

Use of a Spreadsheet Package to Demonstrate Fundamentals of Computational Fluid Dynamics and Heat Transfer*

MARK SCHUMACK

University of Detroit Mercy, Mechanical Engineering Department, P.O. Box 19900, Detroit, MI 48219, USA. E-mail: schumamr@udmercy.edu

This paper describes how the Solver tool in Microsoft Excel can be used to solve nonlinear systems of equations resulting from finite difference discretization of the governing equations for fluid dynamics and heat transfer. Students learning computational fluid dynamics and heat transfer for the first time typically spend an inordinate amount of time struggling with algorithm development and programming issues. The use of the visual presentation and automated solution capabilities of Excel can help bolster the student's understanding of basic discretization issues unencumbered by programming details. Cell formatting can be used to visualize the geometry and how the grid variables interact with one another, a feature that is particularly helpful when dealing with irregular geometries or staggered grids. Results can be easily shown on contour or x-y plots. Although the number of grid points and thus accuracy is limited by the number of equations that the Solver tool can handle, a spreadsheet model is nevertheless valuable for giving students an intuitive feel for discretization before going on to write code. Two applications are described. The first is a two-dimensional conduction problem that is introduced to show how Solver can be used to solve systems of finite difference equations. The second demonstrates the solution of the incompressible Navier-Stokes equations using the staggered grid formulation. The conduction problem is one that is easily incorporated into an undergraduate heat transfer class, while the flow problem is appropriate for a graduate level course.

INTRODUCTION

THE USE OF spreadsheets for solving the finite difference equations resulting from discretization of the governing differential equations for heat conduction is well documented. Mokheimer and Antar [1, 2] describe how the iteration capability of spreadsheets can be used to solve multidimensional steady and transient conduction problems. They also discuss how the contour plot feature can be used to easily visualize results (contour plots in Excel are, however, limited to plotting results in relatively simple geometries). Schumack [3] describes how the Solver tool in Excel can be used to solve two-dimensional steady conduction problems. Solver uses the Generalized Reduced Gradient (GRG2) algorithm, which is designed to solve nonlinear optimization problems. It works by changing values of specified cells until constraints set by the user are satisfied. Two regions are set up in the spreadsheet: a region containing the initial guesses and ultimately the solution, and a region containing the discretized equations. For heat transfer and fluids problems, the regions can be shaped like the physical geometry, facilitating conceptualization of how the nodal

values interact with one another and visualization of results.

Kumar and Al-Shantaf [4] describe how spreadsheets can be used to solve simple fluid statics problems and the modified Bernoulli equation. Wiggins [5] describes a spreadsheet solution of the unsteady Couette flow problem using Gaussian elimination. Morishita [6] solves the incompressible Navier-Stokes equations and the energy equation for a two-dimensional channel flow using the vorticity-stream function formulation along with the circular reference (sometimes referred to as the 'iteration') feature of the spreadsheet package. The vorticity-stream function formulation requires post-processing to obtain velocities—a procedure that can be automated in the spreadsheet—and solution of a Poisson equation for pressure. Morishita *et al.* [7] show how various aeronautical problems can be solved using the matrix inversion and circular reference solution functions.

Although the numbers of changing cells and constraints are limited by Solver, its use provides a straightforward way to solve the discretized heat transfer and fluid flow equations. In this paper we show how Solver can be used to solve a relatively simple but nonlinear heat conduction problem and the incompressible Navier-Stokes equations in primitive variable form on a staggered grid. An

* Accepted 14 July 2004.

advantage of Solver over the circular reference approach is that use of Solver does not require manipulation of the discretized equations to solve for the nodal variables.

SOLUTION OF THE HEAT CONDUCTION EQUATION

In order to demonstrate how the Solver function in Excel is used to solve systems of equations, consider the following conduction problem, given to students in an undergraduate heat transfer class. The problem is to calculate the temperature distribution in a bracket for an automotive wiring harness as seen in Fig. 1. The bracket is made of 3-mm thick AISI 1010 carbon steel with a thermal conductivity of 15.1 W/m K. The bracket is connected to the engine block, which is at 523 K. The temperature of the surrounding air is 343 K with an associated convection coefficient of 25 W/m²K, and the surrounding surfaces are at 373 K. The emissivity of the bracket surface is 0.9. The wiring harness rests in the notch. The upper safe operating limit for the wiring harness is 383 K. Students were asked to calculate the temperature distribution in the bracket and conclude whether the temperature on the notch boundary exceeded the safe operating limit for the harness. The temperature is assumed invariant in the bracket thickness.

Discretization proceeds from either the governing two-dimensional differential equation or by applying an energy balance on the region surrounding each node (a good description of these techniques is found in Incropera and DeWitt [8]). Using the energy balance approach,

the finite difference discretization of the energy equation for an interior node becomes:

$$\begin{aligned} k\delta \frac{\Delta y}{\Delta x} (T_{i-1,j} - T_{i,j}) + k\delta \frac{\Delta y}{\Delta x} (T_{i+1,j} - T_{i,j}) \\ + k\delta \frac{\Delta x}{\Delta y} (T_{i,j+1} - T_{i,j}) + k\delta \frac{\Delta x}{\Delta y} (T_{i,j-1} - T_{i,j}) \\ + 2h\Delta x\Delta y(T_{\infty} - T_{i,j}) \\ + 2\varepsilon\sigma\Delta x\Delta y(T_{sur}^4 - T_{i,j}^4) = 0 \end{aligned} \quad (1)$$

where k is the bracket thermal conductivity, Δx and Δy are the distances between nodes in the x - and y -directions, ε is the bracket surface emissivity, δ is the bracket thickness, and σ is the Stephen-Boltzmann constant.

