AIAA JOURNAL
Vol. 41, No. 9, September 2003

Two-Dimensional "Poor Man's Navier–Stokes Equation" Model of Turbulent Flows

Tianliang Yang,* J. M. McDonough,† and J. D. Jacob‡
University of Kentucky, Lexington, Kentucky 40506-0503

As part of a continuing effort to construct "synthetic velocity" subgrid-scale (SGS) models for large-eddy simulation (LES) techniques utilizing Kolmogorov scaling and discrete dynamic systems (chaotic maps), focus is centered on constructing the chaotic maps from experimental data to demonstrate consistency of the modeling approach with basic physics. Although such efforts have been made previously, all of them used a linear combination of logistic maps to fit one-dimensional experimental data. A two-dimensional chaotic map derived directly from the Navier–Stokes equations is employed, and the bifurcation parameters of this map are determined to best fit two-dimensional experimental velocity data. A genetic algorithm is used as the optimization tool to obtain the required least-squares fit of the time series for a point in a two-dimensional flow behind a turbulator. Results compare reasonably well with the experimental time series, implying that two-dimensional chaotic maps provide viable candidates for producing temporal fluctuations in synthetic-velocity SGS models for LES and additionally as part of real-time control mechanisms.

Nomenclature

A_f	=	real part of the Fourier transform of filtered velocity time series
A_u	=	amplitude factor of subgrid-scale model
a_f	=	real part of Fourier transform of complete velocity time series
B_f	=	imaginary part of Fourier transform of filtered velocity time series
b	=	size of the turbulator, m
b_f	=	imaginary part of Fourier transform of complete velocity time series
D	=	experimental channel width, m
F	=	switching function
f	=	frequency, Hz
H	=	experimental channel height, m
M	=	chaotic time series
m	=	poor man's Navier–Stokes equation
Q	=	least-squares functional
Re	=	Reynolds number based on channel height H
S	=	chaotic function
U, V	=	complete velocity components, m/s
u^*, v^*	=	velocity components of subgrid-scale model, m/s
α	=	amplitude of modeled chaotic time series
β	=	bifurcation parameter related to Reynolds number
γ	=	bifurcation parameter associated with velocity gradient
δp	=	difference of qualification properties between modeled map and data
ζ	=	anisotropy correction of subgrid-scale model
θ	=	"implicitness" factor of chaotic time series
ω	=	frequency of evaluation of chaotic time series

Subscripts

c	=	cutoff
f	=	frequency
i	=	velocity components
j	=	qualification property index in least squares function
l	=	number of terms in map representation
u	=	velocity

Superscript

(n)	=	time (iteration) level in map representation

Introduction

IT is widely believed that large-eddy simulation (LES) in some form will be the method of choice for industrial-level fluid flow calculations in the near future. However, LES is known to exhibit deficiencies because of inaccuracies of its filter-induced subgrid-scale (SGS) models. Recently, Hylin and McDonough[1] introduced a new approach to constructing "synthetic velocity" subgrid-scale (SGS) models for LES based on Kolmogorov scaling and discrete dynamic systems (chaotic maps). In general, synthetic velocities directly model SGS physical variables, for example, velocity components, rather than SGS stress, permitting computation of the stresses from the modeled velocities in some versions, for example, Domaradzki and Saiki[2] and Scotti and Meneveau,[3] or simply using them in the original equations of motion, as done by McDonough et al.[4] and Hylin and McDonough.[1,5]

It is obvious that modeling physical variables should provide a closer connection to actual flow physics than does the usual modeling of SGS stresses. Clearly, the mapping from flow physics to statistics is many-to-one. For example, flows as different as being in completely opposite directions may possess the same second-order statistics (but, obviously, not the same first-order ones). In typical filtered and/or averaged approaches we attempt to recover (calculate) a portion of the flow physics by means of what is formally inversion of the noninvertible physics-to-statistics map; that is, we attempt to predict mean flow behaviors based on the second-order statistics, or, worse yet, models thereof. Use of synthetic velocities avoids this difficulty by providing a specific realization of the SGS velocity field, either to be used directly, as in Refs. 1, 4, and 5, or to construct values of second-order correlations directly from their definitions as in Refs. 2 and 3. At the same time use of synthetic velocities leads to a modeling formalism able to utilize directly experimental data during model construction, rather than simply in the context of post hoc validation.

Received 9 October 2001; presented as Paper 2002-2962 at the AIAA 32nd Fluid Dynamics Conference, St. Louis, MO, 24–27 June 2002; revision received 31 March 2003; accepted for publication 14 April 2003.

*Postdoctoral Scholar, Department of Mechanical Engineering. Member AIAA.

†Professor, Department of Mechanical Engineering. Senior Member AIAA.

‡Associate Professor, Department of Mechanical Engineering. Senior Member AIAA.

The Hylin and McDonough formalism[1] employs SGS models of the form

$$u_i^* = A_i \zeta_i M_i, \qquad i = 1, 2 \tag{1}$$

for two-dimensional flows as will be considered herein; u_i^* is a SGS velocity component; A_i is an amplitude factor; ζ_i is an anisotropy correction, and M_i is a temporal fluctuation obtained from a chaotic time series. Each of the three factors on the right-hand side of Eq. (1) varies across the spatial grid and from one resolved-scale time step to the next.

Expressions for the first two factors have been derived from first principles employing the Kolmogorov (K41) theory of homogeneous turbulence (for a good overview see Ref. 6) and invoking scale similarity, as done in the dynamic SGS models of Germano et al.,[7] as given in Ref. 1 and by Sagaut.[8] The third factor received little specific attention in early investigations. In Ref. 1, this factor was associated with Kolmogorov's "stochastic variable," and the logistic map, "absolute value" logistic map, and tent map were employed as realizations. All appear to give similar results when the same map is used for all velocity components.

Recent work of McDonough et al.[9] demonstrated that one-dimensional chaotic dynamic systems could be curve fit in the sense of reproducing the structural "appearance" of the original data and an extensive set of statistical quantities associated with these data, but without attempting a point-by-point exact fit. Such an approach can be applied to any data set represented by a time series, and the outcome of the curve-fitting process is a model that behaves very much like the original data and exhibiting with the same intrinsic variability (sensitivity to initial conditions) that occurs in physical turbulence. (We remark that exact fits of chaotic time series have often been constructed, for example, see Casdagli and Eubank,[10] but would seem to be inappropriate in the context of turbulence models because the turbulent physical flow can never be exactly reproduced.)

