Chapter 15 – Computational Fluid Dynamics

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

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.

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.

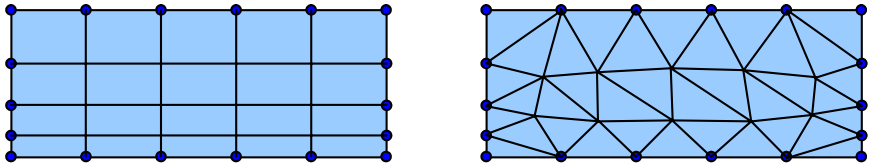
Analysis **Nodes are points along an edge of a computational domain that represent the vertices of cells**. In other words, they are the points where corners of the cells meet. **Intervals, on the other hand, are short line segments between nodes**. Intervals represent the small edges of cells themselves. 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 Fig. 1.



(a) (b)

FIGURE 1
A structured (a) and unstructured (b) grid generated for a given node distribution on the edges of the computational domain of Fig. P15-3.

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 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.

15-6C

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.

15-7C

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.

15-8C

Analysis

(*a*) 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, 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.

15-9C

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.

15-10C

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.

15-11C

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.

15-12C

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 Fig. 1.

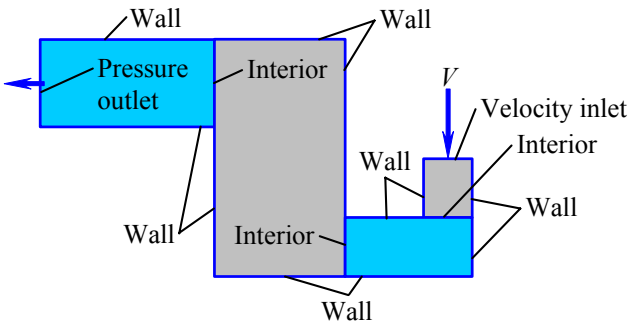


FIGURE 1
Boundary conditions specified on each edge of the 2-D multiblock computational domain of Problem 15-12.

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

15-13C

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 Problem 15-12.

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.

15-14C

Solution We are to list and briefly describe six boundary conditions, and we are to give an example of each.

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

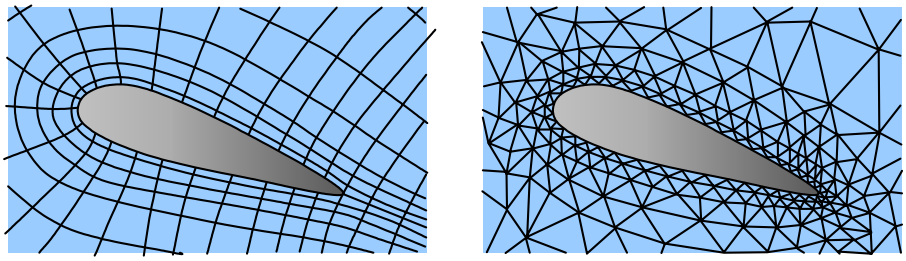
- **Axis**: Used in axisymmetric flows as the axis of rotation. Example: the axis of a torpedo-shaped body.
- **Fan**: An internal edge (2-D) or face (3-D) across which a sudden pressure rise is specified. Example: an axial flow fan in a duct.
- **Interior**: An internal edge (2-D) or face (3-D) across 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.

Discussion There are additional boundary conditions used in CFD calculations, but these are the only ones discussed in this chapter.

15-15

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.



(a) (b)

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.

15-16

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

Analysis We sketch a hybrid grid in Fig. 1. 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.

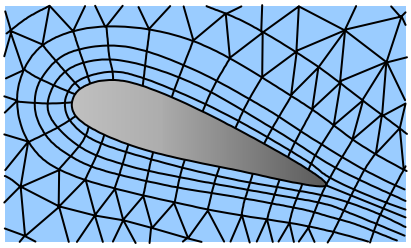


FIGURE 1
A coarse hybrid grid for the airfoil of Problem 15-15.

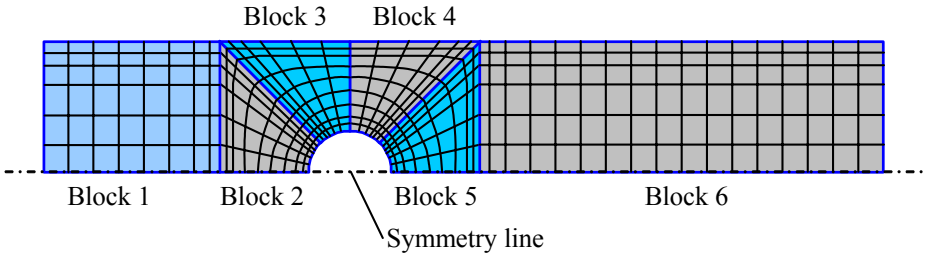
Discussion A structured grid is generally the best choice, but a hybrid grid is often a better option than a fully unstructured grid.

15-17

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**.

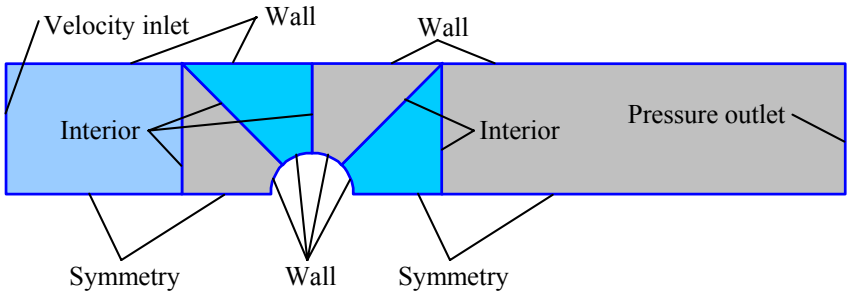
FIGURE 1
A possible blocking topology and coarse structured grid for the 2-D multiblock computational domain of Problem 15-17.



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.

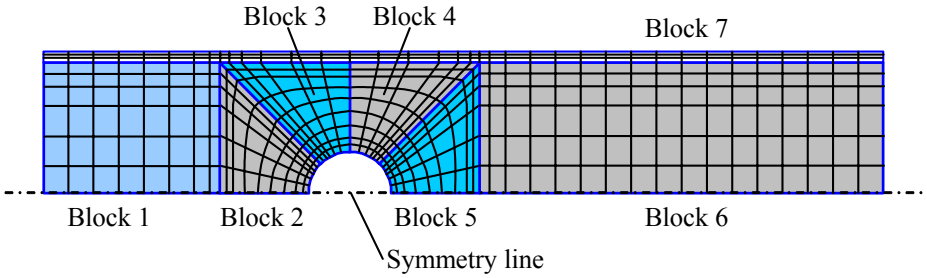
The bottom edge of the computational domain is a line of **symmetry**. The inlet is a **velocity inlet**. The outlet is a **pressure outlet**. The upper edge of the computational domain is a **wall**. The edges that define the cylinder are also **walls**. Finally, there are 5 edges that are specified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

FIGURE 2
Boundary conditions specified on each edge of the 2-D multiblock computational domain of Problem 15-17.



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).

FIGURE 3
An alternative blocking topology and coarse structured grid for the 2-D multiblock computational domain of Problem 15-17. A seventh block is added at the top for better boundary layer resolution near the top plate.



15-18

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).

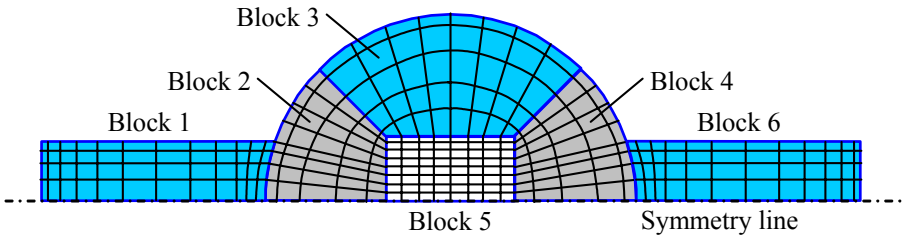


FIGURE 1
The blocking and coarse structured grid for the 2-D multiblock computational domain of Problem 15-18.

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.

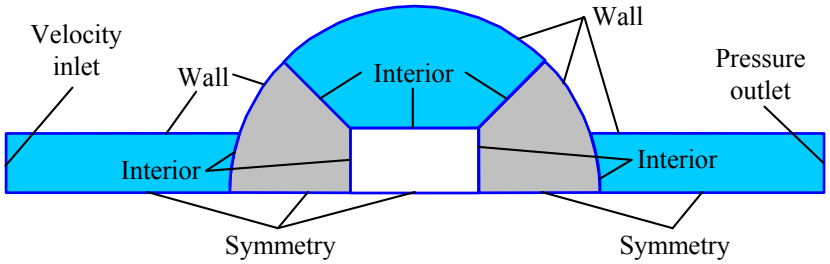


FIGURE 2
Boundary conditions specified on each edge of the 2-D multiblock computational domain of Problem 15-18.

Discussion There are of course, alternative ways to set up the blocking topology.

15-19

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.

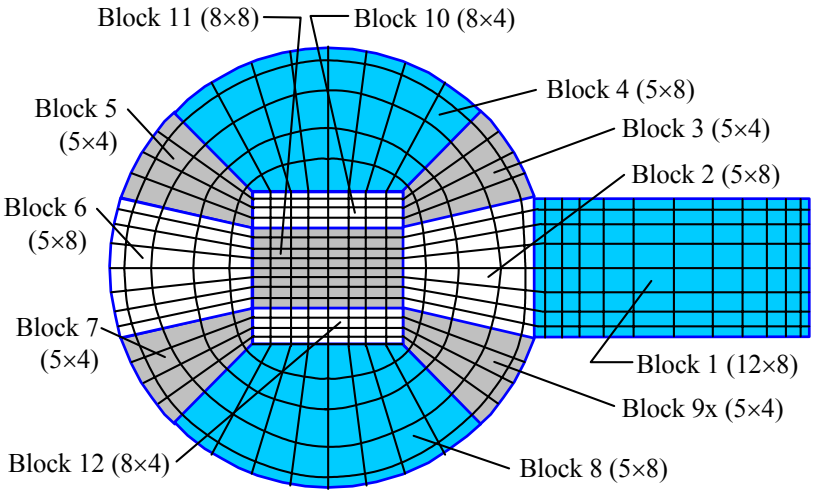


FIGURE 1
The blocking and coarse structured grid for the 2-D multiblock computational domain of Problem 15-19. Only elementary blocks are used in this grid.

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.

15-20

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.

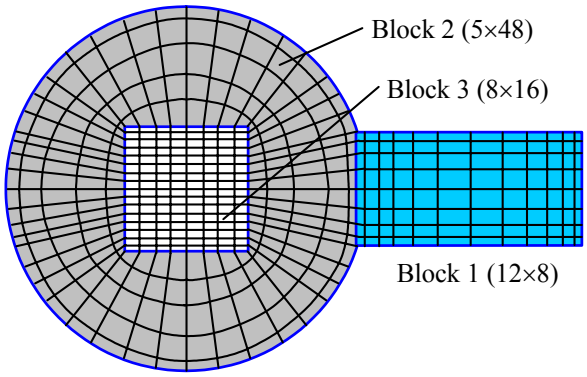


FIGURE 1
The blocking and coarse structured grid for the 2-D multiblock computational domain of Problem 15-19. Only elementary blocks are used in this grid.

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).

15-21

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.

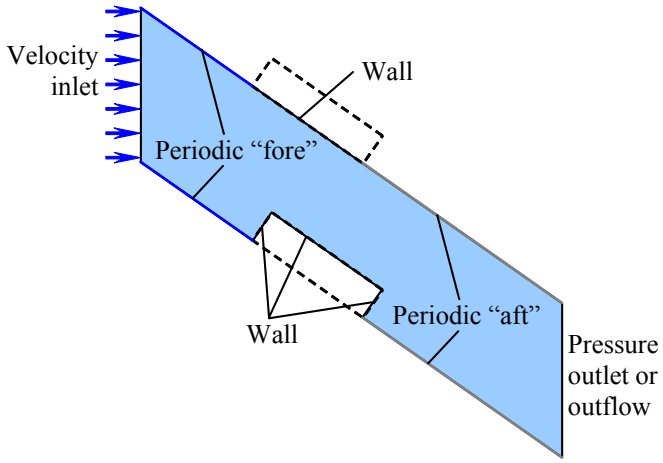


FIGURE 1
A periodic computational domain for Problem 15-21. Boundary conditions are also labeled.

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.

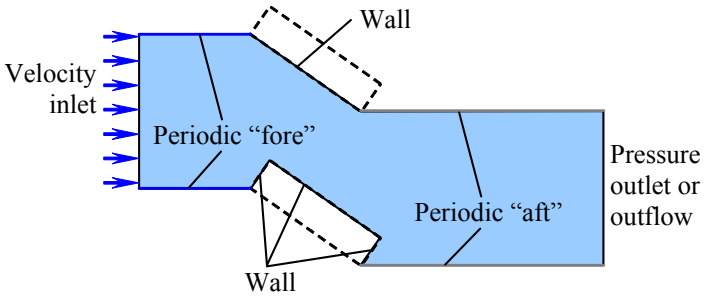


FIGURE 2
An alternative periodic computational domain for Problem 15-21. Boundary conditions are also labeled.

15-22

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, Problem 15-21. 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.

