TURBULENCE ON A DESKTOP

By Mark A. Peterson and Yanir Rubinstein

From the Editors

This column is the last one that we will do as editors. We have been editing the Computer Simulations column continuously since our first column in the July/August 1989 issue of Computers in Physics. We wish to thank all of the authors who have contributed to the column and patiently suffered through our suggestions and revisions, and the editors of Computers in Physics and Computing in Science & Engineering for their help and support. We will really miss editing the column, in part because it gave us a vehicle for learning new physics and algorithms, and in part because we could ask the authors clarifying questions that would have been too embarrassing to ask in other contexts.

Our reason for leaving is that as of 1 July 2001, Jan Tobochnik will become the new editor of the American Journal of Physics and Harvey Gould will become the new associate editor. We look forward to the challenges of editing the American Journal of Physics, and, as usual, we look forward to your contributions and input.

OVER ITS LONG HISTORY, THE PROBLEM OF TURBULENCE HAS BEEN APPROACHED IN MANY DIFFERENT WAYS. NINETEENTH-CENTURY METHODS1 WERE SUFFICIENT TO DESCRIBE SIMPLE LAMINAR FLUID FLOWS (THAT IS, SMOOTH, NOT TURBULENT), such as steady Poisseuille flow through a pipe. This type of flow has a parabolic velocity profile over the cross section of the pipe. However, the experiments of Osborne Reynolds on real fluid flow showed that Poisseuille flow is realized only at low flow rates. For reasons that are subtle, the flow suddenly becomes turbulent at higher flow rates.2 The average velocity profile becomes rather flat over much of the pipe cross section, and the resistance to flow increases. Remarkably, the turbulent resistance of the pipe in this new regime depends strongly on the roughness of the interior wall of the pipe—that is, on details that play no role at all in Poisseuille flow. This behavior is a strong indication that turbulent flow involves many length scales.

This example suggests one of the motivations for wanting to understand turbulence more precisely. The surprising sensitivity of turbulence to small perturbations, such as wall roughness, suggests that we could control turbulence if we understood it better. An example is the dimpling on a golf ball, which lowers the resistance of the golf ball in its flight by inducing a turbulent wake. Needless to say, this effect was discovered empirically.

Understanding turbulence through computation

The computational methods most often used by engineers, even today, are conceptually similar to the methods suggested by the first investigators. In 1897 Joseph Boussinesq suggested replacing the viscosity, a parameter in the Navier-Stokes equation of fluid motion, with a turbulent viscosity that is typically much larger in magnitude and varies with position and time depending on the state of the flow.3 In this and similar methods, the resulting equations are still Navier-Stokes-like, but they differ in the phenomenology of the flow-dependent parameters. Such phenomenology can be refined indefinitely, and ever more experience can be incorporated into it. However, a satisfactory understanding of turbulence still has not been realized, and wind tunnels are still necessary. The reason is probably that these practical, phenomenological methods ignore the many length scales that seem to be an essential, characteristic feature of turbulence.

Turbulent flows are not steady, but fluctuating, and hence some kind of
statistical description is necessary. Indeed, the phenomenology mentioned earlier typically involves averages of fluctuating quantities. Perhaps the most striking success of turbulence theory is the prediction by Andrey Kolmogorov of the statistical distribution of energy over the many length scales in turbulent flow, the \(-5/3\) law.\(^4\) By what amounts to simple dimensional analysis, Kolmogorov argued that in strong, fully developed turbulence, the energy distribution function should follow a power law, with more energy in the large-scale motions (large eddies) and less in the small-scale motions, at least over length scales between the large scale, at which the fluid is forced, and the small scale (now called the Kolmogorov length scale), at which viscosity becomes important. Kolmogorov’s argument does not require the Navier-Stokes equation at all, so it is not really part of a computational method, but it is a striking insight, nonetheless. The energy in real turbulence does, in fact, seem to be distributed according to the Kolmogorov power law.

Digital computers were applied to the turbulence problem as soon as it was feasible, and new phenomena were discovered soon thereafter. For example, the subject of chaos theory owes much to Edward Lorenz and the famous Lorenz attractor, discovered as a byproduct of trying to model the turbulent atmosphere.\(^5\) One reason that chaos became so immediately interesting and popular was the existence of examples such as the Lorenz attractor, involving only a few degrees of freedom and very simple computations. Hence chaos was, in a sense, accessible to everyone, unlike the problem of explicitly simulating turbulence.

