ERROR ESTIMATE OF POLYNOMIAL CHAOS METHODS APPLIED TO CAR SUSPENSION SYSTEM

Duc Thu Dao
Sao Do University
daoducthu85@gmail.com

Dinh Manh Pham
Thanh Dong University
manhvkhln@gmail.com

ABSTRACT

Today, with the development of science and technology, new products are created quickly using numerical simulation technology. One of the effective numerical simulation methods is the Polynomial Chaos method. However, when calculating with this method, there will be errors. When we do not know the exact result, we cannot know whether this error is large or small. In this article, the author develops a method to help determine the error of the Polynomial Chaos method, from which the approximate result can be determined with the exact result.

Keywords: *Sampling method, Polynomial Chaos method, dynamics, random variable.*

1. INTRODUCTION

One of the effective methods used to help solve dynamic problems with uncertain parameters is the Polynomial Chaos (PC) method [1]. This method significantly reduces the calculation time compared to the Monte Carlo method. However, there are still errors in the calculation process using this method. In this paper, we propose a method to estimate the error of the Polynomial Chaos method. The results of this method are compared with the precision error estimate determined as the error between the Polynomial Chaos method and the Monte Carlo method.

2. RESEARCH SUBJECTS AND METHODS

2.1 Research subjects

The object of study is the 1/4 suspension system on cars with uncertain parameters.

2.2 Research methods

2.2.1. Document inheritance method

Collect and collect professional documents related to the dynamics of automobiles to serve as a basis for theoretical research.

2.2.2. Theoretical research methods

We are using automobile theory and engineering mechanics to build dynamic computational model oscillations of the 1/4 suspension on cars.

2.2.3. Method Monte Carlo

This method is named after a city in Monaco. This method will randomly take samples and calculate directly on these samples, so the larger the number of samples taken, the more accurate the results. The accuracy of the results depends on the number and selection of samples. To get accurate results, it is necessary to calculate with a large number of samples, so with a considerable calculation time. Therefore, people only use the Monte Carlo method to verify the results.

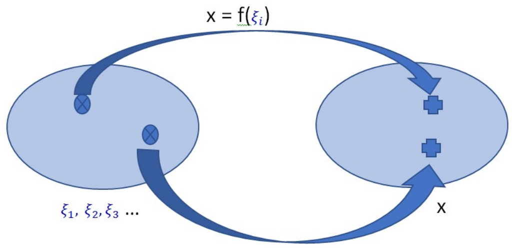


Fig. 1. *Block diagram of the Monte Carlo method*

Within ξ_i the set of samples taken, according to the law of large numbers, the mean is calculated by the formula [2], [3]:

$$x_{MC} = \frac{1}{n_{MC}} \sum_{i=1}^{n_{MC}} x(\xi_i) \quad (1)$$

Where n_{MC} is the number of samples.

2.2.4. Method PC

This is the probability method. With this method, people will choose a small number of samples and calculate on these samples, and the remaining results will be interpolated according to the calculation results in the samples.

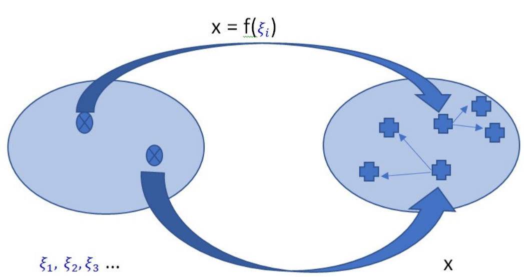


Fig. 2. *Block diagram of the PC method*

With every sample ξ in the episode then Λr, the values of the variables will be approximated according to the formula [4]:

$$x(\underline{\xi},t) \approx x_{PC}(\underline{\xi},t) = \sum_{j=0}^{N_p} \bar{x}_j \phi_j(\underline{\xi},t) \quad (2)$$

ϕj is the polynomial of the PC method, $\underline{\xi} = (\underline{\xi}_1, \ldots, \underline{\xi}_r)$ are independent vectors, where r is the number of uncertain variables. The relationship between these variables and the polynomials of the PC method will be shown in Table 1 [4].