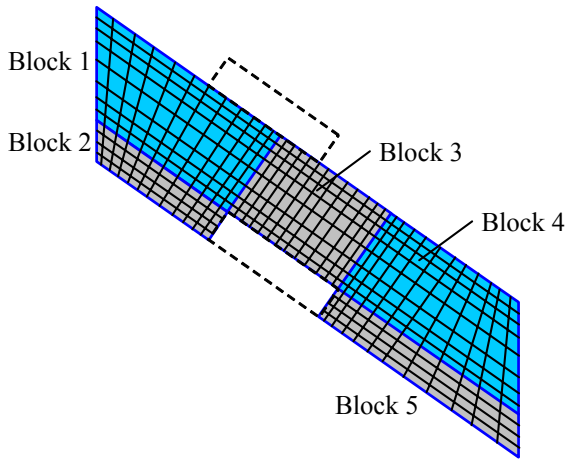


FIGURE 1
The blocking topology and a coarse structured grid for the periodic computational domain of Problem 15-22. This blocking topology applies to CFD codes that allow a block's edges to be split for application of boundary conditions, and do *not* require periodic edge pairs to have identical node 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.

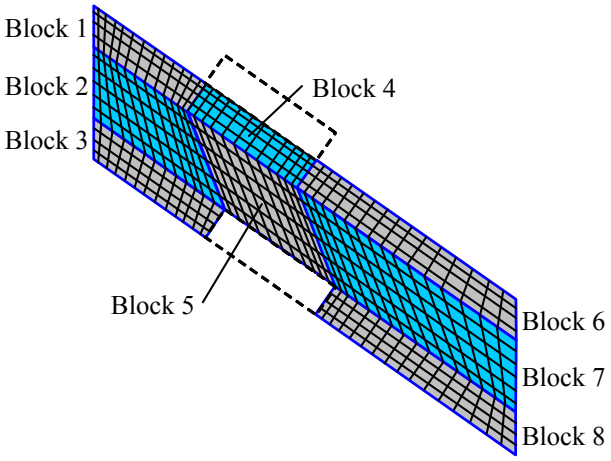


FIGURE 2
The blocking topology and a coarse structured grid for the periodic computational domain of Problem 15-22. This blocking topology applies to CFD codes that require elementary blocks and require periodic edge pairs to have identical node distributions.

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.

15-23

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^o, 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.

15-24

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.

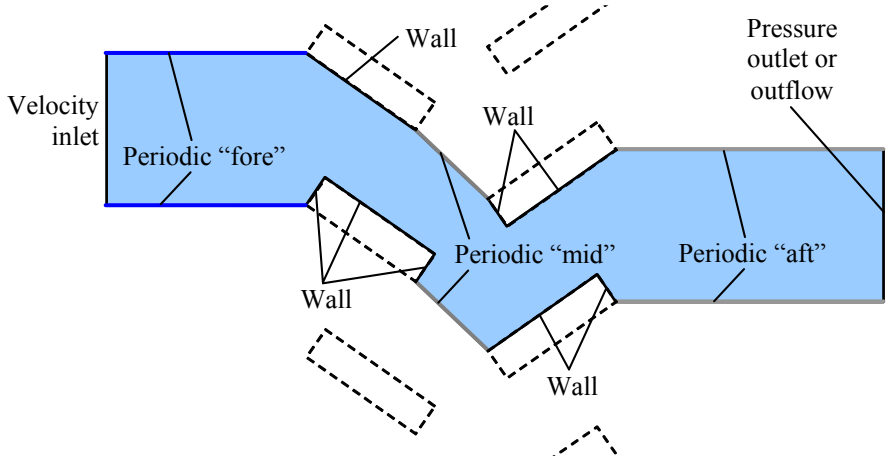


FIGURE 1
A periodic computational domain for Problem 15-24. Boundary conditions are also labeled.

Discussion Many other equally acceptable computational domains are possible.

15-25

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, Problem 15-24. 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**.

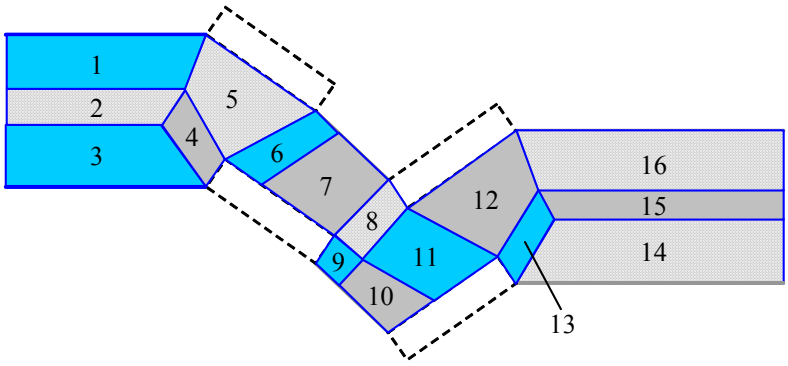


FIGURE 1
The elementary blocking topology for the periodic computational domain of Problem 15-24. This blocking topology applies to CFD codes that do not allow a block's edges to be split for application of boundary conditions, and requires periodic edge pairs to have identical node distributions.

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.

FlowLab Problems*

15-26

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis

(*a*) The Reynolds number is

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

Experimental data indicate that the drag coefficient for this body is $\boldsymbol{C_D \approx 1.9}$ 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 $\mathbf{R/D \approx 100}$ is sufficient to achieve independence of C_D. We report a final value of $\boldsymbol{C_D = 1.34}$.

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

R/D	C_D
5	1.81927
10	1.50662
20	1.41076
50	1.36723
100	1.35282
200	1.34671
500	1.34408

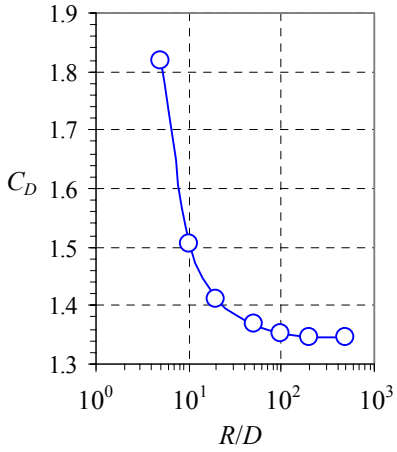
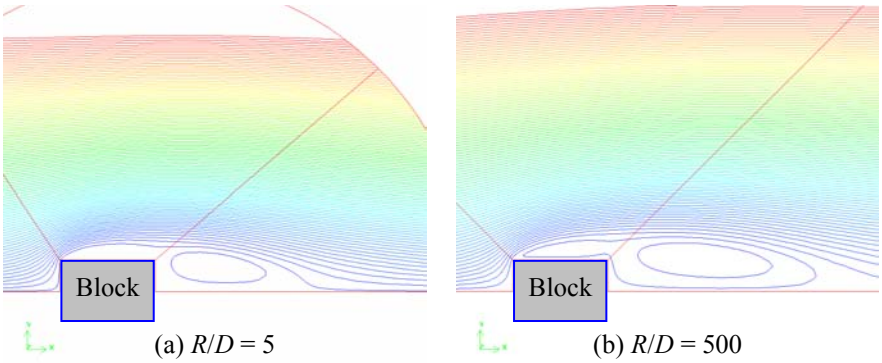


FIGURE 1
Drag coefficient plotted as a function of the normalized extent of the computational domain for turbulent flow over a rectangular block.

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



(a) $R/D = 5$ (b) $R/D = 500$

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 Problem 15-27.

15-27

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis We run the CFD code for 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 $\boldsymbol{C_D = 1.33}$ (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.

TABLE 1
Drag coefficient as a function of number of cells in the computational domain for turbulent flow over a rectangular block.

Number of cells	C_D
3120	1.45505
10400	1.35249
12480	1.34893
18720	1.33839
23920	1.33595
26000	1.33379

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

15-28

Solution We are to repeat the CFD calculation of drag coefficient around a rectangular block for two other fluids – water and kerosene, and we are to compare our results to those obtained using air as the fluid.

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 density and viscosity of the default air are $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s. The density and viscosity of liquid water at $T = 15^{\circ}$C are 998.2 kg/m^3 and 1.003×10^{-3} kg/m·s. The density and viscosity of kerosene at $T = 15^{\circ}$C are 780.0 kg/m^3 and 2.40×10^{-3} kg/m·s.

Analysis A comparison of the CFD calculations for all three fluids is given in Table 1. The drag coefficient is identical to about five digits of precision. We conclude that for incompressible flow without free surface effects, the Reynolds number is the critical parameter; **the type of fluid is irrelevant provided that the Reynolds number is the same**. This reinforces what we learned about dimensional analysis in Chap. 7.

TABLE 1
Drag coefficient as a function of fluid type for the case of turbulent flow over a rectangular block. In all cases, the Reynolds number is the same.

Fluid	Re	C_D
Air	1.37×10^4	1.34344
Water	1.37×10^4	1.34343
Kerosene	1.37×10^4	1.34343

Discussion Newer versions of FlowLab may give slightly different results. Some incompressible CFD codes work with normalized variables from the start, requiring input of a Reynolds number instead of dimensional quantities such as velocity, density, and viscosity.

15-29

Solution We are to generate CFD solutions for drag coefficient as a function of Reynolds number and compare and discuss.

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis We compare six cases in Table 1. We see that C_D levels off to a value of 1.43 to three digits of precision for Re greater than about 5×10^5. Thus, **we have achieved Reynolds number independence**, although the required Reynolds number is somewhat larger than that required experimentally.

The last two cases are peculiar in that the Mach numbers are well beyond the incompressible limit (around 0.3) since the speed of sound in air at room temperature is around 340 m/s. However, even though these flows are unphysical, the CFD code is run as incompressible, and is not "aware" of this problem since the speed of sound is treated as infinite in an incompressible flow solver. The comparison with Re is still valid since we can use *any* incompressible fluid for the calculations, as illustrated in Problem 15-28.

Discussion Newer versions of FlowLab may give slightly different results. Reynolds number independence checks are not always as simple as that shown here, because as Re increases, boundary layer thicknesses tend to decrease, requiring a finer mesh near walls. In the present problem this is not really an issue because the flow separates at the sharp edges of the block, and boundary layer thickness is not an important parameter in calculation of the drag.

TABLE 1
Drag coefficient as a function of Reynolds number for turbulent flow over a rectangular block.

Re	C_D
10000	1.30788
50000	1.40848
100000	1.42215
500000	1.43065
1.00E+06	1.43194
3.00E+06	1.43296
5.00E+06	1.43431

15-30

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis All cases are run at the same Reynolds number, namely 1.37×10^4. 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-ω 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.

TABLE 1
Drag coefficient as a function of turbulence model for flow over a rectangular block. The error is in comparison to the experimental value of 1.9.

Turbulence model	C_D	Error (%)
k-ε (2 eq.)	1.34342	-29.5%
k-ω (2 eq.)	2.536939	33.5%
Spallart-Allmaras (1 eq.)	1.360602	-28.4%
Reynolds stress model (5 eq.)	1.385374	-27.1%

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.

15-31

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis (*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.

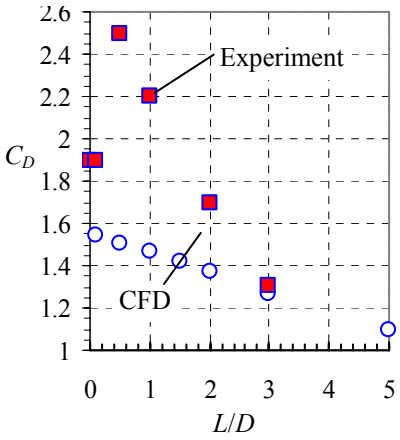


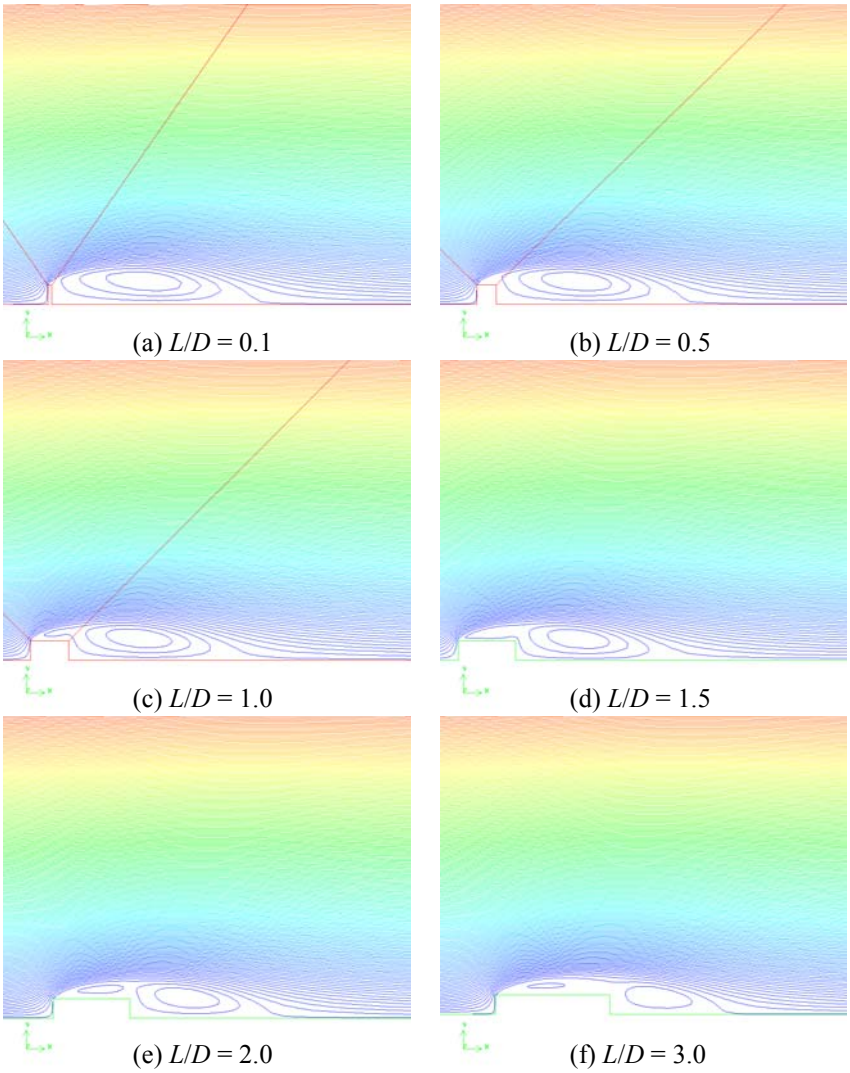
FIGURE 1
Drag coefficient as a function of L/D for turbulent flow over a rectangular block.