The equation for the nodal region on a boundary exposed to convection and radiation depends on whether the node is on an edge or a corner. The boundary condition for an edge node on the left side of the bracket, for example, can be written as:

$$\begin{aligned} k\delta \frac{\Delta y}{\Delta x} (T_{i-1,j} - T_{i,j}) + k\delta \frac{\Delta x}{2\Delta y} (T_{i,j+1} - T_{i,j}) \\ + k\delta \frac{\Delta x}{2\Delta y} (T_{i,j-1} - T_{i,j}) + h\delta\Delta y(T_{\infty} - T_{i,j}) \\ + \varepsilon\sigma\delta\Delta y(T_{sur}^4 - T_{i,j}^4) + h\Delta x\Delta y(T_{\infty} - T_{i,j}) \\ + \varepsilon\sigma\Delta x\Delta y(T_{sur}^4 - T_{i,j}^4) = 0 \end{aligned} \quad (2)$$

If the circular reference solution capability of Excel is used, the finite difference equations must be solved explicitly for the nodal temperatures $T_{i,j}$ before entering them as cell formulas. This extra manipulation is not necessary when using Solver. In order to use Solver, two regions of the

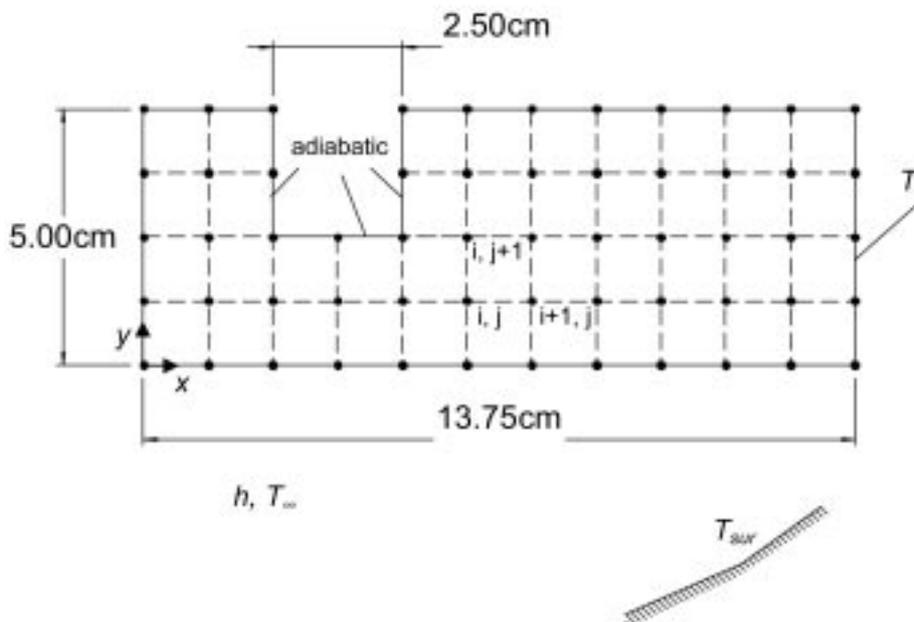


Fig. 1. Wiring harness bracket geometry and boundary conditions. The computational grid is superimposed on the geometry. For this grid, $\Delta x = \Delta y = 1.25$ cm.

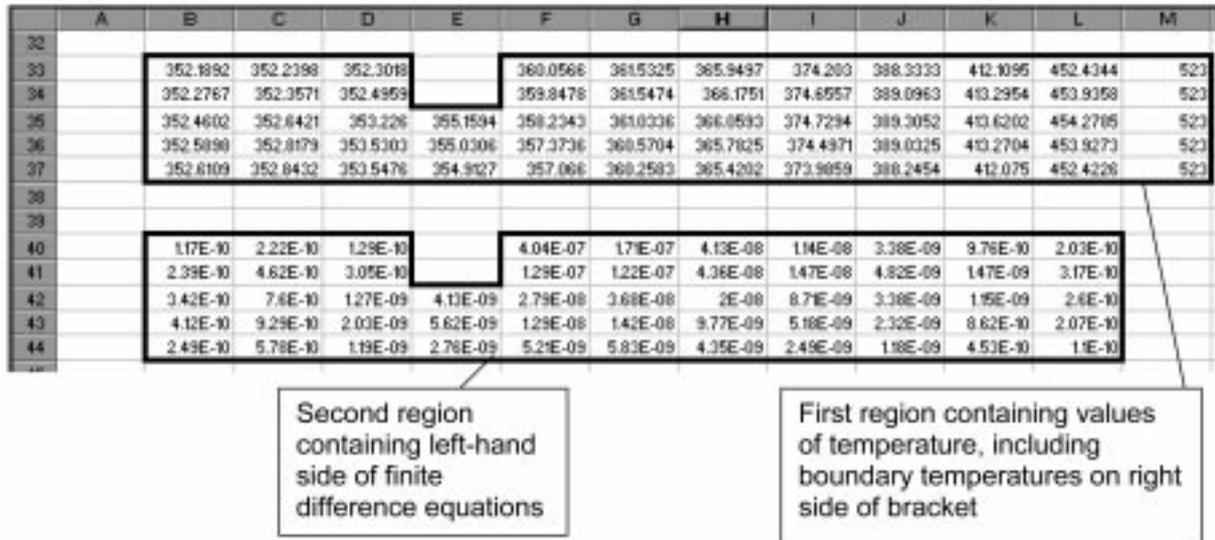


Fig. 2. The two regions on the spreadsheet. The first region contains the temperature values Solver iterated on to satisfy the finite difference equations in the second region.

spreadsheet are identified: one for the values of temperature (which contains initial guesses) and one for the finite difference equations. See Fig. 2. The left-hand sides of the finite difference equations are entered in the second region in a straightforward manner referring to cells containing guesses in the first region. Note that no manipulation of the finite difference equations is necessary before entering as cell formulas; the left-hand sides of Equations (1) and (2) (and similar equations for other edges and corners) are entered exactly as shown. Only one equation need be entered for an internal node; it can subsequently be copied to all other cells containing equations for internal nodes with the relative reference mode of Excel automatically taking care of cell reference adjustment. Similarly, the equation for only one node on an edge needs to be entered; it can subsequently be copied to all other nodes on that edge.

Solver is then chosen from the 'Tools' menu, and

parameters are set as shown in Fig. 3. Note that the 'Set Target Cell' box is left empty; all the finite difference equations are satisfied in the 'Subject to the Constraints' area. When the 'Solve' button is pushed, Solver changes the values in the first region until the constraints are satisfied, ending with a 'Solver found a solution' message.

The solution is seen in Fig. 2. The temperature distribution varies only slightly in the *y*-direction, demonstrating the validity of the one-dimensional assumption for extended surface problems. The solution also indicates that the notch temperatures are within the operating limit of the harness.

