ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1995

Seria: MECHANIKA z. 121 Nr kol. 1266

Anna KUCABA-PIĘTAL, Ludomir LAUDAŃSKI
Politechnika Rzeszowska, Poland

# MODELLING STATIONARY GAUSSIAN LOADS

Summary. The paper presents information about two methods of digital simulation of samples of the stationary Gaussian stochastic processes possessing multi modal spectra. They have been developed in order to imitate dynamic loads arising on an airplane undergoing gusty flying conditions. Therefore the particular spectra typical for an airplane gust study were involved reflecting also elastic properties of the flying vehicle. In an essential part the presented details are devoted to the problem of solving the system of algebraic non linear equations describing desired linear filter. At this stage the presented results can be also applied in studying earthquakes, modelling gusty winds for civil engineering and other purposes.

## MODELOWANIE STACJONARNYCH OBCIĄŻEŃ GAUSSOWSKICH

Streszczenie. Praca zawiera informacje nt. dwóch metod cyfrowej symulacji realizacji stacjonarnych gaussowskich funkcji losowych posiadających wielomodalne spektra. Metody te opracowano pod kątem symulowania dynamicznych obciążeń powstających na samolocie lecącym w burzliwej atmosferze stosując spektra typowe dla tego zagadnienia uwzględniające odkształcalność samolotu. Prezentowane tu wyniki dotyczą problemu rozwiązywania pewnego układu równań algebraicznych nieliniowych drugiego stopnia opisujących liniowy filtr. Z tego punktu widzenia omawiane algorytmy numeryczne mogą być stosowane do opisu np. kinematyki trzęsień ziemi, modelowania podmuchów wiatru dla celów inzynierii lądowej czy innych matematycznie podobnych zagadnień.

## МОДЕЛИРОБАНИЕ ГАУССОВСКИХ СТАЦИОНАРНЫХ НАГРУЗОК

Резюме. Работа касается двух методов цифровой симуляции случайных гауссовских стационарных функций имеющих много модальные спектры. Разработано методы намеренные на имитацию динамических нагрузок возникающих во время болтанки самолета с учетом его упругости - отраженных в виде рассмотренных спектров. Суть работы состоит в решении некоторой системы нелинейных алгебраических уравнений второго порядка с интересными свойствами. Эти решения оформляют фильтр. Результаты работы употребимы при моделировании землетрясений, ветровых нагрузок в строительстве и тем подобных.

## 1. INTRODUCTION - MULTI MODAL PROCESSES

A typical spectrum for considered airplanes/gliders looks like the one presented in Fig.1 showing the response calculated at the centre of gravity for the PZL M-18 (a Polish agricultural airplane) which flies at a speed of 100 km/h horizontally in vertical gusts characterised by the scale of turbulence L=50 m (which means rather severe conditions). We see for the particular airframe typical peaks associated with the sensitive frequencies reflecting its resonances at the particular point chosen on it. We call such spectra - ***multi modal*** spectra, and associated with them processes - multi modal processes - accordingly. For the Gaussian stochastic processes possessing such spectral densities we developed special technique of simulation the appropriate sample functions. It fills a gap in the wide class of publications related to this subject (for references see [1]). Below we present the essence of two numerical methods aiming at such a purpose. They both have the common origin and we begin with showing this origin by stating the numerical problem which has to be solved. Then, we show two algorithms solving this problem. In the end we give a summary of the numerical results.

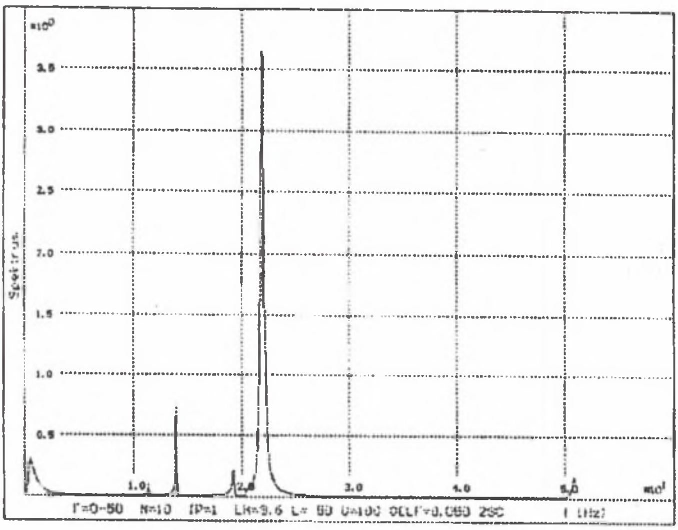


Fig.1. Spectral Density of the Gust Response of the PZL M-18 Agricultural A/P at Its C.G.