Early work by Mukerji et al.[11] and more recent studies by McDonough and Yang[12] and Roclawski et al.[13] have suggested that the third factor in Eq. (1) can be represented with the outcome of a curve-fitting process using experimental turbulence time series. However, these previous efforts focused on using a linear combination of logistic maps to fit one-dimensional experimental data. In this regard, the contribution of the present study is to provide results obtained by fitting a two-dimensional chaotic map (in fact, one derived from the two-dimensional equations of motion) to a two-dimensional experimental velocity field. From both theoretical and practical (computational) standpoints, this is a nontrivial extension; namely, the two-dimensional maps contain many more bifurcation parameters than does the one-dimensional map, thus rendering pure mathematical analysis impossible due to high codimension of the chaotic dynamical system[14] and, at the same time, significantly increasing the computational burden due to the large number of parameters to be determined during the fitting process.

In part due to failures of Eq. (1) to correctly model passive scalars with the corresponding M_i constructed from logistic maps, McDonough and Huang[15,16] derived what we term the "poor man's Navier–Stokes (PMNS) equation" directly from the Navier–Stokes (NS) equations, and showed that this discrete dynamical system (DDS) exhibits the full range of NS temporal behaviors. This obviously motivates the present study of constructing chaotic-map models of two-dimensionl experimental velocity components based on the PMNS equation. In particular, if this can be done, it would be reasonable to employ the PMNS equations as the factors M_i in Eq. (1). Indeed, we will demonstrate that such models show significant potential not only as a factor in SGS models for LES, but also as a portion of a synthetic velocity formalism for Reynolds averaged Navier–Stokes (RANS) approaches, as recently considered by Burton et al.[17] Moreover, as will be clear, evaluation of the PMNS equations is very efficiently performed, making them potentially useful for real-time control applications.

The mentioned work associated with fitting experimental data to chaotic time series[11–13] were very computationally intensive. For example, the study reported in Ref. 11 required several weeks of CPU time with a parallelized code executing on eight processors, and hence, many months of actual CPU time: for a single case, largely because a direct search was employed as the optimization technique due to existence of numerous local minima of the least-squares objective function. It will be evident as we proceed that in order to construct useful SGS models it is advantageous to determine the mapping

$$\text{physical parameters} \longrightarrow \text{PMNS equation bifurcation parameters}$$

where for simple incompressible flows the physical parameters will include the Reynolds number Re and elements of the velocity gradient tensor. It is the purpose of a curve-fitting exercise to ascertain this mapping. Obviously, many data sets representing a wide range of flow conditions, and thus many runs with the curve-fitting procedure, will be needed. To accomplish this in a reasonable amount of time, no worse than "overnight turnaround" will be required. We are currently able to achieve this with parallelized genetic algorithms (GAs).

GAs, for example, see Goldberg and Deb[18] and Davis,[19] are based on an evolution of random tries by "individuals," rather than on a deterministic sequence of steps as in usual optimization algorithms. They comprise a computer implementation of Darwin's theories. Because the whole process is built on randomness but the final effect is not random, it is able to seek the global optimal solution regardless of how many peaks or valleys the objective function possesses. It will be shown, from the nature of the objective function, that the curve-fitting process of the chaotic dynamic system benefits significantly from using the GA optimization method.

The remainder of this paper is organized as follows. We begin in the next section with a brief description of derivation of the PMNS equations and of the nature of the curve-fitting problem that results. This is followed with a section that presents the GAs used in the present study. Finally, we then present results of this research, and end with some concluding remarks.

Nature of Curve-Fitting Problem

In this section we provide an outline of the derivation of the PMNS equation and follow this with details of the nature of the resulting objective function needed for the least-squares curve-fitting procedure. The latter will serve as motivation for use of GAs in this context.

Derivation of the PMNS Equation

The name "poor man's Navier–Stokes equation" is used in deference to remarks by Frisch[6] regarding the relationship of simple quadratic maps to the NS equations. Here (for the sake of completeness), we briefly present what was first demonstrated in Refs. 15 and 16. We begin with the two-dimensional NS equations following Leray projection (for example, see Foias et al.[20]):

$$u_t + (u^2)_x + (uv)_y = (1/Re)\Delta u \tag{2a}$$

$$v_t + (uv)_x + (v^2)_y = (1/Re)\Delta v \tag{2b}$$

where u and v are dimensionless velocity components, Re is the Reynolds number, and Δ is the two-dimensional Laplacian. We assume u and v can be expanded in Fourier series, which can be proven rigorously in two dimensions,[20] and apply the Galerkin procedure to Eqs. (2). We than decimate the result to a single mode per dependent variable, numerically integrate in time, and apply transformations given by May[21] and in Refs. 15 and 16 to obtain

$$a^{(n+1)} = \beta_1 a^{(n)}\left(1 - a^{(n)}\right) - \gamma_{12} a^{(n)} b^{(n)} \tag{3a}$$

$$b^{(n+1)} = \beta_2 b^{(n)}\left(1 - b^{(n)}\right) - \gamma_{21} a^{(n)} b^{(n)} \tag{3b}$$

where a and b are the coefficients for a single (and arbitrary) mode of the Fourier representation of u and v, respectively.

We also observe that β_i, $i = 1, 2$, can be expressed in terms of the Taylor microscale Reynolds number and the arbitrary wave number, which can be directly inferred from the Taylor microscale length scale λ. The SGS numerical time step size also enters this definition, and this can be determined from the SGS energy dissipation rate. It follows that the β_i can be directly computed from LES resolved-scale results (or at least extrapolates thereof) at each grid point and

time step of a simulation. The same holds for γ_{ij}, $i, j = 1, 2$, which are related to the SGS timescale, wave number, and (possibly extrapolated) resolved-scale velocity gradients $\partial u/\partial y$ and $\partial v/\partial x$. Indeed, this entire procedure has by now been successfully implemented in a three-dimensional computational fluid dynamics code, so that the question then is, why attempt correlations to experimental data as being presented here? The answer to this is that the derivation outlined leads to $\mathcal{O}(1)$ scaling constants that may need to be set via experimental data. This is unnecessary for isotropic turbulence but investigations such as the present one are needed to determine to what extent these constants change with class of flow.

