \quad \beta > \gamma \\ c_4 \varepsilon^{-2} (log \varepsilon)^2 & , \quad \beta = \gamma \\ c_4 \varepsilon^{-2-(\gamma-\beta)/\alpha} & , \quad 0 < \beta < \gamma \end{cases}$$

Total variance will be:

$$V = \sum_{l=0}^{L} N_l^{-1} V_l$$

Total cost

$$C = \sum_{l=0}^{L} N_l C_l$$

The mean square error

$$\mathrm{E}\left[\left(\hat{Y} - E[Q]\right)^2\right] = N^{-1} V\left[\hat{Q}\right] + \left(E\left[\hat{Q}\right] - E[Q]\right)^2$$

Great accuracy occurs with larger N and small value of weak error $E\left[\hat{Q}\right] - E[Q]$.

The main idea that we need to estimate $E[Q]$ and we don't know enough information about it, so we can use the control variate variance reduction method that is can estimate it though another estimator that is highly correlated to it.

$$N^{-1}\sum_{n=1}^{N}\{h(\omega^n)-\lambda(g(\omega^n)-E[g])\}$$

So Q represent quantity of interest in a given SDE, that we want to estimate and apply our method on it to converge to the mean. The simulation by Matlab is depending on the MC integration for the stochastic integral of the differential equation. We will replace the conventional $MLMC$ that depend on $randn$ pseudo random generator to generate the normal distributed random variables by the new quasi random generator (CBC). It should increase the convergence rate and also improve the overall cost compared with the conventional $MLMC$. Variance of each level equals $O(2^{-\beta l})$, cost of each level equals $O(2^{\gamma l})$ where β is the exponent of variance and γ is the exponent of cost. Optimal number of samples N_l is proportional to $2^{\left(\frac{-(\gamma+\beta)l}{2}\right)}, l=0,\dots,L$ and therefore the cost on level l is proportional to $2^{\left(\frac{(\gamma-\beta)l}{2}\right)}$.

The relation between β and γ determines the behavior of the cost in coarsest levels or finest levels. During the $MLMC$-routine will plot the mean, variance, kurtosis, consistency check versus the different levels. We are only caring in the computational cost that have clear change when applying the QMC algorithm on the $MLMC$ routine. Successive approximations of quantity of interest Q are $Q_0, Q_1, Q_2, \dots, Q_L$ with increasing accuracy but also increasing cost.

Computational cost C is approximately proportional to ε^{-2} when $\beta>\gamma$ so the multiplication $C\varepsilon^2$ would be approximately constant. So we consider decreasing the cost inspite of increasing the number of levels or accuracy. This is the standard result for a MC approach using i.i.d. samples, to do better would require an alternative approach as: QMC methods where CBC algorithm is apart of it . When $\beta<\gamma$, the dominant computational cost will be on the finest levels only. The computational cost in level L will be $C_L=O\left(\varepsilon^{\frac{-\gamma}{\alpha}}\right)$.

The comparison now is between the computational costs that will be taken in simulation with different number of samples for the stochastic Burger's equation. Decreasing in the computational cost between the conventional standard MC and MLMC in case of using the pseudo random generator and CBC algorithm. The main issue is that MLMC converges in all cases linear or nonlinear with additive or multiplicative noise. Generation of random numbers affects the cost due to increase in the number of samples. Distribution of points enhance the cost despite the method either MC or MLMC. We expect that the cost will reduce in case of nonlinear SPDEs as it reduces in many additive types of SDEs.

4. Component-by-Component algorithm (CBC)

Each figure should have a brief caption describing it and, if necessary, a key to interpret the various lines and symbols on the figure. For approximation of high dimensional integral, we can use equal weight cubature rule based on low discrepancy points. Dirk Nuyens [17, 18] developed a good construction of lattice rule for fast and equi-distributed points in s dimension. We will have a look at the construction of rank-1 lattice rules.