The use of Solver for conduction problems provides students with an intuitive method for obtaining finite difference solutions. Its advantages over the circular reference approach become even more apparent when the staggered grid formulation for the Navier-Stokes equations is utilized, as described in the next section.

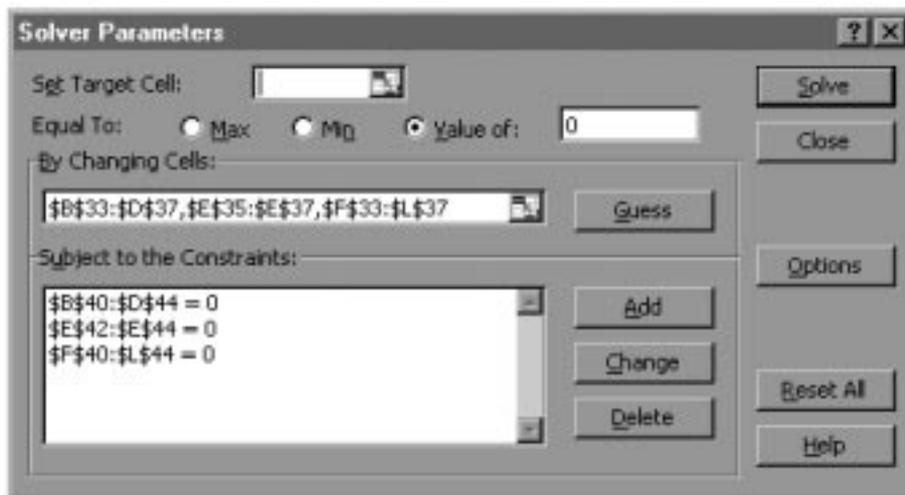


Fig. 3. The Solver tool menu.

SOLUTION OF THE INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

The steady two-dimensional constant property Navier-Stokes equations can be written as follows:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad \text{continuity} \quad (3)$$

$$\begin{aligned} & Re \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) \\ &= -Re \frac{\partial p}{\partial x} + \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \quad \text{x-momentum} \quad (4) \end{aligned}$$

$$\begin{aligned} & Re \left(u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right) \\ &= -Re \frac{\partial p}{\partial y} + \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \quad \text{y-momentum} \quad (5) \end{aligned}$$

where all variables are dimensionless. The Reynolds number is:

$$Re = \frac{\rho VL}{\mu}$$

where ρ is fluid density, V is a characteristic speed, L is a characteristic length, and μ is fluid viscosity.

Students in a graduate-level course in Computational Fluid Dynamics & Heat Transfer were assigned the driven cavity problem. The geometry is illustrated in Fig. 4.

The boundary conditions are:

$$\begin{aligned} u(x, 1) = 1; u(x, 0) = u(0, y) = u(1, y) \\ = v(x, 1) = v(x, 0) = v(0, y) = v(1, y) = 0 \end{aligned}$$

A condition for pressure must also be specified to result in a determined system; it is sufficient to specify pressure at a point, for instance. For iterative solutions of the system of equations resulting from discretization, the specification of pressure at a point is implicitly done through initial guesses for variables at all grid points.

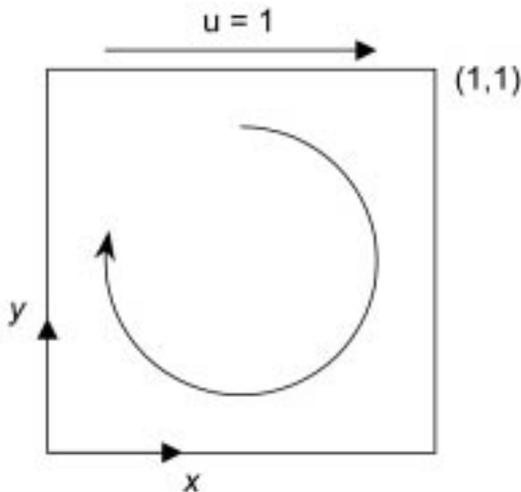


Fig. 4. The driven cavity.

A problem arises in the solution of the incompressible Navier-Stokes equations because there is no equation for pressure. If the equations are discretized in a conventional manner—using central differencing, for instance—and each resulting discretized equation is applied at every interior node, the matrix resulting from the system of equations is singular (in spite of the singularity, numerical solutions for reasonable velocities can be obtained, but the pressure exhibits wild oscillations in a checkerboard pattern; see, for example, Patankar [9]). Several techniques have been developed to deal with this problem. One solution is to recast the governing equations into the vorticity-stream function formulation [6]. This technique, however, has the disadvantage of having to post-process the results to obtain velocities and pressure. Other methods include the artificial compressibility or pressure correction methods [10].

A relatively straightforward solution that avoids the matrix singularity is the staggered grid [9]. The staggered grid, shown in Fig. 5, consists of three sets of grid points. The dependent variables u , v , and p are defined at different points. The continuity equation is applied at the p nodes, the x -momentum equation is applied at the u nodes, and the y -momentum equation is applied at the v nodes.

Referring to Fig. 5, the finite difference formulation of the Navier-Stokes equations becomes:

$$\frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y} = 0 \quad (6)$$

$$\begin{aligned} & Re \left(u_{i+1/2,j} \frac{u_{i+3/2,j} - u_{i-1/2,j}}{2\Delta x} \right. \\ & \left. + \bar{v}_{i+1/2,j} \frac{u_{i+1/2,j+1} - u_{i+1/2,j-1}}{2\Delta y} \right) \\ & + Re \left(\frac{p_{i+1,j} - p_{i,j}}{\Delta x} \right) \\ & - \left(\frac{u_{i+3/2,j} - 2u_{i+1/2,j} + u_{i-1/2,j}}{\Delta x^2} \right. \\ & \left. + \frac{u_{i+1/2,j+1} - 2u_{i+1/2,j} + u_{i+1/2,j-1}}{\Delta y^2} \right) = 0 \quad (7) \end{aligned}$$

$$\begin{aligned} & Re \left(\bar{u}_{i,j+1/2} \frac{v_{i+1,j+1/2} - v_{i-1,j+1/2}}{2\Delta x} \right. \\ & \left. + v_{i,j+1/2} \frac{v_{i,j+3/2} - v_{i,j-1/2}}{2\Delta y} \right) \\ & + Re \left(\frac{p_{i,j+1} - p_{i,j}}{\Delta y} \right) \\ & - \left(\frac{v_{i,j+3/2} - 2v_{i,j+1/2} + v_{i,j-1/2}}{\Delta y^2} \right. \\ & \left. + \frac{v_{i+1,j+1/2} - 2v_{i,j+1/2} + v_{i-1,j+1/2}}{\Delta x^2} \right) = 0 \quad (8) \end{aligned}$$

