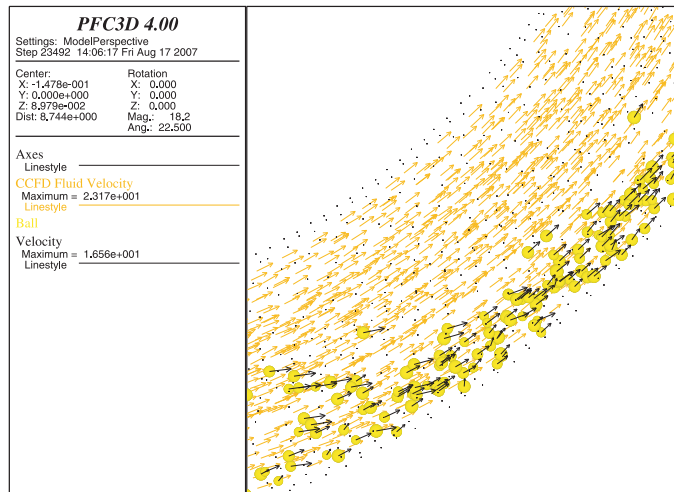


PFC3D

Particle Flow Code in 3 Dimensions

CCFD Add-on



©2008

Itasca Consulting Group Inc.
Mill Place
111 Third Avenue South, Suite 450
Minneapolis, Minnesota 55401 USA

Phone: (1) 612-371-4711
Fax: (1) 612-371-4717
E-Mail: software@itascacg.com
Web: www.itascacg.com

First Edition (Version 1.0) February 2008

TABLE OF CONTENTS

1 Introduction

2 *PFC*^{3D}-CCFD Theory and Implementation

2.1	Theory and Background	2-2
2.1.1	Equations of Motion for the Fluid	2-2
2.1.2	DEM Mechanical Calculation	2-3
2.1.3	Coupling	2-5
2.2	Calculation Cycle	2-6
2.3	Porosity Calculation	2-8
2.4	Practical Considerations	2-9
2.4.1	Timing	2-9
2.4.2	Meshing	2-10
2.5	Commands	2-11
2.6	Plot Items	2-14
2.7	<i>FISH</i> Intrinsic	2-15
2.8	References	2-18

3 Verification Examples

3.1	Single Ball Falling	3-1
3.1.1	Analytical Solution	3-1
3.1.2	<i>PFC</i> ^{3D} -CCFD Modeling	3-3
3.1.2.1	Low Reynolds Number Flow	3-3
3.1.2.2	Data File “pfc_dropTest1.dat”	3-3
3.1.2.3	Higher Reynolds Number Flow	3-5
3.1.2.4	Data File “pfc_dropTest2.dat”	3-5
3.2	Porous Media Flow	3-7
3.2.1	Approximate Solution	3-7
3.2.2	<i>PFC</i> ^{3D} -CCFD Modeling	3-8
3.2.3	<i>PFC</i> ^{3D} Data Files	3-9
3.2.3.1	Data File “pfc_pack.dat”	3-9
3.2.3.2	Data File “pfc_porous1.dat”	3-11
3.2.3.3	Data File “pfc_test.dat”	3-12
3.3	References	3-12

4 Tutorials

4.1	Tutorial 1: Single Steel Ball Dropping in a Vertical Cylinder Filled with Water	4- 1
4.1.1	Part 1: Creating a Cylindrical Volume	4- 1
4.1.1.1	Creating Points	4- 1
4.1.1.2	Creating Lines	4- 2
4.1.1.3	Creating the Base and Volume of the Cylinder	4- 5
4.1.2	Part 2: Fluid Flow Model Setup	4- 10
4.1.2.1	Material Definition	4- 10
4.1.2.2	Problem Data	4- 11
4.1.2.3	Interval Data	4- 13
4.1.2.4	Initial Value Data	4- 15
4.1.2.5	Boundary Conditions	4- 15
4.1.2.6	Mesh Generation	4- 16
4.1.3	Part 3: <i>PFC^{3D}</i> Data File Preparation	4- 19
4.1.3.1	Data File “pfc_Cylinder.dat”	4- 19
4.1.4	Part 4: Launching the Calculation	4- 20
4.1.5	Part 5: Visualizing the Results	4- 22
4.2	Tutorial 2: Particles Convected with the Flow along a Curved Pipe	4- 24
4.2.1	Part 1: Creating an Elbow	4- 24
4.2.1.1	Creating Points	4- 24
4.2.1.2	Creating Lines	4- 25
4.2.1.3	Creating the Inlet Face and the Horizontal Section of the Elbow	4- 31
4.2.1.4	Creating the Curved Section of the Elbow	4- 34
4.2.1.5	Creating the Vertical Section of the Elbow	4- 36
4.2.2	Part 2: Fluid Flow Model Setup	4- 38
4.2.2.1	Material Definition	4- 38
4.2.2.2	Problem Data	4- 39
4.2.2.3	Interval Data	4- 41
4.2.2.4	Initial Value Data	4- 43
4.2.2.5	Boundary Conditions	4- 44
4.2.2.6	Mesh Generation	4- 47
4.2.3	Part 3: Preparation of the Wall Geometry for <i>PFC^{3D}</i>	4- 51
4.2.3.1	Model Cleanup	4- 51
4.2.3.2	Creating an STL File Describing the <i>PFC^{3D}</i> Wall Geometry	4- 52
4.2.4	Part 4: <i>PFC^{3D}</i> Data File Preparation	4- 54
4.2.4.1	Data File “pfc_Elbow.dat”	4- 54
4.2.5	Part 5: Launching the Simulation	4- 57
4.2.6	Part 6: Visualizing the Results	4- 59

TABLES

Table 2.1 *FISH* intrinsic 2 - 15

FIGURES

Figure 1.1	Organization of the CCFD option of <i>PFC^{3D}</i>	1 - 1
Figure 2.1	Flow chart of the calculation cycle	2 - 7
Figure 3.1	Drag coefficient is plotted against Reynolds number for the Stokes law and the empirical drag law used in the CCFD add-on	3 - 2
Figure 3.2	<i>PFC^{3D}</i> output from “dropTest1.gid/” after 0.5 s of calculation. The fluid velocity vectors are shown, along with a plot of the <i>PFC^{3D}</i> particle velocity as a function of time.	3 - 4
Figure 3.3	<i>PFC^{3D}</i> output from “dropTest2.gid/” after 0.5 s of calculation. The fluid velocity vectors are shown, along with a plot of the <i>PFC^{3D}</i> particle velocity as a function of time.	3 - 6
Figure 3.4	<i>PFC^{3D}</i> output for “porous1.gid/” after 0.15 s of calculation. The fluid velocity vectors are shown, along with a plot of the magnitude of the fluid velocity as a function of time.	3 - 8
Figure 4.1	xy-view of the 6 points that have been created	4 - 2
Figure 4.2	Floating arc section created after clicking on two points	4 - 3
Figure 4.3	Two completed arcs	4 - 3
Figure 4.4	Turning the line labels on	4 - 4
Figure 4.5	Four arcs defining the base of the cylinder	4 - 5
Figure 4.6	View of the complete model	4 - 6
Figure 4.7	Surface (in pink) representing the base of the cylinder	4 - 7
Figure 4.8	<i>Copy</i> dialog box showing the parameters used for base extrusion	4 - 8
Figure 4.9	Volume and surfaces resulting from the extrusion of the base	4 - 9
Figure 4.10	A representation of the volumes and their material assignment in the model	4 - 10
Figure 4.11	The General tab of the Problem Data dialog box	4 - 11
Figure 4.12	The Restart, Monitoring point, Scalar and VOF tabs	4 - 12
Figure 4.13	The <i>Problem Data</i> dialog box showing the coupling type and the timestep specification	4 - 13
Figure 4.14	The Control, Gravity and Criterion temp tabs of the <i>Interval Data</i> dialog box	4 - 14
Figure 4.15	Applying initial conditions	4 - 15
Figure 4.16	Summary of all initial and boundary conditions	4 - 16
Figure 4.17	Model after assignment of number of meshing subdivisions in the East-West and South-North directions. Note that the only remaining unassigned (blue) lines are in the bottom-top direction.	4 - 17
Figure 4.18	Structured hexahedral mesh of the cylinder	4 - 18
Figure 4.19	Reading the <i>PFC^{3D}</i> data file	4 - 20
Figure 4.20	Proxy program about to start	4 - 21
Figure 4.21	<i>PFC^{3D}</i> screen launched by the proxy program	4 - 21
Figure 4.22	CCFD screen running in conjunction with <i>PFC^{3D}</i>	4 - 22
Figure 4.23	Contours of fluid pressure using GiD in post-process mode	4 - 23
Figure 4.24	Isometric view of points 1, 2 and 3	4 - 25

Figure 4.25	Straight line passing through points 2 and 3	4 - 26
Figure 4.26	The Copy dialog box used for copying and rotating the vertical line	4 - 27
Figure 4.27	Vertical line and its rotated copy	4 - 28
Figure 4.28	Floating arc section created after clicking on two points	4 - 29
Figure 4.29	Two completed arcs	4 - 29
Figure 4.30	Turning the line labels on	4 - 30
Figure 4.31	Four arcs defining the inlet of the elbow	4 - 31
Figure 4.32	In pink, surface representing the inlet of the elbow	4 - 32
Figure 4.33	Copy dialog box showing the parameters used for copying and extruding the inlet surface	4 - 33
Figure 4.34	Volume and surfaces resulting from the extrusion of the inlet	4 - 34
Figure 4.35	Perspective view of the horizontal section	4 - 34
Figure 4.36	Creating the volume of the curved section	4 - 35
Figure 4.37	Outlet of the curved section of the elbow	4 - 36
Figure 4.38	Creation of the vertical volume	4 - 37
Figure 4.39	A representation of each volume and its material assignment in the model ..	4 - 39
Figure 4.40	The General tab of the Problem Data dialog box	4 - 39
Figure 4.41	The Restart, Monitoring point, Scalar and VOF tabs	4 - 40
Figure 4.42	The Problem Data dialog box showing the coupling type and the timestep specification	4 - 41
Figure 4.43	The Control, Gravity and Criterion temp tabs of the Interval Data dialog box	4 - 42
Figure 4.44	Applying initial conditions	4 - 43
Figure 4.45	Applying the zero velocity wall boundary conditions	4 - 45
Figure 4.46	Summary of all initial and boundary conditions	4 - 46
Figure 4.47	Fixed-pressure boundary condition specified at the outlet	4 - 46
Figure 4.48	Fixed-velocity boundary conditions: 0 in yellow; inlet velocity in green	4 - 47
Figure 4.49	Model after assignment of number of meshing subdivisions in the East-West and South-North directions. Note that the only remaining unassigned (blue) lines are in the bottom-top direction of volume 1.	4 - 48
Figure 4.50	Assigning the number of subdivisions along the curved volume	4 - 49
Figure 4.51	Assigning the number of subdivisions along the vertical volume	4 - 49
Figure 4.52	Structured hexahedral mesh of the elbow with 5 subdivisions across in each direction and 80 subdivisions along the pipe	4 - 50
Figure 4.53	Surfaces 1, 6, 11 and 16 are clearly visible in the model. They need to be removed prior to the creation of the STL file.	4 - 51
Figure 4.54	The Undo dialog box	4 - 52
Figure 4.55	Parameters of size assignment by chordal error	4 - 53
Figure 4.56	Surface mesh of the elbow that will be used for the definition of walls in PFC^{3D}	4 - 53
Figure 4.57	Reading the PFC^{3D} data file	4 - 57
Figure 4.58	Proxy program about to start	4 - 58

Figure 4.59	<i>PFC^{3D}</i> screen launched by the proxy program	4 - 58
Figure 4.60	CCFD screen running in conjunction with <i>PFC^{3D}</i>	4 - 59
Figure 4.61	Contours of fluid pressure using GiD in post-process mode	4 - 60
Figure 4.62	Top-down view of pipe geometry	4 - 61
Figure 4.63	GiD <i>Select & Display Style</i> dialog	4 - 62
Figure 4.64	Contours of drag force magnitude shown on cut-plane surface	4 - 63
Figure 4.65	GiD <i>Animate</i> dialog	4 - 64

1 Introduction

The Coupled CFD (CCFD) add-on for *PFC^{3D}* 4.0 couples the mechanical DEM calculation of *PFC^{3D}* with a computational fluid dynamics code called CCFD.*

A solid modeler/post-processor called GiD† is included with the computational engines. This package allows for interactive creation of arbitrary problem domains, and includes a volume and surface mesher. This package also allows the user to interact with a 3D view of the results.

To conduct a coupled problem, GiD is first used to build a GiD project file. This file describes the problem-domain geometry. The boundary and initial conditions of the fluid are also specified in GiD, along with timing information. The user develops a *PFC^{3D}* data file to be invoked when the coupled calculation begins. The coupled calculation is started from within GiD via the *Calculate* dialog box. GiD starts CCFD, *PFC^{3D}* and a proxy server that brokers communication between CCFD and *PFC^{3D}*. During the coupled calculation, the proxy server controls *PFC^{3D}*, and control is returned to *PFC^{3D}* when the coupled calculation has ended. GiD is then used in post-process mode to view the results. Figure 1.1 shows the organization of these components.

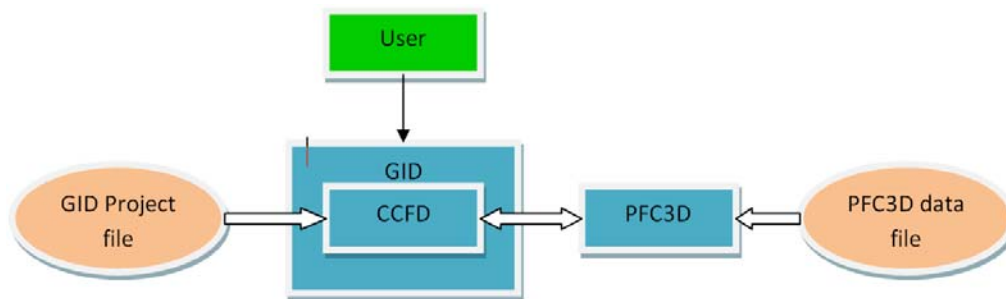


Figure 1.1 Organization of the CCFD option of *PFC^{3D}*

Several documents describe each part of the process in more detail:

This document describes the coupled operation of *PFC^{3D}*-CCFD. Section 2 of this document gives brief theoretical background on CCFD and *PFC^{3D}*, and fully describes the coupling methodology. Section 2 also contains descriptions of the *PFC^{3D}* commands and *FISH* intrinsics that make up this add-on. Section 3 contains two simple verification problems. Section 4 contains two tutorials that give step-by-step instructions for creating a model in GiD, and for running a coupled calculation. An additional source of information about *PFC^{3D}* is the *PFC^{3D}* Manual, which is not included on this CD but can be loaded from the accompanying Itasca CD.

* The Coupled Computational Fluid Dynamics (CCFD) code is a product of ITOCHU Techno-Solution Corporation (CTC) of Tokyo, Japan.

† GiD is a product of the International Center for Numerical Methods in Engineering (Barcelona, Spain).

Three additional documents are provided on this CD to describe CCFD. “CCFD – Theoretical Manual” describes the flow solver. “GiD-CCFD – Operating Manual” describes how to use GiD with CCFD for flow-only calculations (no coupling). (There is some overlap between the “GiD-CCFD – Operating Manual” and [Section 4](#) of this document.) “GiD-CCFD – Exercise Book” contains example problems for flow-only calculations using GiD and CCFD.

GiD has extensive documentation that can be accessed from within GiD via the help menu, or on the Web at <http://gid.cimne.upc.es/>.

2 *PFC^{3D}*-CCFD Theory and Implementation

The coupled CFD (CCFD) option for *PFC^{3D}* 4.0 couples the mechanical DEM calculation of *PFC^{3D}* with a computational fluid dynamics model.

The coupling is volume-averaged and two-way. In general, the CFD code solves the incompressible Navier-Stokes equation on a 3D discretized domain consisting of irregular tetrahedra or mapped hexahedra. The fluid equations (the Navier-Stokes equation and the continuity equation) are formulated with porosity terms, to account for the presence of particles. The force acting on the particles due to the fluid is assigned locally to each particle, based on the fluid conditions in the fluid element that the particle occupies.

A corresponding body force is applied to the fluid as an average over one fluid element. The fluid boundary conditions can be specified as constant pressure, constant velocity, slip, no slip or a logarithmic no-slip condition for turbulent flows.

Other features include:

- user-definable fluid particle interaction force
- user-defined functions for material and time-dependent boundary conditions
- volume of fluid (VOF) method to model free surface flows
- two turbulence models for high Reynolds number flows
- *FISH* access to the internal parameters
- explicit and implicit flow solvers
- scalar fields and thermal logic in the fluid phase

2.1 Theory and Background

A brief summary of the governing equations for the particles and the fluid is given here. For more information on the CCFD formulation, see “CCFD: Theoretical Manual”; for more information on *PFC^{3D}*, see the **Theory and Background** volume of the *PFC^{3D}* manual.

2.1.1 Equations of Motion for the Fluid

CCFD solves incompressible flow equations. However, a compressible formulation is developed here because non-Newtonian fluid rheologies can be used with CCFD.

