INTRODUCTION TO COMPUTATIONAL FLUID DYNAMICS

PROPRIETARY AND CONFIDENTIAL

This Manual is the proprietary property of The McGraw-Hill Companies, Inc. (“McGraw-Hill”) and protected by copyright and other state and federal laws. By opening and using this Manual the user agrees to the following restrictions, and if the recipient does not agree to these restrictions, the Manual should be promptly returned unopened to McGraw-Hill: This Manual is being provided only to authorized professors and instructors for use in preparing for the classes using the affiliated textbook. No other use or distribution of this Manual is permitted. This Manual may not be sold and may not be distributed to or used by any student or other third party. No part of this Manual may be reproduced, displayed or distributed in any form or by any means, electronic or otherwise, without the prior written permission of McGraw-Hill.
Fundamentals, Grid Generation, and Boundary Conditions

15-1C

Solution   We are to list the unknowns and the equations for a given flow situation.

Analysis   There are only three unknowns in this problem, $u$, $v$, and $P$ (or $P'$). Thus, we require three equations: continuity, $x$ momentum (or $x$ component of Navier-Stokes), and $y$ momentum (or $y$ component of Navier-Stokes). These equations, when combined with the appropriate boundary conditions, are sufficient to solve the problem.

Discussion   The actual equations to be solved by the computer are discretized versions of the differential equations.

15-2C

Solution   We are to define several terms or phrases and provide examples.

Analysis   
(a) A computational domain is a region in space (either 2-D or 3-D) in which the numerical equations of fluid flow are solved by CFD. The computational domain is bounded by edges (2-D) or faces (3-D) on which boundary conditions are applied.
(b) A mesh is generated by dividing the computational domain into tiny cells. The numerical equations are then solved in each cell of the mesh. A mesh is also called a grid.
(c) A transport equation is a differential equation representing how some property is transported through a flow field. The transport equations of fluid mechanics are conservation equations. For example, the continuity equation is a differential equation representing the transport of mass, and also conservation of mass. The Navier-Stokes equation is a differential equation representing the transport of linear momentum, and also conservation of linear momentum.
(d) Equations are said to be coupled when at least one of the variables (unknowns) appears in more than one equation. In other words, the equations cannot be solved alone, but must be solved simultaneously with each other. This is the case with fluid mechanics since each component of velocity, for example, appears in the continuity equation and in all three components of the Navier-Stokes equation.

Discussion   Students' definitions should be in their own words.

15-3C

Solution   We are to discuss the difference between nodes and intervals and analyze a given computational domain in terms of nodes and intervals.

Analysis   Nodes are points along an edge of a computational domain that represent the vertices of cells. In other words, they are the points where corners of the cells meet. Intervals, on the other hand, are short line segments between nodes. Intervals represent the small edges of cells themselves. In Fig. P15-3 there are 6 nodes and 5 intervals on the top and bottom edges. There are 5 nodes and 4 intervals on the left and right edges.

Discussion   We can extend the node and interval concept to three dimensions.
15-4C  
**Solution**  
For a given computational domain with specified nodes and intervals we are to compare a structured grid and an unstructured grid and discuss.

**Analysis**  
We construct the two grids in the figure: (a) structured, and (b) unstructured.

There are $5 \times 4 = 20$ cells in the structured grid, and there are 36 cells in the unstructured grid.

**Discussion**  
Depending on how individual students construct their unstructured grid, the shape, size, and number of cells may differ considerably.

15-5C  
**Solution**  
For a given computational domain with specified nodes and intervals we are to compare a structured mesh and a polyhedral mesh and discuss.

**Analysis**  
We construct the two grids in the figure: (a) structured, and (b) unstructured polyhedral. We show two other options in (c) and (d). There are many possible answers for the polyhedral mesh, depending on how large you want your cells to be.

There are $5 \times 4 = 20$ cells in the structured grid. There are 22 cells in polyhedral grid (b). There are some cells with 3 sides, 4 sides, and 5 sides, as required. Compared to the triangular mesh with 36 cells, we have reduced the cell count considerably. In (c) and (d), there are 21 cells and 18 cells respectively. In case (d) we have reduced the cell count below that of even the structured grid. In that case, 3 of the cells have 6 sides each. The cell reduction is particularly useful in large 3-D problems where CPU time and computer memory are important limitations.

**Discussion**  
Note that the node distribution along the boundaries is identical in each case, but we have great flexibility in how we create the grid. Depending on how individual students construct their unstructured grid, the shape, size, and number of cells may differ considerably.
Chapter 15 Computational Fluid Dynamics

15-6C
Solution  We are to summarize the eight steps involved in a typical CFD analysis.

Analysis  We list the steps in the order presented in this chapter:

1. Specify a computational domain and generate a grid.
2. Specify boundary conditions on all edges or faces.
3. Specify the type of fluid and its properties.
4. Specify numerical parameters and solution algorithms.
5. Apply initial conditions as a starting point for the iteration.
6. Iterate towards a solution.
7. After convergence, analyze the results (post processing).
8. Calculate global and integral properties as needed.

Discussion  The order of some of the steps is interchangeable, particularly Steps 2 through 5.

15-7C
Solution  We are to explain why the cylinder should not be centered horizontally in the computational domain.

Analysis  Flow separates over bluff bodies, generating a wake with reverse flow and eddies downstream of the body. There are no such problems upstream. Hence it is always wise to extend the downstream portion of the domain as far as necessary to avoid reverse flow problems at the outlet boundary.

Discussion  The same problems arise at the outlet of ducts and pipes – sometimes we need to extend the duct to avoid reverse flow at the outlet boundary.

15-8C
Solution  We are to discuss the significance of several items with respect to iteration.

Analysis  
(a) In a CFD solution, we typically iterate towards a solution. In order to get started, we make some initial conditions for all the variables (unknowns) in the problem. These initial conditions are wrong, of course, but they are necessary as a starting point. Then we begin the iteration process, eventually obtaining the solution.
(b) A residual is a measure of how much our variables differ from the “exact” solution. We construct a residual by putting all the terms of a transport equation on one side, so that the terms all add to zero if the solution is correct. As we iterate, the terms will not add up to zero, and the remainder is called the residual. As the CFD solution iterates further, the residual should (hopefully) decrease.
(c) Iteration is the numerical process of marching towards a final solution, beginning with initial conditions, and progressively correcting the solution. As the iteration proceeds, the variables converge to their final solution as the residuals decrease.
(d) Once the CFD solution has converged, post processing is performed on the solution. Examples include plotting velocity and pressure fields, calculating global properties, generating other flow quantities like vorticity, etc. Post processing is performed after the CFD solution has been found, and does not change the results. Post processing is generally not as CPU intensive as the iterative process itself.

Discussion  We have assumed steady flow in the above discussions.
15-9C
Solution We are to discuss how the iteration process is made faster.

Analysis
(a) With multigrid, solutions of the equations of motion are obtained on a coarse grid first, followed by successively finer grids. This speeds up convergence because the gross features of the flow are quickly established on the coarse grid, and then the iteration process on the finer grid requires less time.

(b) In some CFD codes, a steady flow is treated as though it were an unsteady flow. Then, an artificial time is used to march the solution in time. Since the solution is steady, however, the solution approaches the steady-state solution as “time” marches on. In some cases, this technique yields faster convergence.

Discussion There are other “tricks” to speed up the iteration process, but CFD solutions often take a long time to converge.

15-10C
Solution We are to list the boundary conditions that are applicable to a given edge, and we are to explain why other boundary conditions are not applicable.

Analysis We may apply the following boundary conditions: outflow, pressure inlet, pressure outlet, symmetry (to be discussed), velocity inlet, and wall. The curved edge cannot be an axis because an axis must be a straight line. The edge cannot be a fan or interior because such edges cannot be at the outer boundary of a computational domain. Finally, the edge cannot be periodic since there is no other edge along the boundary of the computational domain that is of identical shape (a periodic boundary must have a “partner”). The symmetry boundary condition merits further discussion. Numerically, gradients of flow variables in the direction normal to a symmetry boundary condition are set to zero, and there is no mathematical reason why the curved right edge of the present computational domain cannot be set as symmetry. However, you would be hard pressed to think of a physical situation in which a curved edge like that of Fig. P15-9 would be a valid symmetry boundary condition.

Discussion Just because you can set a boundary condition and generate a CFD result does not guarantee that the result is physically meaningful.

15-11C
Solution We are to discuss the standard method to test for adequate grid resolution.

Analysis The standard method to test for adequate grid resolution is to increase the resolution (by a factor of 2 in all directions if feasible) and repeat the simulation. If the results do not change appreciably, the original grid is deemed adequate. If, on the other hand, there are significant differences between the two solutions, the original grid is likely of inadequate resolution. In such a case, an even finer grid should be tried until the grid is adequately resolved.

Discussion Keep in mind that if the boundary conditions are not specified properly, or if the chosen turbulence model is not appropriate for the flow being simulated by CFD, no amount of grid refinement is going to make the solution more physically correct.
**Chapter 15 Computational Fluid Dynamics**

15-12C

**Solution** We are to discuss the difference between a pressure inlet boundary condition and a velocity inlet boundary condition, and we are to explain why both pressure and velocity cannot be specified on the same boundary.

**Analysis** At a pressure inlet we specify the pressure but not the velocity. At a velocity inlet we specify the opposite — velocity but not pressure. To specify both pressure and velocity would lead to mathematical over-specification, since pressure and velocity are coupled in the equations of motion. When pressure is specified at a pressure inlet (or outlet), the CFD code automatically adjusts the velocity at that boundary. In a similar manner, when velocity is specified at a velocity inlet, the CFD code adjusts the pressure at that boundary.

**Discussion** Since pressure and velocity are coupled, specification of both at a boundary would lead to inconsistencies in the equations of motion at that boundary.

15-13C

**Solution** We are to label all the boundary conditions to be applied to a computational domain.

**Analysis** The inlet is a velocity inlet. The outlet is a pressure outlet. All other edges that define the outer limits of the computational domain are walls. Finally, there are three edges that must be specified as interior. These are all labeled in the figure below.

**Discussion** It is critical that each boundary condition be specified carefully. Otherwise the CFD solution will not be correct.

15-14C

**Solution** We are to analyze what will happen to inlet pressure and outlet velocity when a fan is turned on in the computational domain of the previous problem.

**Analysis** Since the fan helps to push air through the channel, the inlet pressure will adjust itself so that less inlet pressure is required. In other words, the inlet pressure will decrease when the fan is turned on. Since the inlet velocity is the same in both cases, the mass flow rate (and volume flow rate since the flow is incompressible) must remain the same for either case. Therefore, outlet velocity will not change.

**Discussion** It may seem at first glance that \( V_{out} \) should increase because of the fan, but in order to conserve mass, the outlet velocity cannot change. The solution is constrained by the specified inlet boundary condition. In a real physical experiment, there is no such restriction. The fan would cause the inlet pressure to decrease, the inlet velocity to increase, and the outlet velocity to increase.
We are to list and briefly describe six boundary conditions, and we are to give an example of each.

In the chapter we list ten, so any six of these will suffice:

- **Axis**: Used in axisymmetric flows as the axis of rotation. Example: the axis of a torpedo-shaped body.
- **Fan**: An internal edge (2-D) or face (3-D) across which a sudden pressure rise is specified. Example: an axial flow fan in a duct.
- **Interior**: An internal edge (2-D) or face (3-D) across which nothing special happens – the interior boundary condition is used at the interface between two blocks. Example: all of the multiblock problems in this chapter, which require this boundary condition at the interface between any two blocks.
- **Outflow**: An outlet boundary condition in which the gradient of fluid properties is zero normal to the outflow boundary. Outflow is typically useful far away from the object or area of interest in a flow field. Example: the far field of flow over a body.
- **Periodic**: When the physical geometry has periodicity, the periodic boundary condition is used to specify that whatever passes through one face of the periodic pair must simultaneously enter the other face of the periodic pair. Example: in a heat exchanger where there are several rows of tubes.
- **Pressure inlet**: An inflow boundary in which pressure (but not velocity) is known and specified across the face. Example: the high pressure settling chamber of a blow-down wind tunnel facility.
- **Pressure outlet**: An outflow boundary in which pressure (but not velocity) is known and specified across the face. Example: the outlet of a pipe exposed to atmospheric pressure.
- **Symmetry**: A face over which the gradients of all flow variables are set to zero normal to the face – the result is a mirror image across the symmetry plane. Fluid cannot flow through a symmetry plane. Example: the mid-plane of flow over a circular cylinder in which the lower half is a mirror image of the upper half.
- **Velocity inlet**: An inflow boundary condition in which velocity (but not pressure) is known and specified across the face. Example: a uniform freestream inlet flow entering a computational domain from one side.
- **Wall**: A boundary through which fluid cannot pass and at which the no-slip condition (or a shear stress condition) is applied. Example: the surface of an airfoil that is being modeled by CFD.

There are additional boundary conditions used in CFD calculations, but these are the only ones discussed in this chapter.
**Solution** We are to sketch a structured and an unstructured grid near the airfoil surface, and discuss advantages and disadvantages of each.

**Analysis** In either case it is wise to cluster cells close to the airfoil surface since we expect that a thin boundary layer will exist along the surface, and we need many tiny cells within the boundary layer to adequately resolve it. Some simple, coarse meshes are drawn in Fig. 1. We would certainly want much higher resolution for CFD calculations.

**FIGURE 1**
A coarse structured (a) and unstructured (b) grid. Notice that the cells are clustered (more fine) near the surface of the airfoil since there is likely to be large velocity gradients there (in the boundary layer).

The structured grid in Fig 1a is called a C-grid since it wraps around the airfoil like the letter “C". The main advantage of the structured grid is that we can get high resolution near the surface with few cells. The main advantage of the unstructured grid is that it is somewhat easier to generate when the geometry is complicated (especially for highly curved surfaces). Furthermore, it is easier to transition between curved and straight edges with an unstructured grid. The main disadvantage of an unstructured grid is that more cells are required for the same spatial resolution.

**Discussion** There are numerous other ways to construct a grid around this airfoil.

---

**Solution** We are to sketch a hybrid grid around an airfoil and explain its advantages.

**Analysis** We sketch a hybrid grid in the figure. Note that the grid is structured near the airfoil surface, but unstructured beyond the surface. **The advantage of a hybrid grid is that it combines the advantages of both structured and unstructured grids.** Near surfaces we can use a structured grid to finely resolve the boundary layer with a minimum number of cells, and away from surfaces we can use an unstructured grid so that we can rapidly expand the cell size. We can also more easily blend the grid into the edges of the computational domain with an unstructured grid.

**Discussion** A structured grid is generally the best choice, but a hybrid grid is often a better option than a fully unstructured grid.
Solution  We are to sketch the blocking for a structured grid, sketch a coarse grid, and label all the boundary conditions to be applied to the computational domain.

Analysis  First of all, we recognize that because of symmetry, we can split the domain in half vertically. We construct four blocks around the half-cylinder to transform from round to square, and then we add simple rectangular blocks upstream and downstream of the cylinder (Fig. 1). There is a total of six blocks.

With the block structure of Fig. 1 no cells are highly skewed, and cells are clustered near the cylinder wall and the upper wall of the duct as desired.

The bottom edge of the computational domain is a line of symmetry. The inlet is a velocity inlet. The outlet is a pressure outlet. The upper edge of the computational domain is a wall. The edges that define the cylinder are also walls. Finally, there are 5 edges that are specified as interior. These are all labeled in Fig. 2.