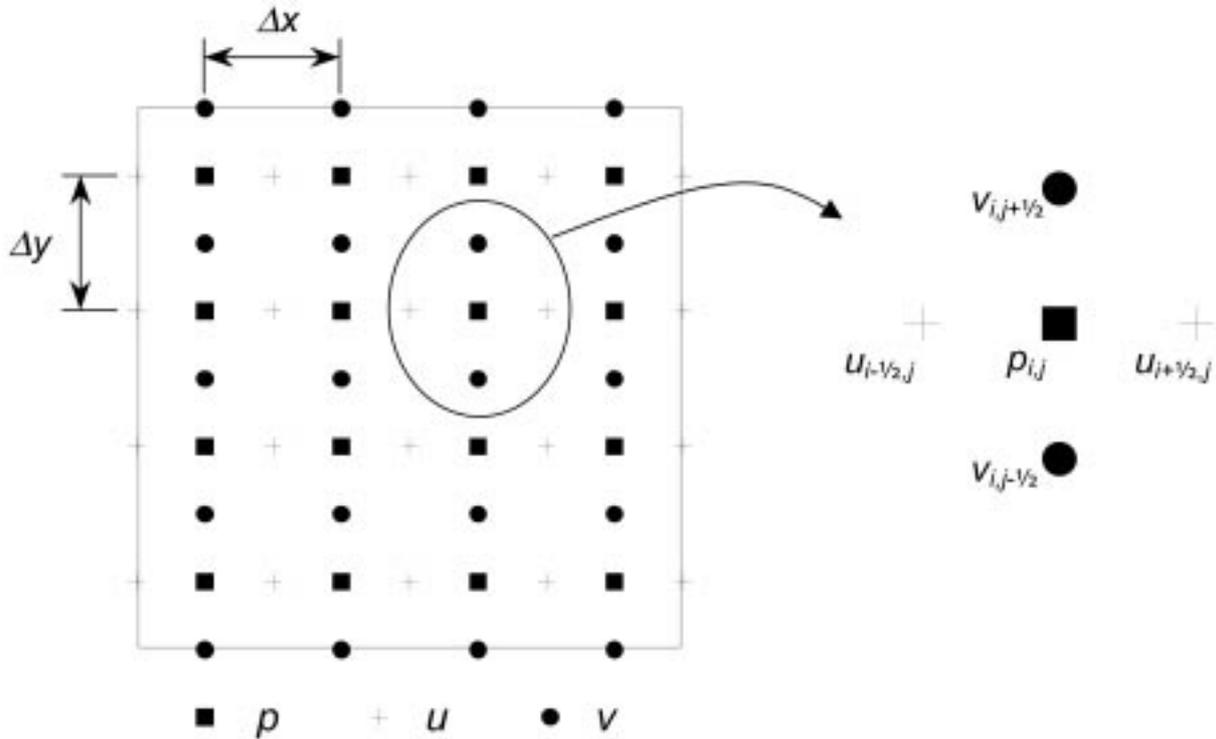


Fig. 5. The staggered grid.

The average velocities and for the case $\Delta x = \Delta y$ are defined as:

$$\bar{u}_{i,j+1/2} = \frac{u_{i+1/2,j} + u_{i+1/2,j+1} + u_{i-1/2,j} + u_{i-1/2,j+1}}{4}$$

and

$$\bar{v}_{i+1/2,j} = \frac{v_{i,j+1/2} + v_{i+1,j+1/2} + v_{i,j-1/2} + v_{i+1,j-1/2}}{4}$$

Equations (6) through (8) utilize central differencing for approximating derivatives in the governing equations, with a corresponding truncation error of $O(\Delta x^2)$.

As seen from observation of Fig. 5, specification of boundary conditions for u at $y=0$ and $y=1$, and v at $x=0$ and $x=1$ presents a problem because there are no grid points for those variables

on those boundaries. The problem is solved by imagining grid points outside the boundaries as shown in Fig. 6. Setting the average of $u_{i,N+1}$ and $u_{i,N}$ equal to the speed of the wall, V_{wall} , leads to the boundary condition $u_{i,N+1} = 2V_{wall} - u_{i,N}$, where V_{wall} is 1 for the top and 0 for every other boundary. Similar conditions are set for v at the left and right boundaries.

Although conceptually elegant, the staggered grid formulation can be an indexing nightmare for the programmer. This is where the spreadsheet solution becomes pedagogically useful. Students can set up the staggered grid in a visibly clear fashion on the spreadsheet, and use cell formatting to distinguish between u , v , and p grid points. Once the finite difference equations have been entered for one u , v , and p grid point, they can be simply copied to the remaining interior points.

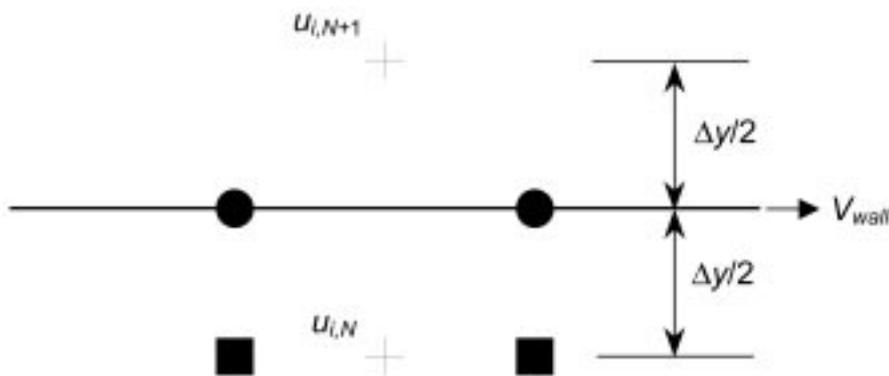


Fig. 6. Treatment of boundary velocities.