The continuity equation describes the conservation of mass in a continuum:

$$\frac{d}{dt} \int_V \rho_f \epsilon dV + \int_S \rho_f \epsilon \vec{v} \cdot ds = 0 \quad (2.1)$$

where V is a control volume, S is a bounding surface, s is an outward normal of that surface, ρ_f is the fluid density, \vec{v} is the fluid velocity and ϵ is porosity.

The conservation of momentum in a compressible viscous fluid can be described as

$$\frac{d}{dt} \int_V \rho_f \epsilon \vec{v} dV + \int_S \rho_f \epsilon \vec{v} \vec{v} \cdot ds = \int_S \epsilon T \cdot ds + \int_V \vec{f}_b dV \quad (2.2)$$

where T is the Cauchy stress tensor and \vec{f}_b is a body force per unit volume. Using Stokes' law, the Cauchy tensor becomes

$$T = 2\mu \dot{D} - \frac{2}{3}\mu (\nabla \cdot (\epsilon \vec{v})) I - pI \quad (2.3)$$

where μ is the dynamic viscosity, p is the fluid pressure, I is the unit tensor and \dot{D} is defined as

$$\dot{D} = \frac{1}{2} \left[\nabla \epsilon \vec{v} + (\nabla \epsilon \vec{v})^T \right] \quad (2.4)$$

For an incompressible fluid, ρ_f is constant. So the continuity equation (Eq. (2.1)) becomes

$$\frac{\partial \epsilon}{\partial t} + \nabla \cdot (\epsilon \vec{v}) = 0 \quad (2.5)$$

Substituting this relation into Eq. (2.3), the Cauchy tensor becomes

$$T = \mu \nabla \vec{v} - \frac{2}{3} \mu \left(\frac{\partial \epsilon}{\partial t} \right) I - p I \quad (2.6)$$

The second term on the right-hand side of this equation describes the viscous losses associated with the fluid movement due to porosity change. This term is ignored in the CCFD solution. Incorporating this form of the Cauchy tensor, and showing the results in the differential form, gives the Navier-Stokes equation,

$$\rho_f \frac{\partial \epsilon \vec{v}}{\partial t} + \rho_f \vec{v} \cdot \nabla (\epsilon \vec{v}) = -\epsilon \nabla p + \mu \nabla^2 (\epsilon \vec{v}) + \vec{f}_b \quad (2.7)$$

2.1.2 DEM Mechanical Calculation

The equations of motion for the PFC^{3D} particles are given by the standard PFC^{3D} equations described in the PFC^{3D} 4.0 **Theory and Background** volume, with an additional forcing term to account for interaction with the fluid:

$$\frac{\partial \vec{u}}{\partial t} = \frac{\vec{f}_{mech} + \vec{f}_{fluid}}{m} + \vec{g} \quad (2.8)$$

$$\frac{\partial \vec{\omega}}{\partial t} = \frac{\vec{M}}{I} \quad (2.9)$$

where \vec{u} is the particle velocity, m is the particle mass, \vec{f}_{fluid} is the total force applied by the fluid on the particle, \vec{f}_{mech} is the sum of additional forces (externally applied forces and contact forces) acting on the particle, \vec{g} is the acceleration due to gravity, $\vec{\omega}$ is the particle angular velocity, I is the moment of inertia and \vec{M} is the moment acting on the particle.

The force applied by the fluid, \vec{f}_{fluid} , is made up of two parts: the drag force, and a force due to the fluid pressure gradient.

The drag force the fluid exerts on the particles is defined individually for each particle, based on the conditions in the fluid element that contains the particle. Depending on how the porosity is calculated, a particle may intersect more than one fluid element. In this case, the forces are distributed based on the fractional overlap between the particle and the fluid element. Note that the fluid-applied force is always applied at the particle centroid, and no rotational moment is applied to the particle.

The drag force \vec{f}_{drag} is defined as

$$f_{drag}^{\vec{}} = \vec{f}_0 \epsilon^{-\chi} \quad (2.10)$$

where \vec{f}_0 is the force on a single particle and ϵ is the porosity. The $\epsilon^{-\chi}$ term is an empirical factor to account for the local porosity. This correction term makes the force applicable to both fixed and fluidized beds for a large range of Reynolds numbers (Di Felice 1994, and Xu and Yu 1997).

The single particle force is defined as

$$\vec{f}_0 = \left(\frac{1}{2} C_d \rho_f \pi r^2 |\vec{u} - \vec{v}| (\vec{u} - \vec{v}) \right) \quad (2.11)$$

where C_d is a drag coefficient, ρ_f is the fluid density, r is the particle radius, \vec{v} is the fluid velocity and \vec{u} is the particle velocity. The drag coefficient is defined as

$$C_d = \left(0.63 + \frac{4.8}{\sqrt{Re_p}} \right)^2 \quad (2.12)$$

where Re_p is the particle Reynolds number. The empirical coefficient χ is defined as

$$\chi = 3.7 - 0.65 \exp \left(- \frac{(1.5 - \log_{10}(Re_p))^2}{2} \right) \quad (2.13)$$

The particle Reynolds number is

$$Re_p = \frac{2 \rho_f r |\vec{u} - \vec{v}|}{\mu_f} \quad (2.14)$$

where μ_f is the dynamic viscosity of the fluid.

Including the force due to the fluid pressure gradient, the total fluid-applied force on a particle is

$$\vec{f}_{fluid} = \vec{f}_0 \epsilon^{-\chi} + \frac{4}{3} \pi r^3 (\nabla p - \rho_f \vec{g}) \quad (2.15)$$

Note that the buoyancy is added explicitly because the pressure field of the fluid does not contain the hydrostatic gradient. The addition of buoyancy can be suppressed by issuing the command **CCFD set buoyancy off**. This is necessary in VOF mode, where the hydrostatic gradient is contained in the pressure gradient.

2.1.3 Coupling

The coupling is implemented as a series of data exchanges between CCFD and PFC^{3D}, conducted at predetermined times.

The porosity, ϵ^i , in each fluid element is determined by PFC^{3D}. The body force per unit volume, \vec{f}_b^i , in each fluid element is determined by PFC^{3D} as

$$\vec{f}_b^i = \frac{\sum_j \vec{f}_{drag}^j}{V^i} \quad (2.16)$$

where the sum is over all the particles in a given fluid element, and V^i is the volume of that element. Note that in the current implementation, the total drag force is determined in PFC^{3D} and divided by volume in CCFD. The fluid velocity, \vec{v}^i , and fluid pressure gradient, ∇p^i , in each element are determined by CCFD.

CCFD applies linear relaxation to the body force and porosity received from PFC^{3D} in each coupling data exchange. For example, the linear relaxation for porosity takes the form

$$\epsilon_t = R_f \epsilon_t + (1 - R_f) \epsilon_{t-dt} \quad (2.17)$$

where R_f is the relaxation parameter (fixed at 0.5), ϵ_t is the current porosity and ϵ_{t-dt} is the porosity from the previous timestep.

The **CCFD set vel_relax** command applies the same form of linear relaxation to the fluid velocity received from CCFD in each coupling data exchange. The command allows any relaxation factor to be specified (see [Section 2.5](#)).

2.2 Calculation Cycle

Each time the coupling information is exchanged, PFC^{3D} sends CCFD the current porosity and the total drag force in each fluid element, and CCFD sends PFC^{3D} the current fluid velocity and fluid pressure gradient in each fluid element. The timing for these data exchanges is given in the Coupling tab in the *Problem Data* dialog of GiD. The boxes are labeled “start,” “end” and “step.” Start and end are the starting and ending times of the coupling in units of time, and step is the coupling interval, t_c , the time between exchanges of data. t_s denotes the total accumulated coupling time. When the coupled calculation is started, the codes exchange information about initial conditions, and then both codes are run forward t_c seconds, and the coupling information is exchanged again. This continues until the specified end of the calculation. [Figure 2.1](#) shows a flow chart of the calculation cycle. The force applied to the PFC^{3D} particles by the fluid, \vec{f}_{fluid}^j , and the porosity in each cell, ϵ^i , are recalculated at given intervals during PFC^{3D} cycling according to the **SET interval** command. Note that these quantities are also recalculated after coupling data is received from CCFD and before coupling data is sent to CCFD.

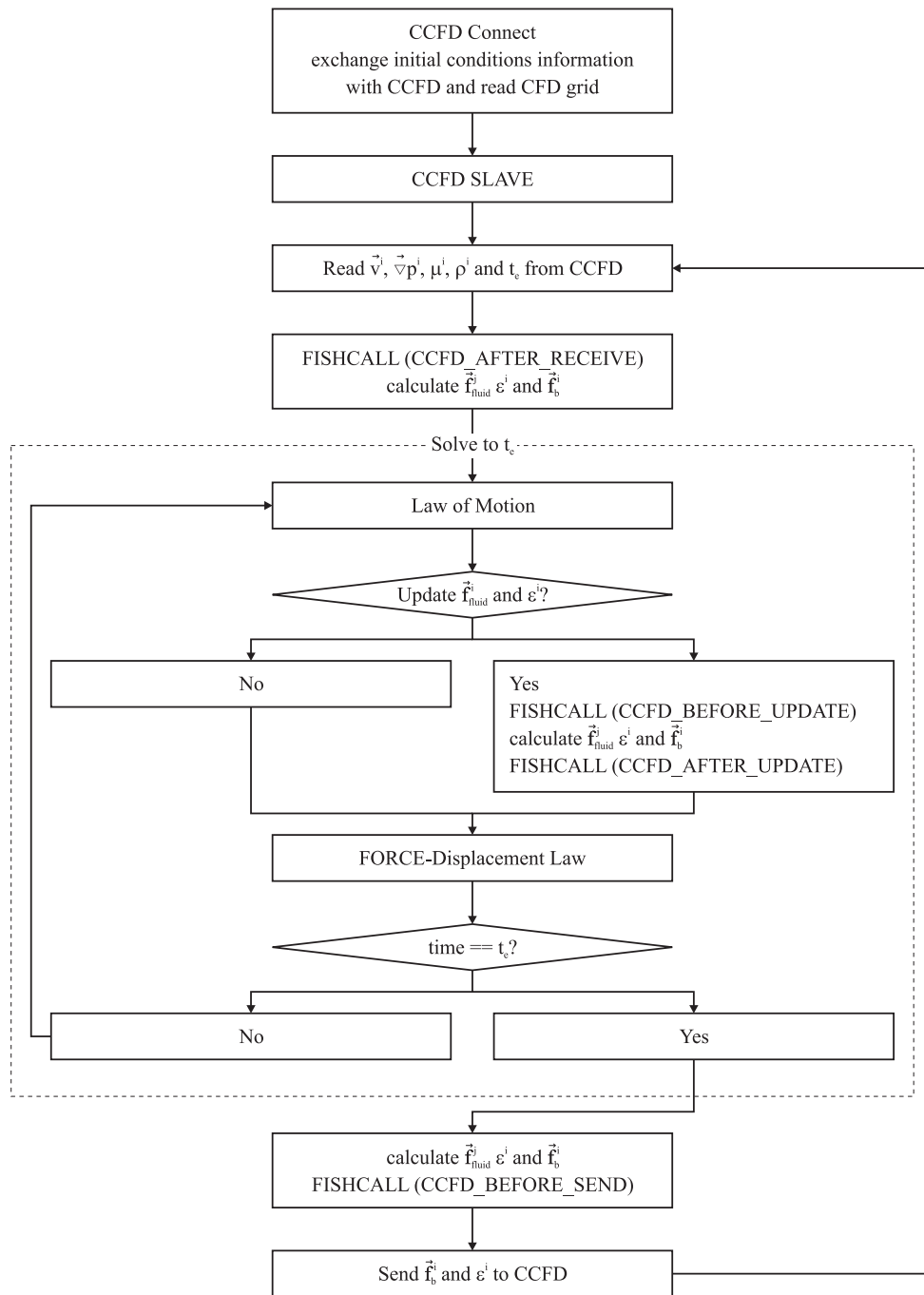


Figure 2.1 Flow chart of the calculation cycle

2.3 Porosity Calculation

There are several options available for calculating the porosity in a fluid element. The command **CCFD set porosity** changes this mode. The available options are: using the location of the centroid, a polyhedron representation of the particles, and an approximate method.

The centroid method considers a particle to be entirely within a fluid element if its centroid is contained within that element. This method is fast and is conservative of particle volume. However, this method can result in jumps in porosity as the particles move, which can reduce the smoothness of the solution. As the particle size decreases relative to the size of the fluid elements, this effect decreases.

The polyhedron method represents a particle as a cube with height, length and width equal to the diameter of the particle. The volume of intersection of this cube with the fluid element is calculated and adjusted to conserve the particle volume. This method is slower, but it has the advantage that the change in porosity is smooth as a particle moves from one fluid element to another.

An experimental approximate method is also available. This method is a heuristic approximation, which is not strictly conservative of particle volume but gives a fast, smooth result. The volume of intersection of a spherical particle with the half-spaces corresponding to each fluid element face is calculated. The ball volume is taken as the volume remaining after all of the element face cuts, plus the original volume times fractional values of each pair of cuts multiplied together. For example, if one element face cuts 20% of the particle, and another element face cuts 15% of the particle, the remaining volume is $65\% + (20\% \text{ of } 15\%) = 68\%$.

A porosity of zero occurring in a fluid element results in a singularity in the governing equations of the fluid. To avoid problems due to the singularity, *PFC^{3D}* limits the porosity to < 0.05 . Note that the full volume of the spherical particle is used in the porosity calculation. No volume adjustment is made for particles that overlap.

2.4 Practical Considerations

This section outlines practical limits on the input parameters.

The fluid grid should be sufficiently fine to resolve flow structures. The following inequality should be met:

$$\frac{d_c}{\Delta x_{cfd}} > 5 \quad (2.18)$$

where d_c is the minimum width of the flow domain, and Δx_{cfd} is the fluid element length.

This coupling methodology is designed to describe the average coupling forces occurring within one fluid element. Flow around the particles is not explicitly represented, as the local porosity is assumed to be evenly distributed within one element. For good results, several PFC^{3D} particles should fit inside one CFD element:

$$\frac{\Delta x_{cfd}}{2r} > 5 \quad (2.19)$$

Where the exact particle micro-properties are not critical to the problem, there is some scope to reduce the particle stiffness to increase the allowable PFC^{3D} timestep. However, particles colliding at maximum velocity with the PFC^{3D} walls should not overlap more than half the radius of the particle.

$$\frac{2}{3}\pi r^3 \rho_p |u_{max}|^2 < \frac{k_n r^2}{8} \quad (2.20)$$

where ρ_p is the particle density, and k_n is the normal contact stiffness of the particle.

2.4.1 Timing

The coupling interval should be small enough to resolve the desired coupling behavior. Coupling information should be exchanged several times as a particle moves across a fluid element. This condition is satisfied when the following inequality is met:

$$\frac{\Delta x_{cfd}}{|\vec{u}|t_c} > 3 \quad (2.21)$$

where t_c is the coupling interval specified within GiD.

When CCFD is used in implicit mode, a timestep has to be specified in the *Interval Data* dialog. In implicit mode, relatively large timesteps can be used, and small timesteps may result in poor convergence. In explicit mode, the timestep is determined automatically. The explicit timestep Δt_{cfd} :

$$\Delta t_{cfd} \approx \frac{1}{2} \frac{\Delta x_{cfd}}{|\vec{v}|} \quad (2.22)$$

In practice, the PFC^{3D} timestep is often smaller than the CFD timestep. Therefore, several cycles of PFC^{3D} are needed to meet one CFD step. It is useful to set the coupling interval, t_c , to the fluid timestep, so the coupling information is exchanged every fluid timestep.

2.4.2 Meshing

The fluid equations are solved in a 3D discretized domain consisting of tetrahedral or mapped hexahedral elements. The mapped hexahedral elements should be used if possible, but the tetrahedral elements have the advantage of being able to mesh any geometry.

If the particles are interacting with the fluid boundaries, then PFC^{3D} walls corresponding to the fluid boundaries need to be created within PFC^{3D} . If the geometry is simple, a combination of planar walls and general walls can be used to produce the particle boundaries. If the boundary geometry is complex, it is possible to mesh the surface of the fluid domain with triangles using GiD, and import them into PFC^{3D} as an STL file. A large number of wall segments can result in slow execution in PFC^{3D} . Often, a combination of the two techniques discussed above can be used (see [Section 4.2](#) for an example).

2.5 Commands

The following commands are available with the CCFD option:

CONFIG ccf	Activate CCFD coupling mode. This command must be called before other CCFD-specific commands are given.
CCFD connect	<port <i>p</i> > Establish the socket connection to the CCFD proxy server. A port number may be specified if the CCFD proxy server has been configured to operate on a port other than the default.
CCFD slave	Issuing this command gives control of PFC ^{3D} cycling to the CCFD proxy server. PFC ^{3D} will return from slave mode when the coupled calculation is complete, or if the <Esc> key is pressed. After pressing <Esc>, giving the command again will rejoin slave mode and continue the coupled calculation. The start, end and interval of the coupled calculation are specified in the GiD program.
CCFD update	Update the porosity, particle-applied force and drag force for all elements and particles in the model. This happens automatically during cycling (see the CCFD set interval command). This command is useful for testing.
CCFD set density	<i>d</i> Set the fluid density. Note that this must be specified here since PFC ^{3D} does not read this information from CCFD. When VOF mode is active, PFC ^{3D} reads the fluid density of each element directly from CCFD, and the value specified with this command is ignored.
CCFD set porosity	centroid, approx, poly sets the porosity calculation mode. centroid is the default.
CCFD set viscosity	<i>ν</i> Set the dynamic viscosity of the fluid. Note that this must be specified here since PFC ^{3D} does not read this information from CCFD. When VOF mode is active, PFC ^{3D} reads the fluid viscosity of each element directly from CCFD, and the value specified with this command is ignored.
CCFD set buoyancy	on off Activate or deactivate the buoyancy term. off is the default.