Discussion  There are alternative ways to set up the blocking topology. For example, at the top we may define a thin block (Block 7) that stretches across the entire horizontal domain so that the boundary layer on the top wall of the channel can be more adequately resolved (Fig. 3).
Solution  We are to sketch the blocking for a structured grid, sketch a coarse grid, and label all the boundary conditions to be applied to the computational domain.

Analysis  First of all, we recognize that because of symmetry, we can split the domain in half vertically. We construct four blocks inside the half-circle, and then we add blocks with one curved edge and three straight edges upstream and downstream of the cylinder (Fig. 1).

The setup of Fig. 1 contains six blocks. With this block structure, no cells are highly skewed, and cells are clustered near upper wall of the duct as desired. Cells are also clustered at the junctions between Blocks 1 and 2 and Blocks 4 and 6, where flow separation may occur.

The bottom edge of the computational domain is a line of symmetry. The inlet is a velocity inlet. The outlet is a pressure outlet. The upper edge of the computational domain is a wall. The edges that define the cylinder are also walls. Finally, there are 5 edges that are specified as interior. These are all labeled in Fig. 2.

Discussion  There are of course, alternative ways to set up the blocking topology.
**Chapter 15 Computational Fluid Dynamics**

**Solution** We are to modify an existing grid so that all blocks are elementary blocks. Then we are to verify that the total number of cells does not change.

**Analysis** The right edge of Block 2 of Fig. 15-11b is split twice to accommodate Block 1. We therefore split Block 2 into three separate elementary blocks. Unfortunately, this process ends up splitting Block 6 twice, which in turn splits Block 4 twice. **We end up with 12 elementary blocks** as shown in Fig. 1.

**FIGURE 1**
The blocking and coarse structured grid for a 2-D multiblock computational domain. Only elementary blocks are used in this grid.

We add up all the cells in these 12 blocks – we get a total of **464 cells**. This agrees with the total of 464 cells for the original 6 blocks in the domain.

**Discussion** Sometimes it is easier to create a grid with elementary blocks, even if the CFD code can accept blocks with split edges or faces.

**Solution** We are to modify an existing grid into a smaller number of non-elementary blocks, and we are to verify that the total number of cells does not change.

**Analysis** We combine Blocks 2, 3, 4, and 5 of Fig. 15-10b. Together, these produce one structured grid that wraps around the square in the middle – there are still 5 \( i \) intervals, but now there are 48 \( j \) intervals. **We end up with 3 non-elementary blocks**, as shown in Fig. 1.

**FIGURE 1**
The blocking and coarse structured grid for a 2-D multiblock computational domain. Only elementary blocks are used in this grid.

We add up all the cells in these 12 blocks – we get a total of **464 cells**. This agrees with the total of 464 cells for the original 6 blocks in the domain.

**Discussion** Block 2 in Fig. 1 is called an **O-grid** (for obvious reasons).
Solution  We are to generate a computational domain and label all appropriate boundary conditions for one stage of a new heat exchanger design.

Analysis  We take advantage of the periodicity of the geometry. There are several ways to create a periodic grid for this flow. The simplest computational domain consists of a single flow passage between two neighboring tubes. We can make the periodic edge intercept anywhere on the front portion of the tube that we desire. We choose the lower surface for convenience and simplicity. The periodic computational domain is sketched in Fig. 1.

Boundary conditions are also straightforward, and are labeled in Fig. 1. For a known inlet velocity we set the boundary condition at the left edge as a velocity inlet. The tube walls are obviously set as walls. The outlet can be set as either a pressure outlet or an outflow, depending on the provided information and how far the outlet region extends beyond the tubes. Finally, we set two pairs of translationally periodic boundaries, one fore and one aft of the tubes. We label them separately to avoid confusion.

Discussion  The fore and aft periodic edges are not horizontal in Fig. 1. This is not a problem since the periodic boundary condition is not restricted to horizontal or even to flat surfaces. An alternative, equally acceptable computational domain is shown in Fig. 2.
**Solution**

We are to sketch a structured multiblock grid with four-sided elementary blocks for a given computational domain.

**Analysis**

We choose the computational domain of Fig. 1 of the previous problem. Since all edges are straight, the blocking scheme can be rather simple. We sketch the blocking topology and apply a coarse mesh in Fig. 1 for the case in which the CFD code does not require the node distribution to be exactly the same on periodic pairs.

**FIGURE 1**

The blocking topology and a coarse structured grid for a periodic computational domain. This blocking topology applies to CFD codes that allow a block’s edges to be split for application of boundary conditions, and do not require periodic edge pairs to have identical node distributions.

Unfortunately, many CFD codes require that the node distribution on periodic pairs of edges be identical (the two edges of a periodic pair are “linked” in the grid generation process). In such a case, the grid of Fig. 1 would not be acceptable. Furthermore, although the edges of the blocks of Fig. 1 are not split with respect to adjacent blocks, the top edges of Block 1 and Block 3 are split with respect to the boundary conditions (part of the edge is periodic and part is a wall). Thus these blocks are not really elementary blocks after all. We construct a more elaborate blocking topology in Fig. 2 to correct these problems. The node distribution on the edges of each periodic pair are identical, at the expense of more complexity (7 instead of 5 blocks) and more cell skewness.

**FIGURE 2**

The blocking topology and a coarse structured grid for a periodic computational domain. This blocking topology applies to CFD codes that require elementary blocks and require periodic edge pairs to have identical node distributions.

**Discussion**

Some of the cells have moderate skewness with the blocking topology of Fig. 2, especially near the corners of Block 2 and Block 7 and throughout Block 5. A more complicated topology can be devised to reduce the amount of skewness.
15-24
Solution  We are to discuss why there is reverse flow in this CFD calculation, and then we are to explain what can be done to correct the problem.

Analysis  Reverse flow at an outlet is usually an indication that the computational domain is not large enough. In this case the rectangular heat exchanger tubes are inclined at 35°, and the flow will most certainly separate, leaving large recirculating eddies in the wakes. Anita should extend the computational domain in the horizontal direction downstream so that the eddies have a chance to “close” and the flow has a chance to re-develop into a flow without any reverse flow.

Discussion  In most commercial CFD codes a warning will pop up on the computer monitor whenever there is reverse flow at an outlet. This is usually an indication that the computational domain should be enlarged.

15-25
Solution  We are to generate a computational domain and label all appropriate boundary conditions for two stages of a heat exchanger.

Analysis  We look for the smallest computational domain that takes advantage of the periodicity of the geometry. There are several ways to create a periodic grid for this flow. The simplest computational domain consists of a single flow passage between two neighboring tubes. We can make the periodic edge intercept anywhere on the fore and aft portions of the heat exchanger that we desire. We choose the periodic computational domain sketched in Fig. 1.

Boundary conditions are also straightforward, and are labeled in Fig. 1. For a known inlet velocity we set the boundary condition at the left edge as a velocity inlet. The tube walls are obviously set as walls. The outlet can be set as either a pressure outlet or an outflow, depending on the provided information and how far the outlet region extends beyond the tubes. Finally, we set three pairs of translationally periodic boundaries, one fore, one mid, and one aft of the tubes. We label them separately to avoid confusion.

Discussion  Many other equally acceptable computational domains are possible.
We are to sketch a structured multiblock grid with four-sided elementary blocks for a given computational domain.

We choose the computational domain of Fig. 1 of the previous problem. We sketch one possible elementary blocking topology in Fig. 1 for the case in which the CFD code requires the node distribution to be exactly the same on periodic pairs. We also assume that we cannot split one periodic edge and not its partner. The blocks are numbered. This topology has 16 elementary blocks.

Figure 1
The elementary blocking topology for a periodic computational domain. This blocking topology applies to CFD codes that do not allow a block's edges to be split for application of boundary conditions, and requires periodic edge pairs to have identical node distributions.

Note that with this blocking topology we had to split the periodic “mid” boundary pair into two edges (the tops of blocks 6 and 7 and the bottoms of blocks 9 and 10). As long as both pairs of each segment are the same size and have the same number of nodes, this is not a problem. In the CFD code we would have to name each periodic pair separately, however. The block numbers are labeled. Notice that most of the blocks are nearly rectangular such that none of the computational cells would have to be highly skewed.

Discussion 
This seems like a rather complicated blocking topology. It would require a bit of work to generate the grid. However, the time spent on developing a good grid is usually well worth the effort. By reducing the amount of cell skewness, we are able to speed up the CFD calculations and obtain more accurate results. This kind of topology also enables us to cluster cells near walls and wakes as needed.
**Solution**  We are to generate CFD solutions for external flow over a 2-D block. Specifically, we are to compare drag coefficient for various values of $R/D$ (the extent of the outer boundary of the computational domain). In addition, we are to compare the calculated value of $C_D$ with experiment.

**Assumptions**  1. The flow is two-dimensional and incompressible. 2. The flow is symmetric about the $x$ axis. 3. The flow is turbulent, but steady in the mean.

**Properties**  The fluid is air with $\rho = 1.225 \text{ kg/m}^3$ and $\mu = 1.7894 \times 10^{-5} \text{ kg/m} \cdot \text{s}$.

**Analysis**  We run FlowLab using template `Block_domain`.

(a) The Reynolds number is

$$Re = \frac{\rho v D}{\mu} = \frac{\left(1.225 \text{ kg/m}^3\right)\left(2.0 \text{ m/s}\right)}{0.10 \text{ m}} = 1.7894 \times 10^4$$

Experimental data indicate that the drag coefficient for this body is $C_D \approx 1.9$ at Reynolds numbers greater than $10^4$ (see Chap. 11).

(b) The CFD code is run for eight values of $R$, all else being equal. $C_D$ is tabulated as a function of $R/D$ in Table 1, and plotted in Fig. 2. As the extent of the computational domain grows in size, the drag coefficient decreases steadily, but levels off to three significant digits of precision by $R/D \approx 200$. Thus, a computational domain extent of $R/D \approx 100$ is sufficient to achieve independence of $C_D$. We report a final value of $C_D = 1.34$.

(c) There are several possible reasons for the discrepancy between the calculated value of $C_D$ (1.34) and the experimentally obtained value of $C_D$ (about 1.9). First of all, the actual flow is most likely unsteady, with vortices being shed into the wake, whereas we are simulating a steady flow. In addition, the unsteady shedding of vortices renders the flow no longer symmetric about the $x$ axis, whereas we are forcing our flow to be symmetric. Furthermore, the grid resolution may not be adequate to achieve grid independence (this is checked in the following problem). Finally (and most importantly), we are using a turbulence model to simulate this flow field. The CFD solution we obtain is only as good as the degree to which the turbulence model correctly models the physics of the turbulence. As discussed in the text, no turbulence model is universally valid for all types of turbulent flows. Discrepancies between experiment and CFD will always exist regardless of how fine the grid or how large the extent of the computational domain.

(d) Streamlines near the body are plotted for $R/D = 5$ and 500 in Fig. 2. We notice that the streamlines for the $R/D = 5$ case are more tight around the body compared to those for $R/D = 500$. This is most likely due to interference from the outer edges of the computational domain, which are too close for the $R/D = 5$ case.

**TABLE 1**

<table>
<thead>
<tr>
<th>$R/D$</th>
<th>$C_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1.81927</td>
</tr>
<tr>
<td>10</td>
<td>1.50662</td>
</tr>
<tr>
<td>20</td>
<td>1.41076</td>
</tr>
<tr>
<td>50</td>
<td>1.36723</td>
</tr>
<tr>
<td>100</td>
<td>1.35282</td>
</tr>
<tr>
<td>200</td>
<td>1.34671</td>
</tr>
<tr>
<td>500</td>
<td>1.34408</td>
</tr>
</tbody>
</table>

**FIGURE 1**

Drag coefficient plotted as a function of the normalized extent of the computational domain for turbulent flow over a rectangular block.
**FIGURE 2**
Streamlines for steady, incompressible, two-dimensional, turbulent flow over a rectangular block at $R/D = (a) \ 5, (b) \ 500$. Only the upper half of the flow is simulated.

**Discussion**  
Newer versions of FlowLab may give slightly different results. It is perhaps surprising how far away the edges of the computational domain must be in order for them not to influence the flow field around the body. No attempt was made here to optimize the grid, and although the comparison between various grid extents is valid, the solution itself may not be grid independent; the actual value of drag coefficient may not be correct due to lack of grid resolution. Grid independence is analyzed in the next problem.
Solution  We are to test for grid independence by running various values of grid resolution, and we are to examine the effect of grid resolution on drag coefficient.

Assumptions  1 The flow is two-dimensional and incompressible. 2 The flow is symmetric about the x axis. 3 The flow is turbulent, but steady in the mean.

Properties  The fluid is air with \( \rho = 1.225 \text{ kg/m}^3 \) and \( \mu = 1.7894 \times 10^{-5} \text{ kg/m} \cdot \text{s} \).

Analysis  We run FlowLab using template Block_mesh. We use several grid resolutions, and we list drag coefficient as a function of total number of computational cells in Table 1. Although some discrepancies exist in the third or fourth digit, the drag coefficient levels off to three significant digits of precision by the sixth row of the table. Further grid refinement may be necessary. We report a final value of \( C_D = 1.33 \) (to three significant digits of precision for comparison with experimental results). This result differs from the experimentally obtained value of \( C_D = 1.9 \) by about 30%. We have achieved grid independence to three digits, and thereby eliminate lack of grid resolution as a source of the discrepancy. The other reasons for the discrepancy between CFD and experiment remain, regardless of how fine the grid. Namely, nonuniversality of the turbulence model, unsteadiness, and nonsymmetry.

We plot the results below – you can see how the drag coefficient levels off as the number of cells increases.

![Graph showing drag coefficient vs. number of cells](image)

Discussion  Newer versions of FlowLab may give slightly different results. This exercise illustrates that grid refinement does not necessarily lead to improved CFD predictions.

<table>
<thead>
<tr>
<th>Number of cells</th>
<th>( C_D )</th>
</tr>
</thead>
<tbody>
<tr>
<td>3120</td>
<td>1.4556754</td>
</tr>
<tr>
<td>10400</td>
<td>1.3524935</td>
</tr>
<tr>
<td>12480</td>
<td>1.349256</td>
</tr>
<tr>
<td>18720</td>
<td>1.3383944</td>
</tr>
<tr>
<td>23920</td>
<td>1.3359489</td>
</tr>
<tr>
<td>26000</td>
<td>1.3337911</td>
</tr>
</tbody>
</table>
**Solution** We are to generate CFD solutions using several turbulence models, and we are to compare and discuss the results, particularly the drag coefficient.

**Assumptions** 1 The flow is two-dimensional and incompressible. 2 The flow is symmetric about the x axis. 3 The flow is turbulent, but steady in the mean.

**Properties** The fluid is air with \( \rho = 1.225 \text{ kg/m}^3 \) and \( \mu = 1.7894 \times 10^{-5} \text{ kg/m/s} \).

**Analysis** We run FlowLab using template `Block_turbulence_model`. All cases are run at the same Reynolds number, namely \( 1.37 \times 10^4 \). We compare drag coefficient for all four cases in Table 1. We see that \( C_D \) depends greatly on turbulence model. Most of the models underpredict the drag coefficient by about 30%, but the \( k-\omega \) model overpredicts \( C_D \) by more than 33%. In terms of percentage error, the Reynolds stress model is closest to the experimental value of 1.9, though it’s prediction is nothing to write home about.