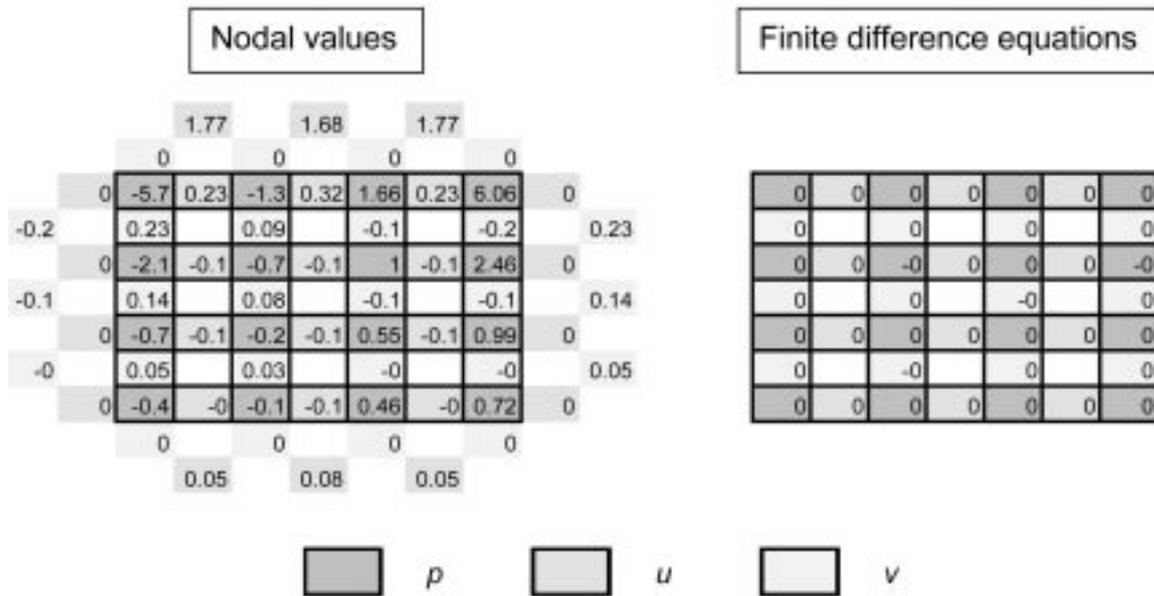


Fig. 7. The two spreadsheet regions for the driven cavity problem. Cell formatting has been used to distinguish pressure and velocity cells. In the region on the left, cells outside the bold borders contain boundary conditions.

Consideration of Equations (6) through (8) will indicate why the spreadsheet circular reference function will not work for the discretization given above. For the circular reference function to work, an equation for the variable at each node must exist. If one were to solve Equation (7) for $u_{i+1/2,j}$ and Equation (8) for $v_{i,j+1/2}$, that would leave Equation (6) for the node corresponding to $p_{i,j}$. Equation (6), however, does not contain pressure! This quandary brings out the essence of the pressure problem for solution of the incompressible flow equations. Matrix inversion could be used, but that requires additional manipulation of the finite difference equations in order to fill the operator matrix. The Solver function is an excellent tool that allows direct solution for the unknowns without muddying the waters with complicated preprocessing.

As with the heat conduction problem, two spreadsheet regions of identical shape are set up. Fig. 7 shows a small grid for illustrative purposes. The region on the left contains the boundary conditions and initially the guesses; the region on the right contains the left-hand side of the finite difference equations. The left side of Equation (6) is written in the *p*-cells, the left side of Equation (7) is written in the *u*-cells, and the left side of Equation (8) is written in the *v*-cells. Any given equation in the region on the right refers to corresponding cells containing values for *u*, *v*, and *p* in the region on the left. Note that Equations (6) through (8) need only be typed in once; afterwards the three equations can simply be copied to the remaining cells, the relative cell reference feature in Excel taking care of adjusting the cell references accordingly. After entering initial guesses (say, every unknown equal to zero), Solver is set up to change the cells corresponding to

interior grid points in the left grid subject to the constraints that the cells containing formulas in the right grid are equal to zero. When Solver has completed its iterations, the grid on the left contains the solution and the grid on the right is filled with zeroes.

The solution algorithm for Solver is sensitive to initial guesses, consequently occasionally Solver cannot find a solution and different initial guesses must be tried. To obtain a solution for non-zero Reynolds numbers, for instance, it may be necessary to use the solution for $Re = 0$ as an initial guess.

Pressure contours are easily plotted by forming another region of the spreadsheet where each cell refers back to a corresponding pressure cell, and adding a row and column to identify the corresponding *x*- and *y*-values. Likewise, streamlines can be plotted by developing another region of the spreadsheet to incorporate the following formulation.

Derivatives of the stream function are defined as follows:

$$\frac{\partial \psi}{\partial x} = -v; \quad \frac{\partial \psi}{\partial y} = u.$$

Integration of the stream function in the *x*- and *y*-directions thus leads to: $\psi_2 - \psi_1 = -\bar{v}\Delta x$ in the *x*-direction and $\psi_2 - \psi_1 = \bar{u}\Delta y$ in the *y*-direction, where \bar{u} and \bar{v} are suitable average quantities. For current purposes, the stream function is found at the pressure grid points as follows (refer to Fig. 8):

$$\begin{aligned} \psi_{i+1,j} &= \psi_{i,j} - \bar{v}_{i+1/2,j}\Delta x; \\ \psi_{i,j+1} &= \psi_{i,j} + \bar{u}_{i,j+1/2}\Delta y \end{aligned}$$

where \bar{u} and \bar{v} were defined earlier.