CCFD set interval	<i>i</i>	Specify the number of <i>PFC^{3D}</i> cycles between updates of porosity, particle-applied force and drag force. Note that these quantities are also calculated before and after the exchange of coupling information with CCFD. A value of zero can be given to specify no updates during <i>PFC^{3D}</i> cycling. The default is 1.
CCFD set coupling	on off	Activate or deactivate the coupling terms in <i>PFC^{3D}</i> . If off , <i>PFC^{3D}</i> cycles forward as usual, but no drag, buoyancy force or porosity is calculated. on is the default.
CCFD set cycling	on off	sets <i>PFC^{3D}</i> cycling on or off. If off , <i>PFC^{3D}</i> receives the coupling information from CCFD, computes a new drag force and porosity, and returns control to CCFD without cycling forward in time. Note that the coupling time, t_s , and <i>PFC^{3D}</i> global time will be still be incremented. on is the default.
CCFD set vof	on off	Enable or disable VOF mode for modeling free surface flows. off is the default.
CCFD set vel_relax	<i>n</i>	sets the relaxation parameter for linear relaxation of the CCFD fluid velocity received during each exchange. (See Section 2.1.3.) A value of 1.0 results in no relaxation – this is the default.
CCFD set use_self_porosity	on off	If off , during the calculation of \vec{f}_{fluid} (Eq. (2.15)), the contribution of the volume of the current ball to the porosity of the current element is removed. In other words, the porosity in an element is adjusted so the influence of the present particle is not felt. It is recommended that the <i>PFC^{3D}</i> particles be small compared to the CFD element size. If large particles must be used, this should be set to off . The default is on .

CCFD set eps_chi_term **on**
 off

If **on**, the $\epsilon^{-\chi}$ term in Eq. (2.15) is included. This term adjusts the particle drag force to account for the increase in drag at low porosities. The default is **on**.

CCFD set gradp_term **on**
 off

If **on**, the ∇P term from Eq. (2.15) is included. The default is **off**.

CCFD set report_unity_porosity **on**
 off

If **on**, PFC^{3D} will always report a porosity of unity to CCFD during coupling. The default is **off**.

CCFD set report_zero_force **on**
 off

If **on**, PFC^{3D} will always report a zero drag force vector during coupling. The default is **off**.

PRINT ccfid **element, nodes, porosity, force, velocity, gradp, information**

prints CCFD-related variables and information.

2.6 Plot Items

The CCFD option for *PFC^{3D}* adds several plot items to the graphical user interface to show the flow quantities and coupling forces:

ccfd drag	A vector is plotted in the center of each fluid element showing the total body force applied to the fluid element.
ccfd elements	A wireframe of the fluid elements is drawn.
ccfd fap	A vector is plotted at the centroid of each ball showing the force applied by the fluid.
ccfd gradp	A vector is plotted in the center of each of the fluid elements showing the fluid pressure gradient.
ccfd velocity	A vector is plotted in the center of each of the fluid elements showing the fluid velocity.

2.7 FISH Intrinsic

Table 2.1 FISH intrinsic

type	function name	modifiable
INT	ccfd_nle()	
INT	ccfd_nnode()	
INT	ccfd_elenode(INT <i>e</i>, INT <i>n</i>)	
FLT	ccfd_xnode(INT <i>i</i>)	
FLT	ccfd_ynode(INT <i>i</i>)	
FLT	ccfd_znode(INT <i>i</i>)	
FLT	ccfd_elevol(INT <i>i</i>)	
FLT	ccfd_xelecent(INT <i>i</i>)	
FLT	ccfd_yelecent(INT <i>i</i>)	
FLT	ccfd_zelecent(INT <i>i</i>)	
INT	ccfd_xyzele(FLT <i>x</i>, FLT <i>y</i>, FLT <i>z</i>)	
INT	ccfd_ballele(B_PTR <i>bp</i>)	
FLT	ccfd_por(INT <i>i</i>)	yes
FLT	ccfd_xvel(INT <i>i</i>)	yes
FLT	ccfd_yvel(INT <i>i</i>)	yes
FLT	ccfd_zvel(INT <i>i</i>)	yes
FLT	ccfd_xdrag(INT <i>i</i>)	yes
FLT	ccfd_ydrag(INT <i>i</i>)	yes
FLT	ccfd_zdrag(INT <i>i</i>)	yes
FLT	ccfd_xgradp(INT <i>i</i>)	yes
FLT	ccfd_ygradp(INT <i>i</i>)	yes
FLT	ccfd_zgradp(INT <i>i</i>)	yes
FLT	ccfd_xballff(B_PTR <i>bp</i>)	yes
FLT	ccfd_yballff(B_PTR <i>bp</i>)	yes
FLT	ccfd_zballff(B_PTR <i>bp</i>)	yes
FLT	ccfd_t.s()	yes
INT	ccfd_fite()	yes
INT	ccfd_fint()	yes
FLT	ccfd_lemu(INT <i>i</i>)	
FLT	ccfd_elerho(INT <i>i</i>)	
INT	ccfd_ennode()	
INT	ccfd_enface()	

ccfd_nele()	returns the number of elements in the CFD mesh.
ccfd_nnode()	returns the number of nodes in the CFD mesh.
ccfd_elenode(<i>e, n</i>)	returns the node index to the <i>n</i> th node of the specified fluid element. The first argument is the fluid element index ($1 \dots N_e$), and the second is the node number ($1 \dots N_n$). The number of nodes per fluid element is given by ccfd_ennode() .
ccfd_xnode(<i>i</i>)	
ccfd_ynode(<i>i</i>)	
ccfd_znode(<i>i</i>)	
	returns the <i>x</i> -, <i>y</i> - or <i>z</i> -component of the position vector of node <i>i</i> .
ccfd_elevol(<i>i</i>)	returns the volume of fluid element <i>i</i> .
ccfd_xelecent(<i>i</i>)	
ccfd_yelecent(<i>i</i>)	
ccfd_zelecent(<i>i</i>)	
	returns the <i>x</i> -, <i>y</i> - or <i>z</i> -component of the centroid position vector of fluid element <i>i</i> .
ccfd_xyzele(<i>x, y, z</i>)	returns the index to the fluid element containing the specified position vector. This function is slow since it checks the point against each fluid element. See ccfd_ballele(ball) .
ccfd_ballele(<i>bp</i>)	Given a ball pointer, this function returns an index to one of the fluid elements, which contains it or zero. This operation is fast. However, the returned fluid element is not necessarily the fluid element which contains the most volume of the particle or the centroid. If the fluid element containing the centroid is required, use ccfd_xyzele() with the ball centroid components as the arguments.
ccfd_por(<i>i</i>)	the porosity of the specified fluid element

ccfd_xvel(*i*)

ccfd_yvel(*i*)

ccfd_zvel(*i*)

returns the *x*-, *y*- or *z*-component of the fluid velocity vector in fluid element *i*. Note that changing this value only affects the behavior of PFC^{3D}; it does not change the value of the velocity in the CFD code.

ccfd_xdrag(*i*)

ccfd_ydrag(*i*)

ccfd_zdrag(*i*)

returns the *x*-, *y*- or *z*-component of the total drag force in fluid element *i*. Changing this value does not directly affect the behavior of PFC^{3D}, as this value is sent to the CFD code.

ccfd_xgradp(*i*)

ccfd_ygradp(*i*)

ccfd_zgradp(*i*)

returns the *x*-, *y*- or *z*-component of the fluid pressure gradient in fluid element *i*. Changing this value only affects the behavior of PFC^{3D}; it does not change the state of the CFD code.

ccfd_xballff(*bp*)

ccfd_yballff(*bp*)

ccfd_zballff(*bp*)

returns the *x*-, *y*- or *z*-component of the drag force applied by the fluid to a ball.

ccfd_t_s()

returns the current coupling time, *t_s*.

ccfd_fite()

number of iterations of coupling that have occurred

ccfd_fint()

returns the number of steps between recalculating the porosity and force. It is equivalent to the command **SET interval int**.

ccfd_lemu(*i*)

returns the viscosity of the fluid in fluid element *i*.

ccfd_elerho(<i>i</i>)	returns the density of the fluid in fluid element <i>i</i> .
ccfd_ennode()	returns the number of nodes in a fluid element: 4 for tetrahedral meshes, and 8 for hexahedral meshes.
ccfd_enface()	returns the number of faces on a fluid element: 4 for tetrahedral meshes, and 6 for hexahedral meshes.

2.8 References

Di Felice, R. D. "The voidage function for fluid-particle interaction systems," *Int. J. Multiphase Flow*, Vol. 20, No. 1, 153-159 (1994).

Xu, B. H., and A. B. Yu. "Numerical simulation of the gas-solid flow in a fluidized bed by combining discrete particle method with computational fluid dynamics," *Chemical Engineering Science*, Vol. 52, No. 16, 2785-2809 (1997).

3 Verification Examples

3.1 Single Ball Falling

In this section, we describe the application of *PFC^{3D}* and the CCFD add-on to the settling of a spherical particle under gravity in a viscous fluid. Tests are conducted for high and low Reynolds numbers, and a comparison with theoretical results is made.

3.1.1 Analytical Solution

The equation of motion for a spherical particle falling under gravity in a viscous fluid is

$$\frac{4}{3}\pi r^3 \rho_p \frac{du_z}{dt} = \frac{4}{3}\pi r^3 (\rho_p - \rho_f) g - \frac{1}{2}\pi r^2 \rho_f C_d u_z^2 \quad (3.1)$$

where g is the acceleration due to gravity, ρ_p and ρ_f are the particle and fluid densities, u_z is the vertical velocity of the settling particle and C_d is the drag coefficient. The drag coefficient, C_d , is a function of the particle Reynolds number.

As the falling ball reaches a steady velocity, $\frac{du_z}{dt}$ becomes zero. This allows us to write

$$\frac{1}{2}\pi r^2 \rho_f C_d u_z^2 = \frac{4}{3}\pi r^3 (\rho_p - \rho_f) g \quad (3.2)$$

It has been shown that for small Reynolds numbers, the drag coefficient is simply (Batchelor 1967)

$$C_d = \frac{24}{Re_p} \quad (3.3)$$

where Re_p is the particle Reynolds number defined as

$$Re_p = \frac{2r\rho_f u_z}{\mu_f} \quad (3.4)$$

With this definition of C_d , Eq. (3.1) can be solved exactly to give Stokes' law:

$$u_z = \frac{2}{9} \frac{r^2 (\rho_p - \rho_f) g}{\mu_f} \quad \text{for } Re_p \ll 1 \quad (3.5)$$

In many cases relevant to the practice of engineering, the particle Reynolds number is greater than one violating the assumptions of this solution. As the Reynolds number increases, laminar flow separation occurs first, followed by the onset of turbulent flow separation. This flow separation results in a change in drag not described by the low Reynolds number solution. Empirical drag laws have been developed to describe drag forces at higher-particle Reynolds numbers. The following empirical relation describes the drag in a spherical particle for the full practical range of Reynolds numbers:

$$C_d = \left(0.63 + \frac{4.8}{\sqrt{Re_p}} \right)^2 \quad (3.6)$$

Figure 3.1 shows the drag coefficient, C_d , as a function of Reynolds number for the Stokes law and for this empirical drag law.

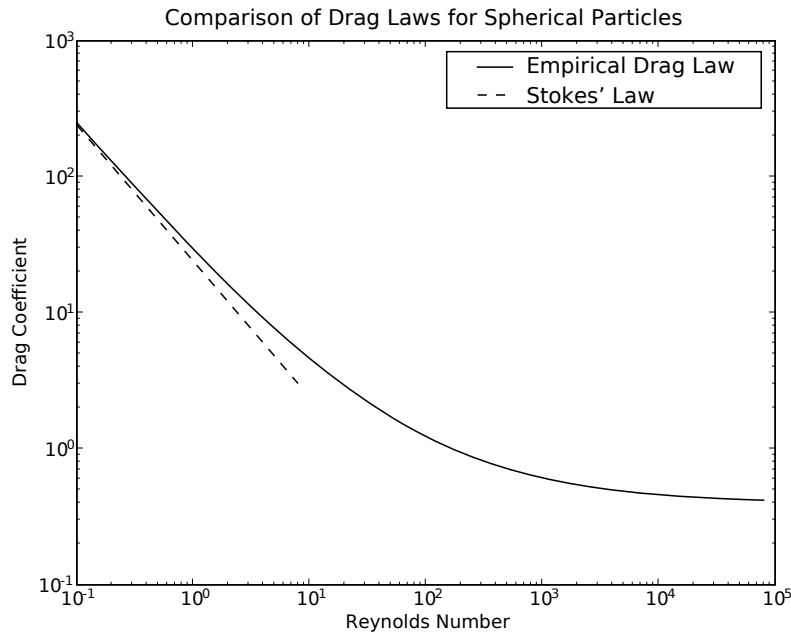


Figure 3.1 Drag coefficient is plotted against Reynolds number for the Stokes law and the empirical drag law used in the CCFD add-on

Substituting Eqs. (3.4) and (3.6) into Eq. (3.2) gives

$$\frac{1}{2} \pi r^2 \rho_f \left(0.63 + \frac{4.8}{\sqrt{\frac{2\rho_f r u_z}{\mu_f}}} \right)^2 u_z^2 = \frac{4}{3} \pi r^3 (\rho_p - \rho_f) g \quad (3.7)$$

This equation can be solved numerically for u_z (see “dropTest1.gid/newton_raphson.dat”).

3.1.2 PFC^{3D}-CCFD Modeling

The model domain consists of a cylinder with a height of 1 m and a diameter of 1 m. No-slip boundary conditions are specified for the circumference and bottom of the cylinder, and a constant pressure condition is specified for the top surface. The cylinder is meshed with 9^3 hexahedral elements. A particle initially at rest is placed 20 cm below the top of the cylinder. The particle is accelerated by gravity to a terminal velocity where the drag and gravitational forces are in equilibrium. The model is run for 0.5 s.

3.1.2.1 Low Reynolds Number Flow

The case of a glass ball 1 cm in diameter settling in glycerol is considered. The density of the particle is 2500 kgm^{-3} , the density and viscosity of glycerol are 1260 kgm^{-3} and $1.5 \text{ Pa}\cdot\text{s}$. For these values, the Stokes law gives a terminal velocity of $-4.50\text{e-}2 \text{ ms}^{-1}$, and the numerical solution of Eq. (3.7) gives $-4.05\text{e-}2 \text{ ms}^{-1}$. This difference is expected: Figure 3.1 shows that the Stokes law predicts less drag than the empirical drag law. This velocity gives a Reynolds number of 0.3, which is approaching the limit of applicability for the Stokes law.

The results of the full coupled numerical solution using PFC^{3D} and CCFD are shown in Figure 3.2. Particle velocity is shown as a function of time. After 0.5 s of coupled calculation, the velocity of the particle is $4.10\text{e-}2 \text{ ms}^{-1}$. The slight deviation from the numerical solution of Eq. (3.7) is the result of the falling particle accelerating the proximal fluid and the small porosity correction term in Eq. (3.10). The GiD project files for this example are located in “dropTest1.gid/.”

3.1.2.2 Data File “pfc_dropTest1.dat”

```
; pfc_dropTest1.dat
set logfile pfccfd.log
set log on ov

damp default local 0.0
ball rad 0.005 x 0. y 0. z .8 id 1
prop dens 2500
prop kn 1e2 ks 1e2

def ftime
  ftime=time
end

history id 1 nstep 2 ball zvel id=1
history id 2 nstep 2 ftime
set grav 0 0 -9.8

config ccfid
```

```

ccfd set bouy on
ccfd set density 1260
ccfd set viscosity 1.5

ccfd connect

plot ball yello
plot add axes
plot set persp off
plot set background white
plot add ccfdelements lgray
plot add ccfdvel blue
plot add hist 1 vs 2

ccfd slave

set plot post
set plot portrait
plot hard file drop1.eps

```

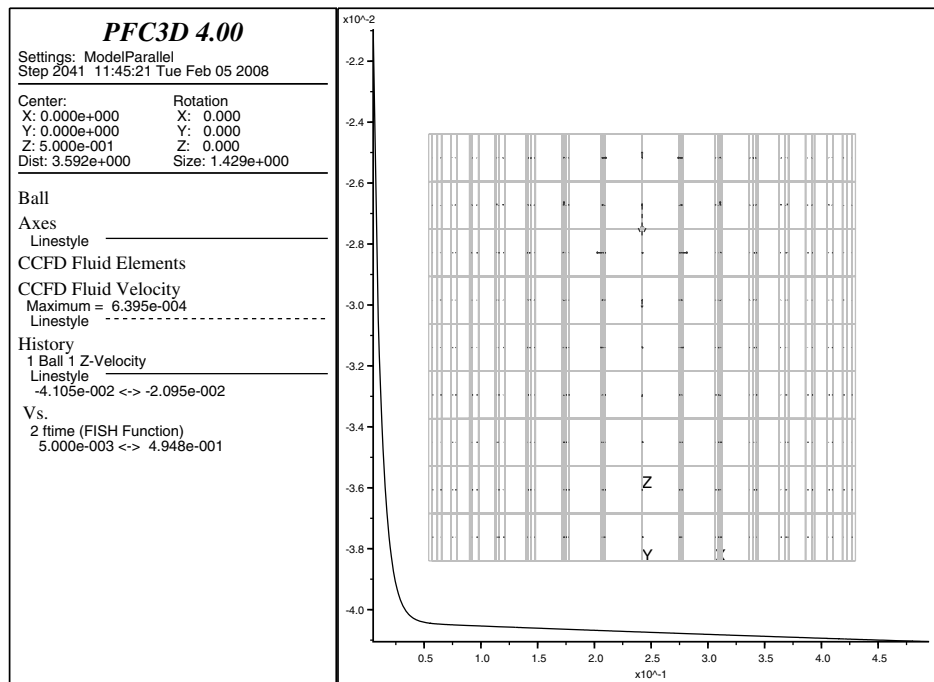


Figure 3.2 *PFC^{3D} output from “dropTest1.gid” after 0.5 s of calculation. The fluid velocity vectors are shown, along with a plot of the PFC^{3D} particle velocity as a function of time.*

3.1.2.3 Higher Reynolds Number Flow

In this case, a spherical particle with a diameter of 1 mm and a density of 2650 kgm^{-3} settles under gravity in water with a density of 1000 kgm^{-3} and a viscosity of $1.0\text{e-}3 \text{ Pa}\cdot\text{s}$. The numerical solution of 7 gives a velocity of -0.142 ms^{-1} . The Reynolds number in this case is 140, well beyond the valid range for the Stokes law.