Objective Function for Least-Squares Fit

The form of the chaotic map M_i in Eq. (1) that will be used for fitting data in the present work is a two-dimensional extension of the form given in Ref. 9:

$$M_i^{(n+1)} = (1-\theta)M_i^{(n)} + \theta \sum_{l=1}^{k} \alpha_{l,i} S_{l,i}^{(n+1)}[\omega_l, m_{l,i}(\beta_l, \gamma_{l,i})] \qquad i = 1, 2, \quad (4)$$

with the indices associated with the two velocity components u and v, or more specifically with a and b, respectively, from Eqs. (3); that is, $m_{l,1}$ is an instance of a calculated from Eq. (3a), and $m_{l,2}$ is similarly associated with b. The $\alpha_{l,i}$ are amplitudes, and θ is an "implicitness" factor. The function $S_{l,i}^{(n+1)}[\omega_l, d_l, m_{l,i}(\beta_l, \gamma_{l,i})]$ is formulated in the following way. We begin by defining the function

$$S_{l,i}^{(n+1)} = \begin{cases} m_{l,i}^{(n)} & \text{if} \quad F_l^{(n+1)} = 0 \\ m_{l,i}^{(n+1)} & \text{otherwise} \end{cases} \quad (5)$$

where the $F_l^{(n+1)}$ are switching functions based on the frequency of evaluation ω_l of map l. They are defined in Ref. 9 as

$$F_l^{(n+1)} \equiv \begin{cases} 1 & \text{if} \quad [(n+1) \bmod \omega_l] = 0 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

As was emphasized in Ref. 9, the goal of constructing the arbitrary function (4) is not to find a time series that exactly coincides with the data but rather one possessing properties to guarantee that the "appearance" of the constructed time series is qualitatively close to the data and that chosen statistical quantities are quantitatively close.

There are many properties by means of which chaotic data may be characterized; in the present study, we use 25 different statistical and mathematical properties to fit the data. For a detailed discussion of these and the methods by which they are computed, see Ref. 9. The method of constructing the modeled time series (4) requires that the minimum value be found for the following least-squares functional corresponding to $N_{p,i}$ properties for each velocity component:

$$Q(\alpha_{l,i}, \beta_l, \gamma_{l,i}, \omega_l, \theta, k) = \sum_{i}^{2} \sum_{j}^{N_{p,i}} (\delta p_{j,i})^2 \quad (7)$$

where $\delta p_{j,i}$ is the difference between the jth property of the ith component of the model and the data. In Eq. (7), a two-dimensional generalization of the objective function given in Ref. 9, $\alpha_{l,i}$, $\beta_{l,i}$, $\gamma_{l,i}$, ω_l, and θ are unknown parameters to be determined by minimizing Q. Note that the proposed objective function is not differentiable with respect to all its unknown parameters, in particular, $\beta_{l,i}$ and $\gamma_{l,i}$, and contains both real ($\alpha_{l,i}$, $\beta_{l,i}$, $\gamma_{l,i}$, and θ) and integer (ω_l and k) unknowns, resulting in difficulties for most standard optimization strategies as can be inferred from a typical plot of this function shown in Fig. 1. (We remark that we usually set values of the parameter k external to the optimization process.) Figure 1 explicitly demonstrates the nondifferentiability with respect to the β and the existence of closely spaced local maxima and minima.

In the present study, the GA optimization method is used to seek values of the unknown parameters by minimizing Eq. (7). Because this is apparently the first application of the GA for optimizing an objective function to fit chaotic maps, it is worthwhile to present this technique in a relatively detailed way. This is done in the next section.

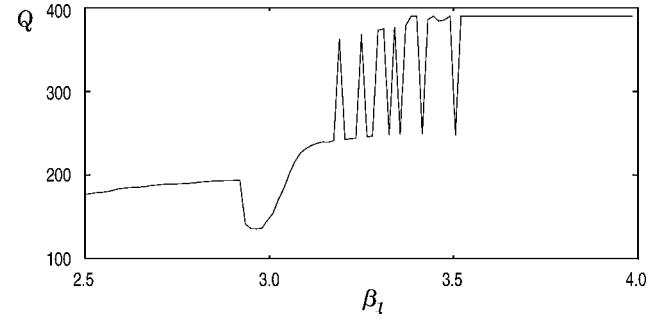

Fig. 1 Variation of objective function Q as one of the bifurcation parameters, β_1 changes from 2.5 to 4.0; three instances of PMNS equations are used, with all other parameters kept constant.

Genetic Algorithms

As inspired by Darwin's theory of evolution, GAs are a part of evolutionary computing. They are started with a set of solutions called a *population*. Normally, each solution in the population is encoded as a set of bits of a binary number, symbols, or decimal numbers. Each of these bits, symbols, or decimal numbers stands in a certain position, and the whole of them form one chromosome. The present study uses a binary encoding method. It converts the searching parameters into a specified number of binary bits. Corresponding to the encoding method, the decoding process in the GA converts chromosomes into solutions of the optimization method. Therefore, the decoding method must conform to the encoding method used in each specific GA.

After the decoding/encoding procedure is processed, the fitness of a chromosome is evaluated. This is accomplished by calculating objective function values corresponding to each solution. The fitness of a chromosome is associated with the value of the objective function. For example, for minimum-value searching problems, a smaller objective function implies better fitness of the corresponding chromosome.