**Discussion** Newer versions of FlowLab may give slightly different results. Turbulence models involve semi-empirical analysis, curve-fits, and simplifications, and no turbulence model is best for every kind of fluid flow. It is not always clear which turbulence model to use for a given problem.

**TABLE 1**

<table>
<thead>
<tr>
<th>Turbulence model</th>
<th>( C_D )</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k-\varepsilon ) realizable (2 eq.)</td>
<td>1.34342</td>
<td>-29.5%</td>
</tr>
<tr>
<td>( k-\omega ) standard (2 eq.)</td>
<td>2.536939</td>
<td>33.5%</td>
</tr>
<tr>
<td>Spallart-Allmaras (1 eq.)</td>
<td>1.360602</td>
<td>-28.4%</td>
</tr>
<tr>
<td>Reynolds stress model (5 eq.)</td>
<td>1.385374</td>
<td>-27.1%</td>
</tr>
</tbody>
</table>
Solution

We are to run CFD simulations of flow over a 2-D rectangular block with various values of $L/D$. We are to compare and discuss streamlines and drag coefficient, and we are to compare our CFD results to experiment.

Assumptions

1. The flow is two-dimensional and incompressible.
2. The flow is symmetric about the $x$ axis.
3. The flow is turbulent, but steady in the mean.

Properties

The fluid is air with $\rho = 1.225 \text{ kg/m}^3$ and $\mu = 1.7894 \times 10^{-5} \text{ kg/m/s}$.

Analysis

We run FlowLab using template Block_length.

(a) Drag coefficient is tabulated as a function of $L/D$ in Table 1. These data are also plotted in Fig. 1, along with experimental data from Chap. 11. We see that the calculations are consistently lower than the experimental values for the smaller lengths, but the agreement is very good at $L/D = 3$ (as high as the table goes). While the experimentally obtained drag coefficient peaks at $L/D = 0.5$, the CFD calculations predict that $C_D$ decays continually with $L/D$.

(b) For several values of $L/D$, we plot streamlines near the block (Fig. 2). We can see why the drag coefficient decreases as $L/D$ increases. Namely, as the block length increases, the large recirculating flow regions in the wake (wake eddies) decrease in size (particularly in the vertical direction). In all cases, the flow separates at the sharp corner of the blunt face, but as $L$ increases, the flow has more time to flatten out along the upper and lower walls of the block, leading to wake eddies of reduced thickness. Indeed, by $L/D = 3.0$ (Fig. 2f) the separated flow along the upper and lower walls appears to reattach just upstream of the back of the block. The wake eddies in this case are much thinner than those of the shorter blocks (compare Fig. 2b and 2e for example). Another way to express this is to say that the longer blocks are more “streamlined” (in a gross sense of the word) than are the shorter blocks, and therefore have less drag. Experiments show that the drag is highest at $L/D = 0.5$. Apparently the vortex shedding process is strong for this case, leading to high drag. Our steady, symmetric CFD model is not able to simulate the unsteady features of the actual flow.

(c) There are many possible reasons for the discrepancy between CFD calculations and experiment. We are modeling the problem as a steady flow that is symmetric about the axis, but experiments reveal that flow over bluff bodies like these oscillate and shed vortices – the flow is neither steady nor symmetric. Furthermore, we are using a turbulence model. As discussed previously, turbulence models are not universal, and may not be applicable to the present problem. A DNS or LES simulation would be required to correctly model the unsteady turbulent eddies.

[ continued on next page → ]
FIGURE 2
Streamlines for steady, incompressible, two-dimensional, turbulent flow over a rectangular block with various values of block length to block height: $L/D =$ (a) 0.1, (b) 0.5, (c) 1.0, (d) 1.5, (e) 2.0, and (f) 3.0.

Discussion  Newer versions of FlowLab may give slightly different results. CFD issues such as grid resolution and the extent of the computational domain do not contribute appreciably to the discrepancy here, since we have set up the computational domain based on results of similar previous problems. As $L/D$ increases, the discrepancy between CFD results and experiment gets smaller. This can be explained by the fact that the shed vortices are reduced in strength as $L/D$ increases – the steady, symmetric CFD simulation thus becomes more physically correct with increasing $L/D$. 
Solution

We are to generate CFD solutions for external flow over a cylindrical block. Specifically, we are to compare drag coefficient for various values of $R/D$ (the extent of the outer boundary of the computational domain). In addition, we are to compare the calculated value of $C_D$ with experiment.

Assumptions

1. The flow is incompressible.
2. The flow is axisymmetric about the $x$ axis.
3. The flow is turbulent, but steady in the mean.

Properties

The fluid is air with $\rho = 1.225 \text{ kg/m}^3$ and $\mu = 1.7894 \times 10^{-5} \text{ kg/m} \cdot \text{s}$.

Analysis

We run FlowLab using template Block_axisymmetric.

(a) The Reynolds number based on cylinder diameter is the same as that of Problem 15-26, namely,

$$\text{Re} = \frac{\rho V D}{\mu} = \frac{(1.225 \text{ kg/m}^3)(2.0 \text{ m/s})(0.10 \text{ m})}{1.7894 \times 10^{-5} \text{ kg/m} \cdot \text{s}} = 1.37 \times 10^4$$

Experimental data indicate that the drag coefficient for this body is $C_D \approx 0.90$ at Reynolds numbers greater than $10^4$ (see Chap. 11).

(b) The CFD code is run for eight values of $R$, all else being equal. $C_D$ is tabulated as a function of $R/D$ in Table 1, and plotted in Fig. 2. As the extent of the computational domain grows in size, the drag coefficient decreases steadily, but is trying to level off by $R/D \approx 500$. Thus, a computational domain extent of $R/D \approx 500$ or more is needed to achieve independence of $C_D$. We report a final value of $C_D = 0.99$. Unfortunately, the program does not allow for $R/D$ values greater than 500.

(c) There are several possible reasons for the discrepancy between the calculated value of $C_D$ and the experimentally obtained value of $C_D$ (about 0.90). First of all, the actual flow is most likely unsteady, with vortices being shed into the wake, whereas we are simulating a steady flow. In addition, the unsteady shedding of vortices renders the flow no longer axisymmetric about the $x$ axis, whereas we are forcing our flow to be axisymmetric. Furthermore, the grid resolution may not be adequate to achieve grid independence. Finally (and most importantly), we are using a turbulence model to simulate this flow field. The CFD solution we obtain is only as good as the degree to which the turbulence model correctly models the physics of the turbulence. As discussed in the text, no turbulence model is universally valid for all types of turbulent flows. Discrepancies between experiment and CFD will always exist regardless of how fine the grid or how large the extent of the computational domain.

(d) Streamlines near the body are plotted for $R/D = 5$ and 500 in Fig. 2. There is surprisingly little observable difference between these two extreme cases. Compared to the 2-D case, it appears that the axisymmetric case requires a greater computational domain extent. This is surprising, because in an axisymmetric flow field, the fluid can flow around the body in all directions, not just over the top and bottom as in 2-D flow. As fluid moves away from the body, the radius also increases there, and more mass can flow through that radial location compared to the 2-D case. In other words an axisymmetric flow is less “confined” or “constrained” than a corresponding two-dimensional flow. Thus, we might have expected the opposite behavior.

We compare our predicted drag coefficient (0.99) with that of experiment (0.90). The discrepancy is only about 10% – much better agreement than the two-dimensional case. The reasons for this improvement is not clear. Axisymmetric flows tend to be less unsteady, and the vortices they shed are generally less coherent and weaker. This may contribute to some of the improvement. It may be merely fortuitous that the turbulence model yields better results for the axisymmetric case as compared to the 2-D case.

### Table 1

Drag coefficient as a function of the normalized extent of the computational domain for turbulent flow over a cylindrical block.

<table>
<thead>
<tr>
<th>$R/D$</th>
<th>$C_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1.0704</td>
</tr>
<tr>
<td>10</td>
<td>1.04053</td>
</tr>
<tr>
<td>20</td>
<td>1.03567</td>
</tr>
<tr>
<td>50</td>
<td>1.02155</td>
</tr>
<tr>
<td>100</td>
<td>1.00781</td>
</tr>
<tr>
<td>200</td>
<td>0.998819</td>
</tr>
<tr>
<td>500</td>
<td>0.992311</td>
</tr>
</tbody>
</table>

### Figure 1

Drag coefficient plotted as a function of the normalized extent of the computational domain for turbulent flow over a rectangular block.
FIGURE 2
Streamlines for steady, incompressible, axisymmetric, turbulent flow over a rectangular block at $R/D = (a) 5, (b) 500$. Only one slice of the flow is shown.

Discussion
Newer versions of FlowLab may give slightly different results. The far field boundaries must be quite far away to achieve results that are independent of the extent of the boundary.
**Solution**  We are to generate CFD solutions for several grid resolutions to test for grid independence for flow through a diffuser. Specifically, we are to compare streamlines and pressure difference at each level of grid resolution.

**Assumptions**  1 The flow is incompressible. 2 The flow is axisymmetric about the x axis. 3 The flow is turbulent, but steady in the mean.

**Properties**  The fluid is air with $\rho = 1.225 \text{ kg/m}^3$ and $\mu = 1.7894 \times 10^{-5} \text{ kg/m} \cdot \text{s}$.

**Analysis**  We run FlowLab using template Diffuser_mesh.

(a) Streamlines are plotted for six grid resolution cases in Fig. 1.

(b) $\Delta P$ is tabulated as a function of cell count in the table for the case with $\theta = 20^\circ$. As grid resolution improves, $\Delta P$ increases, and becomes independent of grid resolution by the fourth or fifth mesh. This is also seen in Fig. 2 where $\Delta P$ is plotted as a function of cell count.

**Discussion**  The unphysical-looking streamlines at the very coarse grid resolution (Fig. 1a) are due to interpolation errors when the CFD code calculates contours of constant stream function. Notice that even though there is gross flow separation in this diffuser, there is still a pressure recovery through the diffuser ($P_{in}$ is less than $P_{out}$). A better design (with even higher pressure recovery) would use a smaller diffuser angle so as to avoid flow separation along the diffuser wall. The outlet of the computational domain is reasonably far downstream (several pipe diameters) to avoid reverse flow at the outlet.
Solution

We are to compare the pressure drop and the pressure distribution at the outlet of a diffuser in a round pipe with two different outlet conditions: pressure outlet and outflow.

Assumptions

1. The flow is incompressible.
2. The flow is axisymmetric about the x axis.
3. The flow is turbulent, but steady in the mean.

Properties

The fluid is air with $\rho = 1.225 \text{ kg/m}^3$ and $\mu = 1.7894 \times 10^{-5} \text{ kg/m/s}$.

Analysis

We run FlowLab using template Diffuser_outflow. For the present case (outflow boundary condition), the pressure difference is $\Delta P = P_{in} - P_{out} = -32.1096 \text{ Pa}$, which agrees to four digits of precision with the result of the previous problem at the same grid resolution (19800 cells; grid refinement number 6), for which $\Delta P = P_{in} - P_{out} = -32.1099 \text{ Pa}$. Thus we conclude that the outlet boundary condition has negligible effect on this flow field.

For the case with the pressure outlet boundary condition, the static pressure at the outlet of the computational domain is forced to be constant (zero gage pressure in the calculations of the previous problem). In the present case however, the outflow boundary condition does not fix static pressure – rather, it forces flow variables to level off as they approach the outlet boundary. We find that $P$ varies by less than 0.001 percent across the boundary. Thus, even though $P$ is not forced to be constant along the outflow boundary, it turns out to be nearly constant anyway.

Discussion

Newer versions of FlowLab may give slightly different results. With a velocity inlet and an outflow outlet, we do not fix the value of pressure at either boundary. Instead, the CFD code assigns $P = 0$ gage pressure at some (arbitrary) location in the flow field (the default location is the origin). Even though the inlet and outlet pressures differ significantly between the two cases, $\Delta P$ is identical to within four significant digits of precision. These results verify the statement we made in Chap. 9: For incompressible flow, it is not pressure itself which is important to the flow field, but rather pressure differences. We conclude that the differences between pressure outlet and outflow are small, provided that the outlet boundary is far enough away from the region of interest in the flow field.
**Solution**  
We are to generate CFD solutions for flow through a diffuser at various values of diffuser half-angle $\theta$. Specifically, we are to compare streamlines and pressure difference for each case, and we are to determine the maximum value of $\theta$ that achieves the stated design objectives.

**Assumptions**  
1. The flow is incompressible.  
2. The flow is axisymmetric about the $x$ axis.  
3. The flow is turbulent, but steady in the mean.

**Properties**  
The fluid is air with $\rho = 1.225$ kg/m$^3$ and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

**Analysis**  
We run FlowLab using template Diffuser_angle.  
(a) Streamlines are plotted in Fig. 1 for all twelve cases.

At $\theta = 5^\circ$ and $7.5^\circ$, the flow does not separate along the diffuser wall (Figs. 1a and 1b), although separation appears imminent near the downstream corner of the diffuser for the latter case. To avoid flow separation, Barb should recommend a diffuser half-angle of $7.5^\circ$ or less. As $\theta$ increases, the boundary layer is unable to remain attached in the adverse pressure gradient, and the flow separates. A very small separation bubble is apparent at $\theta = 10^\circ$ (Fig. 1c). As $\theta$ continues to increase, the separation bubble grows in size, and the separation point moves upstream, closer and closer to the upstream corner of the diffuser (compare Figs. 1d through 1g). By $\theta = 25^\circ$ (Fig. 1h), the flow separates very close to the start of the diffuser. From this point on, the diffuser angle is so sharp that the flow separates right at the upstream corner of the diffuser. The streamline patterns reveal that the separation bubble continues to grow in size as $\theta$ increases (Figs. 1i through 1l). Beyond $\theta \approx 45^\circ$ however, the streamline pattern changes very little in the separation bubble (compare Figs. 1k and 1l).
(b) $\Delta P$ is tabulated as a function of diffuser half-angle in the table (four extra cases are solved for improved clarity). As $\theta$ increases, $P_{in}$ increases, reflecting the effect of the larger separation bubble. Physically, we achieve less and less pressure recovery as the separation bubble grows. We see that $\Delta P$ flattens out at high values of $\theta$, becoming nearly independent of $\theta$ beyond $\theta \approx 60^\circ$. This is also seen in Fig. 2 where $P_{in}$ is plotted as a function of diffuser half-angle. There appears to be a sharp rise in $\Delta P$ beyond $30^\circ$. The reason for this is not certain, but is probably related to the fact that when $\theta$ is greater than about $30^\circ$, the flow separates right at the upstream corner. The pressure rise is greater than 40 Pa for all angles below $15^\circ$. Thus, to ensure a pressure recovery of at least 40 pascals, Barb should recommend a diffuser half-angle less than around $15^\circ$.

**Discussion**  
Newer versions of FlowLab may give slightly different results. The outlet of the computational domain is reasonably far downstream (several pipe diameters) to avoid reverse flow at the outlet. Notice that even for the case of a sudden expansion ($\theta = 90^\circ$) there is still a pressure recovery through the diffuser ($P_{in}$ is less than $P_{out}$).