Figure 3.3 shows the particle velocity as a function of time for the fully coupled solution. After 0.5 s, the particle velocity is 0.142 ms^{-1} . The GiD project files for this example are located in “dropTest2.gid/.”

3.1.2.4 Data File “pfc_dropTest2.dat”

```
set logfile pfc_cfd.log
set log on ov

damp default local 0.0
ball rad 0.0005 x 0.001 y 0.001 z .801 id 1
prop dens 2650
prop kn 1e3 ks 1e3

def ftime
  ftime=time
end

history id 1 ball zvel id=1 nstep 50
history id 2 ftime nstep 50
set grav 0 0 -9.8

config ccf

ccfd set bouy on
ccfd set density 1000
ccfd set viscosity 1.004e-3

ccfd connect

plot ball yello
plot add axes
plot set persp off
plot set background white
plot add ccfdelements lgray
plot add ccfvel blue
plot add hist 1 vs 2

ccfd slave
```

```

set plot post
set plot portrait
plot hard file drop2.eps

```

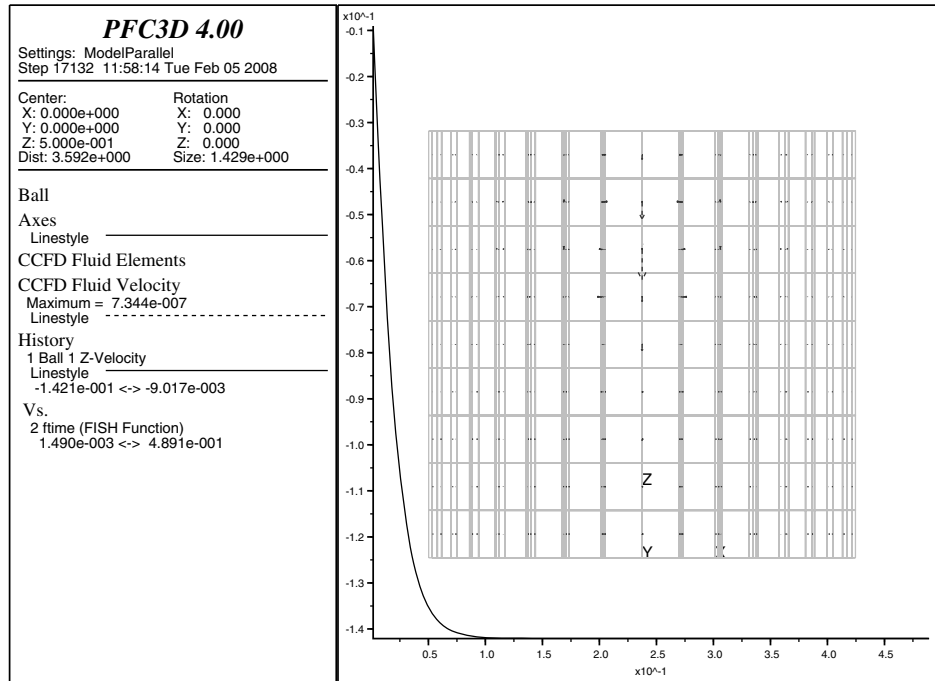


Figure 3.3 *PFC^{3D} output from “dropTest2.gid/” after 0.5 s of calculation. The fluid velocity vectors are shown, along with a plot of the PFC^{3D} particle velocity as a function of time.*

3.2 Porous Media Flow

In this example we consider fully saturated flow through a fixed porous material.

3.2.1 Approximate Solution

Fully saturated flow in a porous medium was first described by Henry Darcy in 1856. Darcy's experiments showed that the rate at which water moves through a column of packed sand is proportional to the cross-sectional area, the hydrodynamic head difference, and is inversely proportional to the length of the column. These results are summarized as Darcy's Law:

$$Q = kA \frac{h_1 - h_2}{L} \quad (3.8)$$

where Q is the volumetric flow rate in m^3s^{-1} , h_1 and h_2 are the hydrodynamic heads in meters, L is the cylinder length in meters, and k is a coefficient of proportionality called the hydraulic conductivity with units of length over time.

Darcy's law can be generalized to give

$$\vec{u}_p = -\frac{K}{\mu_f} (\vec{\nabla} p - \rho \vec{g}) \quad (3.9)$$

where \vec{u}_p is the macroscopic fluid velocity in ms^{-1} , K is permeability with units m^2 , μ_f is the dynamic fluid viscosity in $\text{Pa}\cdot\text{s}$, and $\vec{\nabla} p$ is the pressure gradient vector. The permeability K is a tensor property of the porous medium, independent of the fluid flowing through it. Permeability is often assumed to be isotropic, and is generally proportional to the porosity and the grain size squared.

Several theoretical formulations describe the macroscopic permeability of materials with regular flow pathways. The Kozeny-Carman relation is widely used to estimate the permeability of a porous material in terms of the grain size and porosity:

$$K = B \frac{\epsilon^3}{(1 - \epsilon)^2} d^2 \quad (3.10)$$

where B is a geometric factor taken as $1/180$, d is the grain diameter and ϵ is porosity. This equation is derived by considering the flow through a network of regular pipes (Bear 1972). (For an example of 2D porous flow using *PFC^{2D}*, see *Incorporation of Fluid Coupling in PFC2D* in the **Verification Problems & Example Applications** volume of the *PFC^{2D}* 4.0 manual.)

3.2.2 PFC^{3D}-CCFD Modeling

A PFC^{3D} granular specimen is created with a length of 0.1 m, and height and width of 0.05 m. The particles in the specimen are positioned in a repeated square packing arrangement. (Note that, in general, a regular pack of PFC^{3D} particles is not recommended for mechanical calculations; a regular pack is adopted here for testing purposes.) Each particle has a radius of 1.25e-3 m and is fixed in position. Each hexahedral fluid element contains a 4 × 4 × 4 cube of fixed particles. This packing results in a porosity of 0.476.

The domain is filled with water, and a constant pressure drop of 1e2 Pa is imposed on either end of the domain. Because of the repeating nature of the particles, the porosity, fluid drag force and fluid velocity should be the same in each cell when a steady flow is reached.

Figure 3.4 shows the results of the coupled calculation after 0.15 s of calculation time. The fluid velocity vectors are equal in magnitude and parallel to one another.

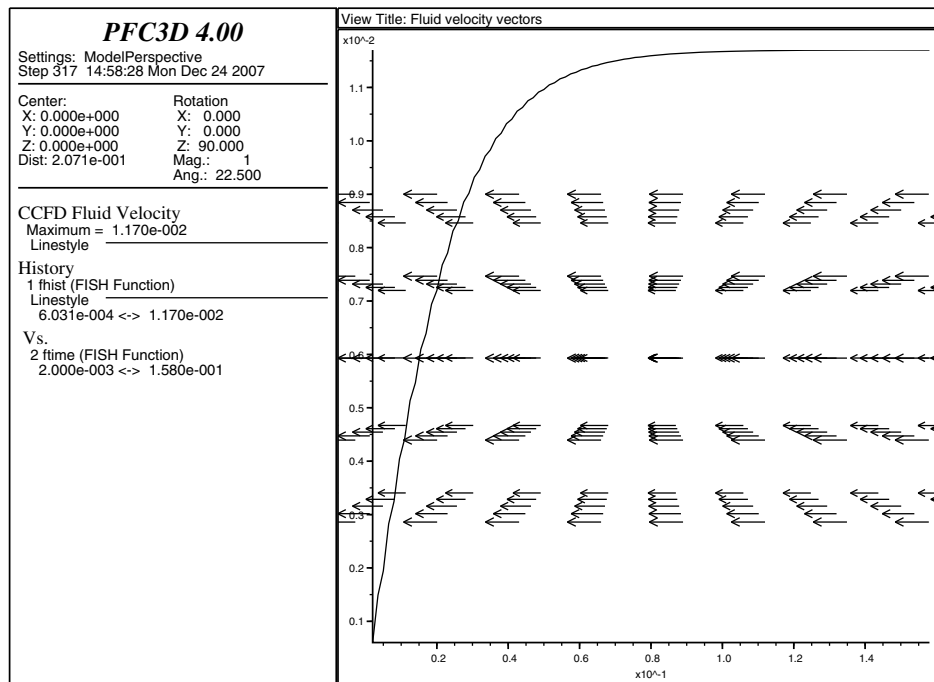


Figure 3.4 PFC^{3D} output for “porous1.gid” after 0.15 s of calculation. The fluid velocity vectors are shown, along with a plot of the magnitude of the fluid velocity as a function of time.

The Kozeny-Carman relation (Eq. (3.10)) predicts a permeability of 1.28e-8 m² for this system. Substituting this permeability into Eq. (3.9) gives a fluid velocity of 1.36e-2 ms⁻¹. The maximum magnitude of the fluid velocity vectors in the simulation is 1.18e-2 ms⁻¹. The difference between these values is a result of different assumptions about the drag on flow through a porous bed. The particle drag force definition given in Eq. (2.10) is not based on the Kozeny-Carman relation. The

comparison of the code output with a complementary theoretical model is given here to show that the model can be applied to low-porosity systems.

At the end of the calculation of this example, the fluid velocity reaches a steady value and is the same in each element. Using this velocity and the regular packing arrangement, we can calculate what the total drag force of each element should be and compare it with the code output.

There is a 4 by 4 by 4 cube of particles in each cell, so the expected total drag force in the y-direction for an element is

$$f_y = 4^3 \left(\frac{1}{2} C_d \rho_f \pi r^2 v_y^2 \right) \epsilon^{-\chi} = -4.83e - 4N \quad (3.11)$$

The command **PRINT ccfid force** can be used to confirm that this value is correct (also see “pfc_test.dat”). The GiD project files for this example are located in “porous1.gid/.”

3.2.3 PFC^{3D} Data Files

To run this example, first run *PFC^{3D}* with “pfc_pack.dat” to create “pack.sav.” The file “pfc_porous1.dat” is used during the coupling. Finally, use “pfc_test.dat” to confirm the result.

3.2.3.1 Data File “pfc_pack.dat”

```
set echo off
set notice off
set warning off

new
def numbers
  minx = -2.5e-2
  maxx = 2.5e-2
  miny = -5.0e-2
  maxy = 5e-2
  minz = -2.5e-2
  maxz = 2.5e-2
  nx = 5
  ny = 10
  nz = 5
  nb = 4
end
numbers

def pack
  dx=(maxx-minx)/float(nx)
  dy=(maxy-miny)/float(ny)
```

```

dz=(maxz-minz)/float(nz)
rad_ = dx/float(2*nb)

loop i(1,nx)
loop j(1,ny)
loop k(1,nz)
  x0_=minx+(i-1)*dx
  y0_=miny+(j-1)*dy
  z0_=minz+(k-1)*dz
  loop ii(1,nb)
  loop jj(1,nb)
  loop kk(1,nb)
    x_= x0_ + rad_ + (ii-1)*2*rad_
    y_= y0_ + rad_ + (jj-1)*2*rad_
    z_= z0_ + rad_ + (kk-1)*2*rad_
    command
      ball rad rad_ x x_ y y_ z z_
    end_command
  end_loop
end_loop
end_loop
end_loop
end_loop
end_loop
end_loop

vol=(maxz-minz)*(maxy-miny)*(maxx-minx)
bvol = nx*ny*nz*nb^3*4./3.*rad_^3*pi
porosity=(vol-bvol)/float(vol)
oo=out(String(vol))
oo=out(String(bvol))
oo=out(String(porosity))
end
pack

set echo on
set notice on
set warning on

sav pack.sav

```

3.2.3.2 Data File "pfc_porous1.dat"

```
res pack.sav
prop kn 1e1 ks 1e1 dens 2600
fix x
fix y
fix z

config ccf
ccfd set bouy off
ccfd set density 998.23
ccfd set viscosity 1.004e-3
ccfd set porosity centroid

def ftime
  ftime=time
end

def fhist
  fhist = ccf_yvel(element)
end

history id 1 fhist nstep 3
history id 2 ftime nstep 3

plot create steady
plot set title text 'Fluid velocity vectors'
plot ccfvel
plot set rot 0 0 90.0
plot set background white
plot add hist 1 vs 2
plot show

ccfd connect

def inline
  element = ccf_xyzele(0,0,0);find the index to the element in the center
end
inline

ccfd slave

set plot post
set plot portrait
```

```
plot hard file porous1.eps
```

```
sav final.sav
```

3.2.3.3 Data File “pfc_test.dat”

```
new
res final.sav

def numbers
  nballs = 4*4*4
  ele = ccfdxzele(0,0,0) ; get center element address
  vy = ccfdyvel(ele) ; y component of fluid velocity
  por = ccfdpore(ele) ; element porosity
  mu = ccfdelemu(ele) ; dynamic viscosity of fluid
  rho = ccfdelerho(ele) ; density of fluid
  r = b_rad(ball_head) ; particle radius
  rep = 2*rho*r*vy/mu ; particle Reynolds number
  cd = (0.63+4.8/sqrt(rep))^2 ; drag coefficient
  chi = 3.7-0.65*exp(-(1.5-log(rep))^2/2.) ; coefficient
  f0 = 0.5*cd*rho*pi*r^2*vy^2 ; force with out porosity correction
  f = -nballs*f0*por^(-chi)
  oo=out('total element force expected : ' + string(f))
  fc = ccfdydrag(ele)
  oo=out('total element force calculated: ' + string(fc))

  KCperm = 1/180.*por^3/(1-por)^2*(2*r)^2
  dx = 0.1
  dp = 1.0e2
  gradp=dp/dx

  uKC = KCperm*gradp/mu
  oo=out('velocity predicted by Kozeny-Carman relation: ' + string(uKC))
  oo=out('velocity output from model: ' + string(vy))
end
numbers
```

3.3 References

Batchelor, G. K. *An Introduction to Fluid Dynamics*. Cambridge University Press, 1967.

Bear, J. *Dynamics of Fluids in Porous Media*. New York: Dover, 1972.

4 Tutorials

4.1 Tutorial 1: Single Steel Ball Dropping in a Vertical Cylinder Filled with Water

4.1.1 Part 1: Creating a Cylindrical Volume

In this section, we will create a model of a cylinder with a diameter of 1 m and a height of 8 m.

4.1.1.1 Creating Points

We are going to create a total of 6 points that will be used in the construction of the base of the cylinder.

1. Start GiD by selecting `PROGRAM FILES/CTC/GID`. Then use `UTILITIES/TOOLS/COORDINATES` to open the *Coordinates window* dialog box.
2. Use `GEOMETRY/CREATE/POINT` to enter point-creation mode. The cursor symbol changes to a + sign, indicating that GiD is ready for input.
3. In the *Coordinates window* dialog, enter (0, 0, 0) for x , y and z , and click to create a point with these coordinates.
4. Then create 5 additional points at coordinates $(-0.5, 0, 0)$, $(0, 0.5, 0)$, $(0.5, 0, 0)$, $(0, -0.5, 0)$ and $(0, 0, 8)$. Click to close the *Coordinates window* dialog box.
5. Use `VIEW/LABEL/ALL` to label the 6 points we have created so far.
6. Adopt an xy -view using `VIEW/ROTATE/PLANE XY`. To better see the points we have created, use `VIEW/ZOOM/IN`, and left-click in the vicinity of the upper-left corner of the model. While holding the left mouse button down, drag the cursor diagonally over the model, then release it ([Figure 4.1](#)).