The next step of the GA is selection. The purpose of the selection step is to choose the pair of parents used to produce a child. This is motivated by an expectation that the new population will be better than the old one. Solutions that are selected to form new solutions (offspring) are selected according to their fitness. In principle, the more suitable the solutions are, the more chances they have to be selected. Because selection is the most important step of a GA, to mimic Darwin's theory of evolution, the selection method of a GA has to retain the random characteristics of natural selection. In the present study, we use the tournament selection method together with the niching technique[22,23] to select parents for performing the crossover. In tournament selection, a set of chromosomes is randomly chosen, and the best (with respect of fitness) chromosome among this set is chosen to be the selected parent.[22,23]

The niching technique (or fitness sharing) aims at promoting the formulation and maintenance of stable subpopulations (niches). It is based on the idea that individual chromosomes in the particular niche have to share the available resources. The more individual chromosomes are located in the neighborhood of a certain individual, the more its fitness is degraded.[22,23] Therefore, use of the niching technique is, in essence, to correct the value of objective function (fitness of the chromosome) according to the distance of the solution (chromosome) to other solutions (chromosomes) in the population. The farther the distance is, the larger/smaller the value of the objective function corresponding to better fitness of the chromosome.[22,23]

The crossover combines the genetic material in two parent chromosomes selected in the previous step to construct the child. Normally, three crossover methods are used by GA practitioners[22,23]: one-point crossover, two-point crossover, and uniform crossover. In the present study, both one-point and uniform crossover methods are used in the optimization process. In the one-point crossover, one crossover point is selected randomly, the binary string from the

beginning of the chromosome to the crossover point is copied from one parent, and the rest is copied from the second parent. In the uniform crossover, bits are randomly copied from the first and/or from the second parent.

After a crossover is performed, mutation takes place. Mutation is used to prevent "falling" of all solutions in the population into a local optimum of the objective function. Mutation randomly changes the new child. For binary encoding, we need switch only a few randomly chosen bits from 1 to 0 or from 0 to 1 to accomplish this. Obviously, different mutation techniques should be employed for different encoding/decoding methods. For example, if the direct value encoding/decoding methods are used in the GA, then each chromosome is represented by a real number associated with an evaluation of the objective function. The mutation method used in this case could be adding or subtracting a small number to or from the chromosome.

Note that not all of the child chromosomes are made by crossover and mutation. Some offspring are exact copies of their parents. This means that there are crossover and mutation probabilities to control how often the crossover and mutation will be performed. If crossover probability is 100%, then all child chromosomes are made by crossover. If it is 0%, a whole new generation is made from the exact copies of chromosomes from the old population (but this does not mean that the new generation is the same because mutation may occur in the next). Similar to the crossover probability, if mutation probability is 100%, the whole chromosome is changed; if it is 0%, nothing is changed. Normally, crossover probability should be relatively high. The best value of the crossover probability is about 60%. On the other hand, mutation probability should be very low. The best value of the mutation probability is about 0.5–1% (Refs. 22 and 23).

The next step of the GA procedure is natural. The replace step places new offspring in a new population for a further run of the algorithm. It is easy to imagine that the chance of losing the best chromosome is very high when we place the new offspring in the new population. Therefore, the idea of elitism is introduced. Elitism is the name given to the method that first copies the best chromosome (or a few best chromosomes) to the new population. Because it prevents losing the best-found solution, elitism can very rapidly increase performance of GAs. Therefore, we use this procedure in the present study.

The preceding description corresponds to only one generation of the evolving process for a GA. Obviously, many generations of evolution are needed to obtain the optimal value of the objective function. Therefore, the process should go back to the decoding/encoding step after the replace step is finished, until the optimum is found.

To apply the described GA to the curve-fitting procedure, we must first decide the number of realization of the PMNS equations [value of k in Eq. (4)] to employ. As was shown in Ref. 15, the PMNS equation represents only one single arbitrary wave vector of the NS motion. Therefore, it is not expected that a single instance of the PMNS equation would be sufficient in general. We initially use three terms to conduct the curve-fitting procedure. Detailed discussion of this will be given in the next section.

After the value of k in Eq. (4) is selected, searching ranges for the unknown parameters in Eq. (4) should be set. The ranges of β and γ are obtained from the regime map of the PMNS equation presented in Ref. 16:

$$2.5 \le \beta_l \le 4.0, \qquad l = 1, 3$$

$$-0.8 \le \gamma_{l,i} \le 0.65, \qquad i = 1, 2, \qquad l = 1, 3$$

The range employed for the β_l is such as to exclude "uninteresting" steady behavior and at the same time avoid divergence of the map interactions for most $\gamma_{l,i}$ in the given range. The ranges of α, θ, and ω are set from our previous experience with data-fitting procedures for chaotic time series[11–13]:

$$-30 \le \alpha_{l,1} \le 30, \qquad l = 1, 3$$

$$-10 \le \alpha_{l,2} \le 10, \qquad l = 1, 3$$

$$0 < \theta \le 0.2, \qquad 1 \le \omega_l \le 32, \qquad l = 1, 3$$

In the present study, each β_l is encoded to 15 binary bits; each $\gamma_{l,i}$ and $\alpha_{l,i}$ is encoded to 14 binary bits. We use one more binary bit to encode the β_l to emphasize their importance in the curve-fitting procedure. Because the searching range of θ is not very large, only nine binary bits are used to encode it. The ω_l are integer values, and they range between 1 and 32; therefore, 5 binary bits can encode all of the possible values of each of them. Thus, 237 binary bits are sufficient to encode one set of unknown parameters. To start the GA for the present curve-fitting procedure, 104 binary numbers of 237 bits each are randomly generated by computer. Random numbers are generated using Knuth's subtractive method (see Ref. 22) to generate 104 chromosomes. We use 104 as the population size instead of 100 to facilitate the parallelization of the GA. The parallelized GA was executed on eight processors. Each processor computes the fitness values for 13 chromosomes. Once it is initiated, the evolutionary procedures comprising execution the GA are conducted as follows:

1) Each chromosome (binary number) is decoded to the values of unknown parameters ($\alpha_{l,i}$, β_l, $\gamma_{l,i}$, ω_l, and θ, $i = 1, 2$, $l = 1, 3$), corresponding to one solution of Eq. (7), according to their searching ranges and number of binary bits employed to encode them (decode step).

2) Evaluate the value Q of Eq. (7) for each solution (fitness step). The fitness of each chromosome in the population is decided according to the value of Q. Smaller values of Q imply better fitness of the corresponding chromosome.

3) Select two parent chromosomes from the population according to their fitness (selection step). The basic principle for this step is the natural selection: the better the fitness, the greater the chance to be selected.