TABLE 1
Drag coefficient as a function of L/D for turbulent flow over a rectangular block.

L/D	C_D
0.1	1.53997
0.5	1.50858
1	1.46466
1.5	1.42077
2	1.36825
3	1.26405
5	1.09386

(*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.

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.



(a) L/D = 0.1 (b) L/D = 0.5

(c) L/D = 1.0 (d) L/D = 1.5

(e) L/D = 2.0 (f) L/D = 3.0

(*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**.

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.

15-32

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis

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

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

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

TABLE 1
Drag coefficient as a function of the normalized extent of the computational domain for turbulent flow over a cylindrical block.

R/D	C_D
5	1.0704
10	1.04053
20	1.03567
50	1.02155
100	1.00781
200	0.998819
500	0.992311

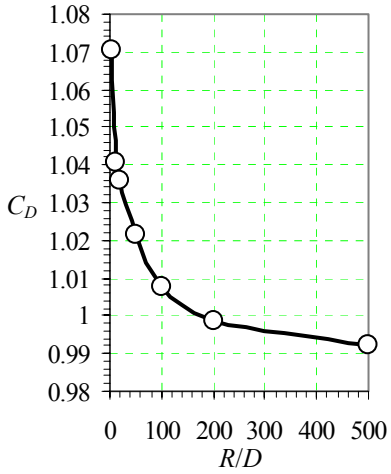


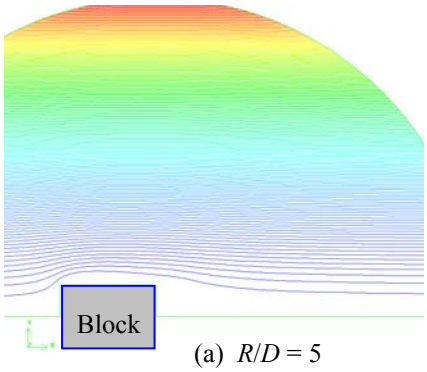
FIGURE 1
Drag coefficient plotted as a function of the normalized extent of the computational domain for turbulent flow over a rectangular block.

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.

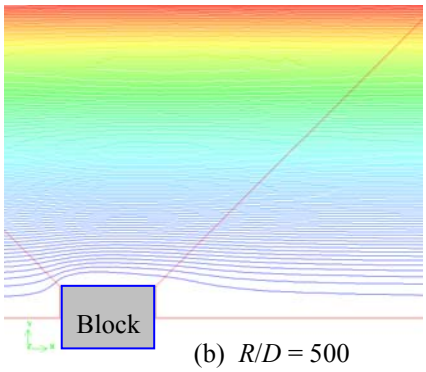
(*b*) The CFD code is run for eight values of R, all else being equal. C_D is tabulated as a function of R/D in Table 1, and plotted in Fig. 2. As the extent of the computational domain grows in size, the drag coefficient decreases steadily, but is trying to level off by $R/D \approx 500$. Thus, a computational domain extent of **R/D ≈ 500 or more** is needed to achieve independence of C_D. We report a final value of $\boldsymbol{C_D = 0.99}$. 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.



(a) $R/D = 5$



(b) $R/D = 500$

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.

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.

15-33

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis

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

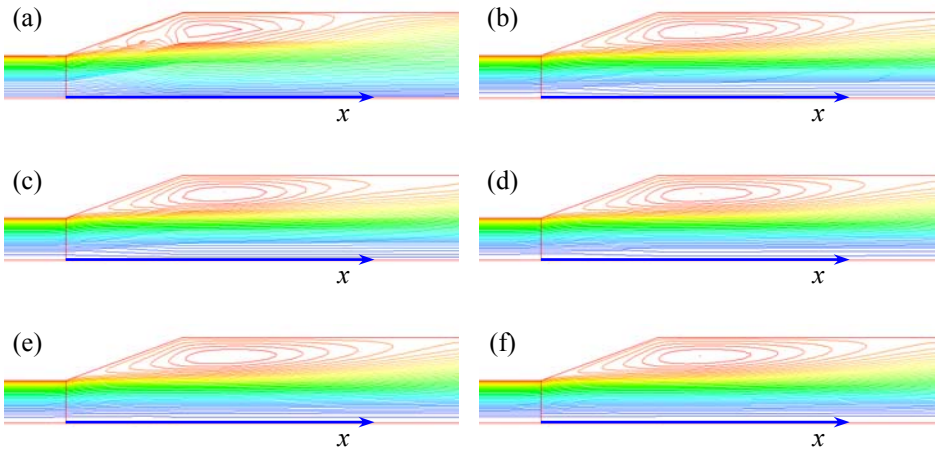


FIGURE 1
Streamlines for steady, incompressible, axisymmetric, turbulent flow through a diffuser at various levels of grid resolution; the number of cells is (a) 880, (b) 3520, (c) 4950, (d) 8800, (e) 13750, and (f) 19800.

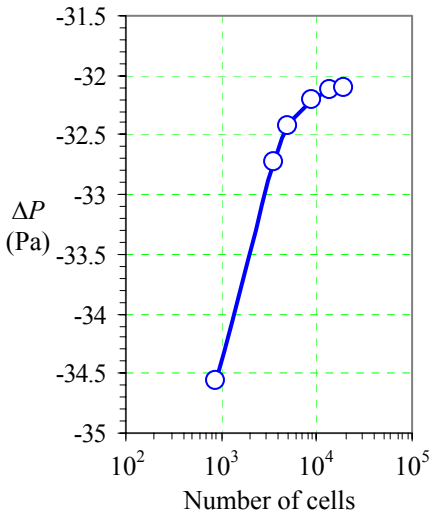


FIGURE 2
Pressure difference as a function of the number of cells in the computational domain. Turbulent flow through an axisymmetric diffuser.

TABLE 1
Pressure difference from inlet to outlet as a function of cell count for a conical diffuser with $\theta = 20^o$.

cell count	ΔP (Pa)
880	-34.5622
3520	-32.7267
4950	-32.4183
8800	-32.2021
13750	-32.1162
19800	-32.1099

At the very coarse grid resolution, the streamlines are not well defined (Fig. 1a), and the calculation is not reliable. As grid resolution improves, details of the flow separation region become more refined – the boundary layer is unable to remain attached in such a strong adverse pressure gradient. From these plots, it appears that grid independence has been achieved by about the fourth case (Fig. 1d), beyond which there is no noticeable change in the shape of the streamlines. We note that simulation of flow separation and separation bubbles is often a very difficult task for a CFD program. In this particular problem we must use an extremely fine grid in order to resolve the details of the flow separation.

(*b*) ΔP is tabulated as a function of cell count in Table 1. As grid resolution improves, ΔP increases, and becomes independent of grid resolution by the fourth or fifth mesh. This is also seen in Fig. 2 where ΔP is plotted as a function of cell count.

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

15-34

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 $\rho = 1.225$ kg/m^3 and $\mu = 1.7894 \times 10^{-5}$ kg/m·s.

Analysis For the present case (outflow boundary condition), the pressure difference is $\Delta P = P_{in} - P_{out} =$ **–32.1096 Pa**, which agrees to four digits of precision with the result of Problem 15-33 at the same grid resolution, for which $\Delta P = P_{in} - P_{out} =$ **–32.1090 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 Problem 15-33). 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. 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, ΔP 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.

15-35

Solution We are to generate CFD solutions for flow through a diffuser at various values of diffuser half-angle θ. Specifically, we are to compare streamlines and pressure difference for each case, and we are to determine the maximum value of θ 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

(*a*) Streamlines are plotted in Fig. 1 for all twelve cases.

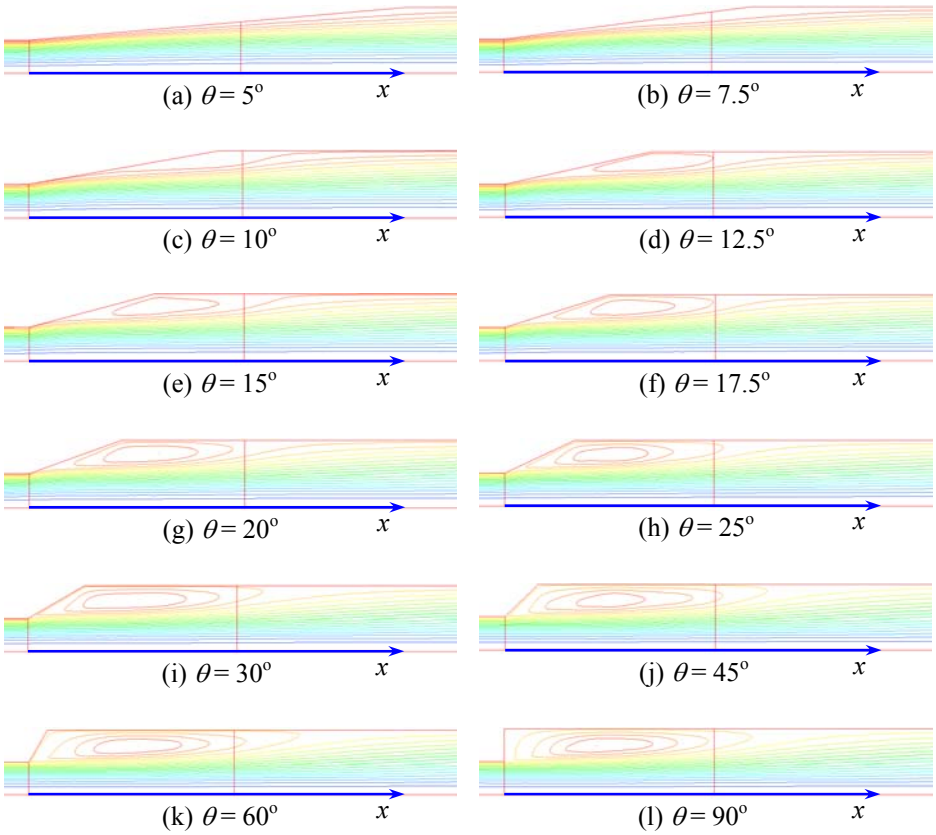


FIGURE 1
Streamlines for steady, incompressible, axisymmetric, turbulent flow through a diffuser at various values of diffuser half-angle: θ = (a) 5°, (b) 7.5°, (c) 10°, (d) 12.5°, (e) 15°, (f) 17.5°, (g) 20°, (h) 25°, (i) 30°, (j) 45°, (k) 60°, and (l) 90°. (The latter case is a sudden expansion.)

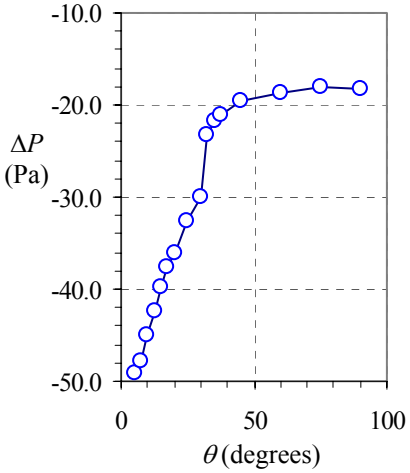


FIGURE 2
Pressure difference from inlet to outlet as a function of diffuser half-angle θ.

At θ = 5° and 7.5°, 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° or less**. As θ 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 θ = 10° (Fig. 1c). As θ 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 θ = 25° (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 θ increases (Figs. 1i through 1j). Beyond $\theta \approx 45°$ however, the streamline pattern changes very little in the separation bubble (compare Figs. 1k and 1l).

TABLE 1
Change in pressure between the inlet and the outlet of a diffuser as a function of diffuser half-angle. Data tabulated from CFD runs for steady, incompressible, axisymmetric, turbulent flow through a conical diffuser. $P_{out} = 0$ in all cases because the outlet boundary condition is a pressure outlet, and $\Delta P = P_{in} - P_{out}$.

θ	ΔP
5	-49.1371
7.5	-47.7787
10	-44.9927
12.5	-42.4013
15	-39.6981
17.5	-37.6431
20	-36.0981
25	-32.7173
30	-29.9919
32.5	-23.2118
35	-21.6434
37.5	-21.0490
45	-19.6571
60	-18.7252
75	-18.1364
90	-18.3018

(*b*) ΔP is tabulated as a function of diffuser half-angle in Table 1 (four extra cases are solved for improved clarity). As θ 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 ΔP flattens out at high values of θ, becoming nearly independent of θ beyond $\theta \approx 60^o$. This is also seen in Fig. 2 where P_{in} is plotted as a function of diffuser half-angle. There is a sharp rise in ΔP beyond 30^o. The reason for this is not certain, but is probably related to the fact that when θ is greater than about 30^o, the flow separates right at the upstream corner. The pressure rise is greater than 40 Pa for all angles below 15^o. Thus, to ensure a pressure recovery of at least 40 pascals, **Barb should recommend a diffuser half-angle of 12.5^o or less**.