Supercomputers of the 1970s finally had sufficient speed and memory for direct numerical simulations of turbulent flow.\(^6,7\) These simulations gave researchers a “microscope” to see every detail of turbulence. Initially, the microscope had fairly low resolving power; that is, the grids on which the simulations were performed were fairly crude. With advances in hardware came finer grids and increased resolution. Now these simulations look like real turbulence. For example, the Kolmogorov power law for the energy spectrum can be observed in sufficiently large simulations.

The Navier-Stokes equations are Newton’s laws of motion applied to a fluid. To derive them one imagines a small volume of the fluid and considers all the forces on it, including most importantly the forces due to the adjoining fluid.\(^1\) Some assumptions must be made, including that there are forces proportional to the rate of strain (viscous forces). The success of the theory in the laminar flow regime for many fluids (Newtonian fluids) is strong evidence that these assumptions are correct. For an incompressible fluid of constant density \(\rho\), the equation for the velocity field \(\vec{u}\) is

\[
\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} = -\nabla P + \nu \nabla^2 \vec{u} + \vec{f}
\]  

where \(\nu\) is the kinematic viscosity, \(P\) is the pressure, and \(\vec{f}\) is an external force per unit mass, such as the acceleration due to gravity.

The Navier-Stokes equations are Newton’s laws of motion applied to a fluid. To derive them one imagines a small volume of the fluid and considers all the forces on it, including most importantly the forces due to the adjoining fluid. Some assumptions must be made, including that there are forces proportional to the rate of strain (viscous forces). The success of the theory in the laminar flow regime for many fluids (Newtonian fluids) is strong evidence that these assumptions are correct. For an incompressible fluid of constant density \(\rho\), the equation for the velocity field \(\vec{u}\) is

\[
\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} = -\nabla P + \nu \nabla^2 \vec{u} + \vec{f}
\]  

where \(\nu\) is the kinematic viscosity, \(P\) is the pressure, and \(\vec{f}\) is an external force per unit mass, such as the acceleration due to gravity.

The flow of an incompressible fluid can be described by a divergenceless velocity vector field \(\vec{u}(t, x, y, z)\), where \((x, y, z)\) is the position. In our simulations positions are defined on a \(32^3\) spatial grid. (We frequently reduced the grid to 24\(^3\).) Our aim is to generate turbulent flow, not to model a realistic experimental geometry. We impose periodic boundary conditions on \(\vec{u}\), so that there are no solid boundaries. The discretized, periodic flow \(\vec{u}\) is represented by a finite Fourier series

\[
\vec{u}(t, x, y, z) = \sum_k \vec{a}_k(t) e^{i(k_x x + k_y y + k_z z)}
\]  

where the sum is over a \(32^3\) lattice (or \(24^3\) lattice) in \(k\)-space. We also consider the vorticity vector field

\[
\vec{\omega} = \nabla \times \vec{u}
\]  

which is the curl of the velocity vector field. The vorticity has a Fourier representation identical in form to Equation 2,

\[
\vec{\omega}(t, x, y, z) = \sum_k \vec{b}_k(t) e^{i(k_x x + k_y y + k_z z)}
\]  

with coefficients

\[
\vec{b}_k = i \kappa \times \vec{a}_k.
\]
Because the flow is divergenceless, we also have
\[ \nabla \cdot a_k = 0, \]
and
\[ (6) \]
We introduce a simple force with just two nonzero Fourier components,
\[ \tilde{f} = f \sin(y) \hat{x} \]
(7)
to do work on the fluid. This force pushes the fluid in the positive \( x \)-direction in half the system and in the negative \( x \)-direction in the other half, so that the fluid is sheared. In terms of the vorticity \( \omega \), the Navier-Stokes equations become
\[ \frac{\partial \omega}{\partial t} = \nabla \times (\hat{u} \times \omega) + \nu \nabla^2 \omega + \nabla \times \tilde{f}. \]
(8)
The advantage of this form over the more familiar form of Equation 1 is that the pressure \( P \) has been eliminated.