<table>
<thead>
<tr>
<th>$\theta$</th>
<th>$\Delta P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>-49.1114</td>
</tr>
<tr>
<td>7.5</td>
<td>-47.6054</td>
</tr>
<tr>
<td>10</td>
<td>-44.9608</td>
</tr>
<tr>
<td>12.5</td>
<td>-42.2056</td>
</tr>
<tr>
<td>15</td>
<td>-39.5764</td>
</tr>
<tr>
<td>17.5</td>
<td>-37.4373</td>
</tr>
<tr>
<td>20</td>
<td>-35.6913</td>
</tr>
<tr>
<td>25</td>
<td>-32.589</td>
</tr>
<tr>
<td>30</td>
<td>-30.189</td>
</tr>
<tr>
<td>45</td>
<td>-19.2195</td>
</tr>
<tr>
<td>60</td>
<td>-18.5275</td>
</tr>
<tr>
<td>90</td>
<td>-18.2946</td>
</tr>
</tbody>
</table>
**Solution**  We are to perform a grid independence test on the 90° diffuser (sudden expansion) case of the previous problem by refining the grid resolution.

**Assumptions**  1 The flow is incompressible. 2 The flow is axisymmetric about the x axis. 3 The flow is turbulent, but steady in the mean.

**Properties**  The fluid is air with \( \rho = 1.225 \) kg/m\(^3\) and \( \mu = 1.7894 \times 10^{-5} \) kg/m\(\cdot\)s.

**Analysis**  We run FlowLab using template Expansion_mesh. We run several levels of grid refinement, tabulate the results, and plot the results. We have achieved grid independence to the third significant digit of precision by the fourth refinement level, namely, 8100 cells. The final value of \( \Delta P \) is reported to three significant digits as \(-18.4 \) Pa.

**Discussion**  Newer versions of FlowLab may give slightly different results. We could try refining the grid even further, but the increase in precision would not be worth the effort. In a flow field such as this, the separation point is fixed at the sharp corner; the CFD code does not have problems identifying the separation point. Notice that even for the case of a sudden expansion (\( \theta = 90° \)) there is still a pressure recovery through the diffuser (\( P_{in} \) is less than \( P_{out} \)).
Solution  We are to generate CFD solutions for several downstream tube extension lengths to see the impact of the downstream boundary. We are also to discuss the variation of $P_{in}$ with $L_{extend}$.

Assumptions  
1. The flow is steady and incompressible. 
2. The flow is axisymmetric about the $x$ axis. 
3. The flow is laminar.

Properties  
The fluid is water with $\rho = 998.2$ kg/m$^3$ and $\mu = 0.001003$ kg/m·s.

Analysis  
We run FlowLab using template Contraction_domain.

(a) Only the first case at $L_{extend}/D_2 = 0.25$ has reverse flow at the pressure outlet. By $L_{extend}/D_2 = 0.50$ the reverse flow problems are gone. The streamlines reveal that the flow separates at the corner, forming a recirculating eddy. The eddy reattaches at approximately $x/D_2 = 0.50$. Streamlines are shown for the first three cases in Fig. 1. The overall streamline shapes appear to be unaffected by the downstream extent of the computational domain.

(b) $P_{in}$, $P_1$, and $\Delta P = P_{in} - P_1$ are tabulated as functions of $L_{extend}/D_2$. $\Delta P$ is independent of $L_{extend}$ (to three significant digits of precision) by $L_{extend}/D_2 = 0.5$.

<table>
<thead>
<tr>
<th>$L_{extend}/D_2$</th>
<th>$P_{in}$ (kPa, gage)</th>
<th>$P_1$ (kPa, gage)</th>
<th>$\Delta P$ (kPa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>0.664955</td>
<td>0.212228</td>
<td>0.452727</td>
</tr>
<tr>
<td>0.5</td>
<td>0.622098</td>
<td>0.171344</td>
<td>0.450754</td>
</tr>
<tr>
<td>0.75</td>
<td>0.597474</td>
<td>0.146778</td>
<td>0.450696</td>
</tr>
<tr>
<td>1</td>
<td>0.589973</td>
<td>0.139247</td>
<td>0.450725</td>
</tr>
<tr>
<td>1.25</td>
<td>0.589224</td>
<td>0.138468</td>
<td>0.450756</td>
</tr>
<tr>
<td>1.5</td>
<td>0.591391</td>
<td>0.140604</td>
<td>0.450787</td>
</tr>
<tr>
<td>2</td>
<td>0.599387</td>
<td>0.148538</td>
<td>0.450849</td>
</tr>
<tr>
<td>2.5</td>
<td>0.609285</td>
<td>0.158375</td>
<td>0.45091</td>
</tr>
<tr>
<td>3</td>
<td>0.619797</td>
<td>0.168828</td>
<td>0.45097</td>
</tr>
</tbody>
</table>

(c) Inlet gage pressure $P_{in}$ is plotted as a function of $L_{extend}/D_2$ in Fig. 2. The inlet pressure drops as $L_{extend}/D_2$ increases from 0.25 to 1.25, but then starts to slowly rise as $L_{extend}/D_2$ continues to increase. We explain this trend with help from the streamlines of Fig. 1. First of all, the data for $L_{extend}/D_2 = 0.25$ are not reliable, since there is reverse flow at the pressure outlet. Thus, we begin our discussion at $L_{extend}/D_2 = 0.50$. The separation bubble (recirculating eddy) at the inlet to the smaller diameter tube forces the flow to converge and accelerate through an effective area that is smaller than the cross-sectional area of the small tube (a vena contracta). This high speed flow leads to a low pressure region near the separation bubble. Downstream of the separation bubble, the pressure tries to increase slightly (the downstream portion of the eddy behaves like a diffuser, recovering some of the pressure loss). However, since the outlet pressure is fixed at zero gage pressure, the inlet pressure must decrease to compensate. This explains why $P_{in}$ decreases when $L_{extend}/D_2$ increases from 0.50 to 1.25. By $L_{extend}/D_2 = 1.25$, the outlet of the computational domain is beyond the region of pressure recovery due to the eddy, and the flow begins its slow development towards fully developed laminar pipe flow. Beyond $x/D_2 \approx 1.25$ the pressure decreases.

FIGURE 1  
Streamlines: $L_{extend}/D_2 = (a) 0.25$, (b) 0.50, and (c) 0.75.

FIGURE 2  
Inlet gage pressure as a function of normalized downstream extent of the computational domain.
axially along the tube because of friction along the pipe wall. But since the exit pressure is atmospheric, the inlet pressure must rise accordingly. In other words, the pressure at the inlet must rise to overcome the increasing pressure drop through the tube. We expect this trend to continue, eventually becoming linear.

Based on all our results taken collectively, we recommend that Shane use a value of $L_{\text{extend}}/D_2$ of 2.0 or more to ensure proper simulation of this flow. If computer speed or memory is a problem, he can get away with a minimum of $L_{\text{extend}}/D_2 = 0.75$ to avoid reverse flow problems at the outlet.

**Discussion** For other geometries, fluids, or speeds, the separation bubble may extend farther than in this example. Thus is wise to extend the computational domain reasonably far downstream (several tube diameters).
Chapter 15 Computational Fluid Dynamics

15-37 FlowLab

Solution We are to simulate flow through a sudden contraction in a tube, using no downstream tube extension. We are to calculate the error in pressure drop and discuss.

Assumptions 1 The flow is steady and incompressible. 2 The flow is axisymmetric about the x axis. 3 The flow is laminar.

Properties The fluid is water with $\rho = 998.2 \text{ kg/m}^3$ and $\mu = 0.001003 \text{ kg/m} \cdot \text{s}$.

Analysis We run FlowLab using template Contraction_zerolength. There is no reverse flow in the CFD simulation. The streamlines (shown in the figure) reveal that since there is no downstream extension, there is obviously no way to have a separation bubble. With the outlet pressure set to a constant value, the streamlines adjust themselves such that the fluid flows through the outlet without reverse flow. Although this may appear to be a good CFD solution, it turns out to be erroneous because the actual pressure across the interface of a sudden contraction is not constant. We obtain a net pressure difference of $\Delta P = P_{in} - P_{out} = 519.6 \text{ Pa}$. Compared to the best values from the previous problem ($\Delta P = 451 \text{ Pa}$), the percentage error is about 15%.

Discussion Newer versions of FlowLab may give slightly different results. The error in $\Delta P$ caused by neglecting the downstream extension is significant. This reinforces our discussion in the text about extending outlets.

15-38 FlowLab

Solution We are to simulate flow through a sudden contraction in a tube, using three different values of outlet pressure at the pressure outlet boundary. We are to compare the pressure drop and discuss.

Assumptions 1 The flow is steady and incompressible. 2 The flow is axisymmetric. 3 The flow is laminar.

Properties The fluid is water with $\rho = 998.2 \text{ kg/m}^3$ and $\mu = 0.001003 \text{ kg/m} \cdot \text{s}$.

Analysis We run FlowLab using template Contraction_pressure. Results for the three cases are summarized in the table for the case in which $L_{extend}/D_2 = 2.0$. We see that the inlet pressure and the pressure at $x = 0$ rise or fall in symphony with $P_{out}$, such that the net pressure difference is the same (to more than four significant digits) in all cases. The results verify a statement we made in Chap. 9: For incompressible flow, it is not pressure itself which is important to the flow field, but rather pressure differences.

Discussion The slight differences (in the fifth digit) in $\Delta P$ are due to numerical inaccuracies in the CFD code (the residuals never go to exactly zero). Note that we are subtracting two large numbers, which inevitably leads to precision errors. If we had run the simulation with double precision arithmetic and adjusted some of the numerical parameters, we could have obtained even better agreement.

<table>
<thead>
<tr>
<th>$P_{out}$ (Pa, gage)</th>
<th>$P_{in}$ (kPa, gage)</th>
<th>$P_{1}$ (kPa, gage)</th>
<th>$\Delta P$ (Pa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-50,000</td>
<td>-49.400613</td>
<td>-49.851461</td>
<td>450.848</td>
</tr>
<tr>
<td>0</td>
<td>0.59938678</td>
<td>0.14853764</td>
<td>450.8491</td>
</tr>
<tr>
<td>50,000</td>
<td>50.599387</td>
<td>50.148539</td>
<td>450.848</td>
</tr>
</tbody>
</table>
**Solution** We are to generate CFD solutions for several downstream tube extension lengths to see the impact of the downstream boundary.

**Assumptions** 1 The flow is incompressible. 2 The flow is axisymmetric about the x axis. 3 The flow is turbulent, but steady in the mean.

**Properties** The fluid is water with $\rho = 998.2 \text{ kg/m}^3$ and $\mu = 0.001003 \text{ kg/m} \cdot \text{s}$.

**Analysis** We run FlowLab using template *Contraction turbulent*. 

(a) Four Reynolds numbers are calculated. Note that since the diameter decreases by a factor of 4 through the contraction, the average velocity in the smaller downstream tube increases by a factor of $4^2 = 16$ compared to that in the larger upstream tube. For the laminar case of Problem 15-37,

Upstream tube:

$$Re_1 = \frac{\rho V_D D_1}{\mu} = \frac{(998.2 \text{ kg/m}^3)(0.05 \text{ m/s})(0.008 \text{ m})}{0.001003 \text{ kg/m} \cdot \text{s}} = 398.1$$

(1)

Downstream tube:

$$Re_2 = \frac{\rho V_D D_2}{\mu} = \frac{(998.2 \text{ kg/m}^3)(0.8 \text{ m/s})(0.002 \text{ m})}{0.001003 \text{ kg/m} \cdot \text{s}} = 1592$$

(2)

Since both of these values are smaller than 2300, the laminar flow assumption for Problem 15-37 is reasonable. For the turbulent pipe flow of the present problem,

Upstream pipe:

$$Re_1 = \frac{\rho V_D D_1}{\mu} = \frac{(998.2 \text{ kg/m}^3)(1.0 \text{ m/s})(0.8 \text{ m})}{0.001003 \text{ kg/m} \cdot \text{s}} = 796,200$$

(3)

Downstream pipe:

$$Re_2 = \frac{\rho V_D D_2}{\mu} = \frac{(998.2 \text{ kg/m}^3)(16.0 \text{ m/s})(0.2 \text{ m})}{0.001003 \text{ kg/m} \cdot \text{s}} = 3,185,000$$

(4)

These Reynolds numbers are clearly high enough that the flow is indeed turbulent.

(b) Our CFD solutions reveal reversed flow for the smallest two cases, i.e. for $L_{\text{extend}}/D_2 = 0.25$ and 0.5. For higher values of $L_{\text{extend}}/D_2$ there is no reverse flow. Comparing to the results of Problem 15-37, apparently the separation bubble for turbulent flow is somewhat longer (proportionally) than that for laminar flow. This is verified by comparing the streamlines of Fig. 1 to those of Fig. 1c of Problem 15-37.

(c) Pressures $P_\text{in}$, $P_1$, and $\Delta P = P_\text{in} - P_1$ are tabulated as functions of $L_{\text{extend}}/D_2$ in the table. Note that we use units of kPa instead of Pa here for convenience. $\Delta P$ is independent of $L_{\text{extend}}$ (to three significant digits of precision) by $L_{\text{extend}}/D_2 = 0.75$.

**Discussion** Newer versions of FlowLab may give slightly different results. The lack of scatter in the turbulent data for $\Delta P$ beyond $L_{\text{extend}}/D_2 = 0.75$ is a rather pleasant surprise; we might have expected more scatter than the laminar flow solution of Problem 15-36, since we have two additional nonlinear transport equations to solve, with their associated interactions that complicate the solution.

<table>
<thead>
<tr>
<th>$L_{\text{extend}}/D_2$</th>
<th>$P_\text{in}$ (kPa, gage)</th>
<th>$P_1$ (kPa, gage)</th>
<th>$\Delta P$ (kPa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>301.44388</td>
<td>117.5592</td>
<td>183.88468</td>
</tr>
<tr>
<td>0.5</td>
<td>271.27503</td>
<td>88.70387</td>
<td>182.571163</td>
</tr>
<tr>
<td>0.75</td>
<td>250.08225</td>
<td>67.56302</td>
<td>182.519234</td>
</tr>
<tr>
<td>1</td>
<td>238.22473</td>
<td>55.70388</td>
<td>182.520847</td>
</tr>
<tr>
<td>1.25</td>
<td>230.67003</td>
<td>48.14458</td>
<td>182.525452</td>
</tr>
<tr>
<td>1.5</td>
<td>225.3785</td>
<td>42.85207</td>
<td>182.526426</td>
</tr>
<tr>
<td>2</td>
<td>218.49866</td>
<td>35.96206</td>
<td>182.536601</td>
</tr>
</tbody>
</table>
Solution  We are to compare the pressure drop through a sudden contraction in a round pipe with two different outlet conditions: pressure outlet and outflow.

Assumptions  1 The flow is incompressible. 2 The flow is axisymmetric about the x axis. 3 The flow is turbulent, but steady in the mean.

Properties  The fluid is water with \( \rho = 998.2 \) kg/m\(^3\) and \( \mu = 0.001003 \) kg/m\( \cdot \)s.

Analysis  We run FlowLab using template *Contraction_outflow*, for which the pipe extension is fixed at \( L_{\text{extend}}/D_2 = 0.80 \). For the case with the outlet boundary condition set to a pressure outlet, the average pressure at \( x = 0 \) is \( P_{\text{in}} = 254.443 \) kPa, the average pressure at \( x = 0 \) is \( P_1 = 68.1982 \) kPa, and the pressure difference is \( \Delta P = P_{\text{in}} - P_1 = 186.245 \) kPa. When we run the FlowLab template with the outlet boundary condition set to an outflow, \( P_{\text{in}} = 129.579 \) kPa, \( P_1 = -56.6647 \) kPa, and \( \Delta P = P_{\text{in}} - P_1 = 186.244 \) kPa. While the actual values of pressure differ throughout the contraction, the pressure difference agrees to 5 digits of precision when comparing the two cases. Thus we conclude that the outlet boundary condition has very little effect on this flow field. In other words, the outlet is sufficiently far downstream that the flow from the large diameter pipe to the small diameter pipe is practically unaffected by the outflow boundary condition.