Table 1. *The relationship between the variable and the polynomial of the method PC*

Variable $\underline{x}$	Polynomial f_j	Value
Gaussienne	Hermites	(-∞, +)
Uniforme	Legendre	[a, b]
Gamma	Legendre	[0,)
Beta	Jacobi	[a, b]

With $\underline{\alpha} = (\alpha_1, \ldots \alpha_r)$ and turn ξ According to Uniforme's law, according to document [4], this polynomial is calculated by the formula:

$$\phi_j(\underline{\xi}) = \phi_{(\alpha_{1,j} \ldots \alpha_{r,j})}(\underline{\xi}) = \prod_{k=1}^{r} L_{i_1(j)}(\underline{\xi}_1) \otimes \ldots L_{i_r(j)}(\underline{\xi}_r) \quad (3)$$

With $L_{i_k(j)}$ (k = 1÷r) is a Legendre polynomial defined by the formula:

$$(n+1)L_{n+1}(x) = (2n+1)xL_n(x) - nL_{n-1}(x) \quad (4)$$

With $L_0(x) = 1$ and $L_1(x) = x$

With the above rule, the variable relationship ξ and polynomial ϕj will depend on the coefficient p of the method polynomial PC(selected by the calculator) and get show according to Figure 3.

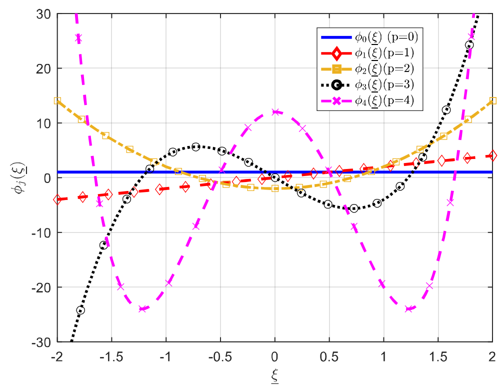


Fig. 3. *Relationship between variable ξ and polynomial ϕj*

For p = 0 and p = 1, then the polynomial ϕj represented as a straight line. For p ≥ 2 then the polynomial ϕj is represented as a curve with the corresponding number of degrees.

According to [5], $\underline{\alpha}$ calculated according to the formula:

$$\|\underline{\alpha}\|_q = (\alpha_1^q + ... + \alpha_r^q) \quad (5)$$

(For the PC method, q=1, for the method Polynomial Chaos Ceux then 0<q<1)

There are two ways to choose the value for $\|\alpha\|_q$[5], [6].

Case 1: $\|\alpha\|_q \le p \times r$, in this case, Np is calculated according to the formula:

$$N_p = (1+p)^r \quad (6)$$

Where r is the number of uncertain parameters, p is the coefficient of the polynomial (chosen by the calculator).

Case 2: $\|\alpha\|_q \le p$, in this case Np is calculated according to the formula:

$$N_p + 1 = \frac{(p+r)!}{p!r!} \quad (7)$$

$\bar{x}_j$ is the coefficient of PC calculated by the formula:

$$\bar{x} = (\phi^T(\underline{\xi}^{(q)})\phi(\underline{\xi}^{(q)}))^{-1}\phi^T(\underline{\xi}^{(q)})x(\underline{\xi}^{(q)}) \quad (8)$$

With $\underline{\xi}^{(q)} = (\underline{\xi}_1^{(q)},...,\underline{\xi}_r^{(q)})$ with $q = 1,...,Q$

$\phi(\underline{\xi}^{(q)})$ is determined by the formula:

$$\phi(\underline{\xi}^{(q)}) = \begin{pmatrix} \phi_0(\underline{\xi}^{(q)}) & \cdots & \phi_{N_{p-1}}(\underline{\xi}^{(q)}) \\ \vdots & \ddots & \vdots \\ \phi_0(\underline{\xi}^{(Q)}) & \cdots & \phi_{N_{p-1}}(\underline{\xi}^{(Q)}) \end{pmatrix} \quad (9)$$

2.2.5. Sampling method

With the experimental method, the quantity and method of sampling are very important in determining the quality of the results. For selected samples, it is necessary to have representative properties for all remaining elements.