If we have a set of points the approximation is given by:

$$\frac{1}{n}\sum_{k=1}^{n} f\left(\left\{\frac{kz}{n}\right\}\right)$$

where z is the randomly generated vector of integers in s dimension,

$$z=(z_1,z_2,z_3,\dots,z_s)$$

$\{.\}$ is the fractional part of the point as if it is fallen outside the cube. The algorithm depends on number of points n which either prime or non-prime. For example, if $n=17,\ s=2$ dimensions, the points shown in Figure 1 are obtained.

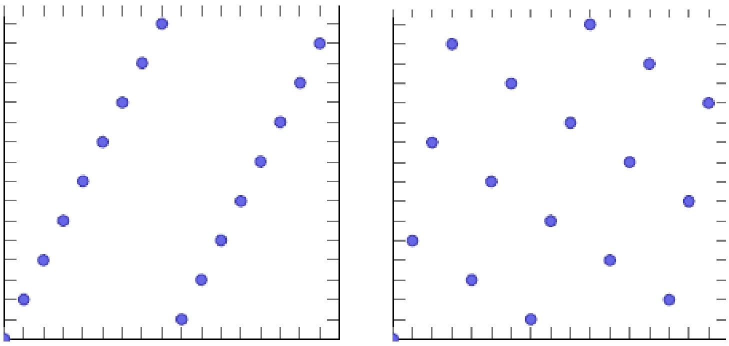

Figure 1. z = (1, 5) left and z = (1, 2) right.

Now we can see, as in Figure 1, the good and bad choice for the distribution of points in the same dimension. CBC algorithm depends on worst-case error that measures mathematically this behaviour in distribution. The worst-case error can be defined as

$$e(Q,f) := \sup_{f\in F,\ \|f\|\leq 1} |I(f) - Q(f)|$$

Where $I(f)$ and $Q(f)$ are the integration and approximation of the function respectively. If we have a set of sample points that represent the stochastic samples, the worst function is the function with worst possible variation with these samples. CBC algorithm depends on the weighted function and the generating random z , as in the algorithm [13].

for $\boldsymbol{s = 1}$ ***to*** $\boldsymbol{s_{max}}$ ***do***
for all $\boldsymbol{Z_s \in U_n}$ ***do***
Calculate $\boldsymbol{e_s^2 = (z_1, z_2, \dots\dots, z_{s-1}, z_s)}$
end for
$\boldsymbol{z_s = \underset{z\in U_n}{argmin}\ e_s^2 = (z_1, z_2, \dots\dots, z_{s-1}, z_s)}$
end for

The choices of z_j are taken relatively prime to n, i.e., from the set:

$$U_n := v \in Z_n : \gcd(v,n) = 1\ , |U_n| = \varphi(n)$$

The worst-case error can be calculated using product weights as:

$$e_s^2(Z_s) = -1 + \frac{1}{n}\sum_{k=0}^{n-1}\prod_{j=1}^{s}\left(1 + \gamma_j\omega(x_j^k)\right)$$

$$= -1 + \frac{1}{n}\sum_{k=0}^{n-1}\ \left(1 + \gamma_s\omega(x_s^k)\right)\prod_{j=1}^{s-1}\left(1 + \gamma_j\omega(x_j^k)\right)$$

$$= -1 + \frac{1}{n}\sum_{k=0}^{n-1}\left(1 + \gamma_s\omega\left(x_s^k\right)\right)p_{s-1}(k)$$

Let us represent the error function using the vector z which is relatively prime to n as the cost of CBC is $O(s\,n^2)$. The new representation in a matrix form decreases the cost using the matrix vector representation as:

$$e_s^2(Z_s) = -1 + \frac{1}{n}\sum_{k=0}^{n-1}\left(1 + \gamma_{\{s\}}\ \omega\left(\frac{k.z_s mod\ n}{n}\right)\right)p_{s-1}(k)$$

The interesting part is:

$$y(z_s) = \sum_{k=0}^{n-1}\left(\omega\left(\frac{k.z_s mod\ n}{n}\right)\right)p_{s-1}(k)$$

Which needs to be calculated for all $z_s \in U_n$, so now have a matrix-vector product:

$$y = \beta_n p_{s-1}$$

According to [19] the complete matrix β_n can be permuted in nested block circulant form for each divisor block with the same permutation. A fast matrix-vector product is possible in $O(n \log n)$ despite it is more complicated. The Dirk Nuyens' Matlab code is to generate this random vector with low processing time that can be used to simulate the different stochastic samples.