## 2. STATING THE PROBLEM

We begin with the definition of the correlation function which is related to the properties of the linear systems while examining their input-output relations. We define the correlation function for the output stochastic process *{y(t)}* which left linear devices called filter being fed by the input stochastic process *{x(t)}*:

$$K(\tau)=\mathscr{E}\left[\sum_{k=0}^{N} h(k)x(t-k) \sum_{n=0}^{N} h(n)x(t+\tau-n)\right] \tag{1}$$

There is now an interesting innovation: we shall look at the output process with the intention to recover the impulse characteristic of the filter $h(t)$ by assuming that the input was a White Noise process. Therefore in this particular case (1) is immediately simplified to:

$$K(\tau) = \sum_{n=0}^{N} h(n)h(k) \qquad with \quad k = 0,1,\ldots,N \tag{2}$$

But for practical purposes (2) can be rewritten in a more useful form as:

$$K(\tau) = \sum_{n=0}^{N-\tau} h(n)h(n+\tau) \qquad with \quad \tau = 0,1,\ldots,N \tag{3}$$

To see what exactly the problem we shall face in this way it is recommended to expand (3) into its explicit form as below:

$$\begin{gathered}
K(0)=h(0)h(0)+h(1)h(1)+h(2)h(2)+\ldots+h(N)h(N) \\
K(1)=h(0)h(1)+h(1)h(2)+h(2)h(3)+\ldots+h(N-1)h(N) \\
K(2)=h(0)h(2)+h(1)h(3)+h(2)h(4)+\ldots+h(N-2)h(N) \\
\vdots \\
K(N-2)=h(0)h(N-2)+h(1)h(N-1)+h(2)h(N-2) \\
K(N-1)=h(0)h(N-1)+h(1)h(N) \\
K(N)=h(0)h(N)
\end{gathered} \tag{4}$$

Equations (4) represent a system of $N$ algebraic equations order two each. The equations are linearly independent i.e. their Jacobian is non zero. Therefore in general the system has $2N$ solutions, say $\alpha_1, \alpha_2, \ldots, \alpha_{2N}$ which can be both real and complex. Each solution of (4) preserving the form $h'(1), h'(2), \ldots, h'(N)$ composes the filter although it is not clear at the beginning whether each solution of (4) possess physical meaning i.e. can be considered as the physically realised filter. There are some basic properties of the solutions which have to be mentioned. If $h'(1), h'(2), \ldots, h'(N)$ is the solution of the system (4) then also $h'(N), h'(N-1), \ldots, h'(1)$ becomes the solution. Also solutions of the form $-h'(1), -h'(2), \ldots, -h'(N)$ and $h'(N), -h'(N-1), \ldots, -h'(1)$ satisfy equations (4). If we call the solution $h'(1), h'(2), \ldots, h'(N)$ the *basic solution* we can state that in general there will always be $N/2$ *basic solutions* either real or complex. Each system (4) has its *normalised* correspondent obtained by dividing (4) side-by-side with value $K(1)$. It is easy to prove that normalised solutions can be obtained dividing ordinary solutions by $\sqrt{K(1)}$ .

## 3. SPECIAL CASE

Let us consider special case of (4) i.e. situation when $N=2$ . For this case one can easily find the analytical solution. Some details of the procedure are shown below. The system of equations (4) has now the form:

$$K(1) = h(1)h(1) + h(2)h(2) \tag{5}$$

$$K(2) = h(1)h(2)$$

Let us denote:

$$D_1 = \sqrt{K(1)+2K(2)} \qquad and \qquad D_2 = \sqrt{K(1)-2K(2)} \tag{6}$$

With this notation we shall get analytical solutions of (5) in the form:

$$h(1)=\frac{1}{2}(D_1 + D_2) \tag{7}$$

$$h(2) = \frac{1}{2}(D_1 - D_2)$$

It is clear now that once:

$$K(1) < 2K(2) \tag{8}$$

value $D_2$ becomes imaginary, therefore the corresponding solution of (5) becomes complex. Such a solution evidently has no physical meaning.