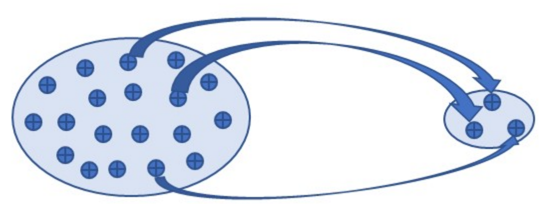


Fig. 4. *Sampling method*

Some typical sample selection methods are:

2.2.5.1. How to select a Monte Carlo (MC) sample

In this way, the samples are numbers taken at random in the sample space [3]. An example of a nine-element sampling with a two-variable uncertainty system with a sampling method is Monte Carlo.

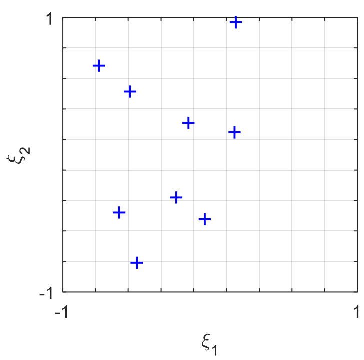


Fig. 5. *Sampling with MC method*

We can see with this approach that the samples are taken at random. Therefore, to reduce the error of the calculation method, the number of samples must be large.

2.2.5.2. How to select a Latin Hypercube sample (LHS)

The Latin Hypercube sampling method is a method developed from the Monte Carlo sampling method [3]. A sample with the LHS selection method is created by dividing the space of the input variables into different subspaces and sampling each of these subspaces. With the LHS method, we obtain a number of samples Q for r random variables in three steps [7]:

Step 1: Simulate the sampling space in the $Q \times r$ cells.

Step 2: We randomly choose r permutations of $\{1, \ldots, Q\}$: $\pi 1$,. . . , πr, determines Q active cells.

Step 3: Generate Q-independent variables uniformly across active cells.

For example, with a sample size of 9, the system has two uncertain parameters, LHS. The samples are shown in Figure 6.

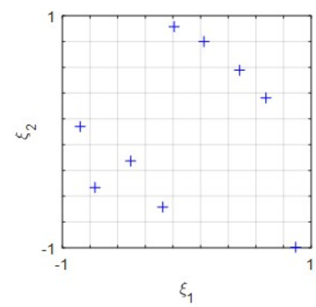


Fig. 6. *Sampling with the LHS method*

2.2.5.3. How to choose a Uniforme sample (USG)

This is a fixed sampling method. It divides the space in a uniform way. Spacing depends on the number of variables. With this sampling method, with the number of variables r and n being the size, the number of samples will be calculated according to the formula [7] : $Q= n^r$ (11)

With this method, sampling is uniform in all directions. That is, the values of the variables are the same and uniform in all directions.

For example, with a sample size of 9 and a system with two uncertain parameters, when selecting according to the USG sampling method, the samples will be represented as shown in Figure 8.

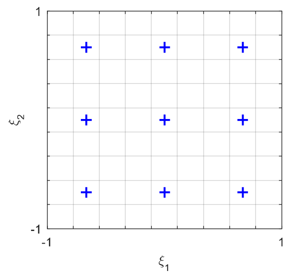


Fig. 7. *Sampling with the USG method*

3. RESULTS AND DISCUSSION

3.1. Modeling a quarter-vehicle suspension model

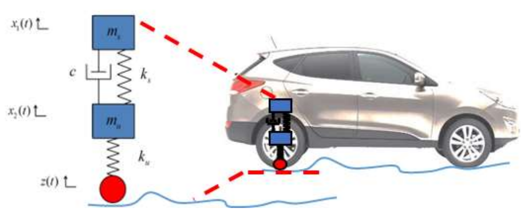


Fig. 8. *Modeling a quarter-vehicle suspension model*

According to the documents [8], [9], the vibration model ¼ on the car is described as follows:

$$\begin{cases} m_s \ddot{x}_1 = -k_s(x_1 - x_2)^3 - c(\dot{x}_1 - \dot{x}_2) \\ m_u \ddot{x}_2 = k_s(x_1 - x_2)^3 + c(\dot{x}_1 - \dot{x}_2) + k_u\{z(t) - x_2\} \end{cases} \quad (10)$$

With

m_s: Mass of the part to be suspended.

m_u: Mass of non-hanging part.

k_s: Suspension stiffness.

c: Coefficient of damping damping.

k_u: Wheel stiffness.