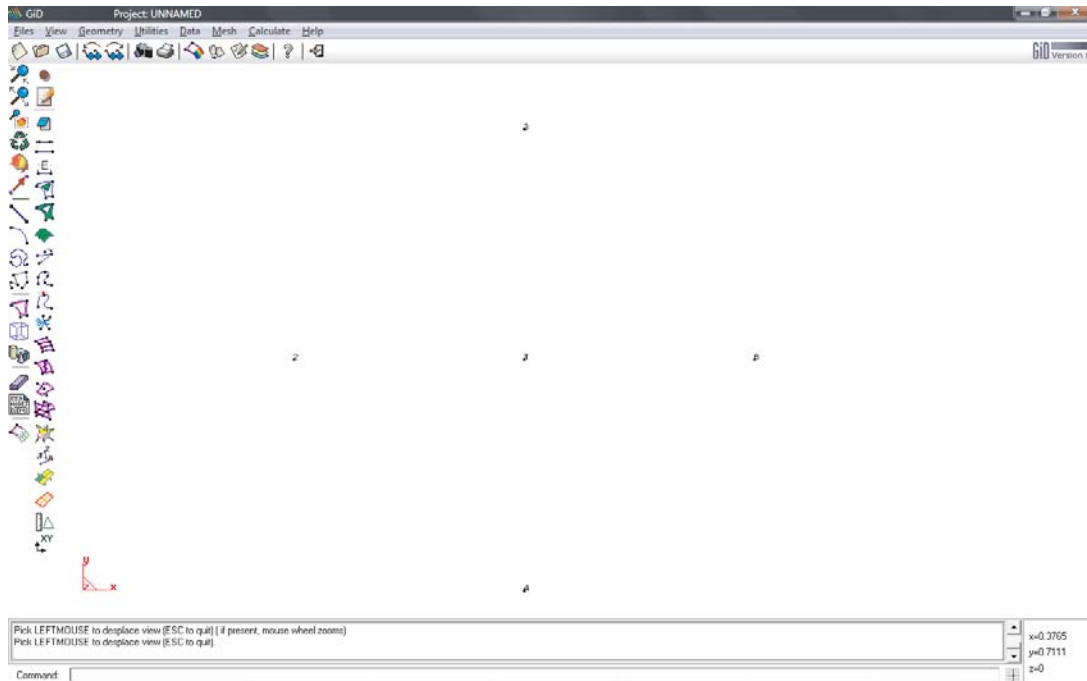


Figure 4.1 *xy-view of the 6 points that have been created*

4.1.1.2 Creating Lines

We will create circular arc sections passing through the points we have created.

1. Select **GEOMETRY/CREATE/ARC/BY 3 POINT** to create an arc passing through 3 points. Note that the cursor symbol changes to a + sign (the default state of the cursor). This cursor symbol means that clicking anywhere on the screen will add that point to the arc. In our case, we want the arc to pass through the points we have already created. In order to make sure that GiD will add the existing point instead of any random coordinate we click on, press **<CTRL-A>** to toggle the cursor to a square symbol. Now, the cursor will only pick existing points. This can also be accomplished by pressing and holding the right mouse button and selecting **CONTEXTUAL/JOIN CTRL A**.
2. Click on points 2 and 3, and note that a floating arc section is created, as shown in [Figure 4.2](#).

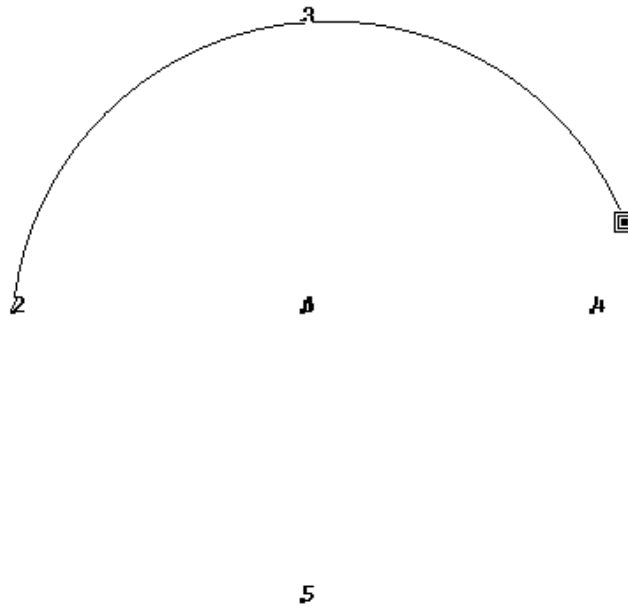


Figure 4.2 *Floating arc section created after clicking on two points*

3. Click on point 4 to complete the arc. To create a second arc, click on point 4, then on point 5, then on point 2. Press <Esc> to exit the arc-creation tool (Figure 4.3).

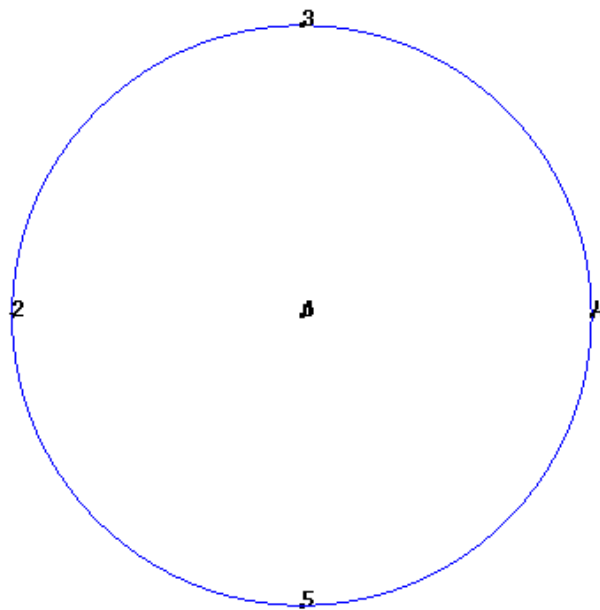


Figure 4.3 *Two completed arcs*

- Use `VIEW/LABEL/OFF` to turn all point labels off. From the drop-down menu in `VIEW/LABEL/ALL IN`, select the picture of the line (the second item from the top), as shown in [Figure 4.4](#).

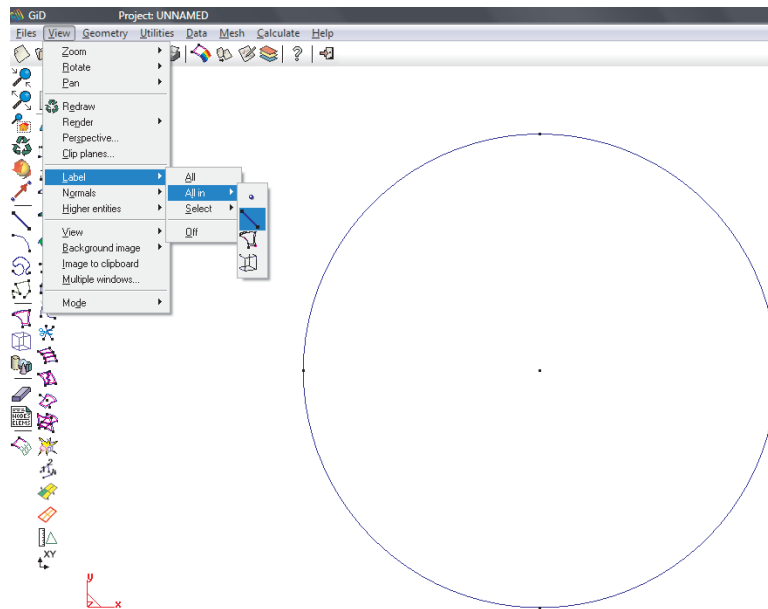


Figure 4.4 *Turning the line labels on*

Note that the upper arc is labeled as line 1, and the lower arc is labeled as line 2.

- We would like to cut each arc into 2 arcs, using a vertical line going through points 3 and 5. To do so, first turn all labels off using `VIEW/LABEL/OFF`.
- Turn all point labels on, using `VIEW/LABEL/ALL IN`, and select the first item (i.e., points).
- Use `GEOMETRY/CREATE/LINE` to create a vertical construction line passing through points 3 and 5. As explained earlier, we can toggle between the + and square cursor symbols by pressing `<CTRL-A>`. The square cursor allows us to pick existing points.
- Click on point 3, then on point 5. Then press `<ESC>` to exit the line-creation tool.
- We are going to cut the upper arc in half by intersecting it with the vertical line. Select `GEOMETRY/EDIT/INTERSECTION/LINE-LINE`. Then click on the vertical line, followed by the upper arc, to complete the intersection operation.
- Then cut the lower arc in two using the same vertical line. Press `<ESC>` to exit the Line-line intersection tool.
- Turn all labels off, then turn all line labels on to label the 4 arcs and the vertical construction line ([Figure 4.5](#)).

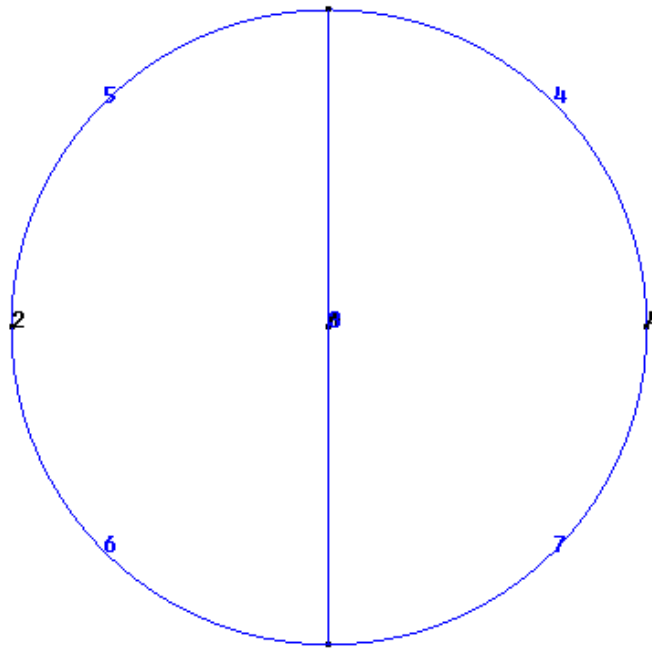


Figure 4.5 *Four arcs defining the base of the cylinder*

4.1.1.3 *Creating the Base and Volume of the Cylinder*

1. Using `GEOMETRY/DELETE/LINE`, select the vertical construction line, then press `<Esc>` to delete it. Turn all labels off, and then turn only line labels on.
2. Using `VIEW/ROTATE/ISOMETRIC`, adopt an isometric perspective view.
3. Using `VIEW/ZOOM/FRAME`, fit the model to our window so we can see everything ([Figure 4.6](#)).



Figure 4.6 View of the complete model

4. To create the surface representing the base of the cylinder, use `GEOMETRY/CREATE/NURBS SURFACE/BY CONTOUR`. Left-click somewhere outside and close to the upper-left corner of the model. While holding down the left mouse button, drag the cursor down and right, over the model, stretching the expanding rectangle. Drag past the lower-right corner of the model and release the left mouse button to select the 4 arc sections representing the base of the cylinder. Press `<Esc>` to exit the surface-creation tool. Note that the resulting surface is represented by a pink oval.
5. Use `VIEW/LABEL/ALL IN`. In the drop-down menu, select the surface (third item from the top) in order to label the newly created surface representing the base of the cylinder, and zoom in ([Figure 4.7](#)). Note that the base is labeled as surface number 1.

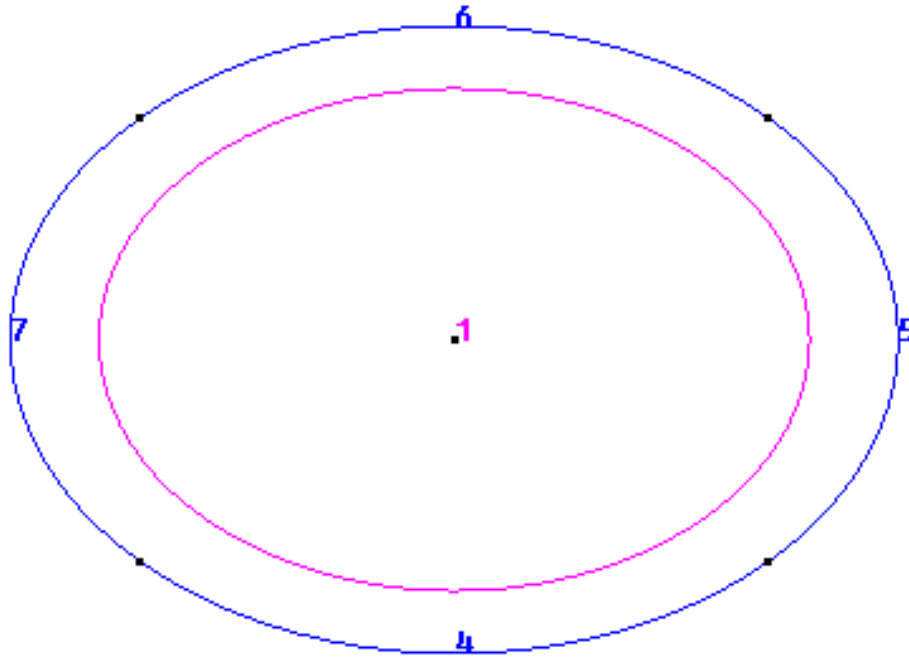


Figure 4.7 Surface (in pink) representing the base of the cylinder

6. Turn all labels off, and then turn all point labels on. We are going to create the volume by extruding the base. We will translate and copy the base from point 1 to point 6. In doing so, we extrude the base to create the volume.
7. Use UTILITIES/COPY to open the *Copy* dialog box. Select Surfaces in the Entities type field, and select Translation in the Transformation field.
8. In the First point field, click on the button located at the bottom of the First point section. Then click on Point 1, which represents the center of the base. In the Second point section, click the button, and then click on point 6 in our model. Note that after each pick operation, the coordinates of that point appear in the corresponding window in the dialog box.
9. Select Volumes in the Do extrude field, and then click (Figure 4.8).

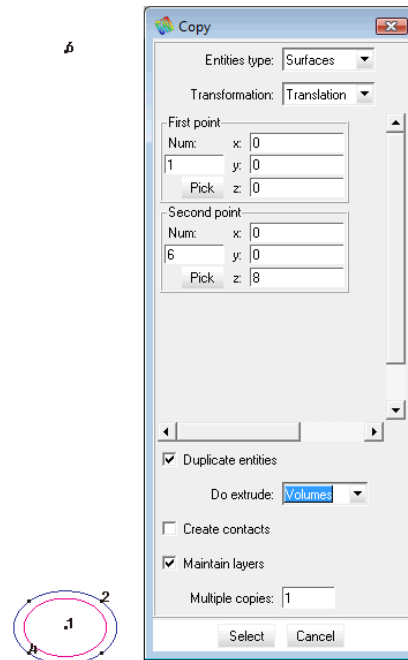


Figure 4.8 Copy dialog box showing the parameters used for base extrusion

Our cursor symbol changes to a square, indicating that we must now select the surface that will be extruded.

10. Click on the pink oval labeled surface number 1. Then press <Esc> to end the selection of surfaces and complete the extrusion. Click **FINISH**, then close the *Copy* dialog box. Label all entities to see the 4 newly formed surfaces (shown in pink) and the volume (shown in light blue and labeled as volume 1) (Figure 4.9). Use FILE/SAVE AS to save our model as “Cylinder.gid.”

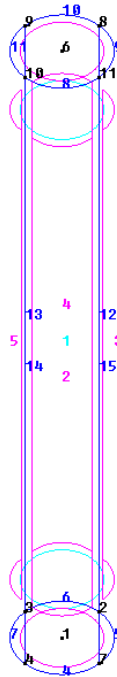


Figure 4.9 *Volume and surfaces resulting from the extrusion of the base*

4.1.2 Part 2: Fluid Flow Model Setup

In this section, we will define the CFD problem, boundary and initial conditions, and the additional parameters needed for the CFD part of the calculation.

4.1.2.1 Material Definition

1. If our model is not open, start GiD. Using `FILE/OPEN`, read in the “Cylinder.gid” project folder. Use `DATA/PROBLEM TYPE/CFD` to indicate that we are going to set up a CFD problem. Use `VIEW/ZOOM/IN` to magnify the details of the model.
2. Use `VIEW/LABEL/ALL` to label everything in the model. Select `DATA/MATERIALS` to open the *Materials* dialog box. Select `WATER` as the material in the uppermost field of the dialog box, click `ASSIGN` and select Volume. The cursor changes to a square symbol, indicating that we must now select a volume to which the material “water” will be assigned.
3. Click on the cylinder volume (drawn in light blue), press `<Esc>` and click `FINISH` in the *Materials* dialog to terminate the material-definition stage.
4. To verify that the correct material has been assigned to the correct volume, while the *Materials* dialog box is still open, click `DRAW` and select All materials to see the current material-assignment state of the model, as shown in [Figure 4.10](#).