For the case with the pressure outlet boundary condition, the static pressure at the outlet of the computational domain is forced to be constant (zero gage pressure in the calculations of the previous problem). In the present case however, the outflow boundary condition does not fix static pressure – rather, it forces flow variables to level off as they approach the outlet boundary.

Note: In the previous problem (pressure outlet boundary condition), the average pressure at the inlet was \( P_{\text{in}} = 250.08 \) kPa, \( P_1 = 67.56 \) kPa, and the pressure difference was \( \Delta P = P_{\text{in}} - P_1 = 182.52 \) kPa. This pressure difference is not the same as that of the present problem, but is within about 2%. The difference is due to the numerical settings of the CFD calculations, especially the convergence criteria, which are different between the two templates, even though the geometry is identical.

Discussion  Newer versions of FlowLab may give slightly different results. With a velocity inlet and an outflow outlet, we do not fix the value of pressure at either boundary. Instead, the CFD code assigns \( P = 0 \) gage pressure at some (arbitrary) location in the flow field (the default location is the origin). Even though the inlet and outlet pressures differ significantly between the two cases, \( \Delta P \) is nearly the same. This result again verifies the statement we made in Chap. 9: For incompressible flow, it is not pressure itself which is important to the flow field, but rather pressure differences.
Solution

We are to generate CFD solutions for several downstream duct extension lengths to see the impact of the downstream boundary.

Assumptions
1. The flow is incompressible.
2. The flow is two-dimensional and symmetric about the $x$ axis.
3. The flow is turbulent, but steady in the mean.

Properties
The fluid is water with $\rho = 998.2 \text{ kg/m}^3$ and $\mu = 0.001003 \text{ kg/m}\cdot\text{s}$.

Analysis
We run FlowLab using template Contraction_2d.

(a) Our CFD solutions reveal reversed flow for the smallest three cases, i.e. for $L_{extend}/D_2 = 0.25, 0.5$, and $0.75$. For higher values of $L_{extend}/D_2$, there is no reverse flow. Comparing to the results of Problem 15-39, the separation bubble for 2-D flow is somewhat longer than that for axisymmetric flow. This is verified by comparing the streamlines of Fig. 1 to those of Fig. 1 of Problem 15-39.

(b) Pressures $P_{in}$, $P_1$, and $\Delta P = P_{in} - P_1$ are tabulated as functions of $L_{extend}/D_2$ in Table 1. $\Delta P$ is independent of $L_{extend}$ (to three significant digits of precision) by $L_{extend}/D_2 = 0.75$.

Discussion
Newer versions of FlowLab may give slightly different results. Comparing $\Delta P$ between this problem (2-D) and Problem 15-39 (axisymmetric), we see that $\Delta P$ for the axisymmetric case is more than 17 times greater than $\Delta P$ for the 2-D case. This is because for the axisymmetric case, the area downstream of the sudden contraction is 16 times smaller than the upstream area, while for the 2-D case, the area changes by only a factor of 4. Nevertheless, the 2-D case generates a longer separation bubble.

\[
\begin{array}{cccc}
L_{extend}/D_2 & P_{in} \text{ (kPa, gage)} & P_1 \text{ (kPa, gage)} & \Delta P \text{ (kPa)} \\
0.25 & 18.227758 & 7.3914395 & 10.8363185 \\
0.5 & 17.786943 & 7.1083252 & 10.6786178 \\
0.75 & 16.127482 & 5.4705503 & 10.6569317 \\
1 & 14.661298 & 4.0061421 & 10.6551559 \\
1.25 & 13.730503 & 3.0757039 & 10.6547991 \\
1.5 & 13.154366 & 2.4999675 & 10.6543985 \\
2 & 12.489565 & 1.8359823 & 10.6538527 \\
3 & 11.882063 & 1.2281641 & 10.6538989 \\
4 & 11.602836 & 0.94779285 & 10.6550432 \\
\end{array}
\]
Solution

We are to generate CFD solutions for several grid resolutions to test for grid independence for flow through a jog in a channel. Specifically, we are to compare streamlines and pressure difference at each level of grid resolution.

Assumptions

1. The flow is incompressible and two-dimensional.
2. The flow is turbulent, but steady in the mean.

Properties

The fluid is air with \( \rho = 1.225 \text{ kg/m}^3 \) and \( \mu = 1.7894 \times 10^{-5} \text{ kg/m}\cdot\text{s} \).

Analysis

We run FlowLab using template Jog_turbulent_mesh.

(a) Streamlines are plotted for six grid resolution cases in Fig. 1. At the very coarse grid resolution, a small separation bubble appears at the lower left corner of the jog (Fig. 1a); flow separation is not obvious at the other corners, but it appears imminent at the most downstream corner. As grid resolution improves, the separation bubble on the lower left corner grows in size (compare Figs. 1a and 1b), and another separation bubble appears downstream of the jog. With continued improvement of the grid, both separation bubbles, especially the downstream one, continue to grow in size (Fig. 1c through 1f). Meanwhile, the streamlines become more rounded near the upper right corner of the jog. By 13,600 cells (Fig. 1e), a small recirculating zone is seen in this corner. The streamline pattern settles down by about 13,600 cells. We note that simulation of flow separation, separation bubbles, and reattachment is often a very difficult task for a CFD program. In this particular problem we must use a fairly fine grid in order to resolve all the details of the flow separation.

(b) \( \Delta P \) is tabulated as a function of cell count in the table. As grid resolution improves, \( \Delta P \) increases, reflecting the effect of the larger separation bubbles. Physically, we achieve less and less pressure recovery as the separation bubbles grow. We see that \( \Delta P \) becomes independent of grid resolution to four significant digits by the fifth level of resolution, namely, at 13,600 cells. This is also seen in Fig. 2 where \( \Delta P \) is plotted as a function of number of cells.

Discussion

Newer versions of FlowLab may give slightly different results. The sharp corners in the streamlines of the very coarse grid resolution case (Fig. 1a) are due to interpolation errors when the CFD code calculates contours of constant stream function. The outlet of the computational domain is reasonably far downstream (ten channel heights) to avoid reverse flow at the outlet.
Solution

We are to generate CFD solutions for several grid resolutions to test for grid independence for flow through a jog in a channel. Specifically, we are to compare streamlines and pressure difference at each level of grid resolution.

Assumptions
1. The flow is incompressible.
2. The flow is two-dimensional.
3. The flow is laminar and steady.

Properties
The fluid is water with \( \rho = 998.2 \, \text{kg/m}^3 \) and \( \mu = 0.001003 \, \text{kg/m} \cdot \text{s} \).

Analysis
We run FlowLab using template Jog_laminar_mesh.

(a) Streamlines are plotted for six grid resolution cases in Fig. 1.

At the very coarse grid resolution (Fig. 1a), the streamlines are not very smooth, but reveal the overall flow pattern. Although there are no closed streamlines, it appears that flow separation is imminent at the lower left and upper right corners of the jog. The streamlines appear fairly similar to those of the turbulent flow of the previous problem. At the grid resolution of Fig. 1b, the flow clearly separates at the sharp corner at the origin. The flow also separates at the downstream inside corner of the jog. The streamlines are not well-enough resolved to show closed separation bubbles. As the grid is further refined, the separation bubbles grow in size (compare Figs. 1b through 1d). In addition, the streamlines are smoother and more rounded. With continued improvement of the grid, the streamlines become more rounded near the upper right corner of the jog. By Fig. 1e, a recirculating flow pattern is seen downstream of the jog. Based on streamline patterns, the grid is fully resolved at a cell count of about 10,000. However, the pressure drop seems to be creeping up a bit as grid resolution is refined even further; it is not clear why.

(b) \( \Delta P \) is tabulated as a function of grid resolution in Table 1 (rounded to the third decimal place). \( \Delta P \) is also plotted as a function of number of cells in Fig. 2. As grid resolution improves, \( \Delta P \) rises at first, reflecting the effect of the larger separation bubbles, but then decreases with further grid refinement, eventually leveling off after about 10,000 cells, implying grid independence by \( 10^4 \) cells.
Discussion

Newer versions of FlowLab may give slightly different results. All cases converge nicely – the flow at this Reynolds number is steady and laminar. There is no sign of instability in the flow. The outlet of the computational domain is reasonably far downstream (ten channel heights) to avoid reverse flow at the outlet.

### TABLE 1
Pressure difference from inlet to outlet as a function of cell count, \( V = 0.10 \text{ m/s} \).

<table>
<thead>
<tr>
<th>cell count</th>
<th>( \Delta P ) (Pa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>272</td>
<td>37.612</td>
</tr>
<tr>
<td>576</td>
<td>37.810</td>
</tr>
<tr>
<td>847</td>
<td>37.503</td>
</tr>
<tr>
<td>1700</td>
<td>36.907</td>
</tr>
<tr>
<td>3388</td>
<td>36.592</td>
</tr>
<tr>
<td>10625</td>
<td>36.410</td>
</tr>
<tr>
<td>14123</td>
<td>36.419</td>
</tr>
<tr>
<td>18645</td>
<td>36.435</td>
</tr>
<tr>
<td>23273</td>
<td>36.464</td>
</tr>
</tbody>
</table>

### FIGURE 2
Pressure difference vs. number of cells, \( V = 0.10 \text{ m/s} \).
**Solution**  We are to generate CFD solutions for several grid resolutions to test for grid independence for flow through a jog in a channel. Specifically, we are to compare streamlines and pressure difference at each level of grid resolution.

**Assumptions**  
1. The flow is incompressible.  
2. The flow is two-dimensional.  
3. The flow is laminar and steady.

**Properties**  
The fluid is water with \( \rho = 998.2 \text{ kg/m}^3 \) and \( \mu = 0.001003 \text{ kg/m} \cdot \text{s} \).

**Analysis**  
We run FlowLab using template \textit{Jog high Re}. 

\((a)\) Streamlines are plotted for six grid resolution cases in Fig. 1.

![Streamlines at various levels of grid resolution](image)

**FIGURE 1**  
Streamlines at various levels of grid resolution: cell count = (a) 272, (b) 576, (c) 847, (d) 1700, (e) 3388, and (f) 14,123. The origin (0,0) is marked on each figure for reference. The inlet velocity for these runs is 1.0 m/s. For cases (e) and (f), the streamline patterns are at an arbitrary point in the solution. For these two cases, the CFD calculations do not converge, and eddies appear in the channel downstream of the jog. As the iterations continue, these eddies are swept downstream, but new ones appear, implying that the flow at this Reynolds number is not steady. The streamline patterns change as the CFD code iterates.

**TABLE 1**  
Pressure difference from inlet to outlet as a function of cell count for laminar flow through a two-dimensional jog, \( V = 1.0 \text{ m/s} \).

<table>
<thead>
<tr>
<th>cell count</th>
<th>( \Delta P ) (Pa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>272</td>
<td>2025.69</td>
</tr>
<tr>
<td>576</td>
<td>2269.26</td>
</tr>
<tr>
<td>847</td>
<td>2324.99</td>
</tr>
<tr>
<td>1700</td>
<td>2482.44</td>
</tr>
<tr>
<td>3388</td>
<td>2582.17</td>
</tr>
<tr>
<td>10625</td>
<td>2515.71</td>
</tr>
<tr>
<td>14123</td>
<td>2484.86</td>
</tr>
<tr>
<td>18645</td>
<td>2468.57</td>
</tr>
<tr>
<td>23273</td>
<td>2511.98</td>
</tr>
</tbody>
</table>
At the very coarse grid resolution, a small separation bubble appears at the lower left corner of the jog (Fig. 1a); flow separation is not obvious at the other corners, but it appears imminent at the most downstream corner. The streamlines appear quite similar to those of the turbulent flow of Problem 15-43. As grid resolution improves, the separation bubble on the lower left corner grows in size (compare Figs. 1a and 1b), the streamlines are smoother and more rounded, and another separation bubble appears downstream of the jog. With continued improvement of the grid, both separation bubbles continue to grow in size (Figs. 1c and 1d). Meanwhile, the streamlines become more rounded near the upper right corner of the jog. By Fig. 1c, a small recirculating zone is seen in this corner. The streamline pattern continues to change somewhat as the grid is further refined. However, just as we appear to be approaching grid independence, the flow develops some instabilities and cannot converge to a steady-state solution. This is first seen in the lower right portion of Fig. 1d – the streamlines drift away from the lower wall. While the streamline pattern in the front portion of the jog do not change appreciably, eddies develop in the recovery zone beyond the jog and are swept downstream as the grid is further refined (Figs. 1e and 1f). At the two finest grid resolutions, the flow is attempting to become unsteady. But since we are iterating towards a steady solution, the development and motion of these eddies is not physical. The solution fluctuates and is unable to converge. The bottom line is that we really should simulate this flow with an unsteady CFD solver. Grid independence is not achieved because the flow becomes unsteady and unstable when a very high resolution grid is used.

(b) $\Delta P$ is tabulated as a function of grid resolution in Table 1. Note: The values shown here may differ slightly from those calculated by the students, depending on the computer and version of FlowLab being used. $\Delta P$ is also plotted as a function of number of cells in Fig. 2. As grid resolution improves, $\Delta P$ does not level off because the flow field is unstable – we should use an unsteady solver instead of a steady solver for this flow.

The Reynolds number for the present problem is

$$ Re = \frac{\rho V D}{\mu} = \frac{(998.2 \text{ kg/m}^3)(1.0 \text{ m/s})(0.001 \text{ m})}{0.001003 \text{ kg/m} \cdot \text{s}} = 995 $$

and for the previous problem, it is

$$ Re = \frac{\rho V D}{\mu} = \frac{(998.2 \text{ kg/m}^3)(0.10 \text{ m/s})(0.001 \text{ m})}{0.001003 \text{ kg/m} \cdot \text{s}} = 99.5 $$

Since the Reynolds number of the present case is ten times larger than that of the previous case, and since it is almost 1000, it is not surprising that unsteadiness and instabilities in the flow start to develop at high grid resolutions.

**Discussion**

Newer versions of FlowLab may give slightly different results. The results here are similar to those of CFD simulations of flow around a circular cylinder. Namely, the flow is naturally unsteady, and leads to convergence difficulties at high grid resolution when the simulation is forced to be steady.
**Solution**

We are to generate CFD solutions for compressible flow of air through a converging-diverging nozzle, and plot pressure and Mach number contours at two values of back pressure.

**Assumptions**  
1. The flow is steady and compressible.  
2. The flow is axisymmetric.  
3. The flow is approximated as inviscid.

**Properties**  
The fluid is air with $k = 1.4$.

**Analysis**

We run FlowLab using template *Nozzle_axisymmetric*. We plot contours of $P$ and $Ma$ in Fig. 1. For the case in which $P_b/P_{0,inlet} = 0.455$ (Fig. 1a), we observe a shock wave in the diverging portion of the nozzle. When $P_b/P_{0,inlet} = 0.977$ (Fig. 1b), however, the flow in the entire nozzle is subsonic, and there are no shock waves.