4) With a crossover probability, combine (crossover step) one part of the first parent with another part of a second parent to form a new offspring (child). If no crossover is performed, offspring is an exact copy of parents.

5) With a mutation probability mutate new offspring (mutation step). In the present study, this step is accomplished by randomly changing the binary bits of the chromosome according to the mutation probability.

6) Place new offspring in a new population and use the newly generated population for a further run of the algorithm (replace step). Note that the elitism technique that first copies the best chromosome to the new population is employed in this step.

7) If the end condition is satisfied, stop, and return the best solution in current population (test step).

8) Go to decode step (repeat step).

Results and Discussion

In this section, we will present results obtained from the curve-fitting process described earlier. The emphasis is on comparison of the results with the experimental time series used to generate the fit. The curve-fitting results are compared using three features, namely, the appearance of the time series, the power spectral density, and the delay map. In addition, we present the various statistical parameter values used to evaluate the objective function.

We simultaneously fit both velocity components obtained from laser Doppler anemometry (LDA) measurements for an experimental flowfield that is essentially two dimensional. The case of turbulator flows corresponding to $Re = 1 \times 10^5$ based on channel height is considered in the present study. The flow configuration is shown in Fig. 2. The test section has a channel height $H = 0.203$ m and width $D = 0.406$ m. A number of square ribs can be placed at different locations in the channel. In the present study, we present results for a single square rib with sides $b = 0.0245$ m. For more detailed descriptions of the flow configuration, see Ref. 13.

The time series employed for curve fitting in this study corresponds to two velocity components at $4.0b$ downstream of the last rib and $1.0b$ above the bottom surface in the center of the channel: one velocity component, U, in the streamwise direction and another, V, corresponding to the velocity component perpendicular to the bottom surface of the channel. The LDA sampling rate was 1 kHz, and 8192 samples were used in the analysis of each velocity

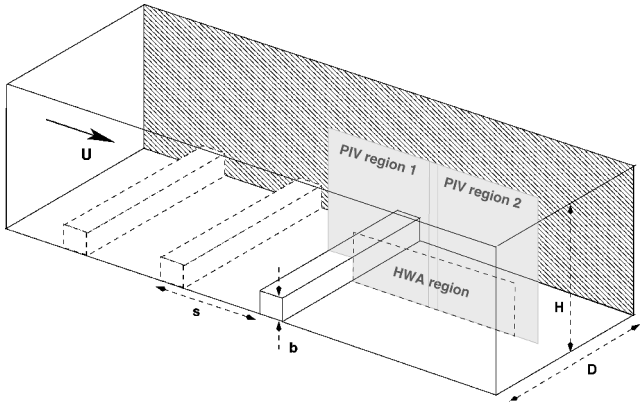

Fig. 2 Flow channel for experimental measurement of velocity time series.

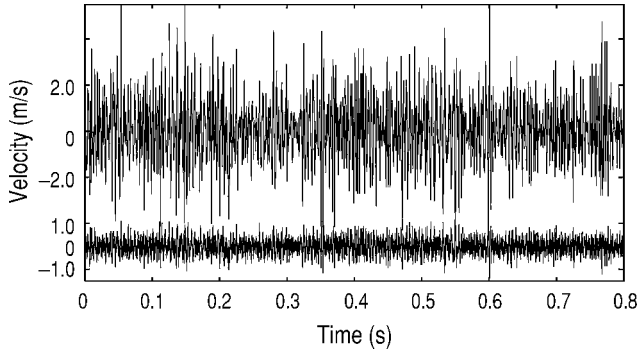

Fig. 3 High-pass filtered experimental velocity time series for $Re = 1 \times 10^5$.

component. These particular data have been employed simply because they represent the only high-resolution LDA measurements available from experiments conducted mainly with particle image velocimetry (PIV).

For the given flow configuration and value of Reynolds number considered, we construct the curve fit of the high-pass filtered data corresponding to the SGS behavior in a LES formalism. In this case,

$$u^*(x,t) = U(x,t) - \tilde{u}(x,t)$$

$$v^*(x,t) = V(x,t) - \tilde{v}(x,t)$$

where the tilde denotes (formally) a spatial filtering. In the present case, however, we are more interested in the temporal behavior of the flow because it is the factor associated with temporal fluctuations in the synthetic velocity SGS models that we wish to construct. The filtering method employed in the present study is, thus, a sharp cutoff Fourier temporal filter implemented as follows. Let a_f and b_f be the real and imaginary parts, respectively, at frequency f of the transformed series of either velocity component. For the case of a high-pass filter operation, new frequency series are obtained as follows:

$$A_f = 0, \qquad B_f = 0, \qquad \text{for all} \qquad f < f_c$$

$$A_f = a_f, \qquad B_f = b_f, \qquad \text{for all} \qquad f \geq f_c$$

where f_c is the high-pass cutoff filtering frequency. In the present study, we set $f_c = 15$ Hz. This implies that all frequencies lower than 15 Hz were removed. Then the inverse Fourier transformation of the new frequency series is performed to obtain the high-pass filtered velocity time series. It is important to recognize that this actually represents most of the measured signal. As a consequence, it is a difficult case to model. Moreover, it is actually much more representative of RANS fluctuations than of those associated with LES.

Each experimental velocity time series was filtered using the described methodology; these are shown in Fig. 3. The top one is u^* in the streamwise direction, whereas the bottom one is v^* corresponding to the velocity component normal to the bottom surface of the channel.

These time series contain a significant amount of information. One of the first things we can estimate directly is the number of terms one might use in the model Eq. (4), that is, the value of k. In particular, because each term in Eq. (4) is a realization of the PMNS equation that corresponds to (formally) only a single wave vector,[15,16] obviously using only one term is not likely to produce a good fit of the time series in Fig. 3. Furthermore, it cannot be expected that all of the frequency information indicated in Fig. 3 will be captured using two terms of the DDS. The evidence of high-level turbulent activity shown in Fig. 3 suggests that we should initially attempt a fit with at least three terms ($k = 3$) in the linear combination of PMNS equations (2). Note that all of the analysis to this point has been rather heuristic, and a more reliable number of terms should be decided through extensive calculations. On the other hand, one should expect that a term representing each of low, medium, and high wave numbers within the inertial subrange might be a good starting point for a model.