The version of Excel used for these results treats the *x* and *y* values in the contour plots as

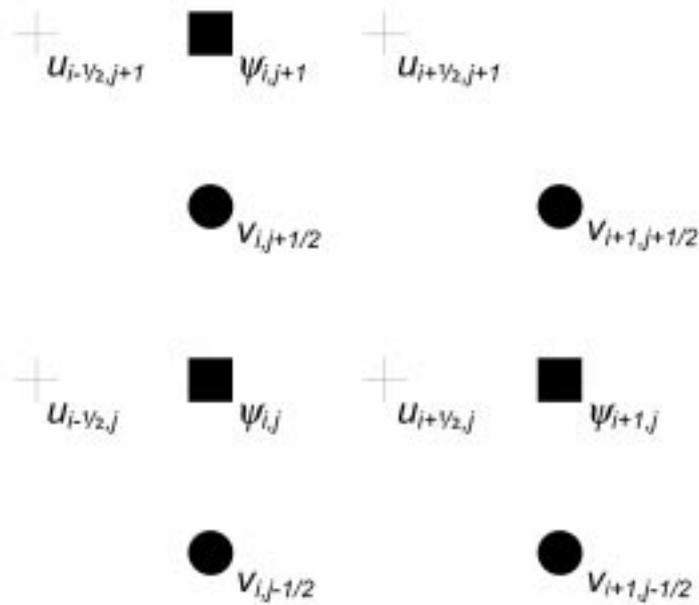


Fig. 8. Grid showing stream function values at pressure grid points and relevant velocity values.

categories, therefore irregularities in grid spacing (as happens when values of ψ on the boundaries are found by using adjacent interior grid points) will not be apparent. Since the spreadsheet results are simply for qualitative analysis, this drawback is not serious.

Because the pressure and stream function regions refer back to the solution, the contour

plots are updated automatically as new solutions are obtained, resulting in the ability to immediately view the results.

A drawback in the Solver function is a relatively low limit on the number of constraints and changing cells. The Solver function in the version of Excel used to calculate results presented here could not handle more than 100 unknowns. For the

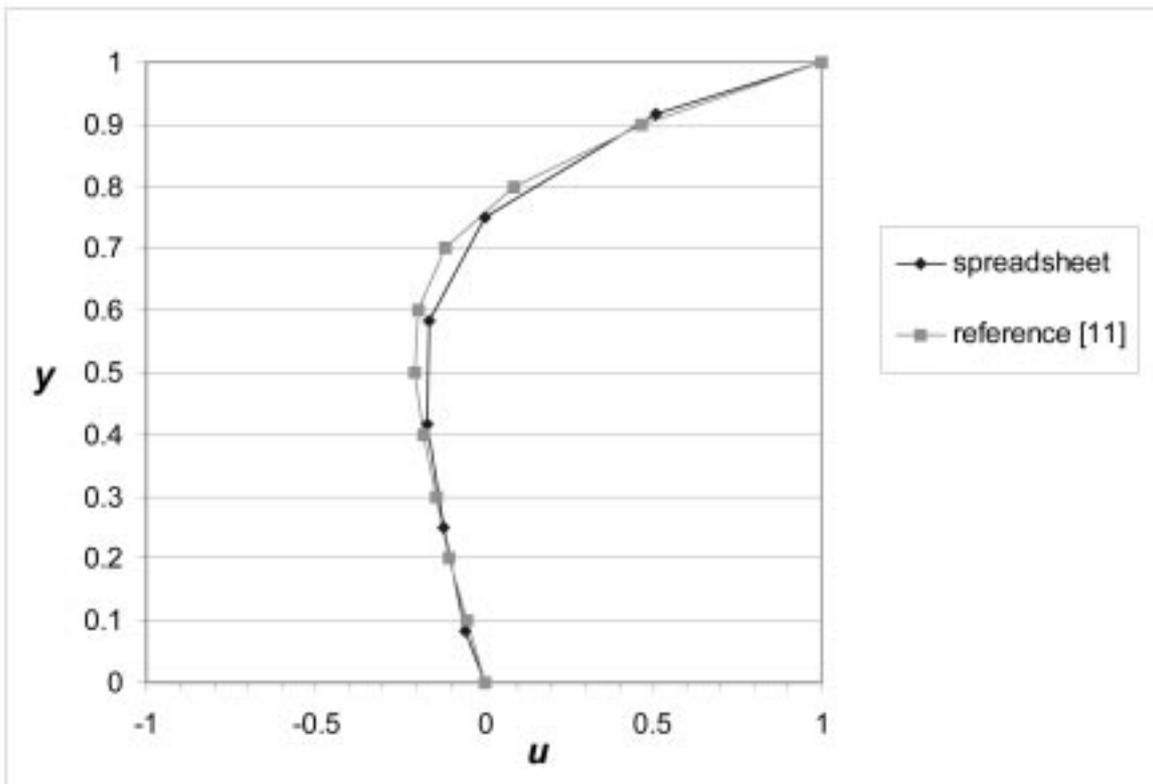


Fig. 9. Comparison of spreadsheet results and results from reference [11] for $Re=0$. The figure shows u along the cavity vertical centerline.

results shown here, the number of grid points for pressure numbered 6×6 , for u 5×6 , and for v 6×5 , for a total of 96 unknowns. This coarse grid is unsuitable for obtaining accurate results, especially considering the strong solution singularities in the upper corners. As mentioned earlier, the usefulness of the method lies in its visual presentation and straightforward solution process. The students, unencumbered by programming details, can develop a clear conceptual understanding of how the grid variables relate to one another before moving on to writing code.

Figure 9 shows how u varies along the cavity centerline at $x=0.5$ for $Re=0$. Note that for $Re=0$ a viscous pressure scaling is used so that Re in front of the pressure gradient terms in Equations (4) and (5) is replaced by unity. The figure compares the results using Solver to those obtained by a highly accurate spectral method as described in reference [11]. The agreement is remarkable considering the coarseness of the spreadsheet solution. Figure 10 shows the streamline plot which illustrates the central vortex and demonstrates the expected symmetry for $Re=0$.

Results for $Re=100$ are compared to those of Burggraf as reported in Goda [12] in Fig. 11. As expected with such a coarse grid, the results are not very close to those from Burggraf's finer grid, but are qualitatively reasonable. The streamline plot in Fig. 12 shows the expected shift of the central vortex to the right as described, for example, in Ghia *et al.* [13].

Ten students in the most recent class offering (Term I, 2003–04) responded to a questionnaire regarding the use of Solver to solve the driven cavity problem. As seen in Fig. 13, students felt the assignment improved their understanding of both the staggered grid and the relationship among the grid points for velocity components and pressure. Students were asked to generate x - y plots of u vs. y and v vs. x at the cavity centerlines, and their responses indicate they felt comfortable visualizing these results on the spreadsheet. Due to time constraints, students were not given an assignment to program a simple Navier-Stokes code, so the effectiveness of the spreadsheet assignment in improving the students' ability to write code could not be directly measured. Six of the ten respondents either disagreed with or were neutral about the statement 'I am good at programming'. This result suggests that the relatively low score for the statement 'Having performed this assignment, I feel I could program a Navier-Stokes code using a programming language' was due more to an insecurity about programming abilities than poor understanding of the finite difference concepts.