**FIGURE 1**
Contour plots of $P$ and $Ma$ for compressible flow of air through a converging-diverging nozzle: $P_b/P_{0,inlet} = (a) 0.455$, and (b) 0.977. A normal shock is seen for the first case, since the flow is choked, but no shock is seen in the second case, since the flow is subsonic everywhere. The pressure scale is 0 (blue) to 220 kPa (red) for both cases. The Mach number scale is 0 (blue) to 2.5 (red) for both cases.

Plots of $P$ and $Ma$ versus $x$ are shown in Fig. 2 for the two cases. One-dimensional inviscid theory predicts that there should be a shock wave in the diverging portion of the nozzle for Case (a) because of the low back pressure. Indeed, we see a sudden jump in pressure and a sudden decrease in Mach number around $x = 0.55$ m, indicating a shock wave at that location in the diverging portion of the nozzle, which also agrees with the contour plots of Figure 1. At the throat ($x = 0.30$ m), the Mach number is 1.02 and the pressure ratio is 0.519. These values agree well with the isentropic one-dimensional approximation at the throat, where $Ma^* = 1.0$ and $P^*/P_{0,inlet} = 0.528$ (choked conditions – see Appendix A-13 at $Ma = 1.0$).

For Case (b), the back pressure is too high for a shock wave. In fact the back pressure is so high that the flow remains subsonic throughout the nozzle, and the Mach number reaches a maximum value of only about 0.52 at the throat (it does not choke). Mach number first increases and then decreases down the nozzle, which agrees with what we would expect in subsonic flow (higher velocity as the cross-sectional area decreases; lower velocity as the cross-sectional area increases). Similarly, the opposite effect – pressure first decreases and then increases down the nozzle. This also agrees with our
expectations for subsonic flow (lower pressure as the cross-sectional area decreases; higher pressure as the cross-sectional area increases).

[Graphs showing plots of pressure and Mach number versus axial distance for two cases.]

**Case (a)**

**FIGURE 2**
Plots of $P/P_{0,\text{inlet}}$ and $Ma$ versus axial distance $x$ down the nozzle for Case (a), $P/P_{0,\text{inlet}} = 0.455$, and Case (b) $P/P_{0,\text{inlet}} = 0.977$. A shock wave is clearly seen in Case (a), but the flow remains subsonic in Case (b).

**Discussion**   Newer versions of FlowLab may give slightly different results. The shock wave calculated by CFD is not straight, but curved, since the calculations are axisymmetric, and therefore show more detail than the simplified one-dimensional approximation.
Chapter 15 Computational Fluid Dynamics

**Solution** We are to examine the effect of the location of the upstream boundary on automobile drag calculations.

**Analysis** We run FlowLab using template *Automobile_domain*. We run five cases, and show the results in Table 1. It turns out that by about 20 heights away, the upper boundary condition no longer impacts the solution significantly to within three significant digits. **The final drag coefficient is reported as 0.184 to three significant digits.**

These data are also plotted in Fig. 1. We use a log scale on the horizontal axis since the range of $H/h$ is fairly large. For three digits of precision, it is necessary that the top boundary be at least 20 car heights tall. If $H/h$ is shorter than this, the upper boundary of the computational domain adversely influences the flow field. In the real-life flow, of course, the upper boundary is nearly infinite.

**TABLE 1**

<table>
<thead>
<tr>
<th>$H/h$</th>
<th>$C_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.214815</td>
</tr>
<tr>
<td>10</td>
<td>0.18877</td>
</tr>
<tr>
<td>20</td>
<td>0.183779</td>
</tr>
<tr>
<td>35</td>
<td>0.183561</td>
</tr>
<tr>
<td>50</td>
<td>0.183635</td>
</tr>
<tr>
<td>100</td>
<td>0.183756</td>
</tr>
</tbody>
</table>

**Discussion** Newer versions of FlowLab may give slightly different results. The bottom wall of the computational domain is not moving in these calculations, so the flow near the ground is not modeled properly. Furthermore, the calculations are two-dimensional, while a real car is of course fully three-dimensional.

**FIGURE 1**

Drag coefficient plotted as a function of the normalized extent of the computational domain for turbulent flow over a two-dimensional automobile body.

15-47

**Solution** We are to compare turbulence models for the calculation of automobile drag.

**Analysis** We run FlowLab using template *Automobile_turbulence_model*. The results from the CFD calculations are presented in Table 1 (to 5 digits). There is some variation in the calculated values of $C_D$ depending on the turbulence model used; $C_D$ ranges from 0.179 for the Spallart-Allmaras model to 0.224 for the RSM model. The range of scatter about the average of 0.20320 is about +/-12%, which is actually not that large for comparison of four very different turbulence models. **It is impossible to say which one, if any, is “better” or more correct**, since all turbulence models are approximations, with calibrated constants. Furthermore, we have no experimental data with which to compare the CFD results.

**TABLE 1**

Predicted drag coefficient on a two-dimensional automobile as a function of turbulence model.

<table>
<thead>
<tr>
<th>Turbulence model</th>
<th>$C_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spallart-Allmaras (1 eq.)</td>
<td>0.17897</td>
</tr>
<tr>
<td>$k-\varepsilon$ (2 eq.)</td>
<td>0.18587</td>
</tr>
<tr>
<td>$k-\omega$ (2 eq.)</td>
<td>0.22390</td>
</tr>
<tr>
<td>Reynolds stress model (7 eq.)</td>
<td>0.22407</td>
</tr>
</tbody>
</table>

**Discussion** Newer versions of FlowLab may give slightly different results. It would be good if we had experimental data with which to compare, but we do not.
Solution  We are to examine entrance the length for laminar pipe flow.

Analysis  We run FlowLab using template Pipeline_laminar_developing. Table 1 shows the entrance length calculations, along with a comparison with theory. The results are very good, for the lower values of Re, as is expected for laminar pipe flow. Note that the results are somewhat subjective since the end of the developing region is hard to estimate. Figure 1 shows the pressure along the axis as a function of downstream distance along the pipe axis for the case in which Re = 500. It is clear that the pressure drop is more severe (higher slope) at the beginning section of the pipe, but it is hard to tell at what axial location the flow becomes fully developed from this type of a plot. Thus, the entrance length is more appropriately determined by studying the velocity profiles.

Figure 2 shows the velocity profiles at Re = 500. We can clearly see that by x = 30, the velocity profiles have pretty much merged from then on – the flow is fully developed. Since pipe diameter $D = 1$ m, this represents an entrance length of approximately $L_e/D = 30$, which agrees with theory. FlowLab does some internal calculations do determine the entrance length, which yields $L_e/D = 28.7$ for this case, as shown in Table 1.

Discussion  Newer versions of FlowLab may give slightly different results. Also, student answers may vary considerably because of the subjectivity of these results, as mentioned above. There are no shed vortices, unsteadiness, or non-symmetries in straight laminar pipe flow, so it is no surprise that the CFD calculations perform so well.

TABLE 1  Entry length vs. Reynolds number for developing laminar pipe flow.

<table>
<thead>
<tr>
<th>Re</th>
<th>Le/D, CFD</th>
<th>Le/D, Theory</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>28.7</td>
<td>30</td>
</tr>
<tr>
<td>1000</td>
<td>58.9</td>
<td>60</td>
</tr>
<tr>
<td>1500</td>
<td>86.0</td>
<td>90</td>
</tr>
<tr>
<td>2000</td>
<td>105.5</td>
<td>120</td>
</tr>
</tbody>
</table>

FIGURE 1  Pressure as a function of axial distance down a long straight pipe at Re = 500.

FIGURE 2  Velocity profiles versus axial distance down a long straight pipe at Re = 500.
Solution  We are to study the entrance region in turbulent pipe flow.

Analysis  We run FlowLab using template Pipe_turbulent_developing. Table 1 shows the entrance length calculations, along with a comparison with the empirical equation. The CFD results are consistently low compared to the empirical results, although the agreement improves as Reynolds number increases. Since the grid is sufficiently resolved and $y'$ values are appropriate, the discrepancies are due to deficiencies in the turbulence model. It is also unclear exactly how the entrance lengths are defined internally in FlowLab.

In the plots below, we show the velocity profiles at various downstream locations, along with the pressure distribution as a function of downstream location. The results are not consistent. The profiles do not seem to become fully developed until very far downstream (at $x/D$ about 30), but the pressure appears to become linear before $x/D = 9$. FlowLab calculates the entrance length internally, and its results are shown in Table 1.

Discussion  Newer versions of FlowLab may give slightly different results.

<table>
<thead>
<tr>
<th>Re</th>
<th>Le/D, CFD</th>
<th>Le/D, Exper.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>10.4657</td>
<td>15.61796</td>
</tr>
<tr>
<td>2500</td>
<td>10.5442</td>
<td>16.20974</td>
</tr>
<tr>
<td>3000</td>
<td>10.6328</td>
<td>16.70986</td>
</tr>
<tr>
<td>5000</td>
<td>10.9546</td>
<td>18.19482</td>
</tr>
<tr>
<td>10000</td>
<td>13.2312</td>
<td>20.42299</td>
</tr>
<tr>
<td>20000</td>
<td>18.3477</td>
<td>22.92403</td>
</tr>
<tr>
<td>50000</td>
<td>21.7255</td>
<td>26.70634</td>
</tr>
</tbody>
</table>
Solution We are to study fully developed laminar pipe flow, and compare CFD calculations of Darcy friction factor with theory.

Analysis We run FlowLab using template Pipe_laminar_developed. Table 1 shows the CFD calculations of \( f \), along with a comparison with theory. The CFD and theoretical results agree very well, since that the grid is well-enough resolved, and the flow is laminar. The values agree to within 0.05%, which is as good as one can ever expect from a CFD analysis.

**TABLE 1**
Darcy friction factor vs. Reynolds number for fully developed laminar pipe flow – comparison between CFD calculations and theory \((f = 64/Re)\).

<table>
<thead>
<tr>
<th>Re</th>
<th>( f ), CFD</th>
<th>( f ), Theory</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>0.127944</td>
<td>0.128</td>
<td>-0.04375</td>
</tr>
<tr>
<td>750</td>
<td>0.0852963</td>
<td>0.085333</td>
<td>-0.0434</td>
</tr>
<tr>
<td>1000</td>
<td>0.0639721</td>
<td>0.064</td>
<td>-0.04359</td>
</tr>
<tr>
<td>1250</td>
<td>0.0511774</td>
<td>0.0512</td>
<td>-0.04414</td>
</tr>
<tr>
<td>1500</td>
<td>0.0426476</td>
<td>0.042667</td>
<td>-0.04469</td>
</tr>
<tr>
<td>2000</td>
<td>0.0319855</td>
<td>0.032</td>
<td>-0.04531</td>
</tr>
</tbody>
</table>

Discussion Newer versions of FlowLab may give slightly different results. There are no shed vortices, unsteadiness, or non-symmetries in straight laminar pipe flow, so it is no surprise that the CFD calculations perform so well.

Solution We are to study fully developed turbulent pipe flow, with smooth walls and compare CFD calculations of Darcy friction factor with experimentally determined values.

Analysis We run FlowLab using template Pipe_turbulent_developed. Table 1 shows the CFD calculations of \( f \), along with a comparison with the empirical formula (the Colebrook equation). The CFD and theoretical results agree reasonably well (within less than 10% for all cases tested), and the agreement improves with increasing Reynolds number. The \( k-\varepsilon \) model performs better at the higher values of Reynolds number because the turbulence model is calibrated for high Re flows.

**TABLE 1**
Darcy friction factor vs. Reynolds number for fully developed turbulent pipe flow.

<table>
<thead>
<tr>
<th>Re</th>
<th>( f ), CFD</th>
<th>( f ), Exper.</th>
</tr>
</thead>
<tbody>
<tr>
<td>5000</td>
<td>0.040007</td>
<td>0.0374</td>
</tr>
<tr>
<td>10000</td>
<td>0.03293</td>
<td>0.0309</td>
</tr>
<tr>
<td>50000</td>
<td>0.022029</td>
<td>0.0209</td>
</tr>
</tbody>
</table>

Discussion Newer versions of FlowLab may give slightly different results. Ten percent agreement is excellent, considering that the Colebrook equation (or the Moody chart) is accurate to only about 15% to begin with.
Solution  We are to compare turbulence models in CFD calculations of a flat plate turbulent boundary layer.

Analysis  We run FlowLab using template Plate_turbulence_models. The calculations for drag coefficient and boundary layer thickness are shown in Table 1 for four turbulence models: standard $k$-$\varepsilon$, standard $k$-$\omega$, Spallart-Allmaras, and the Reynolds stress model (RSM). The first two are two-equation models, the third is a one-equation model, and the fourth is a full Reynolds stress model (5 equations for the two-dimensional case). All turbulence models are approximations, with calibrated constants. The drag coefficients range from 0.0027 for the RSM model to 0.0029 for the $k$-$\varepsilon$ model. The empirical value is 0.0029. The dimensionless boundary layer thickness ranges from 0.012 for the $k$-$\omega$ model to 0.014 for the $k$-$\varepsilon$ model. The empirical value is 0.015. Thus, all the turbulence models do very well at predicting this flow, and the $k$-$\varepsilon$ model performs the best, overall. The RSM model, in spite of its increased complexity, does not do as well as some of the simpler models. Since all turbulence models are calibrated with the turbulent flat plate boundary layer, it is not surprising that all of them give reasonable results.

Discussion  Newer versions of FlowLab may give slightly different results.

<table>
<thead>
<tr>
<th>Fluid</th>
<th>δ_f/L</th>
<th>δ_u/L</th>
</tr>
</thead>
<tbody>
<tr>
<td>air</td>
<td>0.017313</td>
<td>0.015412</td>
</tr>
<tr>
<td>water</td>
<td>0.007664</td>
<td>0.015412</td>
</tr>
</tbody>
</table>

Solution  We are to compare the velocity and thermal boundary layer thicknesses on a heated flat plate, laminar flow.

Analysis  We run FlowLab using template Plate_laminar_temperature. The results are shown in Table 1. We denote the thermal boundary layer thickness as $\delta_f/L$, and compare to the velocity boundary layer thickness, which we denote as $\delta_u/L$. The velocity (or momentum) boundary layer thicknesses are identical, as expected, since the Reynolds number is the same for either fluid. However, since the Prandtl number ($Pr = \nu/\kappa$) of water is close to 1000, which is much greater than one, the velocity boundary layer thickness is much greater than the thermal boundary layer thickness for the water flow. In other words, momentum diffuses into the free-stream flow much more rapidly than does temperature since the Prandtl number is large. The two thicknesses are nearly equal for the air, although the thermal boundary layer thickness is somewhat higher than the momentum boundary layer thickness. This is to be expected since the Prandtl number in air is about 0.70, which means that temperature diffuses more rapidly than does momentum since $Pr < 1$.

Discussion  Newer versions of FlowLab may give slightly different results. Since the flow is laminar and the grid is well resolved with a large computational domain, these results are nearly “exact”.
Solution We are to compare the velocity and thermal boundary layer thicknesses on a heated flat plate, turbulent flow.

Analysis We run FlowLab using template Plate_turbulent_temperature. The results are shown in Table 1. We denote the thermal boundary layer thickness as $\delta_T / L$, and compare to the velocity boundary layer thickness, which we denote as $\delta_u / L$. The velocity (or momentum) boundary layer thicknesses are identical, as expected, since the Reynolds number is the same for either fluid. The velocity boundary layer thickness is slightly greater than the thermal boundary layer thickness for the water flow, while the two thicknesses are nearly equal for the air, although the thermal boundary layer thickness is slightly greater than the momentum boundary layer thickness. These small differences are attributed to differences in the Prandtl number, as explained for the laminar case of the previous problem. However, turbulent diffusion dominates over laminar diffusion, and therefore the effect of Prandtl number is minimal. The thermal and momentum boundary layer thicknesses are nearly equal in both fluids since turbulent diffusion effects dominate laminar (molecular) diffusion effects, especially at high Re.