It is also easy to see that with notation given by (7) the system (5) has *four* solutions:

$$[h(1),h(2)],\quad [h(2),h(1)],\ [-h(1),-h(2)],\ [-h(2),-h(1)] \tag{9}$$

Let us acknowledge by the way that the condition (8) assessing existence of the real solutions of (5) can be physically interpreted as a case with sufficiently rapidly decreasing correlation function $K(\tau)$ with respect to increasing value of its argument.

## 4. SPECIAL NUMERICAL EXAMPLES

Testing and debugging stages require reliable and suitable numerical examples. In this case a special value have examples offering solutions expressed by integers. We enclose below two such examples.

For the case $N=2$ we use:

$$K(1) = 34 \quad \textit{and} \quad K(2) = 15 \tag{10}$$

having the basic solutions as below:

$$h(1) = 5,\ \ h(2) = 3 \tag{11}$$

For the case $N = 4$ we use:

$$K(1) = 30,\ \ K(2) = 20,\ \ K(3) = 11,\ \ K(4) = 4 \tag{12}$$

which possesses a single exact basic solution of the form:

$$h(1) = 1,\ \ h(2) = 2,\ \ h(3) = 3,\ \ h(4) = 4 \tag{13}$$

and another basic solution although this time - approximate - which we give here in the form rounded to two decimal digits after coma:

$$h(1) \approx 1.65,\quad h(2) \approx 1.58,\quad h(3) \approx 4.35,\quad h(4) \approx 2.42 \tag{14}$$

## 5. ALGORITHM MESORS

In the beginning the algebraic problem stated by (4) is reformulated and transformed into a geometrical one. To do that we define the norm:

$$\|K(\tau)-\tilde{K}(\tau)\| = \sum_{\tau=0}^{N}\left|K(\tau) - \sum_{n=0}^{N-\tau} h(n)h(n+\tau)\right|^2 \tag{15}$$

with the different notations for the source process $\{Y(t)\}$ and the simulated one - called the target process $\{\tilde{Y}(t)\}$. Now some formal requirement concerning the accuracy can be specified by demanding :

$$\|K(\tau) - \tilde{K}(\tau)\| \leq \varepsilon \tag{16}$$

Therefore $N=1$ values $h(n)$ can be understood as the arguments of $N+1$ - dimensional function defined by (15) . With the condition (16) we seek for the minima of this multi dimensional geometrical object. It was done numerically by resorting to the Svejgaard algorithm, the main idea of which is based on a gradient searching approach and was developed following the *Algol* implementation given in [2].

## 6. ALGORITHM ANNA

The algorithm solving the system (4) that we are going to present now does it in a direct way. The essence in solving directly system (4) lies in finding such a solution $\alpha^n \in \Re^n$ for which mapping $F \in \Re^n \times \Re^n$ becomes zero. Non-linear character of mapping $F$ suggests the application of an iterative method based upon an algorithm:

$$x^{k+1} = x^k - [J^{-1}(x^k)]F(x^k) \tag{17}$$

where: $J$ is the Jacobian of the mapping matrix $F$
$x^k$ the $k_{th}$ approximation of the vector $x$.

Practical implementation of the algorithm (17) which we call *Anna* is based on the following theorem (see: Fortuna et all. [3]):

Theorem.

Let the function $F(x)$ becomes differentiable according to Frechet within the neighbourhood $K(\alpha,\rho)$ of such a point $\alpha$ that $K(\alpha)=0$. Moreover the first derivative of $F(x)$ is a continuous function at the same point $\alpha$ and non singular. With these circumstances the point $\alpha$ becomes an attractive point of the Newton iterative method:

$$x^{k+1} = x^k - [J^{-1}(x^k)]F(x^k)$$

Strong assumptions imposed upon the function $F(x)$ - continuity, differentiability and non singularity of its derivative at the point $\alpha$ guarantee the local convergence of the method. The system (4) satisfies all the requirements mentioned .