As was described in the preceding section, once the GA is started, it randomly generates a certain number of evaluations of the objective function (7). Figure 4 shows two velocity components modeled using one of these initial solutions. Observe that the appearance of the initial model is far from that of the experimental time series (Fig. 3). This implies that finding the correct parameters is a nontrivial process (and one that requires considerable CPU time). Table 1 provides a comparison of parameter values between the first generation solution (Fig. 4) and the final result (Fig. 5) obtained after about 1×10^6 generations. The differences are very significant, as summarized in the change of objective function value Q. Figure 5 shows two high-pass filtered velocity time series. Two velocity components are arranged in the same way as those in Fig. 3. However, the left part (before 0.4 s) of Fig. 5 corresponds to experimental data, whereas the right part shows the fitting result. It is observed that the appearance of the result is quite close to the

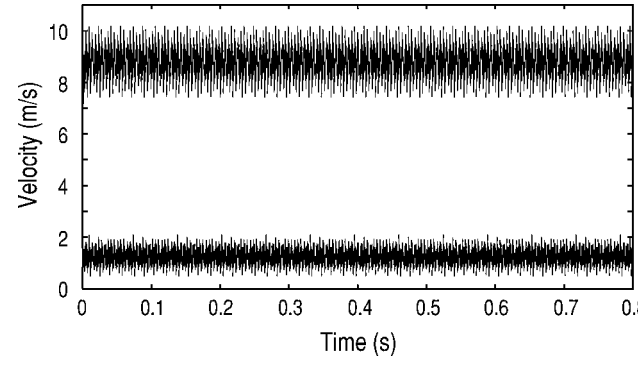

Fig. 4 High-pass filtered velocity time series of Eq. (2) evaluated using an initial guess for the parameters.

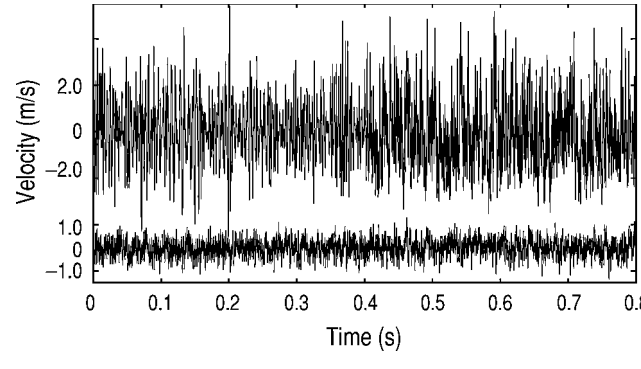

Fig. 5 High-pass filtered velocity time series for $Re = 1 \times 10^5$ with left part (before 0.4 s) experimental data and the right part the final fit.

Table 1 Values of the parameters in the model Eq. (2) for velocity time series at $Re = 1 \times 10^5$

Parameter	Initial solution	Final fit
ω_1	28	13
ω_2	23	3
ω_3	7	23
$\alpha_{1,1}$	17.4602	7.7440
$\alpha_{2,1}$	−23.6459	−27.916
$\alpha_{3,1}$	23.3529	19.4420
$\alpha_{1,2}$	2.1040	9.9084
$\alpha_{2,2}$	9.0649	−4.4369
$\alpha_{3,2}$	−9.2578	−5.1633
β_1	2.5629	3.1798
β_2	3.0791	3.8055
β_3	3.0719	3.0443
$\gamma_{1,1}$	0.32660	−0.2713
$\gamma_{2,1}$	−0.08195	0.4478
$\gamma_{3,1}$	−0.04513	0.3873
$\gamma_{1,2}$	−0.56670	−0.3623
$\gamma_{2,2}$	−0.13266	0.2548
$\gamma_{3,2}$	−0.35942	−0.7450
θ	0.13660	0.1092
Q	111.709	2.3213

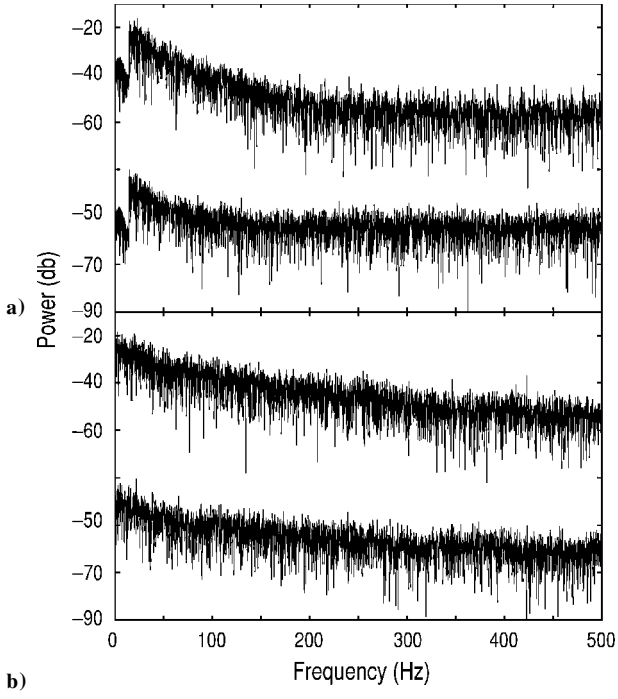

Fig. 6 Power spectra of high-pass filtered velocity time series for $Re = 1 \times 10^5$: a) experimental data and b) final fit of time series.