We express Equation 8 as a system of ordinary differential equations for the time-dependent Fourier coefficients \( b_k \) of the vorticity,
\[ \frac{db_k}{dt} = \tilde{c}_k - \nu k^2 b_k + g_k \]
(9)
where the \( \tilde{c}_k \) in Equation 9 are the Fourier coefficients of the nonlinear term, \( \nabla \times (\hat{u} \times \omega) \) in Equation 8. The forcing terms \( g_k \) are all zero except for
\[ g_k = f / 2. \]
(10)

Once we determine the Fourier coefficients \( \tilde{c}_k \), we have specified the differential equations in Equation 9, and well-known numerical methods can be used to integrate them. (We used a fourth-order Runge-Kutta algorithm.)

The method for finding the \( \tilde{c}_k \) proceeds as follows. At each step of the integration of Equation 9, we know \( b_k \), the Fourier coefficients of \( \omega \), and hence by Equation 6, \( \hat{a}_k \), the Fourier coefficients of \( \hat{u} \). By inverse fast Fourier transforms, we find \( \hat{u} \) and \( \omega \) and hence \( \hat{u} \times \omega \). By a fast Fourier transform we find the Fourier coefficients \( d_k \) of this cross product. Then we find the Fourier coefficients \( \tilde{d}_k \) of this cross product. Then
\[ \tilde{c}_k = i k \times \tilde{d}_k. \]
(11)
Matlab code for this pseudospectral method can be found at www.mtholyoke.edu/~mpeterso/reu/99/code.

If the viscosity \( \nu \) is sufficiently large, a random initial flow becomes laminar, because the viscous term – \( \nu k^2 b_k \) in Equation 9 damps out all the unforced Fourier coefficients. For smaller \( \nu \), the nonlinear term \( c_k \) couples energy from the forced coefficients to the other coefficients faster than the viscous term can damp them out, and they do not approach zero. If \( \nu \) is not too small, a statistically steady state results, in which energy flows from the forced coefficients at small \( k \) to the other coefficients, finally to be damped out at the largest \( k \) values, where the damping is most effective. These are the turbulent solutions. The flow of energy in \( k \) space, accompanied by noisy values for the intermediate Fourier coefficients, is sometimes called the Kolmogorov cascade. If \( \nu \) is chosen to be too small, there is insufficient damping to dissipate the incoming energy, and the numerical solution blows up.

The turbulent solutions in these simple simulations exhibit the characteristic properties of real turbulence. For example, there is anomalous resistance to forcing, as shown in Figure 1. Here the \( z \)-component of the forced Fourier coefficient \( b_{z(0,1,0)} \) has been artificially set to zero at time \( t = 0 \), and we follow its response in time for a turbulent solution. The dashed line shows the exponential approach to its
steady laminar value, the behavior it would have in the absence of turbulence. But because this mode loses energy to other Fourier modes, it never reaches this value, and it levels off noisily at about one-fifth of its laminar steady-state value. An experimentalist would say the resistance was anomalously large by a factor of 5.

We can see how the energy is distributed over various length scales by summing the energy of the modes in concentric spherical shells in $k$-space. We slice $k$-space up into spheres of radius $k_j$ about the origin and thickness $\kappa$, and find the sums

$$\Phi(k_j) = \sum_{k_j - \kappa^2 \leq |k| < k_j + \kappa^2} |\mathbf{k}|^2$$

as shown in Figure 2. This calculation is necessarily rather crude, because we don’t have much $k$-space to work with, just our $32^3$ grid. We can’t take too many spherical shells, or there wouldn’t be enough grid points in each shell to give a good average. However, the distribution of energy looks much the same for any reasonable number of shells and is consistent with a power law, although steeper than the $-5/3$ Kolmogorov law. The slope in the log–log plot is actually close to $-3$. This distribution is not unreasonable, because the smallest-length scale must be near the Kolmogorov length in order for dissipation there to establish the statistically steady state. Beyond this length scale (larger $k$), the energy must fall off faster than any power. We are trapped near this region by hardware limitations, but the distribution still looks like a power law (over an admittedly small range).