Discussion The results here are for an older version of FlowLab, and the results at 20^o are therefore not exactly the same as those of Problem 15-35. Newer versions of FlowLab would give slightly different results, and the agreement would be exact. 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 ($\theta = 90^o$) there is still a pressure recovery through the diffuser (P_{in} is less than P_{out}).

TABLE 1
Pressure difference from inlet to outlet as a function of cell count for a conical diffuser with $\theta = 90^o$ (a sudden expansion).

cell count	ΔP (Pa)
996	-19.3374
2700	-18.5233
4500	-18.4956
8100	-18.4171
13500	-18.3977
17855	-18.3984

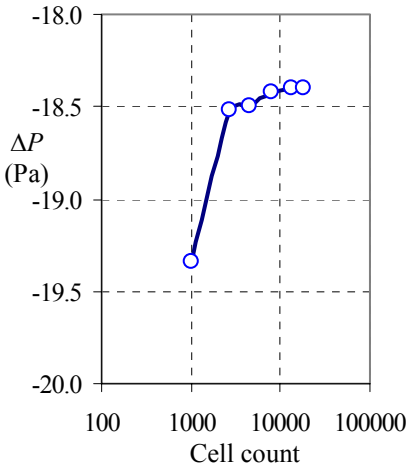


FIGURE 1
Pressure difference from the inlet to the outlet of a sudden expansion as a function of cell count.

15-36

Solution We are to perform a grid independence test on the 90° diffuser (sudden expansion) case of Problem 15-35 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·s.

Analysis We run several levels of grid refinement, tabulate the results in Table 1, and plot the results in Fig. 1. We have achieved grid independence to the third significant digit of precision by the fourth refinement level, namely, 8100 cells. The final value of ΔP 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 ($\theta = 90^o$) there is still a pressure recovery through the diffuser (P_{in} is less than P_{out}).

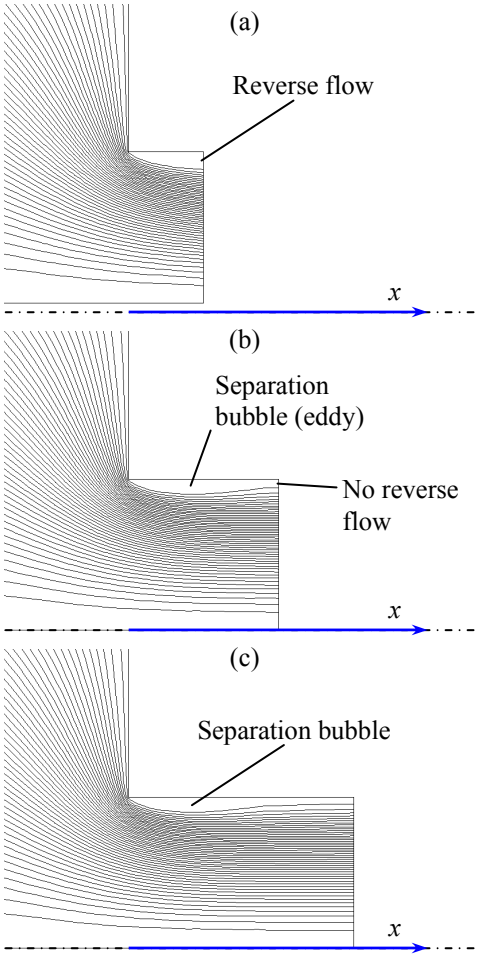


FIGURE 1
Streamlines for steady, incompressible, axisymmetric, laminar flow through a sudden contraction: L_{extend}/D_2 = (a) 0.25, (b) 0.50, and (c) 0.75.

TABLE 1
Inlet pressure, average pressure at $x = 0$, and pressure difference as functions of downstream extent of the computational domain. Data tabulated from CFD runs for steady, incompressible, axisymmetric, laminar flow through a sudden contraction in a tube.

15-37

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
(*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 in Table 1. ΔP is independent of L_{extend} (to three significant digits of precision) by $L_{extend}/D_2 = 0.5$.

L_{extend}/D_2	P_{in} (Pa, gage)	P_1 (Pa, gage)	ΔP (Pa)
0.25	664.924	212.156	452.767
0.5	622.068	171.15	450.918
0.75	597.788	146.831	450.957
1	590.532	139.464	451.068
1.25	590.036	138.86	451.176
1.5	592.523	141.24	451.283
2	601.075	149.576	451.498
2.5	609.822	158.733	451.089
3	620.434	169.248	451.186

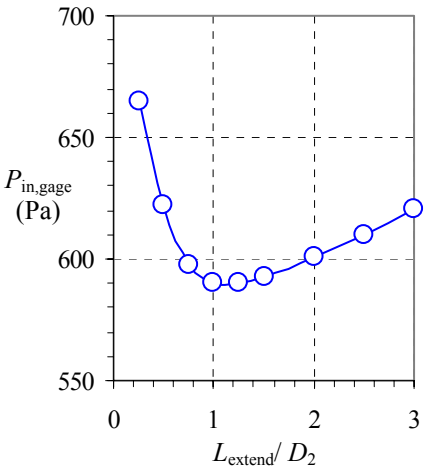


FIGURE 2
Inlet gage pressure as a function of normalized downstream extent of the computational domain.

(*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 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_{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 $L_{extend}/D_2 = 0.75$ to avoid reverse flow problems at the outlet.

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

15-38

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 $\rho = 998.2$ kg/m^3 and $\mu = 0.001003$ kg/m·s.

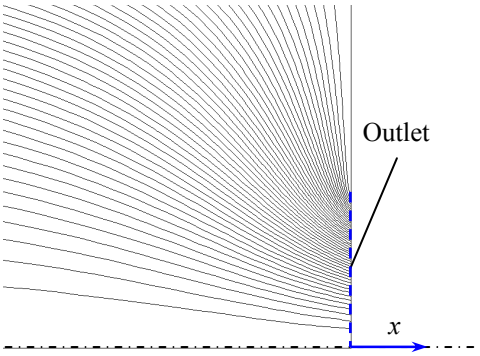


FIGURE 1
Streamlines for steady, incompressible, axisymmetric, turbulent flow through a sudden contraction at $L_{extend}/D_2 = 0$.

Analysis There is no reverse flow in the CFD simulation. The streamlines of Fig. 1 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 Pa**. Compared to the best values from Problem 15-37 ($\Delta P = 451$ Pa), **the percentage error is about 15%**.

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

15-39

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 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 Results for the three cases are summarized in Table 1. 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.

TABLE 1
Inlet pressure, average pressure at $x = 0$, and pressure difference as functions of P_{out} for the case in which $L_{extend}/D_2 = 2.0$. Data tabulated from CFD runs for steady, incompressible, axisymmetric, laminar flow through a sudden contraction in a tube.

P_{out} (Pa, gage)	P_{in} (Pa, gage)	P_1 (Pa, gage)	ΔP (Pa)
-50,000	-49,398.9	-49,850.4	451.500
0	601.075	149.576	451.498
50,000	50,601.1	50,149.6	451.496

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 Newer versions of FlowLab may give slightly different results. The slight differences (in the fifth digit) in ΔP 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.

15-40

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 $\rho = 998.2$ kg/m^3 and $\mu = 0.001003$ kg/m·s.

Analysis (*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,

$$\textit{Upstream tube}: \quad \mathrm{Re}_1 = \frac{\rho V_1 D_1}{\mu} = \frac{\left(998.2 \text{ kg/m}^3\right)\left(0.05 \text{ m/s}\right)\left(0.008 \text{ m}\right)}{0.001003 \text{ kg/m}\cdot\text{s}} = \mathbf{398.1} \qquad (1)$$

and

Downstream tube: $\mathrm{Re}_2 = \dfrac{\rho V_2 D_2}{\mu} = \dfrac{\left(998.2\ \mathrm{kg/m^3}\right)\left(0.8\ \mathrm{m/s}\right)\left(0.002\ \mathrm{m}\right)}{0.001003\ \mathrm{kg/m\cdot s}} = \mathbf{1592}$ (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: $\mathrm{Re}_1 = \dfrac{\rho V_1 D_1}{\mu} = \dfrac{\left(998.2\ \mathrm{kg/m^3}\right)\left(1.0\ \mathrm{m/s}\right)\left(0.8\ \mathrm{m}\right)}{0.001003\ \mathrm{kg/m\cdot s}} = \mathbf{796,200}$ (3)

and

Downstream pipe:

$$\mathrm{Re}_2 = \frac{\rho V_2 D_2}{\mu} = \frac{\left(998.2\ \mathrm{kg/m^3}\right)\left(16.0\ \mathrm{m/s}\right)\left(0.2\ \mathrm{m}\right)}{0.001003\ \mathrm{kg/m\cdot s}} = \mathbf{3,185,000} \quad (4)$$

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

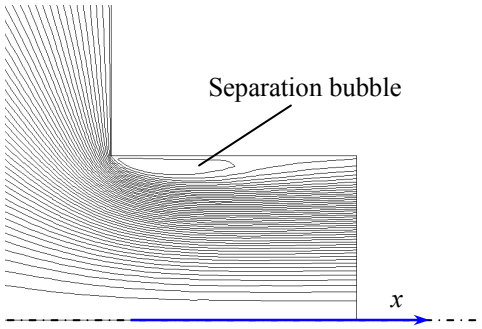


FIGURE 1
Streamlines for steady, incompressible, axisymmetric, turbulent flow through a sudden contraction at $L_{extend}/D_2 = 0.75$.

(*b*) Our CFD solutions reveal reversed flow for the smallest two cases, i.e. for L_{extend}/D_2 = 0.25 and 0.5. For higher values of L_{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_{in}, P_1, and $\Delta P = P_{in} - P_1$ are tabulated as functions of L_{extend}/D_2 in Table 1. Note that we use units of kPa instead of Pa here for convenience. ΔP is independent of L_{extend} (to three significant digits of precision) by L_{extend}/D_2 = 0. 5.

TABLE 1
Inlet pressure, average pressure at $x = 0$, and pressure difference as functions of downstream extent of the computational domain. Data tabulated from CFD runs for steady, incompressible, axisymmetric, turbulent flow through a sudden contraction in a pipe.

L_{extend}/D_2	P_{in} (kPa, gage)	P_1 (kPa, gage)	ΔP (kPa)
0.25	309.31	122.121	187.189
0.5	276.188	90.3473	185.841
0.75	251.176	65.3649	185.811
1	237.423	51.5843	185.839
1.25	228.954	43.0877	185.866
1.5	223.423	37.4649	185.958
2	216.361	30.2144	186.147

Discussion Newer versions of FlowLab may give slightly different results. The lack of scatter in the turbulent data for ΔP beyond $L_{extend}/D_2 = 0.5$ is a rather pleasant surprise; we might have expected more scatter than the laminar flow solution of Problem 15-37, since we have two additional nonlinear transport equations to solve, with their associated interactions that complicate the solution.

15-41

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·s.

Analysis For the previous case (pressure outlet boundary condition), the average pressure at the inlet is P_{in} = **251.18 kPa**, the average pressure at $x = 0$ is P_1 = **65.36 kPa**, and the pressure difference is $\Delta P = P_{in} - P_1$ = **185.82 kPa**. For the present case (outflow boundary condition), the average pressure at the inlet is P_{in} = **127.94** kPa, the average pressure at $x = 0$ is P_1 = **–57.87 kPa**, and the pressure difference is $\Delta P = P_{in} - P_1$ = **185.81 kPa**. While the actual values of pressure differ throughout the contraction, the *pressure difference* agrees to nearly five digits of precision for the two cases. Thus we conclude that **the outlet boundary condition has very little 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 Problem 15-40). 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.

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, ΔP is identical to almost five significant digits of precision. 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*.

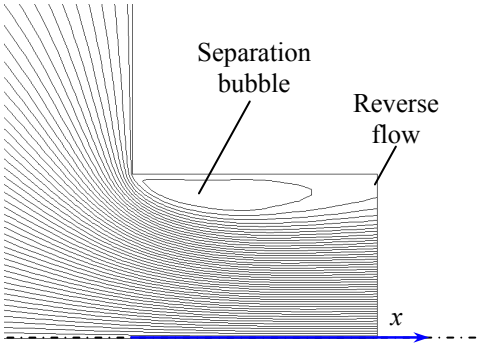


FIGURE 1
Streamlines for steady, incompressible, two-dimensional, turbulent flow through a sudden contraction at $L_{extend}/D_2 = 0.75$.

TABLE 1
Inlet pressure, average pressure at $x = 0$, and pressure difference as functions of downstream extent of the computational domain. Data tabulated from CFD runs for steady, incompressible, two-dimensional, turbulent flow through a sudden contraction in a duct.

15-42

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 $\rho = 998.2$ kg/m^3 and $\mu = 0.001003$ kg/m·s.

Analysis (*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 15-40, 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 15-40.

(*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. ΔP is independent of L_{extend} (to three significant digits of precision) by $L_{extend}/D_2 = 0.5$.

L_{extend}/D_2	P_{in} (kPa, gage)	P_1 (kPa, gage)	ΔP (kPa)
0.25	18.203	7.3241	10.8789
0.5	17.3614	6.65408	10.7073
0.75	15.3649	4.67913	10.6858
1	13.7764	3.09228	10.6841
1.25	12.9757	2.29043	10.6853
1.5	12.5034	1.81676	10.6867
2	11.9787	1.29006	10.6887
3	11.5309	0.845519	10.6853
4	11.3452	0.660043	10.6852

Discussion Newer versions of FlowLab may give slightly different results. Comparing ΔP between this problem (2-D) and Problem 15-40 (axisymmetric), we see that ΔP for the axisymmetric case is more than 17 times greater than ΔP 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.