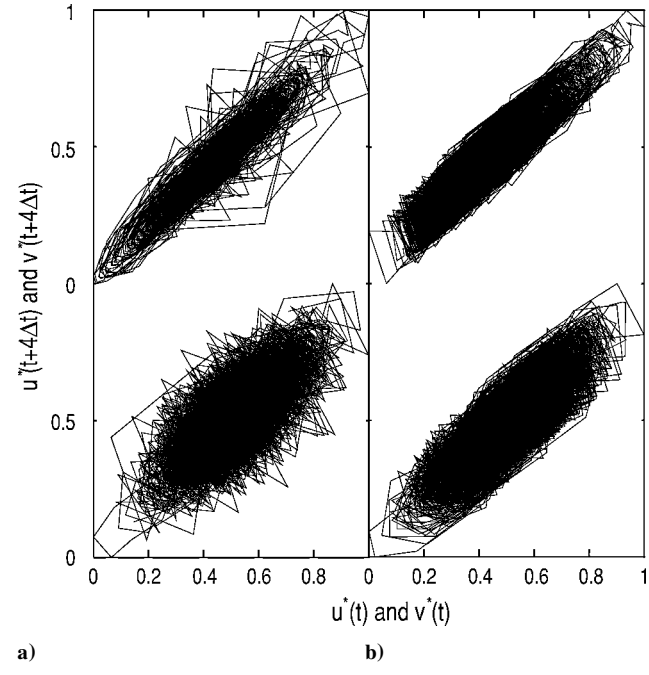

Fig. 7 Delay maps of high-pass filtered velocity time series for $Re = 1 \times 10^5$: a) experimental data and b) final fit of time series.

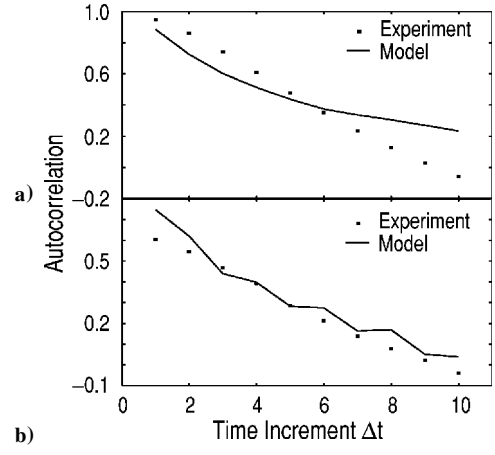

Fig. 8 Autocorrelation of each of the velocity components.

experimental time series. In particular, it is especially difficult to tell the difference between the measurement and data-fitting results for the velocity component v^*.

Further comparisons corresponding to power spectra and delay maps are presented in Figs. 6 and 7, respectively, with Figs. 6a and 7a corresponding to measured data and Figs. 6b and 7b to results of the final fit in each. In each of Figs. 6 and 7, the upper part corresponds to u^* and the lower one to v^*. First, note that the power of the high-pass filtered experimental time series is very low at low frequency ($f < 15$ Hz) and jumps to the actual level at $f = 15$ Hz. Obviously, this is caused by the filtering operation as described earlier, and cannot be avoided with a sharp cutoff Fourier filter. Within the range of $f > 15$ Hz, the two power spectra of Figs. 6a agree well, in terms of both power levels and decay rates, with those in Fig. 6b. The delay maps in Fig. 7a also appear to be topologically equivalent to those in Fig. 7b.

As was true in Ref. 11, neither the power spectra nor the delay maps were included in the data-fitting process; hence, they provide independent checks of goodness of fit. Also note that the previous study[13] of the curve-fitting process using only a logistic map had significant difficulties reproducing the power spectra and delay maps of the experimental data. The PMNS equations, on the other hand, evidently have no such deficiency, implying that a synthetic velocity SGS model based on this DDS is a viable candidate for future LES development.

We also remark that power spectral of the individual velocity components are directly related to the contributions to turbulence kinetic energy of each of these. Thus, one sees that the DDS produced by this fitting process captures this aspect of physical turbulence quite well. Table 2 provides comparisons between models and experimental data for three additional statistical models and experimental data for three additional statistical quantities of importance. The first is the uv correlation, which is not particularly well modeled; the experimental value is nearly a factor of two higher than the prediction. On the other hand, flatness of both velocity components shows good agreement, but skewness is rather poorly represented by the model.

The final comparison we provide is that of the autocorrelation of each of the velocity components: Figures 8a and 8b show horizontal and vertical velocity autocorrelations, respectively. It is observed that agreement between model and experiment is reasonably good for the horizontal component and quite good for the vertical

Table 2 Statistical properties of experimental and modeled velocities

	Experiment		Model	
Property	u	v	u	v
Cross correlation	0.609	0.609	0.350	0.350
Flatness	3.507	3.190	3.200	3.000
Skewness	0.115	0.039	0.630	−0.078

component. When one considers that the modeled results presented here are closer to a RANS model than to LES because our filter cutoff was actually at a quite low frequency, these results must be considered as very satisfactory. Furthermore, we employed the only data set available to us; we have not presented special results selected to make the approach look superior. Thus, we expect that this level of faithfulness would be achieved with similar fits of most other data.

Conclusions

We have focused on constructing chaotic maps from experimental velocity time series. Linear combinations of PMNS equations were fit directly using experimental high-pass filtered two-dimensional turbulent velocity components. The chaotic map curve-fitting method proposed by McDonough et al.[9] utilizing a weighted least-squares functional corresponding to a wide range of statistical quantities as the objective function was adopted. A GA was employed as the optimization method to minimize this complicated objective function containing real, integer, and nondifferentiable features. This is the first known application of a GA technique for fitting chaotic data.

LDA velocity data from a two-dimensional flow behind a turbulator were fit using such linear combinations, and the results were compared with the original experimental data using three diagnostics, namely, the appearance of the time series, the power spectral density, and the delay map. The results compare reasonably well for all three. Overall, other specific statistical quantities were also matched reasonably well with the exception of velocity cross correlation and skewness.

Based on this study, we conclude that synthetic velocity SGS models employing PMNS equations possess significant potential in the context of LES. Indeed, we have by now already completed an initial implementation. Run times for map evaluations over an entire LES grid are roughly the same as needed for resolved-scale equations. However, because large- and small-scale calculations can be done completely in parallel, the necessary wall-clock time is no different than for usual LES. This implementation has been made possible by being able to predict values theoretically of all of the bifurcation parameters found by data fitting in the present study.

However, this does not diminish the utility of such analysis. In particular, as already noted, the PMNS equations possesses potential for use in real-time control, and within such a context, it would be useful to employ laboratory experiments from the device to be controlled to fine tune the parameters of the PMNS equations specifically for the intended application. Moreover, even with regard to LES, such data analyses are useful because they provide a more detailed a priori test of SGS models than has previously been possible, and it should be expected that such tests should be more meaningful with respect to the present formalism (modeling physics) than has been done for usual LES (modeling statistics).