5. Simulations

We start by generating good lattice points using the fast quasi matrix Matlab code for simulating the stochastic term in Burger's equation. The generation of these samples uses one pseudo random that is already in MLMC method for decreasing cost with comparison to standard MC. Quasi numbers should decrease the cost more than pseudo numbers generated normally in any computer program, which is normally distributed. We run MLMC on the stochastic Burger's equation with different values of noise intensity and different number of samples. The discretized Burger's equation is used and the noise term is added using both CBC points and pseudo random numbers.

We shall consider four different test cases according to the parameters used in MLMC. The values of parameters are given and the results are recorded.

Case 1: using MLMC coefficients with CBC and the estimates of key MLMC parameters based on linear regression as:

$\alpha = 1.000000$ (Exponent for MLMC weak convergence)
$\beta = 2.000000$ (Exponent for MLMC variance)
$\gamma = 0.981747$ (Exponent for MLMC cost)

Table 1 and Figure 2 show the results for Case 1. We can notice that the computational cost, despite increasing number of levels, will increase in case of MLMC than standard MC. The CBC points that simulate the stochastic term do not increase the cost which is required to be decreased. Computational cost in CBC points is less than pseudo numbers that depends on MC algorithm.

Case 2: using MLMC coefficients with CBC and the estimates of key MLMC parameters based on linear regression as:

$\alpha = 1.003035$, $\beta = 2.005852$, $\gamma = 1.016542$.

The computational cost (in seconds) are given in Table 2.

Case 3: using MLMC coefficients with CBC and the estimates of key MLMC parameters based on linear regression as:

$\alpha = 0.999314$, $\beta = 1.998694$, $\gamma = 0.953135$.

The computational cost (in seconds) are given in Table 3.

Table 1. Cost for different values of ε using Std MC and MLMC – Case 1.

$\boldsymbol{\varepsilon}$	Std MC-cost	MLMC cost
0.005	100	23.56
0.01	150	34.64
0.02	200	23.76
0.05	250	27.9

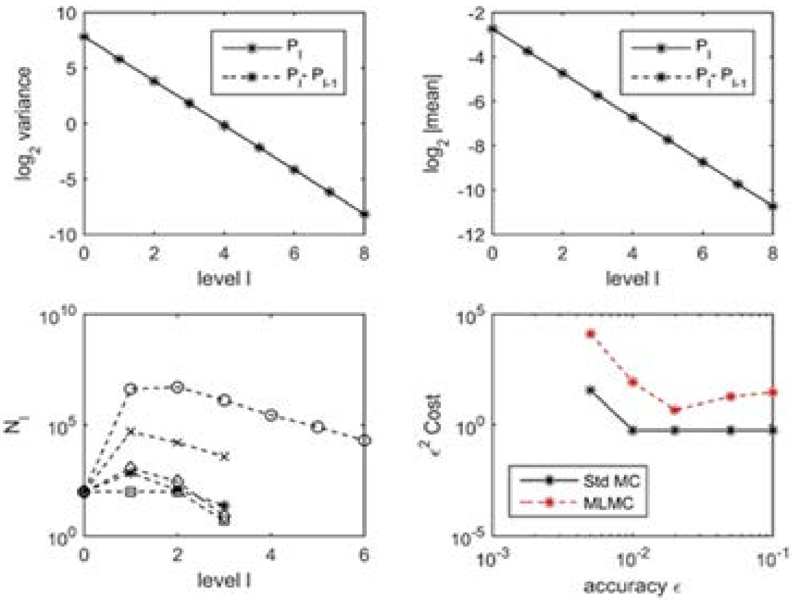

Figure 2. MLMC using CBC algorithm on stochastic Burger with $\sigma = 0.1$, N=10^5 – Case 1

Table 2. Cost for different values of ε using Std MC and MLMC – Case 2

$\boldsymbol{\varepsilon}$	Std MC-cost	MLMC cost
0.005	0.026226	0.006537
0.01	0.006537	0.006537
0.02	0.006537	100.41
0.05	54.4185	47.82
0.1	90.8375	281.7