TABLE 1
Pressure difference from inlet to outlet as a function of cell count for turbulent flow through a two-dimensional jog.

cell count	ΔP (Pa)
900	2.26846
2500	2.63636
5700	2.71292
9100	2.71739
13600	2.71835
15200	2.71828
18000	2.71856
20352	2.71746
24840	2.71544

FIGURE 1
Streamlines for steady, incompressible, two-dimensional, turbulent flow through a jog in a channel at various levels of grid resolution: cell count = (a) 900, (b) 2500, (c) 5700, (d) 9100, (e) 13,600, and (f) 18,000. The origin (0,0) is marked on each figure for reference.

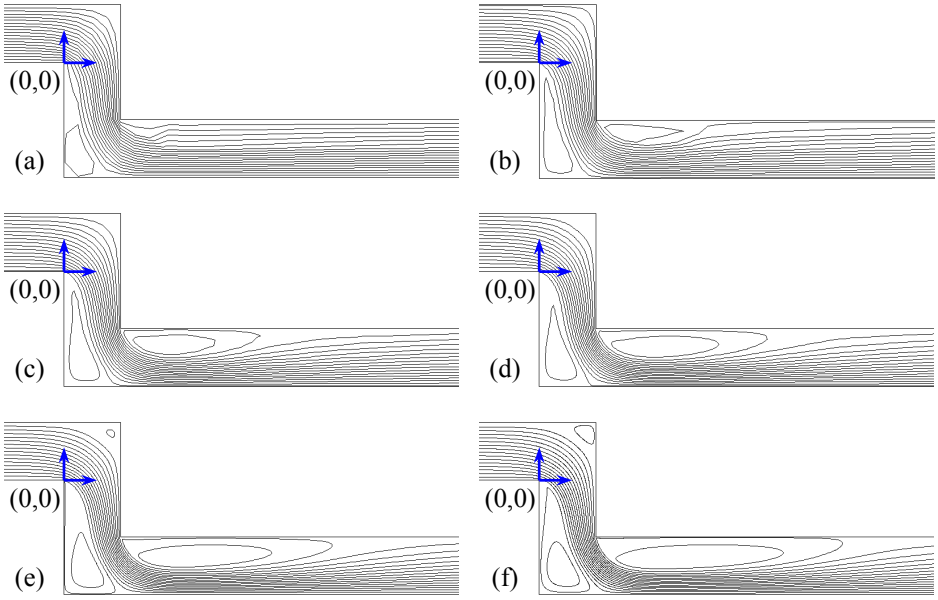
15-43

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



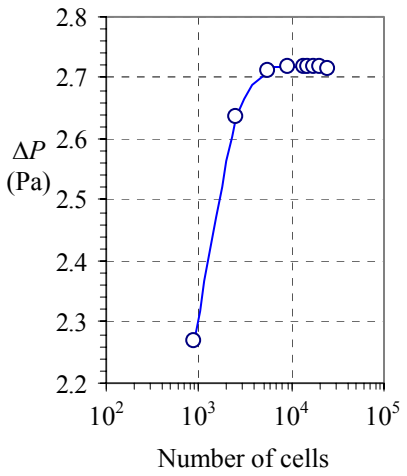


FIGURE 2
Pressure difference as a function of the number of cells in the computational domain; turbulent flow through a jog in a channel.

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*) ΔP is tabulated as a function of cell count in Table 1. As grid resolution improves, ΔP 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 ΔP 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 ΔP 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.

TABLE 1
Pressure difference from inlet to outlet as a function of cell count for laminar flow through a two-dimensional jog, $V = 0.10$ m/s.

cell count	ΔP (Pa)
272	37.6118
576	37.8096
847	37.5031
1700	36.9068
3388	36.5925
10625	36.4095
14123	36.4187
18645	36.435
23273	36.4636

15-44

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 $\rho = 998.2$ kg/m^3 and $\mu = 0.001003$ kg/m·s.

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

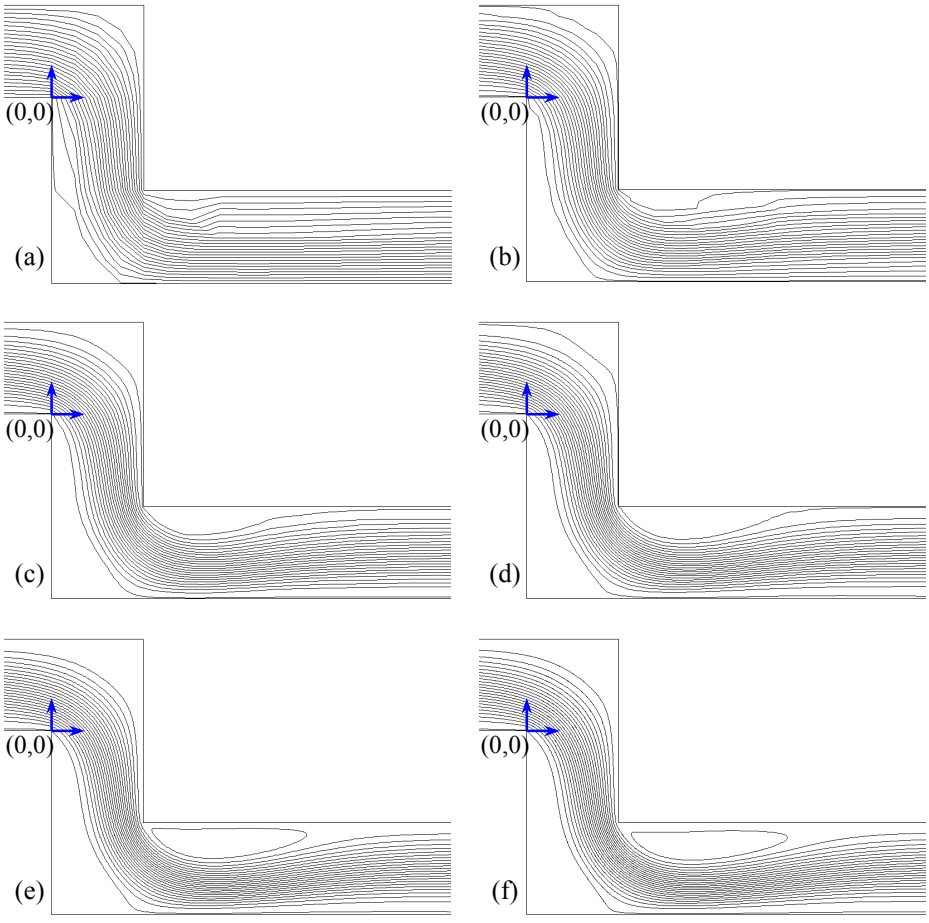


FIGURE 1
Streamlines for steady, incompressible, two-dimensional, laminar flow through a jog in a channel at various levels of grid resolution: cell count = (a) 272, (b) 576, (c) 847, (d) 1700, (e) 3388, and (f) 14,123. The origin (0,0) is marked on each figure for reference. The inlet velocity for these runs is 0.10 m/s.

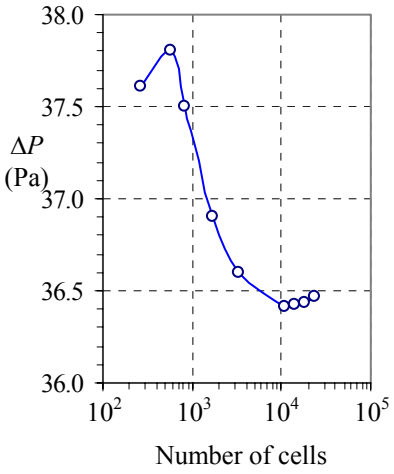


FIGURE 2
Pressure difference as a function of the number of cells in the computational domain; laminar flow through a jog in a channel. $V = 0.10$ m/s.

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 Problem 15-43. 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 refines, 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*) ΔP is tabulated as a function of grid resolution in Table 1. ΔP is also plotted as a function of number of cells in Fig. 2. As grid resolution improves, ΔP 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 for laminar flow through a two-dimensional jog, $V = 1.0$ m/s.

cell count	ΔP (Pa)
272	2025.69
576	2269.26
847	2324.99
1700	2482.44
3388	2582.17
10625	2515.71
14123	2484.86
18645	2468.57
23273	2511.98

15-45

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 $\rho = 998.2$ kg/m^3 and $\mu = 0.001003$ kg/m·s.

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

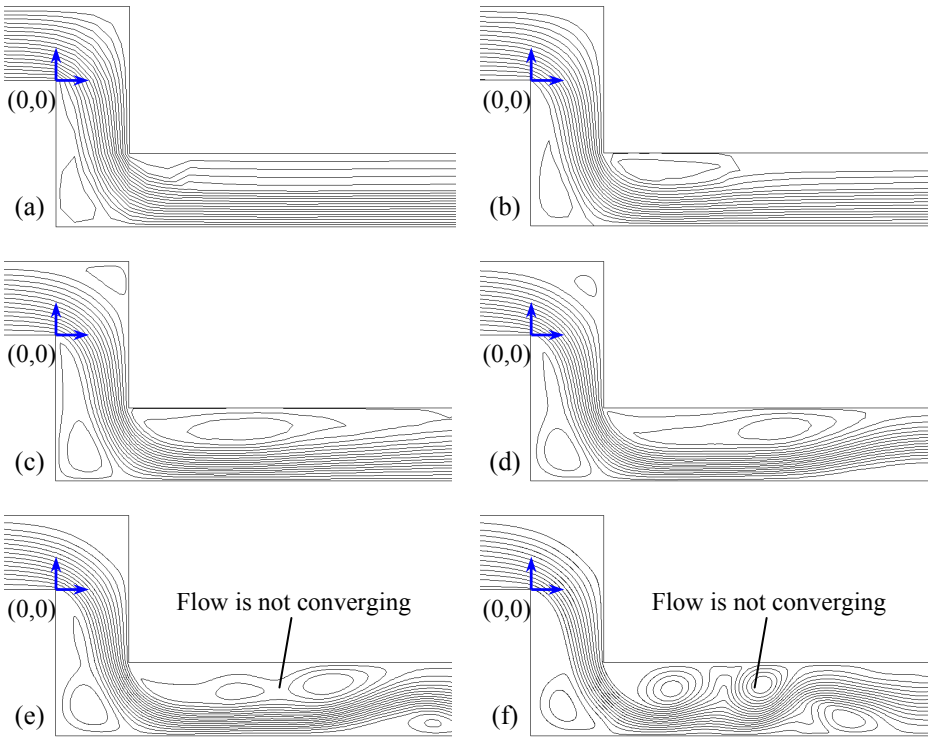


FIGURE 1
Streamlines for steady, incompressible, two-dimensional, laminar flow through a jog in a channel at various levels of grid resolution: cell count = (a) 272, (b) 576, (c) 847, (d) 1700, (e) 3388, and (f) 14,123. The origin (0,0) is marked on each figure for reference. The inlet velocity for these 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.

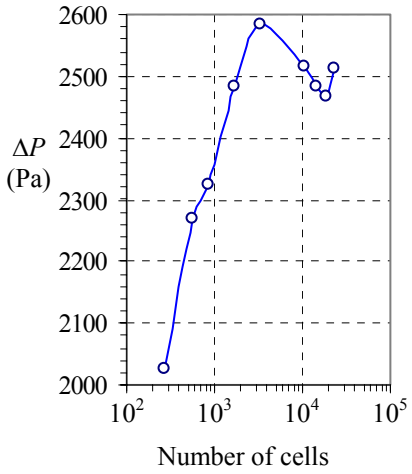


FIGURE 2
Pressure difference as a function of the number of cells in the computational domain; laminar flow through a jog in a channel. $V = 1.0$ m/s.

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*) ΔP is tabulated as a function of grid resolution in Table 1. ΔP is also plotted as a function of number of cells in Fig. 2. As grid resolution improves, ΔP 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

$$\textit{Re at } V = 1.0 \textit{ m/s}: \qquad \mathrm{Re} = \frac{\rho V D}{\mu} = \frac{\left(998.2 \text{ kg/m}^3\right)\left(1.0 \text{ m/s}\right)\left(0.001 \text{ m}\right)}{0.001003 \text{ kg/m}\cdot\text{s}} = \mathbf{995} \qquad (1)$$

and for the previous problem, it is

$$\textit{Re at } V = 0.10 \textit{ m/s}: \quad \mathrm{Re} = \frac{\rho V D}{\mu} = \frac{\left(998.2 \text{ kg/m}^3\right)\left(0.10 \text{ m/s}\right)\left(0.001 \text{ m}\right)}{0.001003 \text{ kg/m}\cdot\text{s}} = \mathbf{99.5} \qquad (2)$$

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.

TABLE 1
Pressure difference from inlet to outlet as a function of cell count for laminar flow through a two-dimensional jog, $V = 1.0$ m/s.