Our preliminary observations suggest that the qualitative features of turbulence are exhibited in these small simulations and that relatively small simulations might yield physical insight. What such an insight might be is unknown, but it would involve both the geometry and the statistics of the flow. A coordinated search for it is conceivable. The limitations of desktop computers could be partially offset by a distributed approach to such computations. In astronomy, number theory, and cryptology, it has been possible to enlist amateurs on significant projects. Perhaps this approach would also work for turbulence theory.

The simulations that we have described do not correct for aliasing, although aliasing is a problem in the pseudospectral method (the application of finite Fourier transforms to nonlinear problems). Aliasing causes a sufficiently high spatial frequency, generated by the nonlinear term, to be mistaken for a low spatial frequency, because of the discrete sampling imposed by the grid. In our solutions it amounts to a small noise source that is not explicitly included in the model but is an artifact of the method. One can rigorously eliminate aliasing by further truncation of the grid, throwing away the largest $k$ values in each time step, but as Figure 2 shows, these coefficients are small anyway. The review article by Parviz Moin and Krishnan Mahesh is useful on this and other numerical issues.

One might think that the choice of initial conditions would be unimportant and that they could be chosen randomly, but this is not strictly true. If one chooses the $b_\ell$ to be random and real, the system is on a rather special, invariant surface in the fluid’s phase space, and such flows apparently do not become turbulent! A systematic way to initiate a turbulent flow is to start with what are called $ABC$ flows. These are eigenstates of the curl operator and look typically like

$$u_x = A \cos y + B \sin z$$  \hspace{1cm} (13a)

$$u_y = B \cos z + C \sin x$$  \hspace{1cm} (13b)

$$u_z = C \cos x + A \sin y.$$  \hspace{1cm} (13c)

A single $ABC$ flow decays exponentially in time, but a superposition of two linearly independent ones becomes turbulent after an initial transient period.
Making sense of the data

Direct numerical simulations of turbulence contain a wealth of information. The challenge is to make sense of it. There are literally thousands of Fourier coefficients, all looking qualitatively like Figure 3. One naturally thinks of using statistical methods. For example, time autocorrelation functions indicate a mixture of randomness and systematic motion. Most correlations decay rapidly, but there is typically an indication of long-term, low-amplitude oscillations as shown in Figure 4.

The Kolmogorov picture gives at least a rough picture of what happens in turbulence in Fourier space, but what does it correspond to in real space? Are there just a few important processes, processes that we can visualize? Is there a probabilistic description of this far-from-equilibrium situation that generalizes the Maxwell-Boltzmann distribution? These are completely open questions. Many approaches are possible.

One approach is to try visualization methods. Figure 5 shows a projection of \( \mathbf{u} \) onto a typical \( z = \text{constant} \) slice of space (the \( z \)-component is not indicated). The overall mean flow pattern corresponding to the systematic forcing term is not really discernible, because it is complicated by whirls and reconnections that are different in different slices and that change in time. It would be possible to follow the path of a marker advected with the flow and visualize this marked region, analogous to observing dye streaks in a real experiment.

One can also construct derived quantities such as the rate-of-strain tensor \( \sigma_{ij} \). The Fourier coefficients of this quantity are

\[
\sigma_{ij} = \mathcal{F}^{-1}\left[ k_j^i \delta_k^j + k_j^i \delta_k^j \right] / 2,
\]

so that \( \sigma_{ij} \) can be constructed with an inverse Fourier transform. If \( \delta_{ij} \) is the Kronecker delta, \( \delta_{ij} + \sigma_{ij} \Delta t \) is the matrix of a quadratic form, defining an ellipsoid at each point of our grid, which is a visualization of the deformation of a small sphere of fluid in the short time \( \Delta t \). An eigenvalue of \( \sigma_{ij} \) is positive if the sphere is being stretched along the corresponding principal axis of the ellipsoid, and it is negative if the sphere is being contracted. Because our fluid is incompressible, the trace of \( \sigma_{ij} \) is zero, and both signs typically occur; that is, the fluid stretches along at least one direction and contracts along at least one direction.