Table 3. Cost for different values of ε using Std MC and MLMC – Case 3

$\boldsymbol{\varepsilon}$	Std MC-cost	MLMC cost
0.005	16801.22	4200.29
0.01	1050.06	262.50
0.02	65.61	314701.42
0.05	248836.81	191601.61
0.1	136809.98	93768.30

We can conclude from the results that despite the minimum cost of CBC, it enhances the speed and the convergence of the solution but it does not decrease the cost. Also using the pseudo random generator, as stochastic sample generator in Burgers' equation does not decrease the cost, which indicates the instability of MLMC in nonlinear SPDE. Therefore, MLMC is not a general case for decreasing cost than MC especially in nonlinear or multiplicative cases of stochastic equations.

IOP Conf. Series: Materials Science and Engineering **610** (2019) 012081 doi:10.1088/1757-899X/610/1/012081

6. References

[1] Barth A, Schwab C and Zollinger N 2011 Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients, *Numer. Math.*, **119** 123-161
[2] Cliffe K, Giles M, Scheichl R and Teckentrup A 2011 Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients, *Computing and Visualization in Science* **14** 3-15
[3] Niederreiter H 1992 *Random Number Generation and Quasi-Monte Carlo Methods (CBMS-NSF Regional Conference Series in Applied Mathematics)*, SIAM
[4] Funaki T, Surgailis D and Woyczynski W 1995 Gibbs-Cox random fields and Burgers' turbulence, *Annals of Applied Probability*, **5** 461-492
[5] Da Prato G, Debussche A and Temam R 1994 Stochastic Burgers' equation, *Nonl.Differ. Eq, Appl.*, **1** 389-402.
[6] Catuogno P, Colombeau J and Olivera C 2018 Generalized solutions of the multidimensional stochastic Burgers equation, *J. Math. Anal. Appl.*, **464** 375-1382
[7] Bec J and Khanin K 2007 Burgers turbulence, *Physics Reports*, **447** 1-66
[8] Chekhlov A and Yakhot V 1995 Kolmogorov turbulence in a random-force Burgers' equation, *Phys. Rev. E*, *52* 5681-5684
[9] Woyczynski W 1998 *Burgers-KPZ Turbulence: Gottengen Lectures*, Springer-Verlag, New York
[10] Wang J 1974Wiener-Hermite expansion and the inertial subrange of a homogeneous isotropic turbulence, *Phys. Fluids*, **17** 1130
[11] Meecham W 1999 Scaleless algebraic energy spectra for the incompressible Navier–Stokes equation; relation to other nonlinear problems, *J. Mar. Sys.*, **21** 113-130
[12] Lee C 1982 Application of the Wiener–Hermite expansion to turbulence of moderate Reynolds numbers, *Phys. Fluids*, **25** 1322
[13] Luo W 2006 Wiener chaos expansion and numerical solutions of stochastic partial differential equations, Ph.D. thesis, California Institute of Technology, Pasadena, California, United States
[14] Zhang Z Rozovskii B, Tretyakov M and Karniadakis G 2012 A multistage Wiener chaos expansion method for advection-diffusion equations, *SIAM J. Sci. Com.*, **34** A914–A936
[15] Giles M 2008 Multilevel Monte Carlo path simulation, *Oper. Res.* **256** 981-986
[16] Giles M 2015 Multilevel Monte Carlo methods, *Acta Numer.* **24** 259-328
[17] Nuyens D 2014 The construction of good lattice rules and polynomial lattice rules, In uniform distribution and Quasi-Monte Carlo methods, *Radon Series on Computational and Applied Mathematics,* **15** 223-256
[18] Nuyens D and Cools R 2006 Fast algorithms for component-by-component construction of rank-1 lattice rules in shift-invariant reproducing kernel Hilbert spaces, *Math. Comp.* **75** 903-920
[19] Nuyens D and Cools R 2006 Fast component-by-component construction, a reprise for different kernels, *Monte Carlo and quasi-Monte Carlo methods 2004,* Springer, Berlin.