$z_{(t)}$: The undulation of the road surface.

$x_{1(t)}$: Coordinate the system attached to the vehicle body.

$x_{2(t)}$: Coordinate the system attached to the wheel.

During the car's oscillation, some parameters of the car's suspension system are nonlinear such as tire stiffness, rubber mound, etc. Therefore, we propose the value of suspension stiffness and Wheel stiffness varies by about 10%.

Table 2. *Parameters of surveyed cars [8]*

Parameter	**Value**
ks	400 N/m ± 10%
ku	2000 N/m ± 10%
ms	40 kg
mu	20 kg
c	600 Ns/m
Zmax	0.2 m

3.2. Error estimation method of the PC method

To generalize, the system of equations (10) can be rewritten to equation (11):

$$M\ddot{x}(t) + f(x) = F(t) \quad (11)$$

With :

$$M = \begin{pmatrix} m_s & 0 \\ 0 & m_u \end{pmatrix} \quad 12)$$

$$f(x) = [f_1(x), f_2(x)]^T \quad (13)$$

$$f_1(x) = -k_s(x_1 - x_2)^3 - c(\dot{x}_1 - \dot{x}_2) \quad (14)$$

$$f_2(x) = k_s(x_1 - x_2)^3 + c(\dot{x}_1 - \dot{x}_2) - k_u x_2 \quad (15)$$

$$F(t) = [k_u z(t), 0]^T \tag{16}$$

Suppose we can calculate exactly the result of the system of equations as x_{cx}, then:

$$M\ddot{x}_{cx}(t) + f(x_{cx}) = F(t) \tag{17}$$

With x_{PC} is the result of the system of equations (11) with the PC method, then:

$$M\ddot{x}_{PC}(t) + f(x_{PC}) \neq F(t) \tag{18}$$

Thus, we can calculate the residuals of the PC method:

$$R = M\ddot{x}_{PC}(t) + f(x_{PC}) - F(t) \tag{19}$$

Assuming e is an error in the calculation by the PC method, then:

$$e = x_{PC} - x_{cx} \tag{20}$$

From the formulas (12), (18), (20), (21) we can deduce:

$$M\ddot{e}(t) + f(x_{PC} - e) + f(x_{PC}) = R \tag{21}$$

Using the MC method to calculate the formula (22), we will find the error in the calculation process by the PC method, from which we will find the result of the problem.

Specifically, with the car suspension system simulated as the system of equation (1), we can calculate R_1 and R_2 according to formula (23).

$$\begin{cases} R_1 = m_s\ddot{x}_{1CP} + k_s(x_{1CP} - x_{2CP})^3 + c(\dot{x}_{1CP} - \dot{x}_{2CP}) \\ R_2 = m_s\ddot{x}_2 + k_s(x_{1CP} - x_{2CP})^3 - c(\dot{x}_{1CP} - \dot{x}_{2CP}) - k_u(z(t) - x_{2CP}) \end{cases}$$

(22)

With e_1 and e_2 defined by:

$$\begin{cases} e_1 = x_{1CP} - x_{1cx} \\ e_2 = x_{2CP} - x_{2cx} \\ \dot{e}_1 = \dot{x}_{1CP} - \dot{x}_{1cx} \\ \dot{e}_2 = \dot{x}_{2CP} - \dot{x}_{2cx} \\ \ddot{e}_1 = \ddot{x}_{1CP} - \ddot{x}_{1cx} \\ \ddot{e}_2 = \ddot{x}_{2CP} - \ddot{x}_{2cx} \end{cases} \tag{23}$$