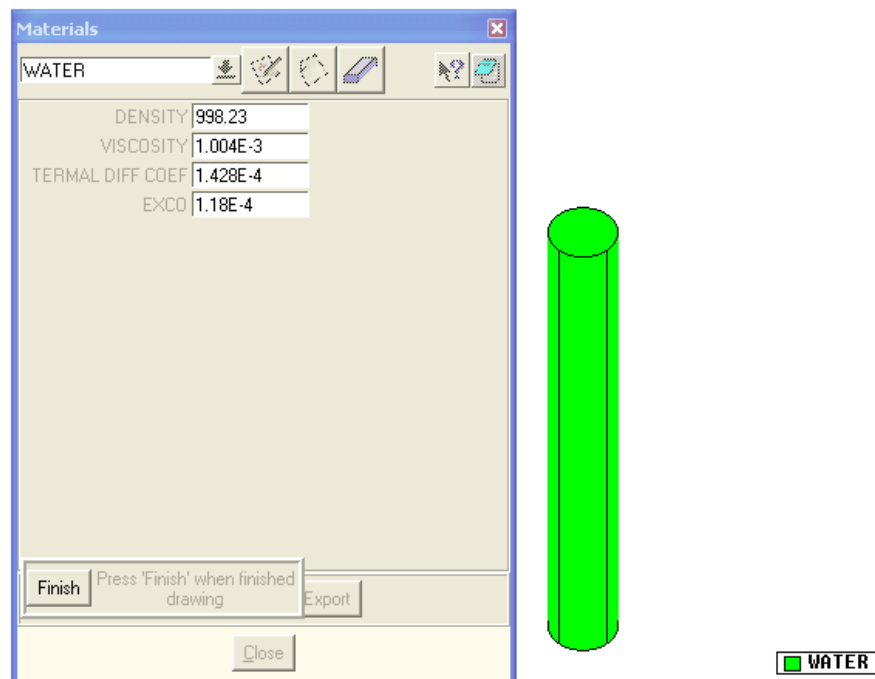


Figure 4.10 A representation of the volumes and their material assignment in the model

5. Click `FINISH`, then `CLOSE`, to exit material definition.

4.1.2.2 Problem Data

1. Use DATA/PROBLEM DATA to open the *Problem Data* dialog box. Select the General tab.
2. Enter Cylinder in the TITLE field, choose EXPLICIT in the Type of problem field, choose NO in the Turb (Turbulence) model field, enter 1 in the Sweep field, enter 0.2 in the Relax factor P field, and enter 0.0 in the Scheme(0.0-1.0) field. Click to complete the General tab (Figure 4.11).

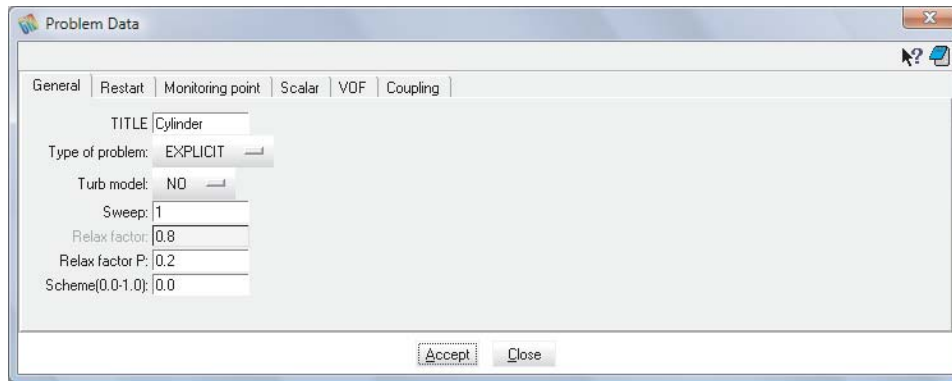


Figure 4.11 The General tab of the Problem Data dialog box

The fields in the Restart, Monitoring point, Scalar and VOF tabs should be left at their default values, as shown in [Figure 4.12](#).

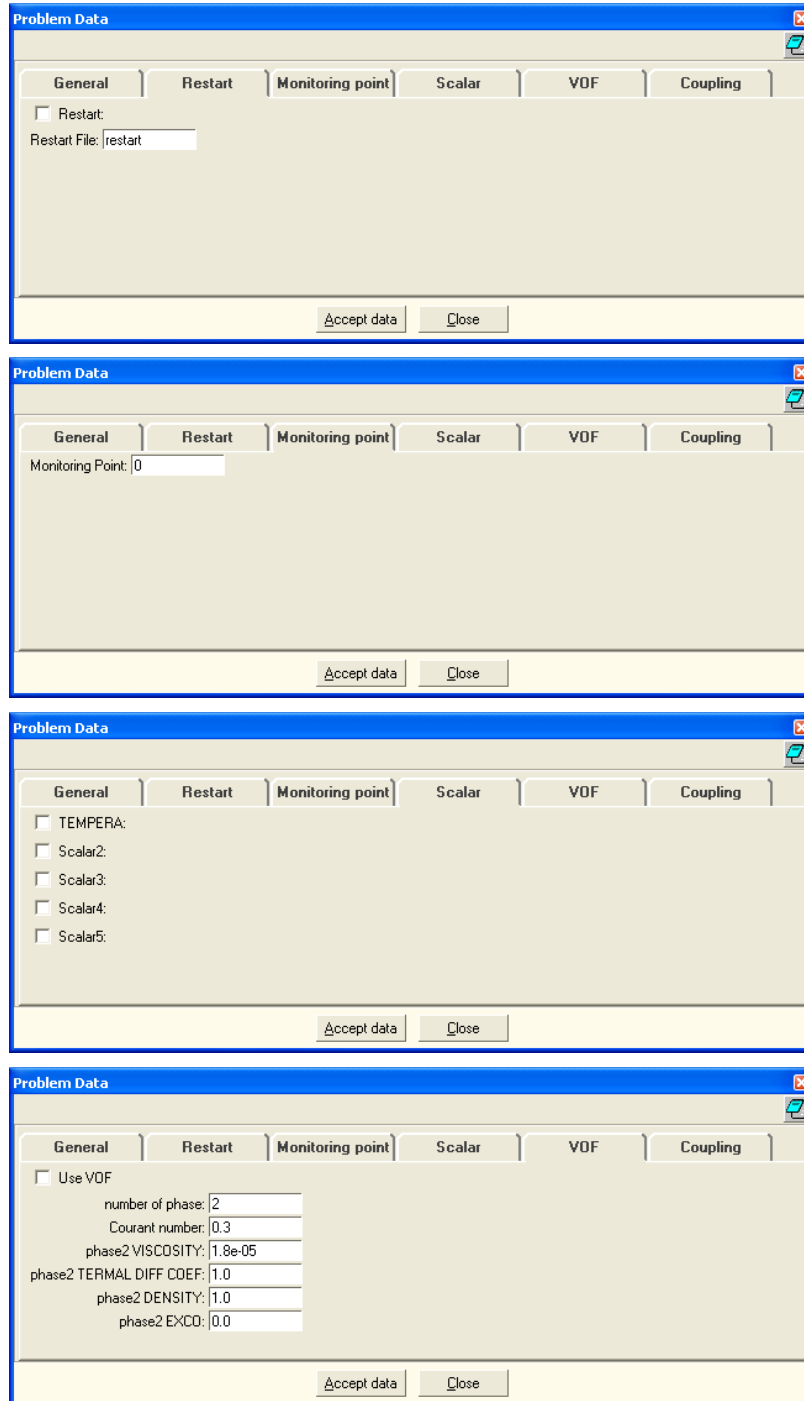


Figure 4.12 The Restart, Monitoring point, Scalar and VOF tabs

3. In the Coupling tab, check the Use Coupling field. In the Coupling Type field, select the third entry (i.e., CFD [unsteady] <-> FINAS [dynamic]).
4. Clicking on any of the Time Table buttons opens 3 fields. Enter 0 for start, 1 for end, and 0.01 for step (Figure 4.13). Click Problem Data dialog box.

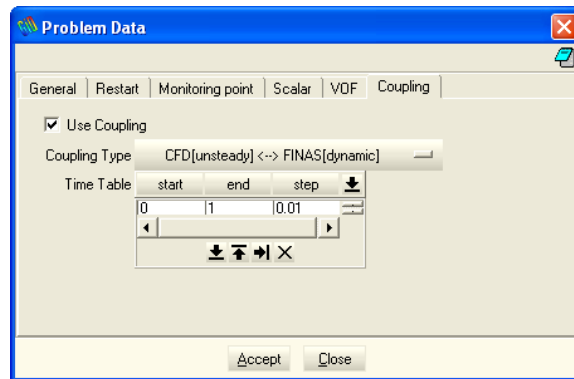


Figure 4.13 *The Problem Data dialog box showing the coupling type and the timestep specification*

4.1.2.3 Interval Data

1. Use DATA/INTERVAL DATA to open the *Interval Data* dialog box. In the Control tab, leave the Number of Steps set at 1. Enter 1.0 in the Time step increment field, and enter 1 in the Step freq. post process field.
2. The fields in the Gravity and Criterion temp tabs should be left at their default values (Figure 4.14).

NOTE: “Start” and “end” in the Coupling tab of the *Problem Data* dialog represent, respectively, the start and end time of coupling between *PFC^{3D}* and CCFD. “Step” represents the timestep between each data exchange. If the coupling between the fluid flow and particles is strong (i.e., if the flow strongly affects particle velocities and distribution, or if particles strongly affect the flow), “step” should remain small.

If “start” is greater than 0, the CCFD calculation will start at time 0 and proceed until coupling begins at time “start.” “Start” should be less than “end,” and “step” should be a whole fraction of (“end” – “step”).

“Number of Steps” in the Control tab of the *Interval Data* dialog represents the number of fluid flow calculation steps, and “Time step increment” represents the duration of each computational step. If the fluid flow computation is explicit and the Time step increment is larger than the one dictated by the CFL criterion, CCFD will perform several inner explicit calculation steps in the course of a Time step increment. For practical purposes, you could choose to complete the CFD calculation in 1 large step. “Step freq. post process” represents the frequency of result output for post processing.

By setting Step freq. post process equal to 1, you can effectively control the number of outputs independently using Number of Steps. You should then select an appropriate Time step increment so that the CCFD simulation lasts a total of (Time step increment) \times (Number of Steps).

The CCFD calculation lasts for (Number of Steps) \times (Time step increment). If the CCFD calculation terminates before the end of the coupling (end), the *PFC^{3D}* calculation will wait for data from CCFD. On the other hand, if the coupling ends before the end of the CCFD calculation, the latter will continue without coupling, and will end normally.

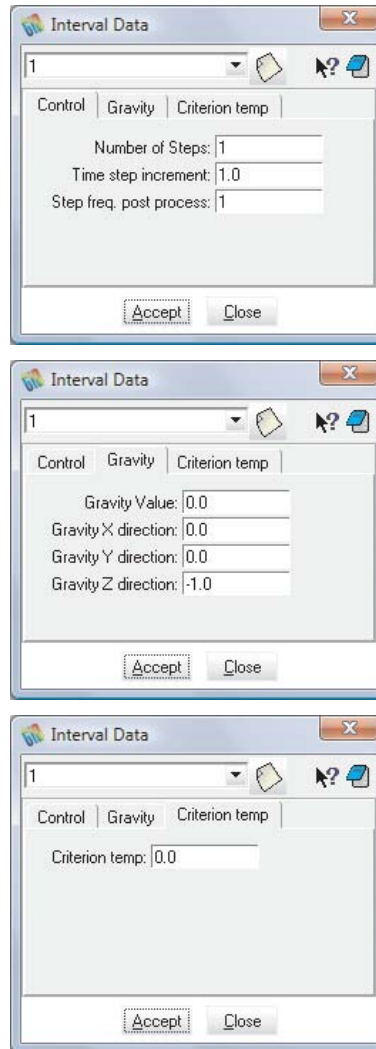


Figure 4.14 *The Control, Gravity and Criterion temp tabs of the Interval Data dialog box*

4.1.2.4 Initial Value Data

1. Use **DATA/CONDITION**, and select **Initial** to open the *Initial* dialog box. In this dialog box, we can specify the initial conditions of the CFD problem. In this problem, the water is still, so the initial conditions are trivial.
2. Click **ASSIGN** to assign the default values appearing in the dialog box to a volume. Click on volume 1 (drawn in light blue) to select it, and then click **FINISH**. To check where initial conditions have been applied, click **DRAW** and select **All initial** to see the initial value applied to the center of volume 1 (Figure 4.15). Click **FINISH** to close the dialog box.

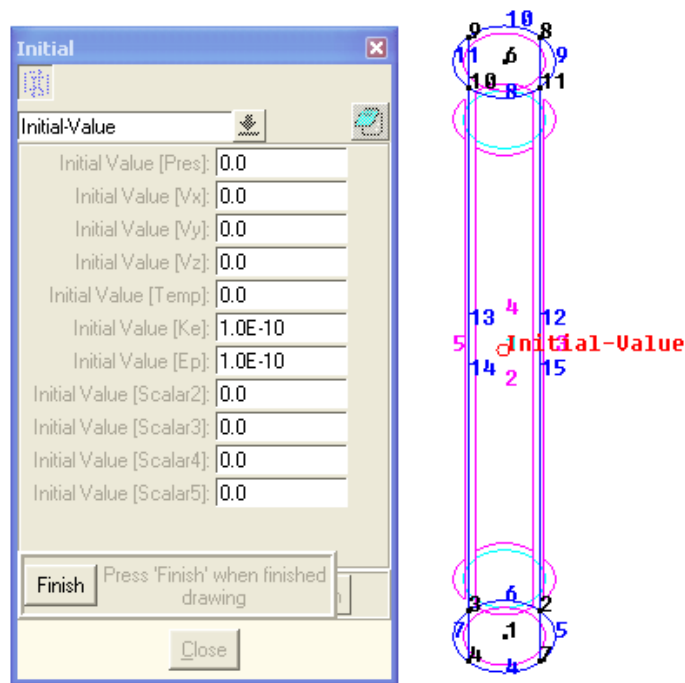


Figure 4.15 Applying initial conditions

4.1.2.5 Boundary Conditions

1. Use **DATA/CONDITION** and select **Boundary** to open the *Boundary* dialog box. In the drop-down menu, select **FIXED-VELOCITY** and enter 0 in the X-Value, Y-Value and Z-Value fields.
2. Click **ASSIGN**. We must now select all the surfaces on which this boundary condition will be applied. In this example, the fixed-velocity boundary condition is applied to all surfaces except the base, for a total of 5 surfaces. Please note that surfaces are represented in pink. Selecting a surface will change the color of the lines representing the surface from pink to red.

3. Click on the top and the 4 lateral surfaces. Press <Esc> or click to exit the surface-selection tool.
4. In the drop-down menu at the top of the *Boundary* dialog box, select FIXED-PRESSURE and set its value to 0. Click and click on the base of the cylinder to apply this boundary condition to this surface. Then press <Esc> or click to exit the surface-selection tool.
5. To view all of the boundary and initial conditions, click and select All Boundary (Figure 4.16). Click to close the dialog box.

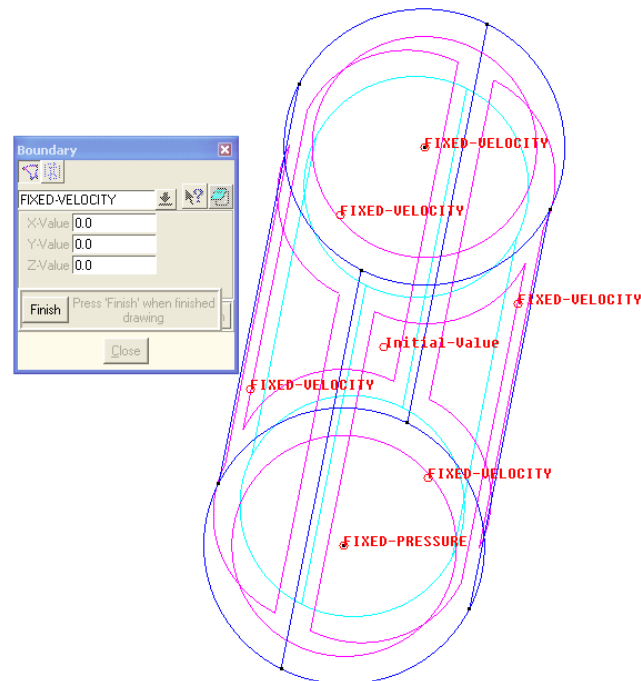


Figure 4.16 Summary of all initial and boundary conditions

6. Save our model as “Cylinder.gid.”

4.1.2.6 Mesh Generation

In CCFD, we can use both hexahedral and tetrahedral meshes. In this example, we will create a structured, or mapped, hexahedral mesh.

1. Continue on from the previous section or read in the “Cylinder.gid” project file. Using MESH/ELEMENT TYPE/HEXAHEDRA, select volume 1 in order to specify our choice of elements.
2. Use VIEW/ROTATE/ISOMETRIC and VIEW/ZOOM/IN to adopt a clear and detailed view of our model, enabling us to pick any line. Turn all labels off, and remember that points are

represented as black dots, lines are in blue, surfaces are in pink and volumes are in light blue.

3. A mapped mesh is composed of a collection of cubes, deformed to match the shape of the cylinder. We will mesh this volume as a single block. This is also called a structured mesh and has a natural orientation along the bottom-top, East-West and North-South directions. Use `MESH/STRUCTURED/VOLUMES`. The cursor symbol changes, and we see at the bottom-left of the screen that we must select volumes bound by six faces. Here, we have only one volume. Click on it to select it, then press `<Esc>`. Exiting volume selection immediately opens an Enter value window.
4. We only need three numbers to mesh this cylinder as a stretched cube. This dialog box enables us to specify the number of subdivisions in the bottom-top, East-West and South-North directions. Enter 5, then click `ASSIGN`, and select a segment along the East-West direction. (In fact, any of the 4 arcs forming the base of the cylinder would do.) As soon as we click on that line, we will notice that the 3 other lines (the quarter arc located opposite the one we chose, as well as the two corresponding arcs on the top lid of the cylinder) turn red. This means that they have been selected, and that the number of subdivisions on each selected arc has been set to 5.
5. While we are still in the line-selection mode, select an arc adjacent to the one we clicked, and set the number of subdivisions in the South-North direction to 5 ([Figure 4.17](#)).