Acknowledgments

The financial support from the Air Force Office of Scientific Research under Grant F49620-00-1-025 and from NASA/EPSCoR Grant WKU-522635-00-10 is gratefully acknowledged by the authors. Work of Tianliang Yang also has been partially supported by the University of Kentucky Center for Computational Sciences. In addition, the authors are grateful to David L. Carroll for the genetic algorithm FORTRAN driver and to the University of Kentucky Computing Center for use of their Hewlett–Packard N-4000 and SuperDome computers for all required calculations.

References

[1]Hylin, E. C., and McDonough, J. M., "Chaotic Small-Scale Velocity Fields as Prospective Models for Unresolved Turbulence in an Additive Decomposition of the Navier–Stokes Equations," *International Journal of Fluid Mechanics Research*, Vol. 26, 1999, pp. 539–567.

[2]Domaradzki, J. A., and Saiki, E. M., "A Subgrid-Scale Model Based on the Estimation of Unresolved Scales of Turbulence," *Physics of Fluids*, Vol. 9, 1997, pp. 2148–2164.

[3]Scotti, A., and Meneveau, C., "A Fractal Method for Large Eddy Simulation of Turbulent Flow," *Physica D*, Vol. 127, 1997, pp. 198–232.

[4]McDonough, J. M., Yang, Y., and Hylin, E. C., "Modeling Time-Dependent Turbulent Flow over a Backward-Facing Step via Additive Turbulent Decomposition and Chaotic Maps," *Proceedings of First Asian Computational Fluid Dynamics Conference*, edited by W. H. Hui, Y.-K. Kwok, and J. R. Chasnov, Dept. of Mathematics, Hong Kong Univ. of Science and Technology, Hong Kong, PRC, 1995, pp. 747–752.

[5]Hylin, E. C., and McDonough, J. M., "Chaotic Map Models for the Small-Scale Quantities in an Additive Decomposition of the Navier–Stokes Equations. Part 1. Theory," Mechanical Engineering Rept. CFD-01-94, Univ. of Kentucky, Lexington, KY, Jan. 1994.

[6]Frisch, U., *Turbulence: The Legacy of A.N. Kolmogorov*, Cambridge Univ. Press, Cambridge, England, U. K., 1995, p. 31.

[7]Germano, M., Piomelli, U., Moin, P., and Cabot, W. H., "A Dynamic Subgrid-Scale Eddy Viscosity Model," *Physics of Fluids A*, Vol. 3, 1991, pp. 1760–1765.

[8]Sagaut, P., *Large Eddy Simulation for Incompressible Flow*, Springer-Verlag, Berlin, 2001, pp. 191–194.

[9]McDonough, J. M., Mukerji, S., and Chung, S., "A Data Fitting Procedure for Chaotic Time Series," *Applied Mathematics and Computation*, Vol. 95, 1998, pp. 219–243.

[10]Casdagli, M., and Eubank, S. (eds.), *Nonlinear Modeling and Forcasting*, Santa Fe Inst. Studies in the Sciences of Complexity, Proceedings Vol. 12, Addison–Wesley, Redwood City, CA, 1992.

[11]Mukerji, S., McDonough, J. M., Mengüç, M. P., Manickavasagam, S., and Chung, S., "Chaotic Map Models of Soot Fluctuations in Turbulent Diffusion Flames," *International Journal of Heat and Mass Transfer*, Vol. 41, 1998, pp. 4095–4112.

[12]McDonough, J. M., and Yang, T., "Parallelization of a Chaotic Dynamical Systems Analysis Procedure." *Parallel Computational Fluid Dynamics, Practice and Theory*, edited by P. Wilders, A. Ecer, J. Periaux, N. Satofuka, and P. Fox, North-Holland Elsevier, 2002, pp. 253–260.

[13]Roclawski, H., Jacob, J. D., Yang, T., and McDonough, J. M., "Experimental and Computational Investigation of Flow in Gas Turbine Blade Cooling Passages," AIAA Paper 2001-2925, June 2001.

[14]Guckenheimer, J., and Holmes, P., *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer-Verlag, New York, 1983, pp. 120–123.

[15]McDonough, J. M., and Huang, M. T., "A Low-Dimensional Model of Turbulence–Chemical Kinetics Interactions," Third International Symposium on Scale Modelling, Nagoya, Japan, Sept. 2000.

[16]McDonough, J. M., and Huang, M. T., "A 'Poor Man's Navier–Stokes Equation: Derivation and Numerical experiments—the 2-D Case," Mechanical Engineering Rept. CFD-03-01, Univ. of Kentucky, Lexington, KY, 2001; also *International Journal for Numerical Methods in Fluids* (to be published).

[17]Burton, G. C., Dahm, W. J. A., Dowling, D. R., and Powell, K. G., "A New Multifractal Subgrid-Scale Model for Large-Eddy Simulation," AIAA Paper 2002-0983, Jan. 2002.

[18]Goldberg, D. E., and Deb, K., "A Comparitive Analysis of Selection Schemes Used in Genetic Algorithms," *Foundations of Genetic Algorithms*, edited by G. J. E. Rawlins, Morgan Kaufmann, San Mateo, CA, 1991.

[19]Davis, L., *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991.

[20]Foias, C., Manley, O., and Teman, R., *Navier–Stokes Equations and Turbulence*, Cambridge Univ. Press, Cambridge, England, U.K., 2001, pp. 36–40.

[21]May, R. M., "Simple Mathematical Models with Very Complicated Dynamics," *Nature*, Vol. 261, 1976, pp. 459–467.

[22]Carroll, D. L., "Genetic Algorithms and Optimizing Chemical Oxygen–Iodine Lasers," *Developments in Theoretical and Applied Mechanics*, Vol. 18, 1996, pp. 411–424.

[23]Carroll, D. L., "Chemical Laser Modeling with Genetic Algorithm," *AIAA Journal*, Vol. 34, 1996, pp. 338–346.

R. M. C. So
Associate Editor