Discussion Newer versions of FlowLab may give slightly different results. Mass (species), momentum (velocity), and energy (temperature) diffuse nearly equally in a turbulent flow since the large turbulent eddies cause rapid mixing through the boundary layer. Turbulence models do not actually calculate the details of the unsteady flow caused by these turbulent eddies; rather, they model the increased diffusion effects with approximations that enable the calculations to be performed in reasonable time on a computer.

---

Solution We are to calculate and plot pressure drop down a pipe with an elbow in turbulent flow.

Analysis We run FlowLab using template Elbow. A plot of pressure as a function of axial distance along the upstream pipe, through the elbow, and through the downstream pipe is shown in Fig. 1. The pressure losses are nearly linear through the entrance region. In the elbow itself, and just downstream of it, the pressure drops rapidly. The rate of pressure drop in the pipe section downstream of the elbow is about the same as that upstream. Therefore, it appears that most of the pressure drop occurs near the region of the elbow.

Discussion Newer versions of FlowLab may give slightly different results.
Solution  
We are to study the counter-rotating vortices in a turbulent pipe flow downstream of an elbow.

Analysis  
We run FlowLab using template Elbow. Velocity vectors at several cross sections are plotted in Fig. 1. There are no counter-rotating eddies upstream of the elbow. However, they are formed as the fluid passes through the elbow, and are very strong just downstream of the elbow. These vortices decay in strength down the pipe after the elbow, but they persist for a very long time, and may influence the accuracy of flow meters downstream of an elbow. This is why many manufacturers of pipe flow meters recommend that their flow meter be installed at least 10 or 20 pipe diameters downstream of an elbow – to avoid influence of the counter-rotating eddies.

Discussion  
The counter-rotating eddies lead to additional irreversible head loss as they dissipate.
Solution  We are to calculate the minor loss coefficient through a pipe elbow in turbulent flow.

Analysis  We run FlowLab using template Elbow. The value of $K_L$ given in Chap. 8 is 0.30. For the pipe with the elbow, the pressure drop calculated by the CFD code is 0.2805 Pa. For the straight pipe, the pressure drop is calculated to be 0.2245 Pa. Subtracting these yields 0.0560 Pa. We convert this pressure drop to a minor loss coefficient using the equation

$$K_L = \frac{h_i}{V^2/2g} \quad \text{and} \quad h_i = \frac{\Delta P}{\rho g} \quad \rightarrow \quad K_L = \frac{2\Delta P}{\rho V^2}$$

In this problem, $\rho = 998.2 \text{ kg/m}^3$, $\mu = 1.003 \times 10^{-3} \text{ kg/m} \cdot \text{s}$, the Reynolds number is 20,000, and the pipe diameter is 1.0 m. Thus,

$$\text{Re} = \frac{\rho V D}{\mu} \quad \rightarrow \quad V = \frac{\mu \text{Re}}{\rho D} = \frac{1.003 \times 10^{-3} \text{ kg/m} \cdot \text{s} (20,000)}{998.2 \text{ kg/m}^3 (1.0 \text{ m})} = 0.020096 \text{ m/s}$$

Thus,

$$K_L = \frac{2\Delta P}{\rho V^2} = \frac{2 \left( 0.0560 \frac{\text{N}}{\text{m}^2} \right)}{\left( 998.2 \frac{\text{kg}}{\text{m}^3} \right) \left( 0.020096 \frac{\text{m}}{\text{s}} \right)^2} \approx 0.27778$$

Note: This value also agrees with the calculations performed in FlowLab when you follow the directions in the panel called “How to Operate?”. So, the value of $K_L$ predicted by our CFD calculation for the standard $k$-$\varepsilon$ turbulence model (to three significant digits) is 0.278 – a difference of around 7 percent when compared to the value of 0.30 from Chapter 8, and well within the accuracy of the Colebrook equation, which is about 15%, and the accuracy of the tabulated minor loss coefficients, which is often much greater than 15%. This agreement is better than expected, considering that this is a very complex 3-D flow, and turbulence models may not necessarily apply for such problems.

Discussion  The agreement here is excellent – CFD does not always match so well with experiment, especially in turbulent flow when using turbulence models, since turbulence models are approximations that often lead to significant error.
15-58

**Solution** We are to compare various turbulence models – how well they predict the minor loss in a pipe elbow.

**Analysis** We run FlowLab using template *Elbow*. The results for four turbulence models are listed in Table 1.

**TABLE 1** Minor loss coefficient as a function of turbulence model for flow through a 90° elbow in a pipe. The error is in comparison to the experimental value of 0.30.

<table>
<thead>
<tr>
<th>Turbulence model</th>
<th>$\Delta P_{\text{elbow}}$</th>
<th>$\Delta P_{\text{pipe}}$</th>
<th>FlowLab $K_L$</th>
<th>Calculated $K_L$</th>
<th>% error compared to $K_L = 0.30$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k-\varepsilon$ (2 eq.)</td>
<td>0.28048927</td>
<td>0.22449793</td>
<td>0.277788362</td>
<td>0.2778</td>
<td>-7.40</td>
</tr>
<tr>
<td>$k-\omega$ (2 eq.)</td>
<td>0.31350237</td>
<td>0.24385084</td>
<td>0.345560304</td>
<td>0.3456</td>
<td>15.19</td>
</tr>
<tr>
<td>Reynolds stress model (7 eq.)</td>
<td>0.28830543</td>
<td>0.22364271</td>
<td>0.320809452</td>
<td>0.3208</td>
<td>6.94</td>
</tr>
<tr>
<td>Spallart-Allmaras (1 eq.)</td>
<td>0.276444</td>
<td>0.212898</td>
<td>0.315269098</td>
<td>0.3153</td>
<td>5.09</td>
</tr>
</tbody>
</table>

The Spallart-Allmaras turbulence model does the best job in predicting $K_L$. The standard $k-\omega$ turbulence model does the worst job. The Spallart-Allmaras turbulence model calculations are the best, even though this is the simplest of the models. One would hope that the more complicated model (Reynolds stress model) would do a better job than the simpler models, but this is not the case in the present problem (it does worse than the simplest model, but better than the other two, although the comparison is somewhat dubious given the accuracy of the published value of $K_L$ for an elbow. All turbulence models are approximations, with calibrated constants. While one model may do a better job in a certain flow, it may not do such a good job in another flow. This is the unfortunate state of affairs concerning turbulence models.

**Discussion** Newer versions of FlowLab may give slightly different results. Although the RSM model does not seem to be too impressive based on this comparison, keep in mind that this is a very simple flow field. There are flows (generally flows of very complex geometries and rotating flows) for which the RSM model does a much better job than any 1- or 2-equation turbulence model, and is worth the required increase in computer resources.

15-59

**Solution** We are to examine the effect of grid resolution on airfoil stall at a given angle of attack and Reynolds number.

**Analysis** We run FlowLab using template *Airfoil_mesh*. The CFD results are shown in Table 1 for the case in which the airfoil is at a 15° angle of attack at a Reynolds number of $1 \times 10^7$. The lift coefficient levels off to a value of 1.44 to three significant digits by a cell count of about 17,000. The drag coefficient levels off to a value of 0.849 to three significant digits by a cell count of about 22,000. Thus, we have shown how far the grid must be refined in order to achieve grid independence. Thus, we have achieved grid independence for a cell count greater than about 20,000.

As for the effect of grid resolution on stall angle, we see that with poor grid resolution, flow separation is not predicted accurately. Indeed, when the grid resolution is poor (under 10,000 cells in this particular case), stall is not observed even though the angle of attack (15°) is above the stall angle (14°) for this airfoil at this Reynolds number ($1 \times 10^7$). When the cell count is about 15,000, however, stall is observed. Thus, yes, grid resolution does affect calculation of the stall angle – it is not predicted well unless the grid is sufficiently resolved.

**Discussion** Newer versions of FlowLab may give slightly different results.

**TABLE 1** Lift and drag coefficients as a function of cell count (higher cell count means finer grid resolution) for the case of flow over a 2-D airfoil at an angle of attack of 15° and $Re = 1 \times 10^7$.

<table>
<thead>
<tr>
<th>Cells</th>
<th>$C_L$</th>
<th>$C_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>672</td>
<td>1.09916</td>
<td>0.242452</td>
</tr>
<tr>
<td>1344</td>
<td>1.00129</td>
<td>0.224211</td>
</tr>
<tr>
<td>2176</td>
<td>1.06013</td>
<td>0.196212</td>
</tr>
<tr>
<td>6264</td>
<td>1.02186</td>
<td>0.189023</td>
</tr>
<tr>
<td>12500</td>
<td>1.42487</td>
<td>0.0869799</td>
</tr>
<tr>
<td>16800</td>
<td>1.4356</td>
<td>0.0857791</td>
</tr>
<tr>
<td>21700</td>
<td>1.43863</td>
<td>0.0848678</td>
</tr>
<tr>
<td>24320</td>
<td>1.43719</td>
<td>0.0852304</td>
</tr>
<tr>
<td>27200</td>
<td>1.43741</td>
<td>0.0854769</td>
</tr>
</tbody>
</table>
Solution  We are to study the effect of computational domain extent on the calculation of drag in creeping flow.

Analysis  We run FlowLab using template Creep_domain. The drag coefficient is listed as a function of $R/L$ in the table. The data are also plotted. From these data we see that for $R/L$ greater than about 50, the drag coefficient has leveled off to a value of about 278 to three significant digits. This is rather surprising since the Reynolds number is so small, and the viscous effects are expected to influence the flow for tens of body lengths away from the body.

<table>
<thead>
<tr>
<th>$R/L$</th>
<th>$C_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>334.067</td>
</tr>
<tr>
<td>10</td>
<td>289.152</td>
</tr>
<tr>
<td>50</td>
<td>278.291</td>
</tr>
<tr>
<td>100</td>
<td>277.754</td>
</tr>
<tr>
<td>500</td>
<td>277.647</td>
</tr>
<tr>
<td>1000</td>
<td>277.776</td>
</tr>
<tr>
<td>2000</td>
<td>278.226</td>
</tr>
</tbody>
</table>

Discussion  When analyzing creeping flow using CFD, it is important to extend the computational domain very far from the object of interest, since viscous effects influence the flow very far from the object. This effect is not as great at high Reynolds numbers, where the inertial terms dominate the viscous terms.
15-61

Solution  We are to generate three different coarse grids for the same geometry and node distribution, and then compare the cell count and grid quality.

Analysis  The three meshes are shown in Fig. 1. The node distributions along the edges of the computational domain are identical in all three cases, and no smoothing of the mesh is performed. The structured multi-block mesh is shown in Fig. 1a. We split the domain into four blocks for convenience, and to achieve cells with minimal skewing. There are 1060 cells. The unstructured triangular mesh is shown in Fig. 1b. There is only one block, and it contains 1996 cells. The unstructured quad mesh is shown in Fig. 1c. It has 833 cells in its one block. Comparing the three meshes, the triangular unstructured mesh has too many cells. The unstructured quad mesh has the least number of cells, but the clustering of cells occurs in undesirable locations, such as at the outlets on the right. The structured quad mesh seems to be the best choice for this geometry – it has only about 27% more cells than the unstructured quad mesh, but we have much more control on the clustering of the cells. Skewness is not a problem with any of the meshes.

FIGURE 1
Comparison of three meshes: (a) structured multiblock, (b) unstructured triangular, and (c) unstructured quadrilateral.

Discussion  Depending on the grid generation software and the specified node distribution, students will get a variety of results.
Solution We are to run a laminar CFD calculation of flow through a wye, calculate the pressure drop and how the flow splits between the two branches.

Analysis We choose the structured grid for our CFD calculations. The back pressure at both outlets is set to zero gage pressure, and the average pressure at the inlet is calculated to be $-8.74 \times 10^{-5}$ Pa. The pressure drop through the wye is thus only $8.74 \times 10^{-5}$ Pa (a negligible pressure drop). The streamlines are shown in Fig. 1. For this case, 57.8% of the flow goes out the upper branch, and 42.2% goes out the lower branch.

Discussion There appears to be some tendency for the flow to separate at the upper left corner of the branch, but there is no reverse flow at the outlet of either branch. This case is compared to a turbulent flow case in the following problem.

Solution We are to run a turbulent CFD calculation of flow through a wye, calculate the pressure drop and how the flow splits between the two branches.

Analysis We choose the structured grid for our CFD calculations. The back pressure at both outlets is set to zero gage pressure, and the average pressure at the inlet is calculated to be $-3.295$ Pa. The pressure drop through the wye is thus $3.295$ Pa (a significantly higher pressure drop than that of the laminar flow, although we note that the inlet velocity for the laminar flow case was 0.002 times that of the turbulent flow case). The streamlines are shown in Fig. 1. For this case, 54.4% of the flow goes out the upper branch, and 45.6% goes out the lower branch. Compared to the laminar case, a greater percentage of the flow goes out the lower branch for the turbulent case. The streamlines at first look similar, but a closer look reveals that the spacing between streamlines in the turbulent case is more uniform, indicating that the velocity distribution is also more uniform (more “full”), as is expected for turbulent flow.

Discussion There appears to be some tendency for the flow to separate at the upper left corner of the branch, but there is no reverse flow at the outlet of either branch.

Solution We are to keep refining a grid until it becomes grid independent for the case of a laminar boundary layer.

Analysis Students will have varied results, depending on the grid generation code, CFD code, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

Solution We are to keep refining a grid until it becomes grid independent for the case of a turbulent boundary layer.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.
Solution We are to study ventilation in a simple 2-D room using CFD, and using a structured rectangular grid.

Analysis Students will have varied results, depending on the grid generation code, CFD code, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

Solution We are to repeat the previous problem except use an unstructured grid, and we are to compare results.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

Solution We are to use CFD to analyze the effect of moving the supply and/or return vents in a room.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

Solution We are to use CFD to analyze a simple 2-D room with air conditioning and heat transfer.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

Solution We are to compare the CFD predictions for 2-D and 3-D ventilation.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.
Chapter 15 Computational Fluid Dynamics

15-71
Solution We are to use CFD to study compressible flow through a converging nozzle with inviscid walls. Specifically, we are to vary the pressure until we have choked flow conditions.

Analysis Students will have varied results, depending on the grid generation code, CFD code, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-72
Solution We are to repeat the previous problem, but allow friction at the wall, and also use a turbulence model. We are then to compare the results to those of the previous problem to see the effect of wall friction and turbulence on the flow.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-73
Solution We are to generate a low-drag, streamlined, 2-D body, and try to get the smallest drag in laminar flow.

Analysis Students will have varied results, depending on the grid generation code, CFD code, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-74
Solution We are to generate a low-drag, streamlined, axisymmetric body, and try to get the smallest drag in laminar flow. We are also to compare the axisymmetric case to the 2-D case of the previous problem.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-75
Solution We are to generate a low-drag, streamlined, axisymmetric body, and try to get the smallest drag in turbulent flow. We are also to compare the turbulent drag coefficient to the laminar drag coefficient of the previous problem.

Analysis Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.
15-76
Solution  We are to use CFD to study Mach waves in supersonic flow. We are also to compare the computed Mach angle with that predicted by theory.