CONCLUSIONS

The Solver tool in Excel provides an intuitive way for students to learn the fundamentals of computational fluid dynamics and heat transfer. A spreadsheet solution allows students to

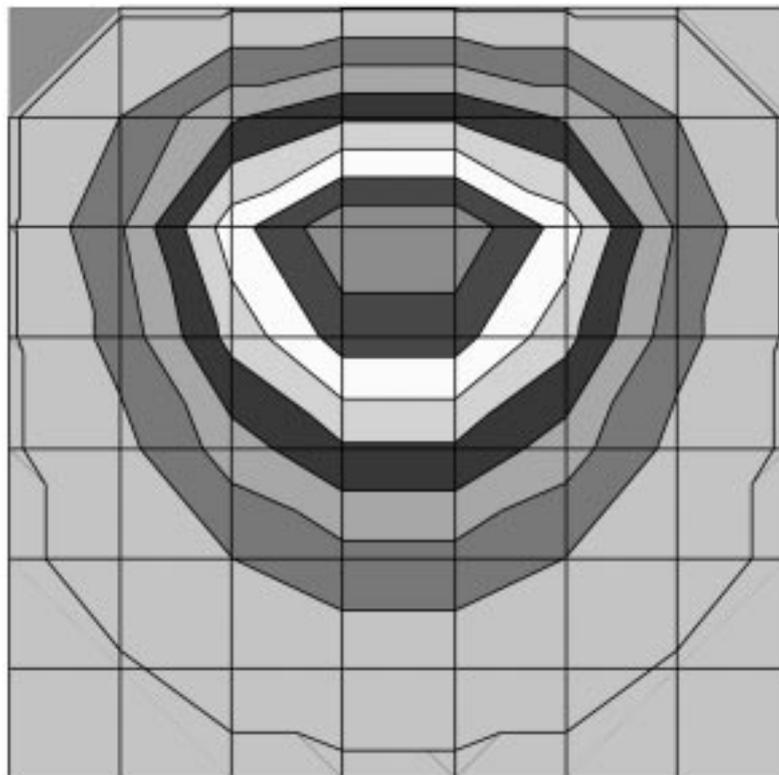


Fig. 10. Streamlines for $Re=0$.

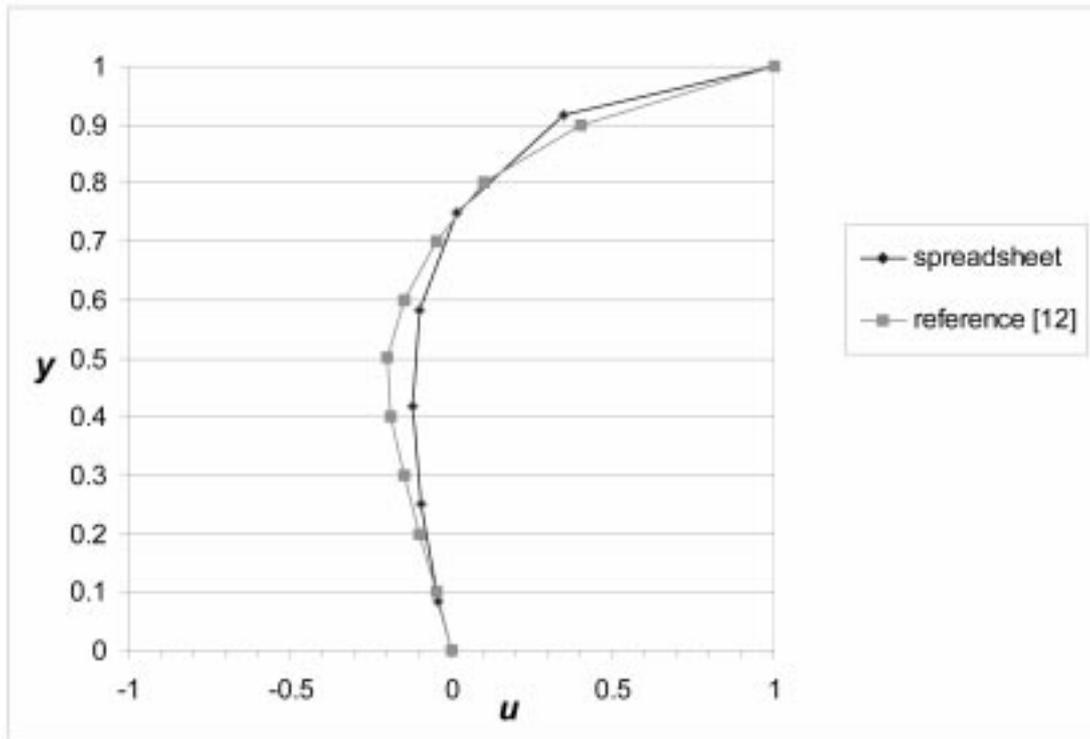


Fig. 11. Comparison of spreadsheet results and results from reference [12] for $Re = 100$. The figure shows u along the cavity vertical centerline.

clearly conceptualize the interrelationships among the nodal variables in the discretization without the burden of having to write and debug

code. The spreadsheet links between plots and cell values allows instant visualization of parametric changes.