Methods based on the above *Theorem* having the Jacobian given numerically are called quasi-Newtonian methods. One particular example of such a method offers a hybrid method of Powell (see for instance: [4] and [5]). Here the Jacobian is calculated by the method of finite differences. This method in order to improve the convergence supposes moreover:

$$\|F(x^{k+1})\|_2 > \|F(x^k)\|_2 \tag{18}$$

$$x^{k+1} = x^k + p^k \tag{19}$$

To find $p^k$ which is necessary to determine the next iteration $x^{k+1}$ must be solved the system of equations given below:

$$J(x^k)p^k = -F(x^k) \tag{20}$$

The idea of the algorithm *Anna* implementing iterations with respect to $k$ can be briefly explained as below:

| | |
|---|---|
| 1-st step: | If $F(x^k) \rightarrow STOP$ |
| 2-nd step: | Calculate $p^k$ by solving (19) |
| 3-rd step: | If $\|F(x^{k+1}+p^k)\|_2 < \|F(x^k)\|_2$ the step is accepted resulting in: $x^{k+1} = x^k + p^k$ then $k \Leftarrow k+1$ and return to step 1 otherwise: |
| 4-th step: | Calculations are interrupted. |

## 7. NUMERICAL RESULTS, CONCLUSIONS

The algorithm *Mesors* was implemented into a program written first in Fortran IV and widely used in numerous calculations by resorting to the electronic computers of the generations IBM 360, PDP 11/70 and CDC 6600. They imposed hard restrictions upon the volume of the problem which related to the speed of computations did not reach at that time above N=32. Arrival and fast development of Personal Computers gave rise to new implementations by using Fortran 77 and by using special compilers like the NDP Fortran. Both implementations: the earlier from the mid 70-ties and the actual one were successful in the sense that they lead to right solutions - what became completely evident lastly by testing this algorithm with the examples described above in paragraphs 3 and 4 developed only now. To complete successful computations for *N=64* by using *IBM PC 486-50* requires about 3-5 minutes. Increase in time goes approximately in such a way that doubling the volume N leads to about computations ten times longer. It is worth mentioning that the nature of the calculations within the airplane gust response studies needs even values *N=512* or may be some times as big as *N=1024*. Completing solution to the system (4) by using the Svejgaard algorithm became a difficult problem of reaching the desired accuracy for the results derived in this way. Especially it concerns those distanced (latest) components of the approximate numerical solution which are usually about hundreds of times smaller than the greatest initial values. We guess that the accuracy in their estimation may become dramatically low. The question is: whether they *have to* be derived *exactly*? Or in other words - which level of errors in their estimations can be accepted, and which not? And for these questions we do not have a satisfying answer untill now.

The algorithm *Anna* was developed at the end of 1993 and our numerical experience with its application to the particular aeronautical purposes is just at its beginning. There is no convincing way to compare both algorithms quite literally. Seemingly the direct solution of *Anna* goes about ten times faster than geometrical approach which follows *Mesors*. In both cases crucial for the successful solution is the choice of the initial point. It is so far only possible to say that *Mesors* almost always produces a solution of (4) disregarding the particular choice of the initial point. There are some doubts - as we said before - about the accuracy of these solutions. Nevertheless through out long-lasting numerical practice there were only few cases of divergent behaviour observed. On the other hand - *Anna* solves (4) only when the choice of the initial point becomes extremely luckily done, so - there were only few cases that the solution was obtained. Moreover the solutions produced by *Anna* go significantly faster and their accuracy is high. Therefore both algorithms remain as a potential field of the further future considerations. Some numerical results will also be shown during the oral presentation in case of our presence at the *Sympozjon*.

## ACKNOWLEDGMENT

The results presented in this paper derived between 1992-93 were financed by the KBN grant No.770039203.

LITERATURE

[1] L.M.Laudanski: An Introduction to Stochastic Dynamics of Gliders (book published in Polish language twice). Rzeszów Technical University Press, Rzeszów 1992 (1978).

[2] W.M.Turski: Usage of Digital Computers in Scientific Engineering Centres (book published in Polish language). PWN, Warsaw 1973.

[3] Z.Fortuna, B.Macugow, J.Wysowski: Numerical Methods (book published in Polish language), WNT, Warsaw 1982.

[4] M.J.D.Powell: A Hybrid Method for Nonlinear Equations. In book Methods for Nonlinear Algebraic Equations, Editor - P.Rabinowitz, Gordon and Break 1970.

[5] D.Kahaner, C.Moler, S.Nash: Numerical Methods and Software. Prentice-Hall 1988.

Recenzent: prof dr hab. inż. W. Gutkowski

Wpłynęło do Redakcji w grudniu 1994 r.