$P_b/P_{0,\text{inlet}}$	$\dot{m}$ (kg/s)
0.454545	9.08358
0.545455	9.08356
0.636364	9.08355
0.727273	9.08357
0.818182	9.08356
0.909091	9.08354
0.954545	9.03556
0.977273	6.91699
0.988636	4.97062
0.990909	4.45686
0.995455	3.16752

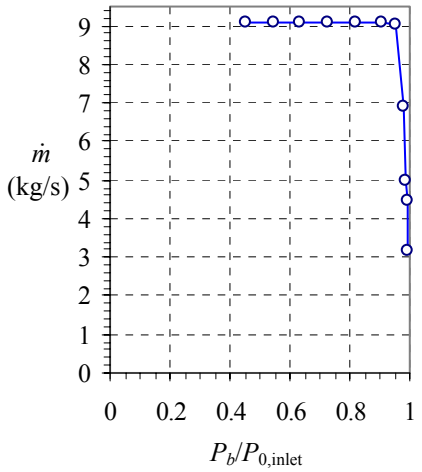


FIGURE 1
Mass flow rate as a function of back pressure ratio in a converging-diverging nozzle.

15-46

Solution We are to generate CFD solutions for compressible flow of air through a converging-diverging nozzle, and compare mass flow rate at various 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 tabulate and plot $\dot{m}$ as a function of $P_b/P_{0,\text{inlet}}$ in Table 1 and Fig. 1 respectively. We see that $\dot{m}$ **is constant as long as the flow through the throat is sonic, because the flow is choked** ($P_b/P_{0,\text{inlet}} < 0.95$). For values of $P_b/P_{0,\text{inlet}}$ around 0.95 or higher, however, the flow in the diverging section becomes subsonic (no shock waves, and not choked), and the mass flow rate decreases with increasing $P_b/P_{0,\text{inlet}}$ from there on, as expected, since the flow is subsonic.

Discussion The small variations in $\dot{m}$ in Table 1 (in the sixth digit of precision) are due to the fact that the residuals in the CFD solution do not go to zero. A finer grid and longer run times would correct this.

15-47

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

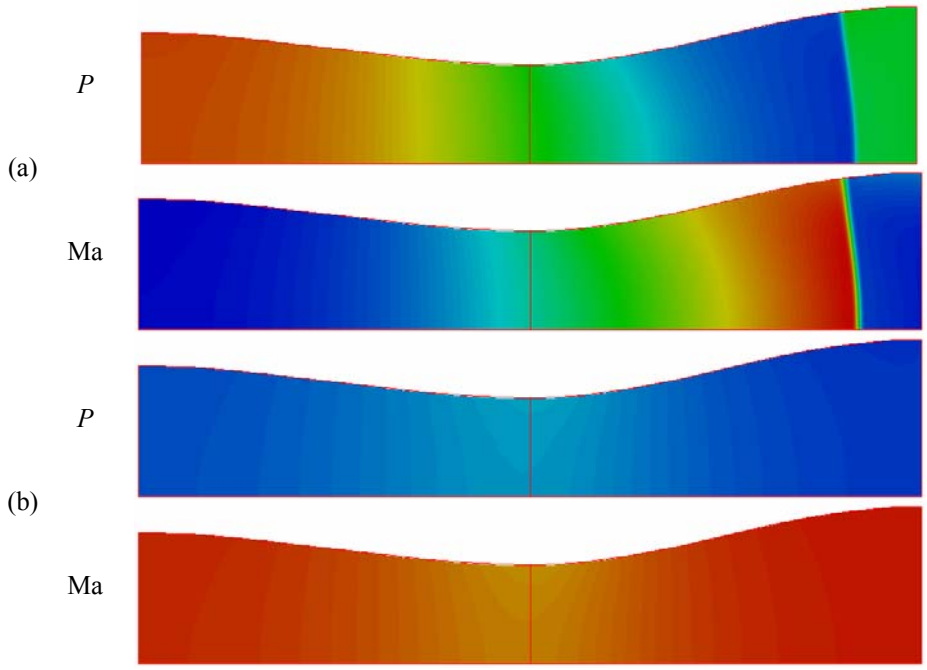


FIGURE 1
Contour plots of P and Ma for compressible flow of air through a converging-diverging nozzle: $P_b/P_{0,\text{inlet}}$ = (a) 0.455, and (b) 0.977. A normal shock is seen for the first case, since the flow is choked, but no shock is seen in the second case, since the flow is subsonic everywhere. 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. Also shown are calculations from one-dimensional inviscid theory. The agreement is excellent.

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.

15-48

Analysis The results are expected to be similar, except that for the 2-D case, the area change is much less significant. Therefore it the value of back pressure that causes the flow to choke will be different than the axisymmetric case. We should still be able to observe a normal shock in the flow at low-enough values of back pressure.

Discussion Newer versions of FlowLab may give slightly different results.

15-49

Analysis The template provides five different geometries. Model 1 has a very blunt rear end, kind of like a station wagon. As the model number increases, the rear end gets more slanted and rounded. Table 1 shows the calculated drag coefficient for each model.

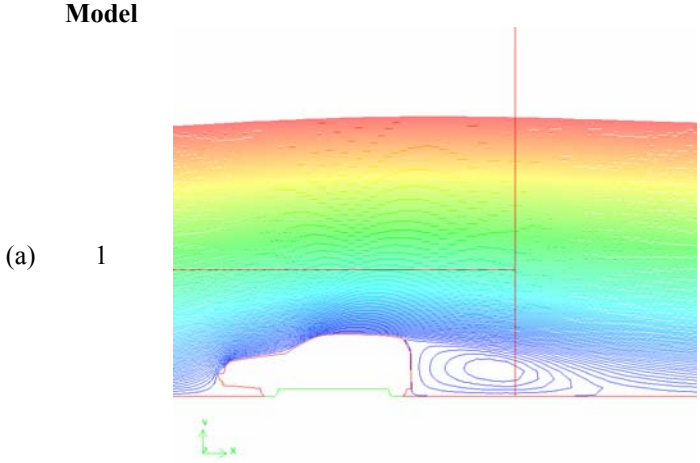
TABLE 1
Drag coefficient as a function of model number, where the rear end of the car is modified according to model number.

Model number	Description	C_D
1	Blunt rear end, like a station wagon	0.320
2	Back window with a short trunk section	0.298
3	Similar to model 2, but with a longer trunk	0.276
4	More rounded back end with a short notch for the trunk	0.180
5	Fully rounded back end	0.211

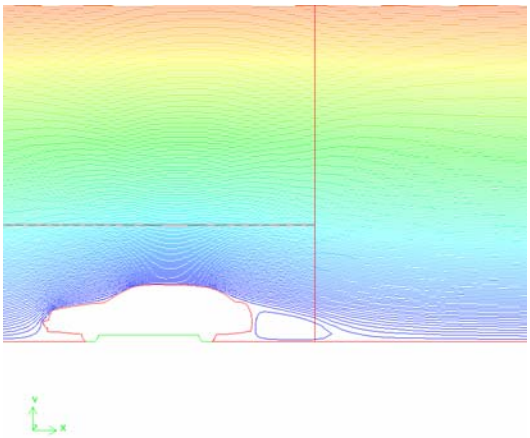
It turns out that **Model number 4 has the lowest drag coefficient**, and Model number 1 has the highest. We probably would have guessed the latter, but not the former. Namely, the car with the most blunt rear end (Model 1) has the highest drag as we might expect, but most people would predict that Model 5, which is the most rounded, would have the lowest drag. Model 4 has a short notch for the trunk, and the aerodynamics turn out such that it has the lowest drag.

Streamline plots are shown in Fig. 1 for Models 1 and 4, the highest and lowest drag cases, respectively. It is clear from these plots that the blunted rear body has a larger separation bubble in the wake (low pressure in the wake which leads to large drag). On the other hand, Model 4 has a much smaller wake and therefore much less drag.

FIGURE 1
Streamlines for two representative two-dimensional automobile shapes: (a) Model 1, and Model 4.



(b) 4



Discussion Newer versions of FlowLab may give slightly different results. It is not always immediately obvious whether a shape will have more or less drag than another shape. These results are two-dimensional, whereas actual automobiles, of course, are three-dimensional [see Problem 15-52].

TABLE 1
Drag coefficient as a function of the normalized extent of the computational domain for turbulent flow over a cylindrical block.

H/h	C_D
5	0.2101
10	0.1847
20	0.1798
50	0.1797
100	0.1798

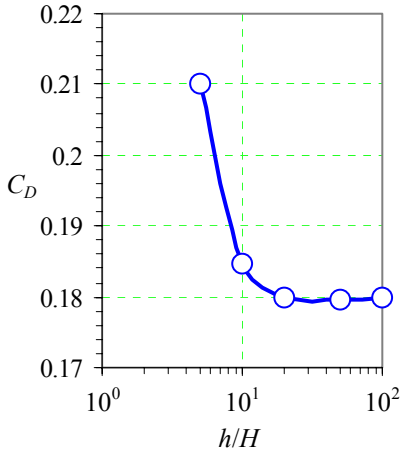


FIGURE 1
Drag coefficient plotted as a function of the normalized extent of the computational domain for turbulent flow over a two-dimensional automobile.

TABLE 1
Predicted drag coefficient on a two-dimensional automobile as a function of turbulence model.

15-50

Analysis 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 0.180 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.

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.

15-51

Analysis The results from the CFD calculations are presented in Table 1. There is some variation in the calculated values of C_D depending on the turbulence model used; **C_D ranges from 0.175 for the Spallart-Allmaras model to 0.223 for the RSM model**. The range of scatter 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 correct**, since all turbulence models are approximations, with calibrated constants. Furthermore, we have no experimental data with which to compare the CFD results.

Turbulence model	C_D
Spallart-Allmaras (1 eq.)	0.175
k-ε (2 eq.)	0.182
k-ω (2 eq.)	0.221
Reynolds stress model (7 eq.)	0.223

Discussion Newer versions of FlowLab may give slightly different results. It would be good if we had experimental data with which to compare.

15-52

Analysis The 3-D drag coefficient is **0.785**, significantly higher than the 2-D case, since flow separates off the sides of the car as well as off the top. Note too that the car model used in this analysis is more like a truncated 2-D model rather than a truly three-dimensional model. If all the sharp corners were rounded off, the drag coefficient would be much lower than that calculated here.

Discussion Newer versions of FlowLab may give slightly different results. This is not a very good drag coefficient for a modern car, and there is much improvement possible by further streamlining.

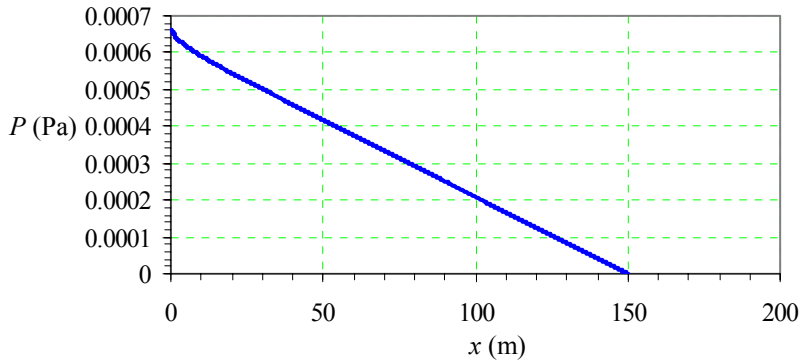
15-53

Analysis Table 1 shows the entrance length calculations, along with a comparison with theory. The results are very good, as is expected for laminar pipe flow. Note that the results are somewhat subjective since the end of the developing region is being judged "by eye". 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, as above.

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

Re	*Le*/*D*, CFD	*Le*/*D*, Theory
500	20	30
1000	50	60
1500	86	90
2000	1050	120

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



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 turbulent pipe flow.

Re	*Le*/*D*, CFD	*Le*/*D*, Exper.
2000	10	15.6
5000	11	18.2
10000	13	20.4
20000	18	22.9
50000	22	26.7

15-54

Analysis 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.

Discussion Newer versions of FlowLab may give slightly different results.

15-55

Analysis 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 ($f = 64/\mathrm{Re}$).

Re	*f*, CFD	*f*, Theory	Error (%)
500	0.127944	0.128	-0.04375
750	0.0852963	0.085333	-0.0434
1000	0.0639721	0.064	-0.04359
1250	0.0511774	0.0512	-0.04414
1500	0.0426476	0.042667	-0.04469
2000	0.0319855	0.032	-0.04531

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.

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

Re	*f*, CFD	*f*, Exper.
5000	0.040007	0.0374
10000	0.03293	0.0309
50000	0.022029	0.0209

15-56

Analysis 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-ε model performs better at the higher values of Reynolds number because the turbulence model is calibrated for high Re flows.

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.

TABLE 1
Darcy friction factor vs. dimensionless roughness height for fully developed turbulent pipe flow at Re = 1×10^6.

ε/D	f, CFD	f, Exper.
0.00005	0.0113382	0.0126
0.00025	0.0134436	0.0152
0.0005	0.016288	0.0172
0.001	0.0204516	0.0199
0.0015	0.0231213	0.0220
0.002	0.0246173	0.0236

15-57

Analysis 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 15% for all cases tested), and the agreement improves with increasing roughness. Since the agreement is within the known inaccuracy of the Colebrook equation (about 15%), these CFD results are considered adequate.

Discussion Newer versions of FlowLab may give slightly different results.

15-58

Analysis For a Reynolds number of 1×10^5, the CFD calculations give a nondimensional boundary layer thickness of δ/x = **0.0154** and a drag coefficient of C_D = **0.00435**. The theoretical values are obtained from equations in Chaps. 10 and 11, namely,

$$\frac{\delta}{x} = \frac{4.91}{\sqrt{\mathrm{Re}_x}} = \frac{4.91}{\sqrt{1\times10^5}} = 0.0155 \qquad (1)$$

and

$$C_D = C_f = \frac{1.33}{\sqrt{\mathrm{Re}_x}} = \frac{1.33}{\sqrt{1\times10^5}} = 0.00421 \qquad (2)$$

The agreement is excellent for both values, the discrepancy being less than 1% for δ/x and about 3% for drag coefficient.