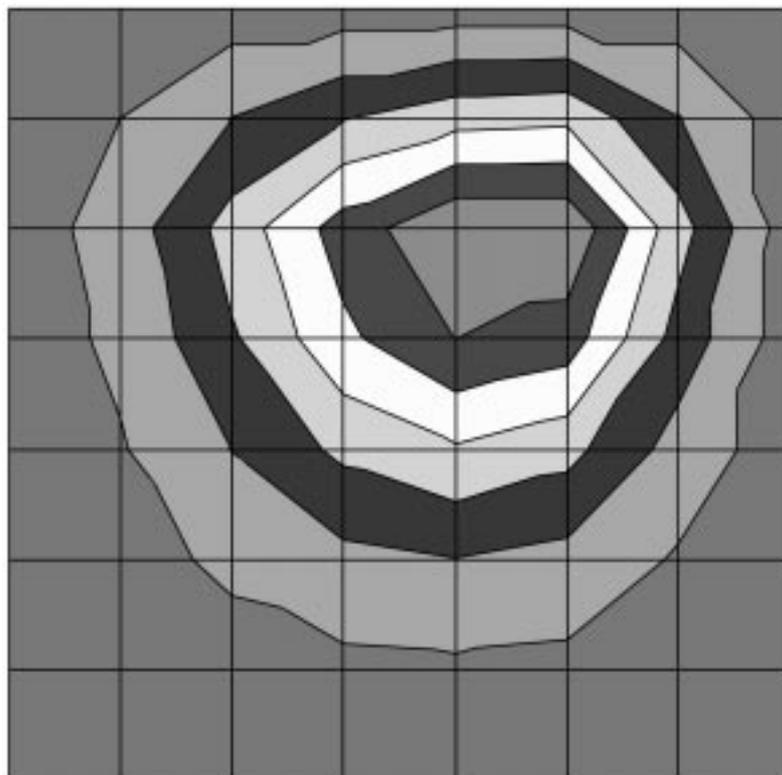


Fig. 12. Streamlines for $Re = 100$.

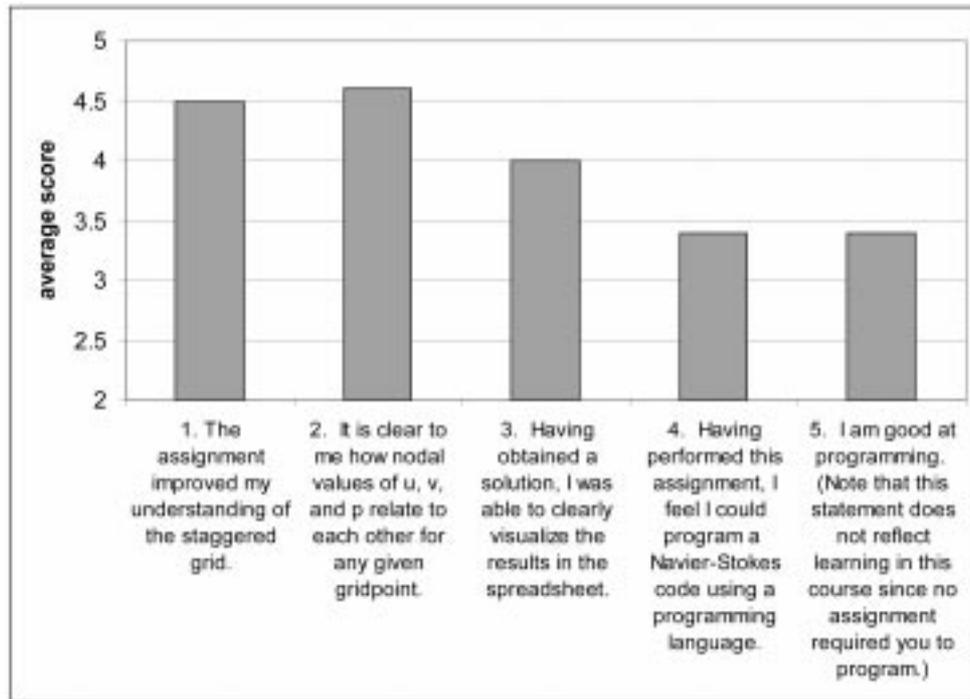


Fig. 13. Student assessment of using Solver to solve the driven cavity problem. Possible scores were 5 = strongly agree, 4 = agree, 3 = neutral, 2 = disagree, 1 = strongly disagree.

REFERENCES

1. E. M. A. Mokheimer and M. A. Antar, On the use of spreadsheets in heat conduction analysis, *Int. J. Mech. Eng. Educ.*, **28**(2), 1998, pp. 113–139.
2. M. A. Antar and E. M. Mokheimer, Spreadsheet modelling of transient three dimensional heat conduction with various standard boundary conditions, *Int. J. Mech. Eng. Educ.*, **30**(1), 2000, pp. 17–34.
3. M. R. Schumack, Teaching heat transfer using automotive-related case studies with a spreadsheet analysis package. *Int. J. Mech. Eng. Educ.*, **25**(3), 1997, pp. 177–196.
4. R. Kumar and A. Al-Shantaf, Spreadsheet analysis of fluid mechanics problems, *1997 ASEE/IEEE Frontiers in Education Conf. Proc.*
5. E. G. Wiggins, Computational fluid dynamics on a spreadsheet, *Computers in Education J.*, **7**(2), 1997, pp. 7–11.
6. E. Morishita, Spreadsheet fluid dynamics, *J. Aircraft*, **36**(4), 1999, pp. 720–723.
7. E. Morishita, Y. Iwata, K. Yuki and H. Yoshida, Spreadsheet fluid dynamics for aeronautical course problems, *Int. J. Eng. Educ.*, **17**(3), 2000, pp. 294–311.
8. F. P. Incropera and D. P. DeWitt, *Introduction to Heat Transfer*, 4th ed., John Wiley & Sons (2002), ch. 4.
9. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corp. (1980), ch. 6.
10. J. C. Tannehill, D. A. Anderson, and R. H. Pletcher, *Computational Fluid Mechanics and Heat Transfer*, 2nd ed., Taylor & Francis (1997), ch. 9, sect. 3.
11. M. R. Schumack, W. W. Schultz and J. P. Boyd, Spectral method solution of the Stokes Equations on nonstaggered grids, *J. Computational Physics*, **94**(1), 1991, pp. 30–58.
12. K. Goda, A Multistep Technique with implicit difference schemes for calculating two- or three-dimensional cavity flows, *J. Computational Physics*, **30**, 1979, pp. 76–95.
13. U. Ghia, K. N. Ghia, and C. T. Shin, High-Re solutions for incompressible flow using the Navier-Stokes Equations and a Multigrid Method, *J. Computational Physics*, **48**, 1982, pp. 387–411.

Mark Schumack is Associate Professor of Mechanical Engineering at the University of Detroit Mercy. He teaches courses in heat transfer, thermodynamics, fluid mechanics, and energy systems. His research interests lie in thermal/fluid modeling using computational techniques, including applications in the automotive and manufacturing fields. He has also developed instructional multimedia modules for manufacturing engineering candidates in the NSF-sponsored Greenfield Coalition for New Manufacturing Education. Dr. Schumack earned his BS, MS, and Ph.D. degrees in Mechanical Engineering from the University of Michigan.