The orientation of the vorticity vector \( \omega \) with respect to the maximal stretching direction has been the subject of much investigation. An early suggestion that vorticity would tend to
align with the direction of maximal stretching was shown in direct numerical simulations in the late 1980s to be not generally true, and in fact in certain situations there is a clear tendency for vorticity to align with the eigenvector of the intermediate eigenvalue of the rate of strain. This eigenvalue could have either sign, and suggests that what is going on is more subtle than simple, unambiguous stretching of vorticity.

A scalar quantity related to the local rate of stretching of vorticity is

\[ I = \sum_{j} \omega_{j} \sigma_{j} \omega_{j} \]  

(14)

We find \( I \) to be concentrated, at any given moment, in a few localized regions that show up as “hot spots” (regions where \( I \) is large) in slices through the fluid. Another scalar quantity that is nonuniformly distributed in a qualitatively similar way, with local concentrations, is the helicity, \( \vec{u} \cdot \vec{\omega} \). Could quantities like these, visible in the turbulence microscope, be keys to a better understanding of turbulence?

Order out of chaos

Over the last few decades, questions that originated in the turbulence problem have given rise to chaos theory, a new paradigm. The Lorenz attractor, distilled out of a system of equations for meteorology (that is, turbulent flow of the atmosphere), was perhaps the first widely discussed example of a simple chaotic system. Now it is possible to turn these still relatively new methods back upon the problem that inspired them, turbulence itself.

One simple method for discovering patterns in a chaotic time series \( \{x(t)\} \) is reconstruction. The triplets, \( \{x(t), x(t+k), x(t+2k)\} \), are plotted using three axes for some suitable delay \( k \). As an example, a periodic function would generate points that fall on a single closed curve in 3-space. Chaotic functions, although not periodic, might still generate recognizable patterns in space, not so different from closed curves, and readily distinguishable from uniform, random distributions. Figure 6 shows the data of Figure 3 graphed in this way and viewed from two different “directions” in this abstract space. The points of the time series are not randomly distributed but lie predominantly in a plane. Within the

Figure 5. Projection of a typical turbulent velocity field onto a \( z = \) constant plane.

Figure 6. The time series in Figure 3 is reconstructed as described in the article and viewed from two different angles. The points seem to lie approximately in a plane, which is viewed edge-on in the lower graph. The pattern is viewed face-on in the upper graph and shows a repetitive but not periodic behavior, typical of a strange attractor.
plane the curve loops systematically in a way that is not periodic but does not look merely random, either. Such patterns are candidates for one of the basic objects of chaotic dynamics, a “strange attractor.”

A lower bound on the dimension of a strange attractor is the number of positive Lyapunov exponents. The existence of at least one positive Lyapunov exponent is the standard criterion for chaos. We estimated the largest Lyapunov exponent for one of our flows and found that the largest Lyapunov exponent is convincingly positive, and hence that our turbulent flows are chaotic.

In this computation we regard a flow at any moment as given by its Fourier coefficients, and hence by a point in the 243-dimensional space of these coefficients. Over time, the flow traces out a trajectory in this space. We give the space the usual Euclidean norm \( | \cdot | \). That is, if \( \{ b_j \} \) and \( \{ B_j \} \) are the Fourier coefficients of two flows \( P_b \) and \( P_B \), we say the flows are separated by a distance

\[
| P_b - P_B | = \left[ \sum_j (b_j - B_j)^2 \right]^{1/2}.
\]

By forward integration of Equation 9 from some initial point \( P(0) \), we find the trajectory \( P \). At the same time, we start a second trajectory at \( Q_0(0) \), with

\[
| P(0) - Q_0(0) | = \varepsilon.
\]

After one integration time step, we compute the new distance between the two trajectories:

\[
| P(\Delta t) - Q_0(\Delta t) | = \lambda_1 \varepsilon.
\]

Here \( \lambda_1 \) is the factor by which the initial distance grew, and its logarithm is already a (very crude) estimate for the largest Lyapunov exponent. Because the second trajectory \( Q_0 \) might get too far away from \( P \), we start a new trajectory \( Q_1 \) by moving the point \( Q_0(\Delta t) \) toward \( P(\Delta t) \) until it is again just the distance \( \varepsilon \); that is, we have

\[
Q_1(\Delta t) = P(\Delta t) + \frac{Q_0(\Delta t) - P(\Delta t)}{\lambda_1 \varepsilon}.
\]

Then we repeat and find \( \lambda_2 \) from

\[
| Q_1(2\Delta t) - P(2\Delta t) | = \lambda_2 \varepsilon
\]

and so forth. The largest Lyapunov exponent \( \lambda \) is then

\[
\lambda_N = \lim_{N \to \infty} \frac{1}{N\Delta t} \sum_j \log(\lambda_j)
\]

where \( N \) is the number of iterations.