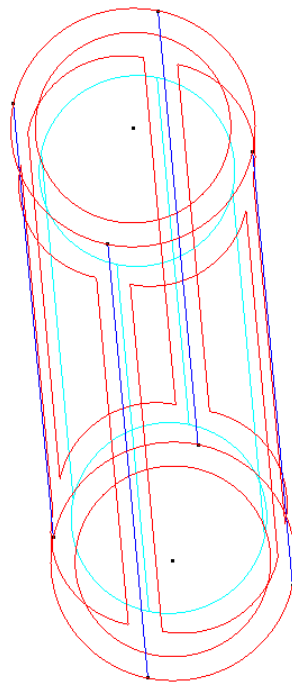


Figure 4.17 *Model after assignment of number of meshing subdivisions in the East-West and South-North directions. Note that the only remaining unassigned (blue) lines are in the bottom-top direction.*

6. Press <Esc> to exit the line-selection mode. The Enter value window reappears. Enter 40, click and select any of the 4 remaining straight blue lines to assign the number of subdivisions along the bottom-top direction. Note that the 3 remaining bottom-top lines also turn red.
7. Press <Esc> to exit the line-selection mode. Press <Esc> again or click to close the Enter value window.
8. We have now completed the subdivision assignment and are ready to generate the mesh. Use MESH/GENERATE MESH. A window appears, reporting the progress of mesh generation, followed by an informational window, reporting that 1000 hexahedral elements and 1476 nodes were created. Click to see the resulting mesh ([Figure 4.18](#)).

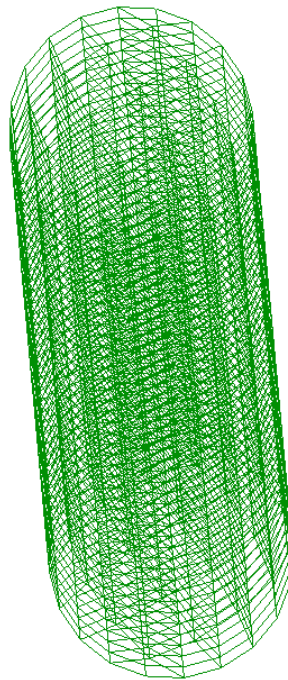


Figure 4.18 *Structured hexahedral mesh of the cylinder*

9. Use FILE/SAVE to save the geometry (points, lines, surface and volume), all operating conditions and the mesh of our model.

4.1.3 Part 3: PFC^{3D} Data File Preparation

The PFC^{3D} portion of the computation needs a data file called “pfc_Cylinder.dat” (located in the “Cylinder.gid” folder), which describes the location, size and properties of particles.

4.1.3.1 Data File “pfc_Cylinder.dat”

```
set logfile pfcccfid.log
set log on ov

config ccfid

damp default local 0.0
ball rad 0.004 x 0.2 y 0.2 z 7.8 id 1
prop dens 7.0e3
prop kn 1e2 ks 1e2

history id 1 ball zvel id=1

ccfid set bouy on
ccfid set density 998.23
ccfid set viscosity 1.004e-3

set grav 0 0 -9.8

ccfid connect

plot ball yello
plot add axes
plot set persp off
plot set background white
plot add ccfdelements lgray
plot add ccfdffap black
plot add ccfidvel blue
plot add hist 1

pause
ccfid slave
```

4.1.4 Part 4: Launching the Calculation

In this section, we will learn to launch *PFC^{3D}* with the CCFD option.

1. Start GiD and open the “Cylinder.gid” project.
2. Using CALCULATE/CALCULATE, open the *FINAS-CFD* dialog box. Click , and after a short time the *Open* dialog box opens. This box lists all of the files with the extension “.dat” that are located in the “Cylinder.gid” project.
3. We must now read the *PFC^{3D}* data file needed for the computation. Select the file “PfcCylinder.dat” and click (Figure 4.19).

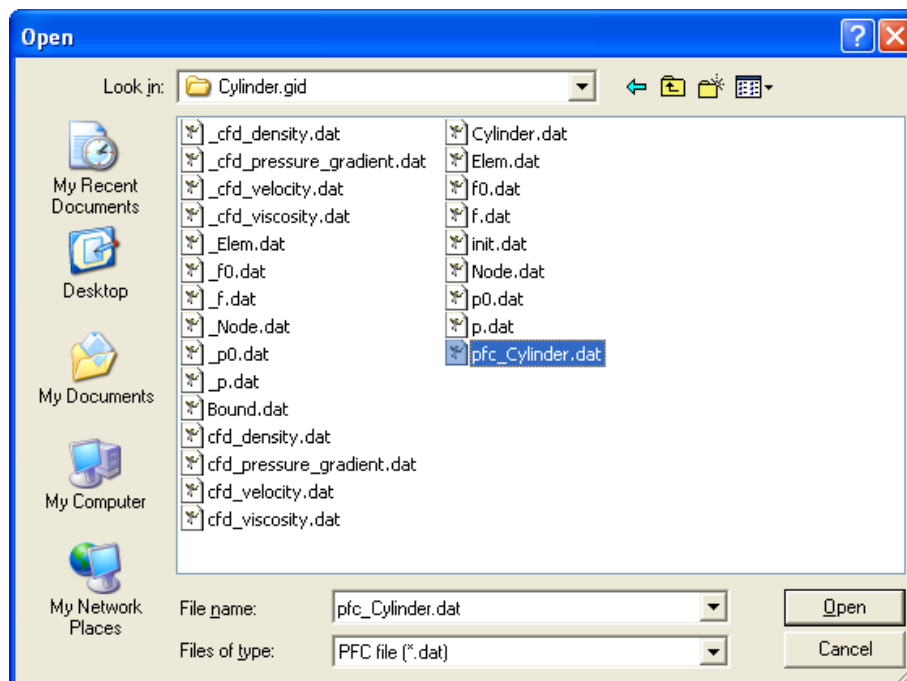


Figure 4.19 Reading the *PFC^{3D}* data file

4. After reading the *PFC^{3D}* data file, a proxy program which manages communications between *PFC^{3D}* and the CCFD solvers (Figure 4.20) is initiated.

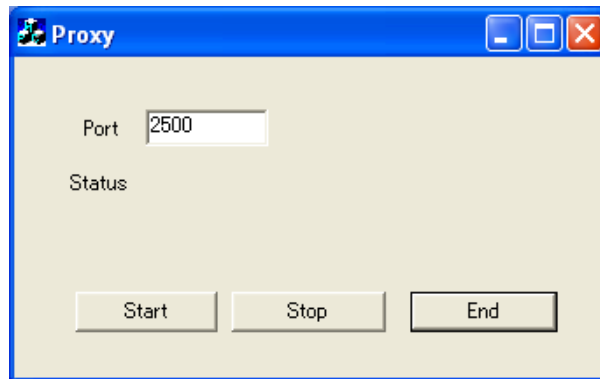


Figure 4.20 Proxy program about to start

- Clicking launches the coupled PFC^{3D} -CCFD computation. A PFC^{3D} window opens, and we will notice that several commands are executed, until the word “pause” at the bottom of the PFC^{3D} data file is reached (Figure 4.21).

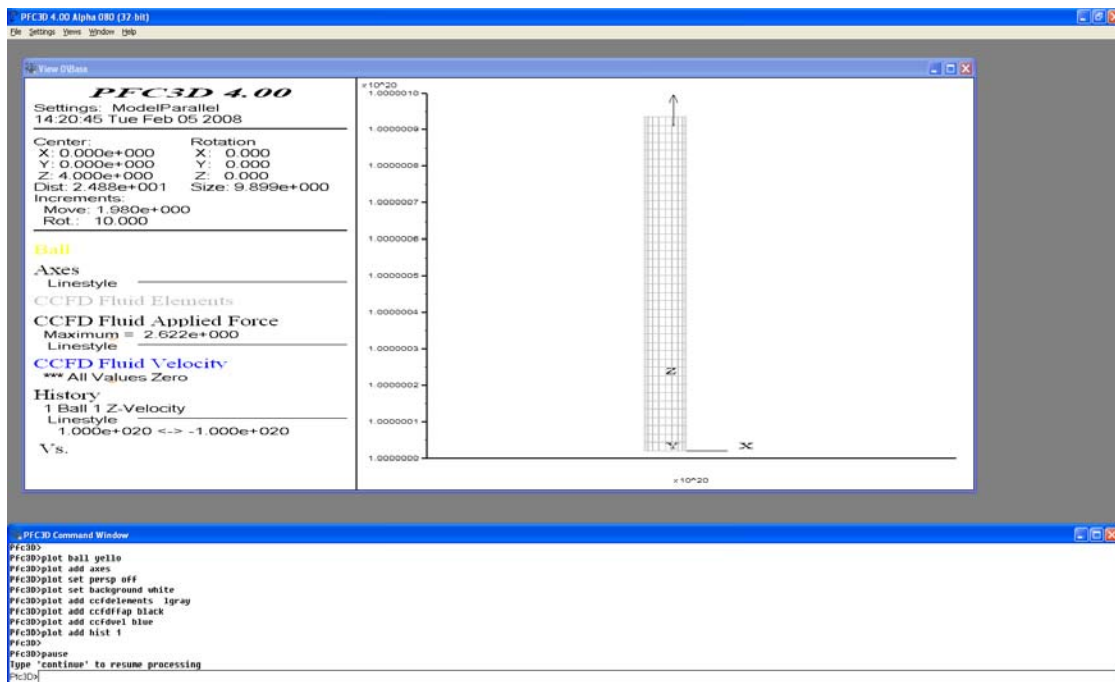


Figure 4.21 PFC^{3D} screen launched by the proxy program

- Type “continue” in the command window of PFC^{3D} to resume the computation. At pre-set time intervals (defined earlier in Figure 4.13), PFC^{3D} and CCFD exchange information. Use the *Windows* taskbar to see the CCFD screen displaying the current state of the CFD computation (Figure 4.22).

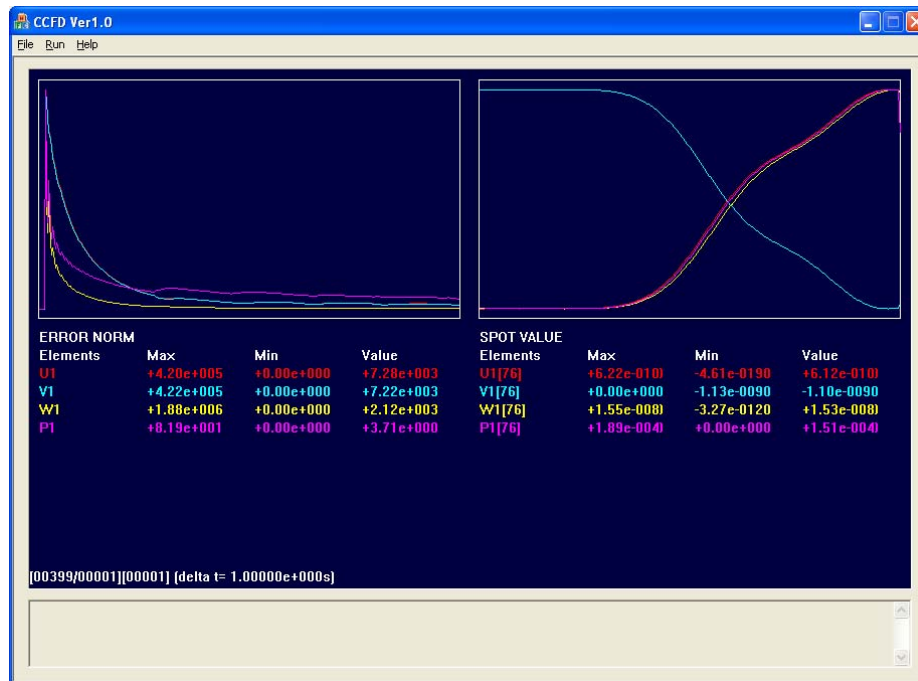


Figure 4.22 CCFD screen running in conjunction with PFC^{3D}

7. The left-hand graph shows the normalized residual errors. The right-hand graph monitors quantities at a mesh point whose label has been previously entered into the Monitoring point tab of the *Problem Data* dialog box. We can select the monitoring mesh point by displaying the mesh and turning point labels on. The computation will stop after one second of problem time, as specified in the Coupling tab of the *Problem Data* dialog box (Figure 4.13).

4.1.5 Part 5: Visualizing the Results

In this section, we will use GiD to visualize the flow results.

1. Click in the proxy server window.
2. Click on in the dialog that appears, to invoke post-process mode in GiD.
3. Use VIEW/ROTATE/PLANE XZ and VIEW/ZOOM/FRAME to get a view of the cylinder.
4. Use VIEW RESULTS/CONTOUR FILL/PRESSURE to draw contours of the fluid pressure (Figure 4.23).

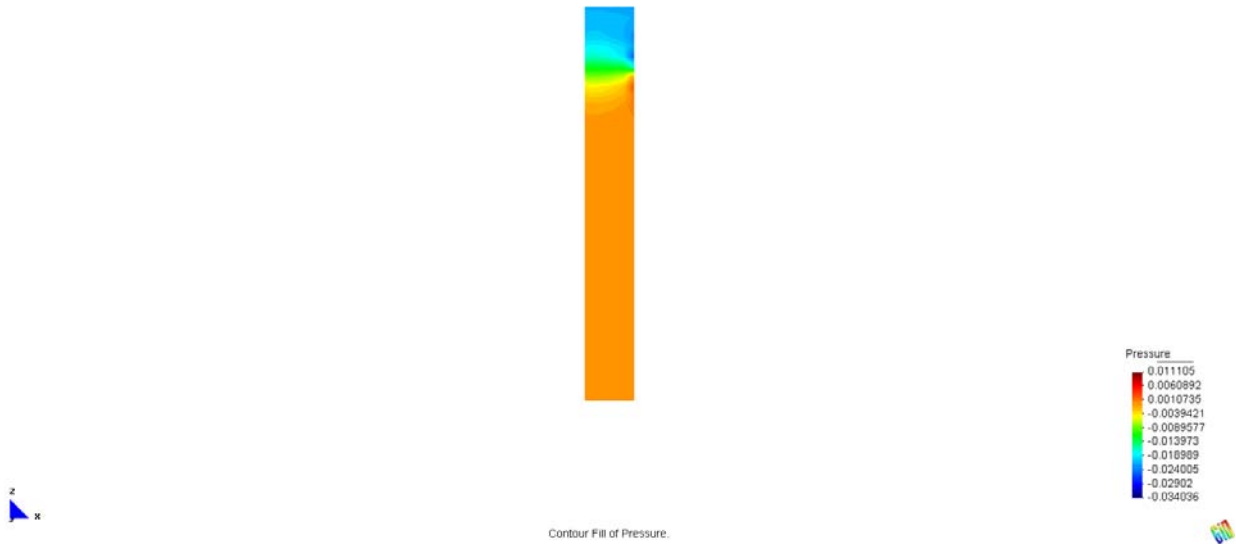


Figure 4.23 *Contours of fluid pressure using GiD in post-process mode*

5. Use `VIEW RESULTS/CONTOUR FILL/VELOCITY/MAG` to draw contours of the fluid velocity magnitude.
6. For more information about post-processing using GiD, access the GiD documentation via the Help menu.

4.2 Tutorial 2: Particles Convected with the Flow along a Curved Pipe

4.2.1 Part 1: Creating an Elbow

In this section, we will create a model of a 90° elbow, featuring a long horizontal inlet and a short vertical outlet.

4.2.1.1 Creating Points

We are going to create a total of 7 points that will be used in the construction of the elbow.

1. Start GiD and use UTILITIES/TOOLS/COORDINATES WINDOW to open the *Coordinates window* dialog box.
2. Use GEOMETRY/CREATE/POINT to enter point-creation mode. The cursor symbol changes to a + sign, indicating that GiD is ready for input.
3. In the *Coordinates window* dialog, enter $(-2, 0, 0)$ for x , y and z , and click to create a point.
4. Then, create 6 additional points at coordinates $(-2, 0, 0.05)$, $(-2, 0, -0.05)$, $(-0.4, 0, 0)$, $(-0.4, 0, 0.4)$, $(0, 0, 0.4)$ and $(0, 0, 1)$. Click to close the *Coordinates window* dialog box.
5. Use VIEW/LABEL/ALL to label the 7 points we have created.
6. Adopt an isometric view using VIEW/ROTATE/ISOMETRIC. To better see the points we have created, use VIEW/ZOOM/IN, and left-click close to the upper-left corner of the group of points labeled 1, 2 and 3. While holding the left mouse button down, drag the cursor diagonally, stretching the expanding rectangle. Then release the left button when the 3 points are well inside the rectangle, in order to get a good view of them ([Figure 4.24](#)).