From formulas (21), (22), (23) we get a system of equations to determine the error of the PC method according to formula (24):

$$\begin{cases} m_s\ddot{e}_1 = R_1 - k_s(x_{1CP} - x_{2CP})^3 + k_s(x_{1CP} - x_{2CP} - e_1 + e_2)^3 - c(\dot{x}_{1CP} - \dot{x}_{2CP}) \\ m_u\ddot{e}_2 = R_2 - k_s(x_{1CP} - x_{2CP})^3 - c(\dot{x}_{1CP} - \dot{x}_{2CP}) - k_u(z(t) - x_{2CP}) \end{cases} \tag{24}$$

3.3. Research results

To compare between the calculation methods, the paper uses the method of comparing the difference of the mean square of the comparative quantity according to the expression:

$$T = \frac{\int_0^t e_1^2 dt}{\int_0^t e_{cx1}^2 dt} \tag{25}$$

With e_1 is the result of error estimation, e_{cx1} is the error between the PC method and the control method, *t* is the calculation time.

With the above system of equations, using the Matlab software program, we get the simulation results corresponding to the case where the input is the simulated road surface, as shown in Figure 9:

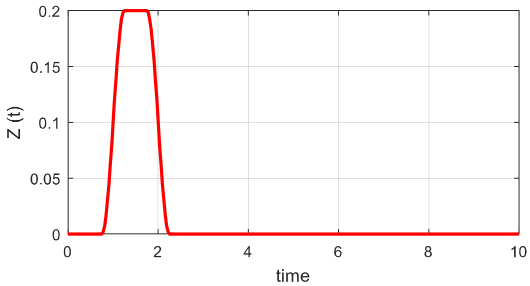
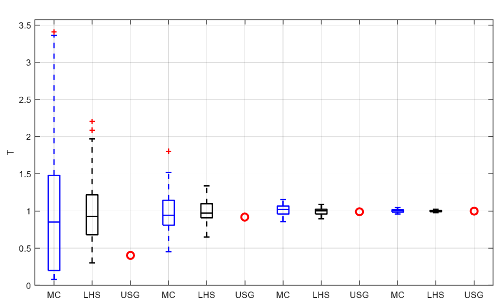


Fig. 9. *Road surface undulation*

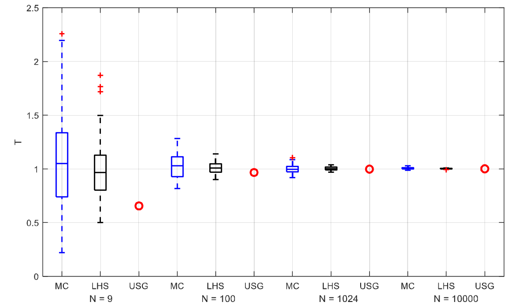
Through the above figure, we see that in the first second, the car moves on a flat road surface, then encounters a bump with a height of 0.2 m, then the car continues to travel on a flat road surface.

With the above calculation conditions, the author calculates with the input conditions: when calculating with the PC method, the sampling method is MC, p = 2, 3, 4, 5; When using the MC method to estimate the error, the sampling method is MC, LHS, USG with the number of samples: 9, 100, 1024, 10000.

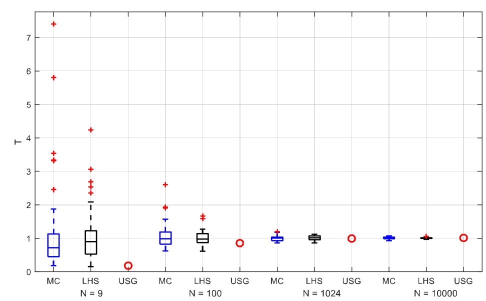
Because the calculation results after each calculation are different, in order to observe the error area of the calculated cases, the author has performed the calculations repeatedly 100 times.



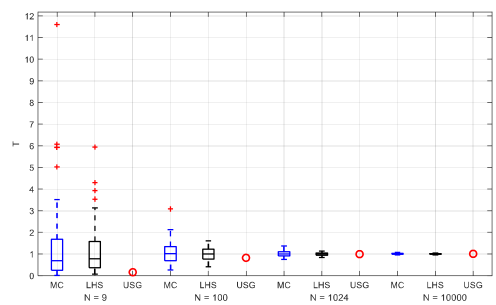
a. p=2



b. p=3



c. p=4



d. p=5

Fig. 10. *Error estimation*

With MC sampling method, the sample selected is random in the sampling space. Thus, with the same number of samples, when taking using this method, it will produce results that are not identical in the process of repetition. Hence the repeated calculation will produce different results, which are shown as blue cubes as shown in Figure 10.