Our intuitive picture of a strange attractor suggests that \( \lambda \) should be insensitive to the choices of \( \varepsilon, \Delta t, P(0), \) and \( Q_0(0) \), but in reality it is not even clear that the limit exists. Two attempts to calculate the largest Lyapunov exponent are shown in Figure 7. Each calculation seems to be approaching a limit, but they do not appear to be the same limit. In both cases \( \varepsilon = 0.1 \), but they differ in the choice of \( P(0), Q_0(0), \) and \( \Delta t \). The result in both cases is that \( \lambda_N \) is convincingly positive. This calculation is an example of one where many desktop computers working separately could perhaps best clarify the pattern, because the computations

![Figure 7. Two determinations of the largest Lyapunov exponent of one of the turbulent flows. The quantity \( \lambda_N \) on the right side of Equation 20 before the limit \( N \to \infty \) is taken is shown as a function of \( t = N\Delta t \). In both cases, \( \varepsilon = 0.1 \). For \( \Delta t = 0.005 \), \( \lambda_N \approx 2 \), and for \( \Delta t = 0.02 \), \( \lambda_N \approx 13 \). The two trajectories analyzed here had different starting points.](image)
Turbulence theory has a well-earned reputation for being difficult, but, paradoxically, it is so difficult that it is actually accessible. Precisely because it is difficult to make progress analytically, an increasingly important part of turbulence theory is computational. The peculiarities of real turbulence suggest that new mathematics and new physics are waiting to be discovered. The power of even modest computers now opens the door to this avenue of investigation. 

Acknowledgments
This work was supported by the NSF Research Experience for Undergraduates program, Grant DMS-9732228, and by the Robert and Elizabeth Hutchcroft '35 Endowment Fund to the Mount Holyoke College Mathematics Department. The REU team included Emily Clark, Lubomira Ivanova, Aaron Koll, and Jenny Ross, who all contributed substantially to the project described here. All computations were done in Matlab, funded by a gift from Anne Van Lent to the Mount Holyoke College Mathematics Department.

References
10. G.K. Batchelor and A.A. Townsend, “Turbulent Diffusion,” Surveys in

Suggestions for Further Study
Chaos theory has introduced the notion of the Lyapunov exponent. In the main article we described an algorithm to compute the largest Lyapunov exponent for our turbulent flows, but what we found was puzzling. It would be useful to know what the “largest Lyapunov exponent” really means for these flows. This question lies behind the following suggestions.

1. Initialize your simulation using the ABC flows found in initbabc.m at www.mtholyoke.edu/~mpeterso/reu/99/code. For a fixed value of $\Delta t = 0.005$, run the simulation from $t = 0$ to $t = 50$ to equilibrate the flow. Save the parameters of this flow to reuse as an initial turbulent flow. Choose a value of $\epsilon$ to obtain a starting point for a nearby flow $Q(0)$ in the notation of Equation 16 in the main article. Next compute the largest Lyapunov exponent $\lambda$ as described in the main article by averaging from $t = 50$ to $t = 150$. Repeat with different choices for $Q(0)$. Is the value of $\lambda$ sensitive to $Q(0)$?

2. Initialize the flow as in the preceding question, but change one of the ABC parameters in initbabc.m by $\pm0.1$. This change will produce a different equilibrated flow at $t = 50$, and hence a different initial flow $P(0)$ in the notation of Equation 16. Recompute $\lambda$ by averaging from $t = 50$ to $t = 150$. Is the value of $\lambda$ sensitive to $P(0)$?

3. Determine whether the value of $I$ computed in Problems 1 and 2 is sensitive to the choice of $\Delta t$. In particular, repeat your computations for $\Delta t = 0.01$ and $\Delta t = 0.02$. If the computed value of $\lambda$ seems to behave like a random variable, how is it distributed?