Discussion Newer versions of FlowLab may give slightly different results.

15-59

Analysis For a Reynolds number of 1×10^7, the CFD calculations give a nondimensional boundary layer thickness of δ/x = **0.0140** and a drag coefficient of C_D = **0.00292**. The empirical values are obtained from equations in Chaps. 10 and 11, namely,

$$\frac{\delta}{x} = \frac{0.38}{\mathrm{Re}_x^{1/5}} = \frac{0.38}{\left(1\times10^7\right)^{1/5}} = 0.015 \qquad (1)$$

and

$$C_D = C_f = \frac{0.074}{\mathrm{Re}_x^{1/5}} = \frac{0.074}{\left(1\times10^7\right)^{1/5}} = 0.0029 \qquad (2)$$

The agreement is excellent for both values, the discrepancy being about 7% for δ/x and negligible (within 2 significant digits) for drag coefficient.

Discussion Newer versions of FlowLab may give slightly different results. We report our empirical values to only two significant digits in keeping with the level of precision and accuracy of turbulent flows.

15-60

Analysis The calculations for drag coefficient and boundary layer thickness are shown in Table 1 for four turbulence models: standard k-ε, standard k-ω, 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-ε model. The empirical value is 0.0029. The dimensionless boundary layer thickness ranges from 0.012 for the k-ω model to 0.014 for the k-ε model. The empirical value is 0.015. Thus, all the turbulence models do very well at predicting this flow, and the k-ε 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 1
Drag coefficient and boundary layer thickness for a flat plate at $\mathrm{Re} = 1 \times 10^7$, as predicted by CFD with various turbulence models.

ε/D	C_D	δ/L
k-ε	0.00291589	0.0140152
k-ω	0.00281271	0.0122085
S-A	0.00276828	0.013329
RSM	0.00274706	0.0131455

TABLE 1
Nondimensional temperature and boundary layer thicknesses for a heated flat plate at $\mathrm{Re} = 1 \times 10^5$, as predicted by CFD (laminar flow).

Fluid	δ_T/L	δ_u/L
air	0.017313	0.015412
water	0.007664	0.015412

15-61

Analysis The results are shown in Table 1. We denote the thermal boundary layer thickness as δ_T/L, and compare to the velocity boundary layer thickness, which we denote as δ_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 ($\mathrm{Pr} = \nu/\kappa$) 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 $\mathrm{Pr} < 1$.

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".

TABLE 1
Nondimensional temperature and boundary layer thicknesses for a heated flat plate at $\mathrm{Re} = 1 \times 10^7$, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

Fluid	δ_T/L	δ_u/L
air	0.014527	0.014015
water	0.013162	0.014014

15-62

Analysis The results are shown in Table 1. We denote the thermal boundary layer thickness as δ_T/L, and compare to the velocity boundary layer thickness, which we denote as δ_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.

15-63

Analysis 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**.

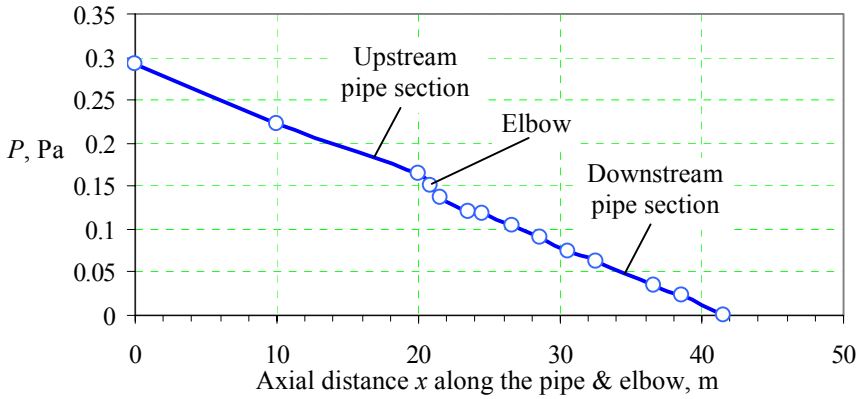


FIGURE 1
Pressure along the axis of a pipe with an elbow at Re = 2×10^4, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

Discussion Newer versions of FlowLab may give slightly different results.

15-64

Analysis 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**.

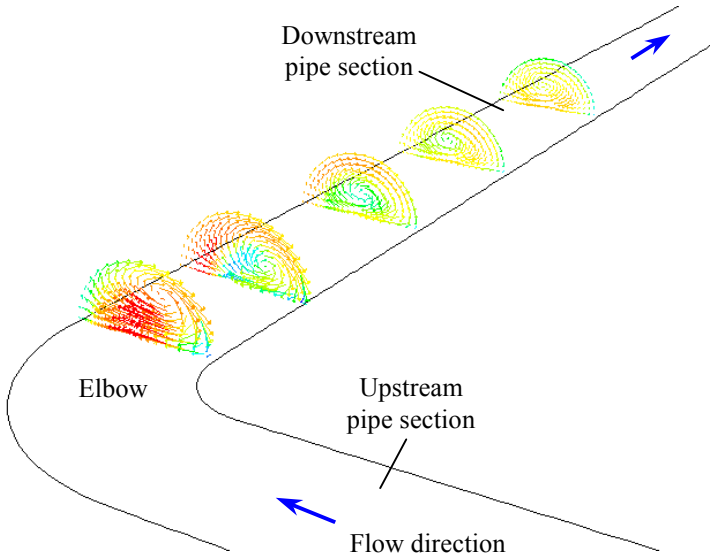


FIGURE 1
Velocity vector plots at several cross sections of a pipe with an elbow at Re = 2×10^4, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

Discussion Newer versions of FlowLab may give slightly different results.

15-65

Analysis 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 0.284 kPa. For the straight pipe, the pressure drop is calculated to be 0.224 kPa. Subtracting these and converting to minor loss coefficient, the value of K_L predicted by our CFD calculation for the standard k-ε turbulence model is **0.295** – a difference of less than two percent, 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 Newer versions of FlowLab may give slightly different results.

15-66

Analysis The results for four turbulence models are listed in Table 1. **The standard k-ε turbulence model does the best job** in predicting K_L. **The standard k-ω turbulence model does the worst job**. The Spallart-Allmaras turbulence model calculations are not the worst, 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 k-ε model, but better than the other two. 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.

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.

Turbulence model	K_L	Error (%)
Spallart-Allmaras (1 eq.)	0.204	-32%
k-ε (2 eq.)	0.295	-1.7%
k-ω (2 eq.)	0.401	34%
Reynolds stress model (7 eq.)	0.338	13%

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.

TABLE 1
Lift and drag coefficients as functions of angle of attack for an airfoil at Re = 1×10^7, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

α (degrees)	C_L	C_D
-2	0.138008	0.0153666
0	0.348498	0.0148594
2	0.560806	0.0149519
4	0.769169	0.0170382
5	0.867956	0.0192945
6	0.967494	0.0210042
8	1.14544	0.0275433
10	1.29188	0.0375832
12	1.39539	0.0522318
14	1.44135	0.0725146
16	1.41767	0.100056
18	1.34726	0.140424
20	1.29543	0.274792

15-67

Analysis The lift coefficient rises to 1.44 at α = 14°, beyond which the lift coefficient drops off. So, **the stall angle is about 14°**. Meanwhile, the drag coefficient increases slowly up to the stall location, and then rises significantly after stall. The data are also plotted in Fig. 1.

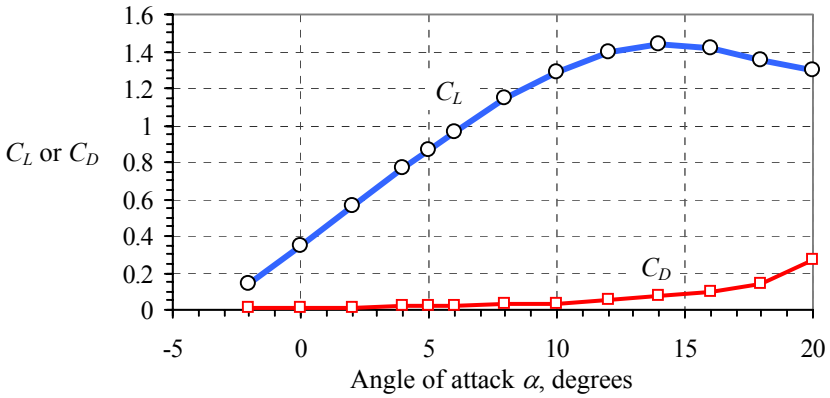


FIGURE 1
Lift and drag coefficients as functions of angle of attack for an airfoil at Re = 1×10^7, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

Discussion Newer versions of FlowLab may give slightly different results. We note that this airfoil is not symmetric, as can be verified by the fact that the lift coefficient is nonzero at zero angle of attack. The lift coefficient does not drop as dramatically as is observed empirically. Why? The flow becomes *unsteady* for angles of attack beyond the stall angle. However, we are performing *steady* calculations. For higher angles, the run does not even converge; the CFD calculation is stopped because it has exceeded the maximum number of allowable iterations, not because it has converged. Thus the main reason for not capturing the sudden drop in C_L after stall is because we are not accounting for the transient nature of the flow. The airfoil used in these calculations is called a *ClarkY airfoil*.

TABLE 1
Lift and drag coefficients as functions of angle of attack for an airfoil at Re = 3×10^6, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

α (degrees)	C_L	C_D
0	0.221797	0.0118975
4	0.65061	0.0166523
6	0.858744	0.0212052
8	1.05953	0.0273125
10	1.2501	0.0351061
12	1.42542	0.0447038
14	1.57862	0.0562746
16	1.69816	0.0702321
18	1.76686	0.0875881
20	1.75446	0.111326
22	1.70497	0.178404

FIGURE 1
Lift and drag coefficients as functions of angle of attack for an airfoil at Re = 3×10^6, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

15-68

Analysis For a different airfoil, and for the case in which Re = 3×10^6, the lift coefficient rises to about 1.77 at about 18°, beyond which the lift coefficient drops off (Table 1). So, **the stall angle is about 18°**. Meanwhile, the drag coefficient increases slowly up to the stall location, and then rises significantly after stall. The data are also plotted in Fig. 1.

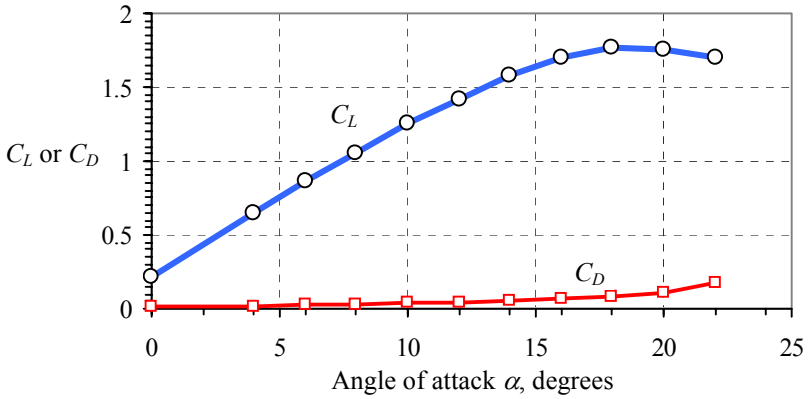
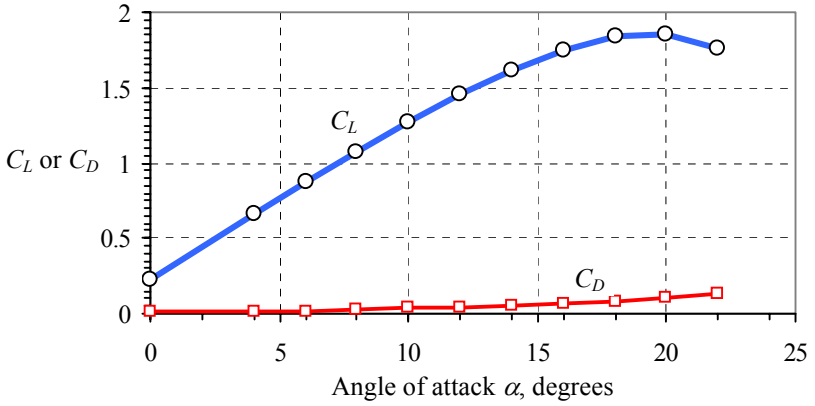


TABLE 2
Lift and drag coefficients as functions of angle of attack for an airfoil at Re = 6×10^6, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).

α (degrees)	C_L	C_D
0	0.226013	0.0106894
4	0.659469	0.015384
6	0.870578	0.0198087
8	1.07512	0.0256978
10	1.27094	0.0331355
12	1.4533	0.0422083
14	1.61589	0.0530114
16	1.74939	0.0657999
18	1.83901	0.0812925
20	1.85799	0.101563
22	1.76048	0.138806

For the case in which Re = 6×10^6, lift coefficient rises to about 1.86 at about 20°, beyond which the lift coefficient drops off (Table 2). So, **the stall angle is about 20°**. Meanwhile, the drag coefficient increases slowly up to the stall location, and then rises significantly after stall. The data are also plotted in Fig. 2. The maximum lift coefficient and the stall angle have both increased somewhat compared to those at Re = 3×10^6 (half the Reynolds number). Apparently, the higher Reynolds number leads to a more vigorous turbulent boundary layer that is able to resist flow separation to a greater downstream distance than for the lower Reynolds number case. For all angles of attack, the drag coefficient is slightly smaller for the higher Reynolds number case, reflecting the fact that the skin friction coefficient decreases with increasing Re along a wall, all else being equal. [Airfoil drag (before stall) is due mostly to skin friction rather than pressure drag.]