Analysis  Students will have varied results, depending on the grid generation code, CFD code, and their choice of computational domain, etc.

Discussion  Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-77
Solution  We are to study the effect of Mach number on the Mach angle in supersonic flow, and we are to compare to theory.

Analysis  Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion  Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.
Review Problems

15-78C
Solution
(a) **False**: If the boundary conditions are not correct, if the computational domain is not large enough, etc., the solution can be erroneous and nonphysical no matter how fine the grid.
(b) **True**: Each component of the Navier-Stokes equation is a transport equation.
(c) **True**: The four-sided cells of a 2-D structured grid require less cells than do the triangular cells of a 2-D unstructured grid. (Note however, that some unstructured cells can be four-sided as well as three-sided.)
(d) **True**: Turbulence models are approximations of the physics of a turbulent flow, and unfortunately are not universal in their application.

15-79C
Solution
We are to discuss right-left symmetry as applied to a CFD simulation and to a potential flow solution.

*Analysis*  In the time-averaged CFD simulation, we are not concerned about top-bottom fluctuations or periodicity. Thus, top-bottom symmetry can be assumed. However, fluid flows do not have upstream-downstream symmetry in general, even if the geometry is perfectly symmetric fore and aft. In the problem at hand for example, the flow in the channel develops downstream. Also, the flow exiting the left channel enters the circular settling chamber like a jet, separating at the sharp corner. At the opposite end, fluid leaves the settling chamber and enters the duct more like an inlet flow, without significant flow separation. **We certainly cannot expect fore-aft symmetry in a flow such as this.**

On the other hand, potential flow of a symmetric geometry yields a symmetric flow, so it would be okay to cut our grid in half, invoking fore-aft symmetry.

*Discussion*  If unsteady or oscillatory effects were important, we should not even specify top-bottom symmetry in this kind of flow field.
Chapter 15 Computational Fluid Dynamics

15-80C Solution  We are to discuss improvements to the given computational domain.

Analysis (a) Since Gerry is not interested in unsteady fluctuations (which may be unsymmetric), he could eliminate half of the domain. In other words, **he could assume that the axis is a plane of symmetry between the top and bottom of the channel**. Gerry’s grid would be cut in size by a factor of two, leading to approximately half the required CPU time, but yielding virtually identical results.

(b) The fundamental flaw is that **the outflow boundary is not far enough downstream**. There will likely be flow separation at the corners of the sudden contraction. With a duct that is only about three duct heights long, it is possible that there will be reverse flow at the outlet. Even if there is no reverse flow, the duct is nowhere near long enough for the flow to achieve fully developed conditions. Gerry should extend the outlet duct by many duct heights to allow the flow to develop downstream and to avoid possible reverse flow problems.

Discussion The inlet appears to be perhaps too short as well. If Gerry specifies a fully developed channel flow velocity profile at the inlet, his results may be okay, but again it is better to extend the duct many duct heights beyond what Gerry has included in his computational domain.

15-81C Solution  We are to discuss a feature of modern computer systems for which nearly equal size multiblock grids are desirable.

Analysis The fastest computers are multi-processor computers. In other words, the computer system contains more than one CPU – a **parallel computer**. Modern parallel computers may combine 32, 64, 128, or more CPUs or *nodes*, all working together. In such a situation it is natural to let each node operate on one block. If all the nodes are identical (equal speed and equal RAM), the system is most efficient if the blocks are of similar size.

Discussion In such a situation there must be communication between the nodes. At the interface between blocks, for example, information must pass during the CFD iteration process.

15-82C Solution  We are to discuss the difference between multigridding and multiblocking, and we are to discuss how they may be used to speed up a CFD calculation. Then we are to discuss whether multigridding and multiblocking can be applied together.

Analysis **Multigridding** has to do with the resolution of an established grid during CFD calculations. With multigridding, **solutions of the equations of motion are obtained on a coarse grid first, followed by successively finer grids**. This speeds up convergence because the gross features of the flow are quickly established on the coarse grid (which takes less CPU time), and then the iteration process on the finer grid requires less time.

**Multiblocking** is something totally different. It refers to the creation of **two or more separate blocks or zones, each with its own grid**. The grids from all the blocks collectively create the overall grid. As discussed in the previous problem, multiblocking can have some speed advantages if using a parallel-processing computer. In addition, some CFD calculations would require too much RAM if the entire computational domain were one large block. In such cases, the grid can be split into multiple blocks, and the CFD code works on one block at a time. This requires less RAM, although information from the dormant blocks must be stored on disk or solid state memory chips, and then swapped into and out of the computer’s RAM.

There is no reason why multigridding cannot be used on each block separately. Thus, **multigridding and multiblocking can be used together**.

Discussion Although all the swapping in and out requires more CPU time and I/O time, for large grids multiblocking can sometimes mean the difference between being able to run and not being able to run at all.
We are to discuss why we should spend a lot of time developing a multiblock structured grid when we could just use an unstructured grid.

There are several reasons why a structured grid is “better” than an unstructured grid, even for a case in which the CFD code can handle unstructured grids. First of all the structured grid can be made to have better resolution with fewer cells than the unstructured grid. This is important if computer memory and CPU time are of concern. Depending on the CFD code, the solution may converge more rapidly with a structured grid, and the results may be more accurate. In addition, by creating multiple blocks, we can more easily cluster cells in certain blocks and locations where high resolution is necessary, since we have much more control over the final grid with a structured grid.

As mentioned in this chapter, time spent creating a good grid is usually time well spent.

Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.
Chapter 15 Computational Fluid Dynamics

15-87
Solution  We are to study the effect of heating element angle of attack on heat transfer through a two-stage heat exchanger.

Analysis  Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion  Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-88
Solution  We are to study the effect of spin on a cylinder using CFD, and in particular, analyze the lift force.

Analysis  Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion  Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-89
Solution  We are to study the effect of spin speed on a spinning cylinder using CFD, and in particular, analyze the lift force as a function of rotational speed in nondimensional variables.

Analysis  Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

Discussion  Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.
Solution  We are to use CFD to calculate the minor loss coefficient due to a bump in laminar flow through a pipe, and to compare two cases – bump zone treated as a solid and as a fluid.

Assumptions  1 The flow is steady, laminar, and incompressible.

Analysis  We first run template Pipe_3d_bump_zone at Re = 100 and with the bump turned off (the bump zone is a fluid). We get \( \Delta P = 0.400977 \) Pa. Note that the pressure drop is positive since we define \( \Delta P \) as positive when there is a pressure drop down the pipe. Next we run the template with the bump turned on (the bump zone is a solid). We get \( \Delta P = 0.479955 \) Pa. The results from both runs are summarized below.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>No bump</th>
<th>With bump</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Delta P ) (Pa)</td>
<td>0.400977</td>
<td>0.47993</td>
</tr>
<tr>
<td>number of cells</td>
<td>54400</td>
<td>54400</td>
</tr>
<tr>
<td>CPU time (s)</td>
<td>≈ 250 to 450</td>
<td>≈ 200 to 300</td>
</tr>
</tbody>
</table>

Note: CPU time will vary significantly from machine to machine.

The average velocity through the pipe is calculated from the Reynolds number and the known fluid properties,

\[
Re = \frac{\rho V D}{\mu} \quad \Rightarrow \quad V = \frac{\mu Re}{\rho D} = \frac{0.001003 \text{ kg/m} \cdot \text{s}}{(998.2 \text{ kg/m}^3)(0.01 \text{ m})} = 0.010048 \text{ m/s}
\]

\((a)\) The number of cells is identical for both cases (54400 cells). This is because the cells inside the bump are there in either case – they are simply defined as either solid or fluid.

\((b)\) The CPU time for the case without the bump takes about 25% to 50% longer to converge, depending on the computer used and how busy it is. It is not intuitively obvious that this should be the case. We would expect the case without the bump to be “easier” to calculate since it is a simpler flow, but apparently that is not the case. For some unknown reason, the more complicated flow converges faster.

\((c)\) To calculate the minor loss coefficient, we take the difference between the two pressure drops, which represents the additional pressure drop due to the bump, since the pipes are the same length and everything else is identical except for the bump,

\[
\Delta P_{bump \, \text{effects \, only}} = \Delta P_{\text{developing + bump}} - \Delta P_{\text{developing \, only}} = 0.47993 \text{ Pa} - 0.400977 \text{ Pa} = 0.078953 \text{ Pa}
\]

Finally, by the definition of minor loss coefficient,

\[
K_{L, \, \text{entrance region}} = \frac{h_{L, \, \text{bump}}}{V^2 / (2g)} = \frac{\Delta P_{\text{bump}} / (\rho g)}{V^2 / (2g)} = \frac{\Delta P_{\text{bump}}}{\rho V^2}
\]

\[
= \frac{0.078953 \text{ Pa}}{(998.2 \text{ kg/m}^3)
\left(0.010048 \text{ m/s}\right)^2} = 0.78341 \cong 0.783
\]

This minor loss coefficient is about the same as a re-entrant inlet, but this is not a proper comparison since the tabulated data are for turbulent pipe flow whereas the calculations here are for laminar pipe flow. Nevertheless, the result seems reasonable.

Discussion  The minor loss coefficient may vary with Reynolds number – this can be checked by repeating the CFD calculations at different values of Re. CPU times will vary greatly depending on the computer used for the calculations.
15-91

**Solution**  We are to use CFD to compare two grids for laminar flow through a pipe with a bump in.

**Assumptions**  1 The flow is steady, laminar, and incompressible.

**Analysis**  We first run template `Pipe_3d_bump` at Re = 100 and with the bump turned on. We get $\Delta P = 0.479955$. Next we run template `Pipe_3d_bump_zone` at Re = 100 and with the bump turned on (the bump zone is a solid). We get $\Delta P = 0.47993$ Pa. Note that the pressure drop is positive since we define $\Delta P$ as positive when there is a pressure drop down the pipe. The results from both runs are summarized below.

<table>
<thead>
<tr>
<th>Parameter</th>
<th><code>Pipe_3d_bump</code></th>
<th><code>Pipe_3d_bump_zone</code></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta P$ (Pa)</td>
<td>0.479955</td>
<td>0.47993</td>
</tr>
<tr>
<td>number of cells</td>
<td>52160</td>
<td>54400</td>
</tr>
<tr>
<td>CPU time (s)</td>
<td>198</td>
<td>197</td>
</tr>
</tbody>
</table>

CPU times will vary greatly depending on the computer used for the calculations.

Comparisons:

(a) The cell count is not the same for the two cases (54400 cells for the zoned case and 52160 cells for the case without the separate zone). This is because the cells inside the bump are there in the zoned case – they are simply defined as either solid or fluid. However, in the single-zoned case, the volume corresponding to the bump is notched out of the computational domain – no cells inside the bump are included in the computational domain, and therefore we save some cells.

(b) The CPU time for the two cases take nearly the same time to converge. We would expect the case without the bump zone to take less time since it has fewer cells, but apparently that is not the case. For some unknown reason, the larger-celled case converges faster. I am not sure why. Note: These runs were done on a different day than those of the previous problem – hence the difference in CPU times. It seems that the CPU time varies widely from one run to another, even on the same computer. I suspect that the CPU time is greatly influenced by how many other programs are running simultaneously with FlowLab.

(c) The pressure drop is nearly identical for the two cases.

**Discussion**  CPU times will vary greatly depending on the computer used for the calculations. In fact, the CPU time varies by several seconds even when I ran the case a few times on the same machine back-to-back.

15-92

**Solution**  We are to study flow into a slot along a wall using CFD.

**Analysis**  Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

**Discussion**  Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.

15-93

**Solution**  We are to calculate laminar flow into a 2-D slot, compare with irrotational flow theory, and with results of the previous problem, and discuss the vorticity field.

**Assumptions**  1 The flow is steady and 2-D. 2 The flow is laminar.

**Analysis**  The flow field does not change much from the previous problem, except that a thin boundary layer shows up along the floor. The vorticity is confined to a region close to the floor – vorticity is negligibly small everywhere else, so the irrotational flow approximation is appropriate everywhere except close to the floor.

**Discussion**  The irrotational flow approximation is very useful for suction-type flows, as in air pollution control applications (hoods, etc.).
**Solution** We are to model the flow of air into a vacuum cleaner using CFD, and we are to compare the results to those obtained with the potential flow approximation.

**Analysis** We must include a second length scale in the problem, namely the width $w$ of the vacuum nozzle. For the CFD calculations, we set $w = 2.0$ mm and place the inlet plane of the vacuum nozzle at $b = 2.0$ cm above the floor (Fig. 1). Only half of the flow is modeled since we can impose a symmetry boundary condition along the $y$-axis. We use the same volumetric suction flow rate as in the example problem, i.e., $\dot{V}/L = 0.314$ m$^3$/s, but in the CFD analysis we specify only half of this value since we are modeling half of the flow field.

Results of the CFD calculations are shown in Fig. 2. Fig. 2a shows a view of streamlines in the entire computational plane. Clearly, the streamlines far from the inlet of the nozzle appear as rays into the origin; from “far away” the flow feels the effect of the vacuum nozzle in the same way as it would feel a line sink. In Fig. 2b is shown a close-up view of these same streamlines. Qualitatively, the streamlines appear similar to those predicted by the irrotational flow approximation. In Fig 2c we plot contours of the magnitude of vorticity. Since irrotationality is defined by zero vorticity, these vorticity contours indicate where the irrotational flow approximation is valid – namely in regions where the magnitude of vorticity is negligibly small.

We see from Fig. 2c that vorticity is negligibly small everywhere in the flow field except close to the floor, along the vacuum nozzle wall, near the inlet of the nozzle, and inside the nozzle duct. In these regions, the irrotational flow approximation is not valid. The streamlines in these regions are instead suggestive of a rotational flow field.

**Figure 1**
CFD model of air being sucked into a vertical vacuum nozzle; the $y$-axis is a line of symmetry (not to scale – the far field is actually much further away from the nozzle than is sketched here).

**Figure 2**
CFD calculations of flow into the nozzle of a vacuum cleaner; (a) streamlines in the entire flow domain, (b) close-up view of streamlines, (c) contours of constant magnitude of vorticity illustrating regions where the irrotational flow approximation is valid, and (d) comparison of pressure coefficient with that predicted by the irrotational flow approximation.

We see from Fig. 2c that vorticity is negligibly small everywhere in the flow field except close to the floor, along the vacuum nozzle wall, near the inlet of the nozzle, and inside the nozzle duct. In these regions, the irrotational flow approximation is not valid.
regions, net viscous forces are not small and fluid particles rotate as they move; the irrotational flow approximation is not valid in these regions. Nevertheless, it appears that the irrotational flow approximation is valid throughout the majority of the flow field. Finally, the pressure coefficient predicted by the irrotational flow approximation is compared to that calculated by CFD in Fig. 2d.

**Discussion** For \( x^* \) greater than about 2, the agreement is excellent. However, the irrotational flow approximation is not very reliable close to the nozzle inlet. Note that the irrotational flow prediction that the minimum pressure occurs at \( x^* \approx 1 \) is verified by CFD.

---

**15-95**

**Solution** We are to compare CFD calculations of flow into a vacuum cleaner for the case of laminar flow versus the inviscid flow approximation.

**Analysis** Students will have varied results, depending on the grid generation code, CFD code, turbulence model, and their choice of computational domain, etc.

**Discussion** Instructors can add more details to the problem statement, if desired, to ensure consistency among the students’ results.