4. If it is possible to understand the systematics of $\lambda$, one should look at its dependence on viscosity. At sufficiently large viscosity, the flow becomes laminar and stable, and hence $\lambda \leq 0$. How does the change of $\lambda$ happen? Is there a discontinuous transition at some critical viscosity, or is it continuous?

5. How do hot spots, concentrations of quantities expected to be associated with turbulence, such as $I = \omega \sigma(\omega)$ and $J = \vec{u} \cdot \vec{\omega}$, evolve in time? Do they form and disappear, or do they have a continuous existence and move around? Do they seem to interact with each other? Use the initial flow suggested in Problem 1, interrupt it at regular intervals, and examine the distribution of the scalar quantities $I$ and $J$ in space using the Matlab command slice. Keep a record of the time evolution by printing the resulting pictures.
MEMBERSHIP

The IEEE Computer Society is the world’s largest association of computing professionals, and is the leading provider of technical information in the field.

PURPOSE

Members receive the monthly magazine COMPUTER, discounts, and opportunities to serve (all activities are led by volunteer members). Membership is open to all IEEE members, affiliate society members, and others interested in the computer field.

BOARD OF GOVERNORS


Term Expiring 2002: Yanir Rubinstein, Mark A. Peterson, others interested in the computer field. He majored in mathematics at the Technion in Israel. Contact him at the Dept. of Mathematics, Weizmann Inst. of Science, Rehovot 76100, Israel; yanir@aluf.technion.ac.il.

Mark A. Peterson is a professor of mathematics and physics at Mount Holyoke College. His research interests include fluid dynamics in biology and the history of science. He received his PhD in physics from Stanford. Contact him at Mount Holyoke College, South Hadley, MA 01075; mpeterso@mtholyoke.edu.

Yanir Rubinstein is a graduate student at the Weizmann Institute of Science. His research interests include fluid dynamics in biology and the history of science. He received his PhD in physics from Stanford. Contact him at the Dept. of Mathematics, Weizmann Inst. of Science, Rehovot 76100, Israel; yanir@aluf.technion.ac.il.

EXECUTIVE COMMITTEE

President: BENJAMIN W. WAH* University of Illinois Coordinated Sci Lab 1308 W. Main St Urbana, IL 61801-2307 Phone: +1 217 333 3516 Fax: +1 217 244 7175 b.wah@computer.org


PUBLICATIONS

EXECUTIVE STAFF

Acting Executive Director: ANNE MARIE KELLY Publisher: ANGELA BURGESS Acting Director, Volunteer Services: MARY KATE RADA Chief Financial Officer: VIOLET S. DOAN Director, Information Technology: DEBORAH K. SCHERRER Manager, Research & Planning: JOHN C. KEATON

COMPUTER SOCIETY WEB SITE

The IEEE Computer Society’s Web site, at http://computer.org, offers information and samples from the society’s publications and conferences, as well as a broad range of information about technical committees, standards, student activities, and more.

COMPUTER SOCIETY OFFICES

Headquarters Office

1730 Massachusetts Ave NW Washington, DC 20036-1992 Phone: +1 202 371-0101 • Fax: +1 202 728 9614 E-mail: bsqf@computer.org

Publications Office

4000 Kissena Blvd., PO Box 3014 Los Alamitos, CA 90720-1314 General Information: Phone: +1 714 821-8380 E-mail: membership@computer.org

Membership and Publication Orders: Phone: +1 714 821-6657 • Fax: +1 714 821 4641 E-mail: order.ofc@computer.org

European Office

13, Ave. de L’Aquilon B-1200 Brussels, Belgium Phone: +32 2 770 21 98 • Fax: +32 2 770 85 05 E-mail: euro.ofc@computer.org

Asia/Pacific Office

Watanabe Building, 1-4-2 Minami-Aoyama, Minato-ku, Tokyo 107-0062, Japan Phone: +81 3 3408 3110 • Fax: +81 3 3408 3553 E-mail: tokyo.ofc@computer.org

IEEE OFFICERS


COMPUTER SIMULATIONS