FIGURE 2
Lift and drag coefficients as functions of angle of attack for an airfoil at Re = 6×10^6, as predicted by CFD (turbulent flow, using the standard k-ε turbulence model).



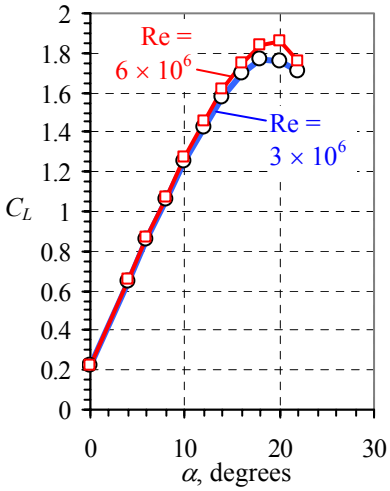


FIGURE 3
Lift coefficient plotted as a function of angle of attack of an airfoil at two different values of Reynolds number.

A comparison of the lift coefficient as a function of angle of attack for the two Reynolds numbers is provided in Fig. 3. The airfoil clearly performs better at the higher Reynolds number.

Discussion Newer versions of FlowLab may give slightly different results. The behavior of the lift and drag coefficients beyond stall is not as dramatic as we might have expected. Why? The flow becomes *unsteady* for angles of attack beyond the stall angle. However, we are performing *steady* calculations. For higher angles, the run does not even converge; the CFD calculation is stopped because it has exceeded the maximum number of allowable iterations, not because it has converged. Thus the main reason for not capturing the sudden drop in C_L after stall is because we are not accounting for the transient nature of the flow. We note that the airfoil used in this problem (a *NACA2415 airfoil*) is different than the one used in Problem 15-67 (a *ClarkY airfoil*). Comparing the two, the present one has better performance (higher maximum lift coefficient and higher stall angle, even though the Reynolds numbers here are lower than that of Problem 15-67. At higher Re, this airfoil may perform even better.

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 $Re = 1 \times 10^7$.

Cells	C_L	C_D
672	1.09916	0.242452
1344	1.00129	0.224211
2176	1.06013	0.196212
6264	1.02186	0.189023
12500	1.42487	0.0869799
16800	1.4356	0.0857791
21700	1.43862	0.0848678
24320	1.43719	0.0852304
27200	1.43741	0.0854769

15-69

Analysis 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 1×10^7. 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 (1×10^7). 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
Drag coefficient as a function of the normalized extent of the computational domain for flow over a microorganism.

R/L	C_D
3	334.067
10	289.152
50	278.291
100	277.754
500	277.647
1000	277.776
2000	278.226

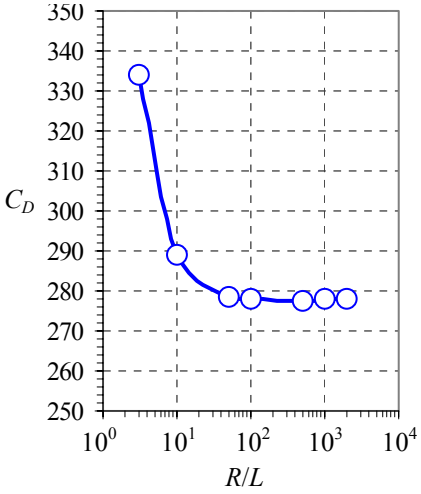


FIGURE 1
Drag coefficient plotted as a function of the normalized extent of the computational domain for creeping flow over a microorganism.

15-70

Analysis The drag coefficient is listed as a function of *R*/*L* in Table 1. It is also plotted in Fig. 1. 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.

Discussion Newer versions of FlowLab may give slightly different results.

TABLE 1
Drag coefficient as a function of Reynolds number for flow over a 2×1 ellipsoid.

Re	C_D
0.1	552.89
0.5	114.634
5	14.3342
20	4.89852
50	2.61481
100	1.691

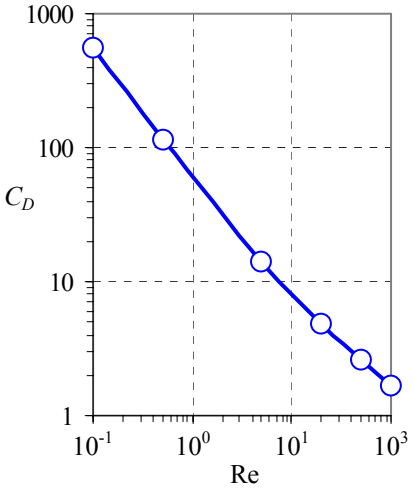


FIGURE 1
Drag coefficient plotted as a function of Reynolds number for creeping flow over a 2×1 ellipsoid.

15-71

Analysis The velocity profile for creeping flow (Re < 1) shows a very slowly varying velocity from zero at the wall to V eventually. At high Re, we expect a thin boundary layer and flow that accelerates around the body. However, in creeping flow, there is negligible inertia, and the flow does not accelerate around the body. Instead, the body has significant impact on the flow to distances very far from the body. As Re increases, the drag coefficient drops sharply, as expected based on experimental data (see Chap. 11). At the higher values of Re (here, for Re = 50 and 100), inertial effects are becoming more significant than viscous effects, and the velocity flow disturbance caused by the body is confined more locally around the body compared to the lower Reynolds number cases. If Re were to be increased even more, very thin boundary layers would develop along the walls.

The data are also plotted in Fig. 1. The drop in drag coefficient with increasing Reynolds number is quite dramatic as Re ranges from 0.1 to 100. (C_D decreases from more than 500 to nearly 1 in that range). Thus, we use a log-log scale in Fig. 1.

Discussion Newer versions of FlowLab may give slightly different results.

General CFD Problems*

15-72

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.

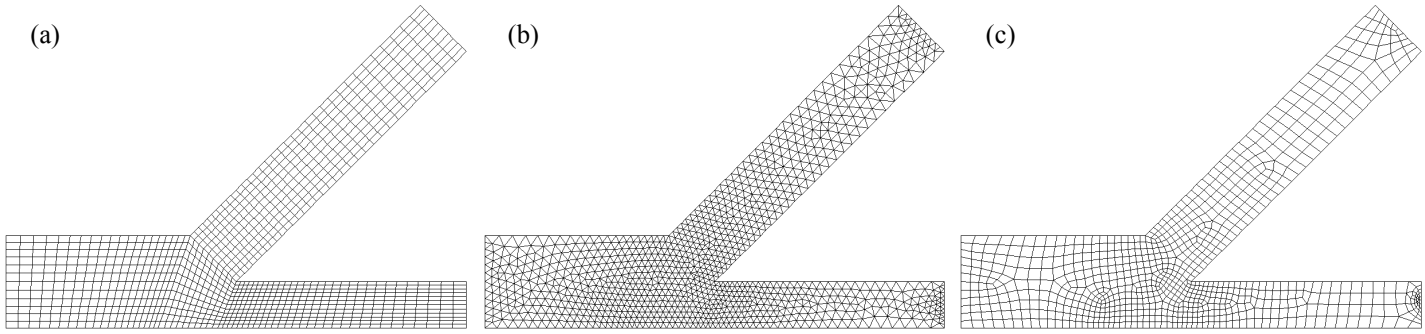


FIGURE 1
Comparison of three meshes: (a) structured multiblock, (b) unstructured triangular, and (c) unstructured quadrilateral.

Discussion Depending on the grid generation software and the specified node distribution, students will get a variety of results.

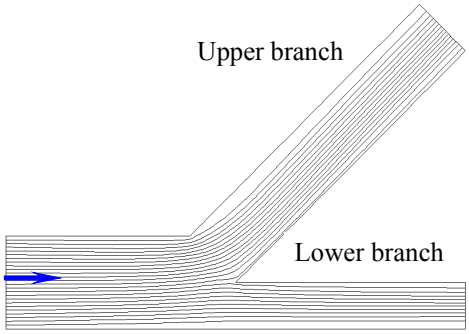


FIGURE 1
Streamlines for laminar flow through a wye.

15-73

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 -8.74×10^{-5} Pa. The pressure drop through the wye is thus only 8.74×10^{-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.

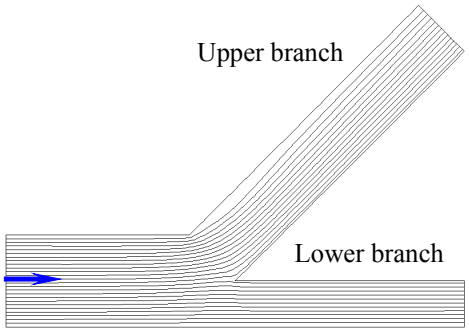


FIGURE 1
Streamlines for turbulent flow through a wye. The k-ε turbulence model is used.

15-74

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.

15-75

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.

15-76

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.

15-77

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.

15-78

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.

15-79

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.

15-80

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.

15-81

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.

15-82

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.

15-83

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.

15-84

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.

15-85

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.

15-86

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.

15-87

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.

15-88

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

15-89C

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.

15-90C

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.

15-91C

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.

15-92C

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.

15-93C

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 Problem 15-92C, 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.

15-94C

Solution 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.

Analysis 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.

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

15-95

Solution We are to calculate flow through a single-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.

15-96

Solution We are to study the effect of heating element angle of attack on heat transfer through a single-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.

15-97

Solution We are to calculate flow through a single-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.

15-98

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.

15-99

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.

15-100

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.

15-101

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.

15-102

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 flowfield 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.).

15-103

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^2/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, 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.

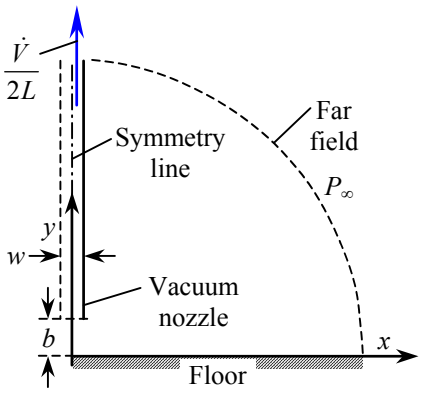
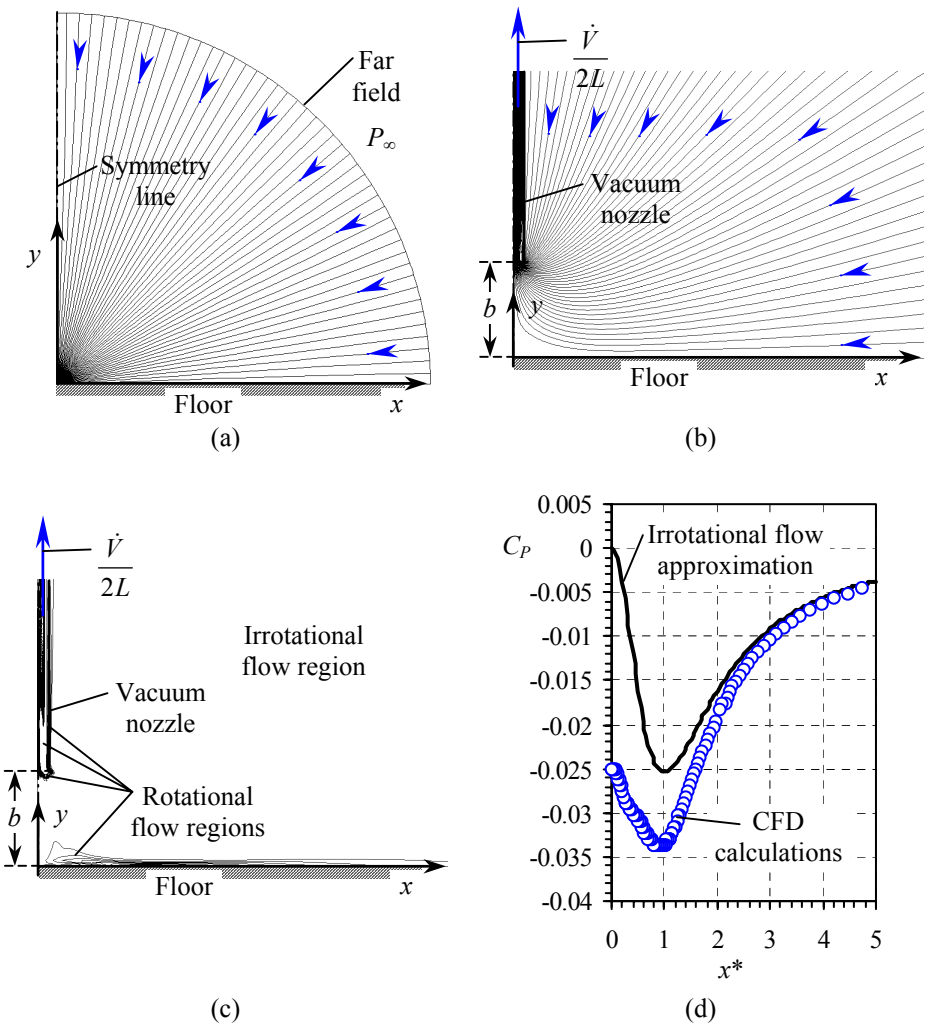


FIGURE 1
CFD model of air being sucked into a vertical vacuum nozzle; the y-axis is a line of symmetry (not to scale –the far field is actually much further away from the nozzle than is sketched here).

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.



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.

15-104

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.