The Hypercube Latin sampling method is a method developed from the Monte Carlo sampling method. With the LHS sampling method, the sample is created by dividing the space of the input variables into different subspaces and randomly taking each of these subspaces. Thus, when repeating this method for sampling, different sets of samples will be generated. Therefore, with the same number of samples, repeated calculations will produce different results shown as black blocks.

With the USG sampling method, only one set of samples can be identified for each sampled quantity. Therefore, when repeating the calculation, only one result is obtained for each sampled quantity.

Based on Figure 10, we see that as p increases, the quality of the results tends to decrease. As the sample size of the error estimate increases, so does the quality of the result.

The way in which the sample is selected affects the error estimation quality. When using the USG sampling method, the quality of the results is the best.

Based on the results in Figure 10, we see that when calculating with a sample size of 1024, T≈1 (or the calculation result is close to the correct result).

To observe the calculation time, the author gives the total calculation time (including the calculation time by the method with the PC method and the calculation time for error estimation) with the sampling method as USG.

Table 3. *Calculation time (seconds)*

N \ p	2	3	4	5
9	3.6	3.8	4.2	4.4
100	56.9	57.4	58.8	59.2
1024	390.1	393.5	397.1	401.4
10000	3026.6	3034.7	3041.3	3061.5

Based on Table 3, we see that the computation time is proportional to p and the number of samples. When calculating with parameters: p=2, N=1024, the calculation result will be close to the correct result. The total calculation time is 390.1 seconds (very small compared to the calculation time of the verification method 28454 seconds).

4. CONCLUSION

The author has introduced a tool to help estimate the error of the PC method, applied to nonlinear problems with many uncertain parameters. With the obtained results, it is found that using this method will greatly reduce the computational cost but still ensure the quality of the results. When using this method, it is recommended to calculate with the following parameters: when calculating with the PC method with small p (namely p=2), the sampling method of the error estimate is USG.

REFERENCES

[1] Dao Duc Thu, Pham Van Trong, Tran Quang Thanh (2019), *Research on the Polynomial Chaos method applied to car suspension systems*, Scientific Research Journal of Sao Do University, No. 2.

[2] Wiener N. (1938), *The Homogeneous Chaos*, American Journal of Mathematics. American Journal of Mathematics, Vol. 60, No. 4. 60 (4): 897–936. doi:10.2307/2371268.

[3] H. Niederreiter (1992), *Random Number Generation and Quasi-Monte Carlo Methods*, CBMS-NSF Regional Conference Series in Applied Mathematics, Society for Industrial and Applied Mathematics

[4] Dongbin Xiu and George Em Karniadakis (2002), *The wiener–askey polynomial chaos for stochastic differential equations*, SIAM Journal on Scientifific Computing, 24(2):619–644, 2002.

[5] Géraud Blatman and Bruno Sudret (2009), *Anisotropic parcimonious polynomial chaos expansions based on the sparsity-of-effects principle*. In Proc ICOSSAR'09, International Conference in Structural Safety and Relability.

[6] Ling Feng, Ma Ze-Yu,Tang Zheng-Fei, Chen Yong-Fu (2013), *Uncertainty Analysis of Vehicle Suspension Systems Based on Polynomial Chaos Methods*.

[7] M. D. McKay, R. J. Beckman, W. J. Conover (1979), *A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code*, Technometrics 21 (2) 239–245. doi:10.2307/1268522.

[8] Gaurav Kewlania, Justin Crawfordb and Karl Iagnemmaa (2012), *A polynomial chaos approach to the analysis of vehicle dynamics under uncertainty*.

[9] Dao Duc Thu, Luong Quy Hiep, Pham Van Trong (2021), *Research on the influence of sampling method on the quality of polynomial chaos method applied to automobile suspension systems*, Research Journal Science Red Star University, No. 1.