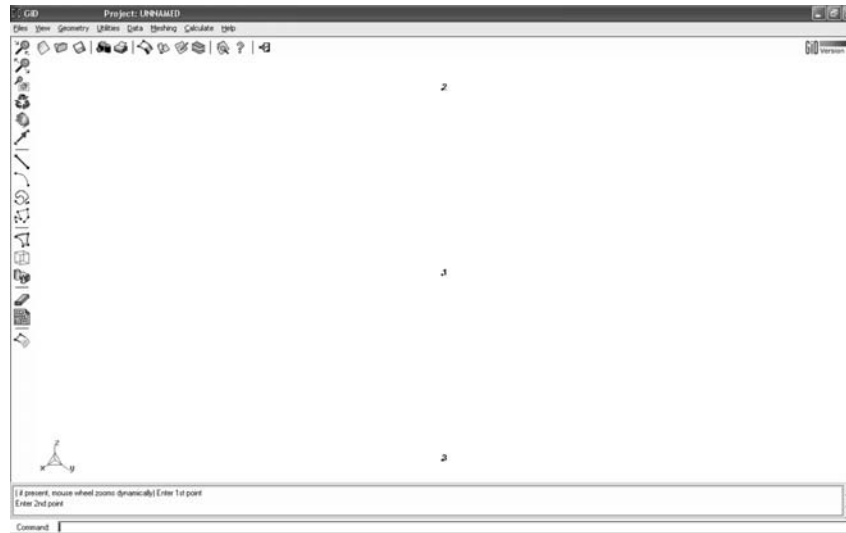


Figure 4.24 Isometric view of points 1, 2 and 3

4.2.1.2 Creating Lines

We will create 4 circular arcs forming a circle centered at point 1, orthogonal to the x -axis and passing through points 2 and 3.

1. Select `GEOMETRY/CREATE/LINE` to activate the polyline-creation tool. Note that the cursor symbol changes to a + sign, which is the default state of the cursor. This cursor symbol means that clicking anywhere on the screen will add that point to the arc. In our case, we want the arc to pass through the points we have already created. In order to make sure that GiD will add the existing point instead of a random coordinate, press `<CTRL-A>` to toggle the cursor to a square symbol. Now, the cursor will only pick existing points.
2. Click on point 2, then on point 3. Press `<Esc>` once to specify that we will not be adding any new points to this polyline. Press `<Esc>` again to exit the polyline-creation tool (Figure 4.25).



Figure 4.25 *Straight line passing through points 2 and 3*

3. We need to create a copy of this line and rotate it along an x -axis passing through point 1. Select UTILITIES/COPY to open the *Copy* dialog box.
4. Enter Lines in the Entities type field, and enter Rotation in the Transformation field. Select an angle of 90 degrees.
5. To specify the rotation axis, click on the button located at the bottom of the First point section. The cursor changes to a square symbol. Click on point 1. In the Second point section, click the button, then click on point 1 again.
6. Having selected point 1 as each of the two end points of the rotation axis, we have an axis of length zero. Click in the x -field of the Second point and change the -2 to 0 . Click in the Num field of the Second point, and delete label 1 (appearing in this field as a result of clicking on point 1). Now the two points are distinct, and the rotation axis is along the x -direction.
7. Leave Duplicate entities checked, and leave No in the Do extrude field. Click to start selecting the entities we want to copy ([Figure 4.26](#)).

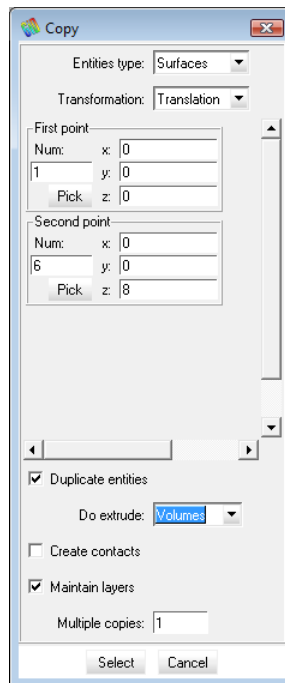


Figure 4.26 *The Copy dialog box used for copying and rotating the vertical line*

8. Click on the line joining points 2 and 3, then press <Esc> or click **FINISH** to exit the *Copy* dialog box. [Figure 4.27](#) shows the result of copying and rotating the vertical line about the horizontal axis.

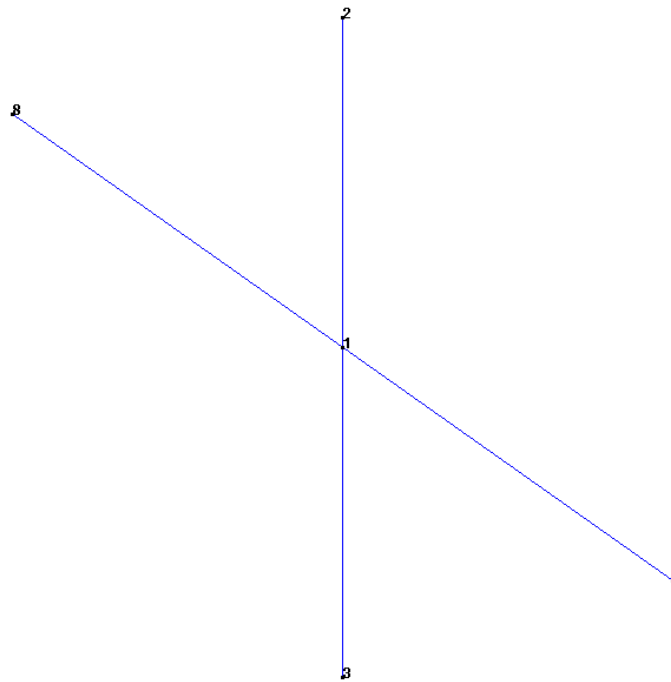


Figure 4.27 Vertical line and its rotated copy

9. Turn all point labels on and select `GEOMETRY/CREATE/ARC/THREE POINTS` to create an arc passing through 3 points. Note that the cursor symbol changes to a + sign, the default state of the cursor. This cursor symbol means that clicking anywhere on the screen will add that point to the arc. In our case, we want the arc to pass through the points we have already created. In order to make sure that GiD will add the existing point instead of a random coordinate, we press `<CTRL-A>` to toggle the cursor to a square symbol. Now the cursor will only pick existing points.
10. Click on points 8 and 2, and note that a floating arc section is created, as shown in [Figure 4.28](#).

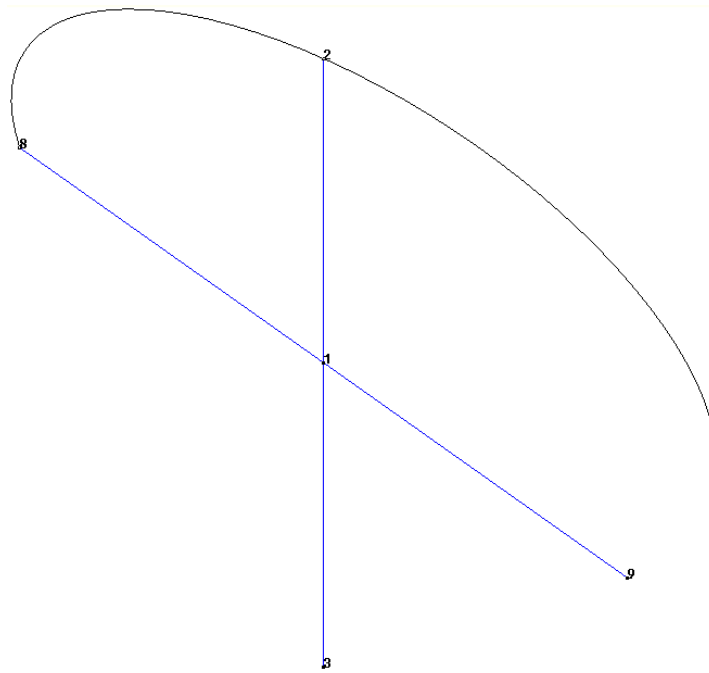


Figure 4.28 *Floating arc section created after clicking on two points*

11. Click on point 9 to complete the arc. To create a second arc, click on point 9, then on point 3, then on point 8. Press <Esc> to exit the arc-creation tool (Figure 4.29).

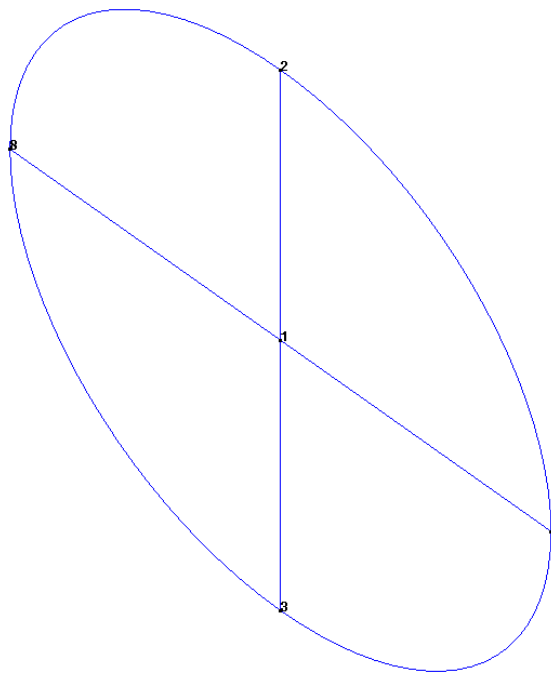


Figure 4.29 *Two completed arcs*

12. Use `VIEW/LABEL/OFF` to turn all point labels off. Then use `VIEW/LABEL/ALL IN`, and select the picture of the line (the second item from the top), as shown in [Figure 4.30](#).

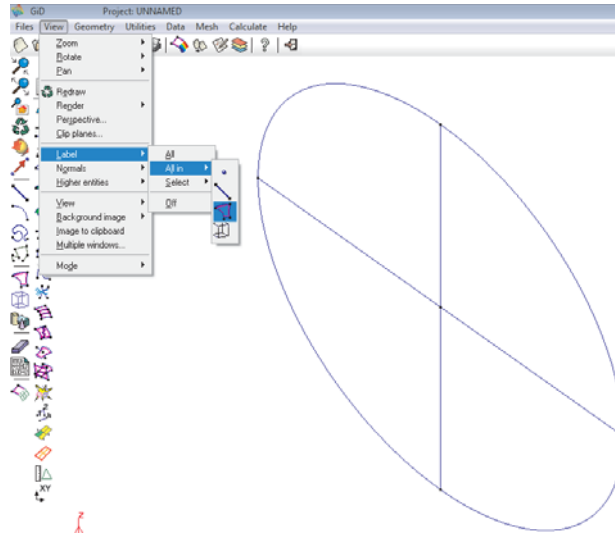


Figure 4.30 *Turning the line labels on*

Note that the upper arc is labeled as line 3, the lower arc is labeled as line 2, and the vertical and horizontal lines are labeled as 1 and 2.

13. We would like to cut each arc into 2 arcs, using the vertical line. To do so, first turn all labels off, using `VIEW/LABEL/OFF`.
14. Turn all point labels on, using `VIEW/LABEL/ALL IN`, selecting the first item of the drop-down menu (i.e., points).
15. We are going to cut the upper arc in half by intersecting it with the vertical line. To complete the intersection operation, use the Geometry Create Intersect Lines icon on the left side of the screen, click first on the vertical line, and then click on the upper arc (line 3). At the bottom-left of our screen, we see a message indicating that the two lines are now 3 lines, and that the program is ready for the next intersection.
16. First click on the vertical line, then on the lower arc, to cut it. Press `<Esc>` to exit the Line-line intersection tool.
17. Use `GEOMETRY/DELETE/LINE`, then click on the vertical and horizontal lines and press `<Esc>` to complete the deletion of the 2 construction lines.
18. Turn all labels off, then turn all line labels on, to label the 4 arcs ([Figure 4.31](#)).

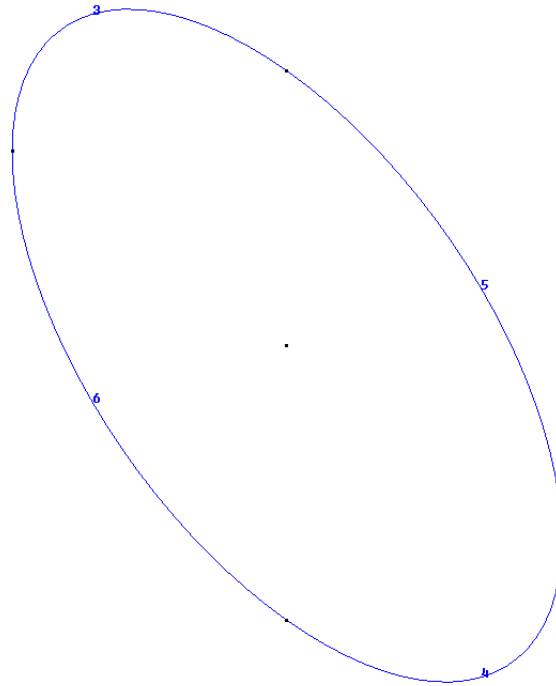


Figure 4.31 Four arcs defining the inlet of the elbow

4.2.1.3 Creating the Inlet Face and the Horizontal Section of the Elbow

1. Turn all labels off, and turn only line labels on.
2. Use VIEW/ROTATE/ISOMETRIC to adopt an isometric perspective view.
3. Using VIEW/ZOOM/FRAME, fit the model to our window so we can see everything (Figure 4.31).
4. To create the surface representing the inlet of the elbow, use GEOMETRY/CREATE/NURBS SURFACE/BY CONTOUR and left-click somewhere outside and close to the upper-left corner of the 4 arcs. While holding down the left mouse button, drag the cursor diagonally until the stretching rectangle contains the 4 arcs, and release the left mouse button to select the 4 arc sections. Press <Esc> to exit the surface-creation tool. Note that the resulting surface is represented by a pink oval.
5. Use VIEW/LABEL/ALL IN. In the drop-down menu, select the surface (third item from the top) in order to label the newly created surface representing the inlet (Figure 4.32). Note that this surface is labeled as surface number 1.

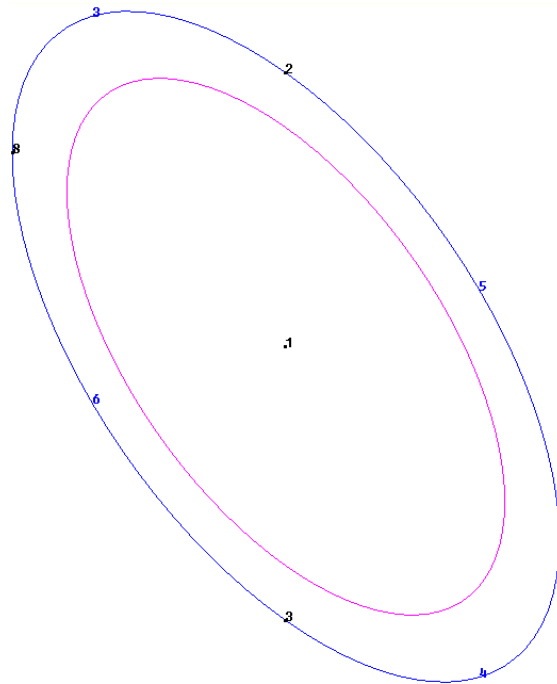


Figure 4.32 In pink, surface representing the inlet of the elbow

6. Turn all labels off, and then turn all point labels on. We are going to create the horizontal part of the elbow by extruding the inlet. We will translate and copy the inlet from point 1 to point 4. In doing so, we extrude the inlet to create the volume.
7. Adopt a perspective point-of-view to clearly see the surface of the inlet, as well as points 1 and 4.
8. Use UTILITIES/COPY to open the *Copy* dialog box. Select Surfaces in the Entities type field, and select Translation in the Transformation field.
9. Click on the button in the First point field, then click on point 1, which is located at the center of the inlet surface. Click on the button in the Second point field, and then click on point 4. Note that after each operation, the coordinates of that point appear in the corresponding window in the dialog box.
10. Select Volumes in the Do extrude field, and then click (Figure 4.33).

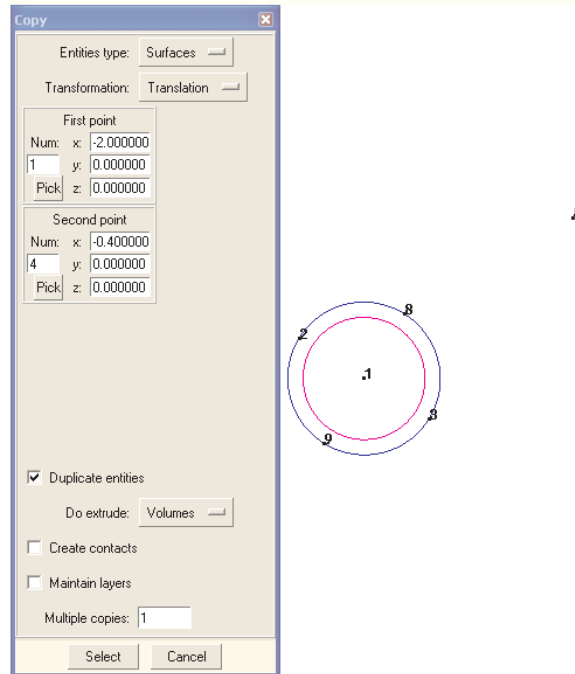


Figure 4.33 Copy dialog box showing the parameters used for copying and extruding the inlet surface

Our cursor symbol changes to a square, indicating that we must now select the surface(s) that will be extruded.

11. Click on the pink oval surface (note that it turns red upon selection), and press <Esc> to end the selection of surfaces and complete the extrusion. Close the *Copy* dialog box. Label all entities to see the 4 newly formed surfaces (shown in pink) and the volume (shown in light blue and labeled as volume 1) (Figure 4.34). Use FILE/SAVE AS to save our partially completed model as “Elbow.gid.”

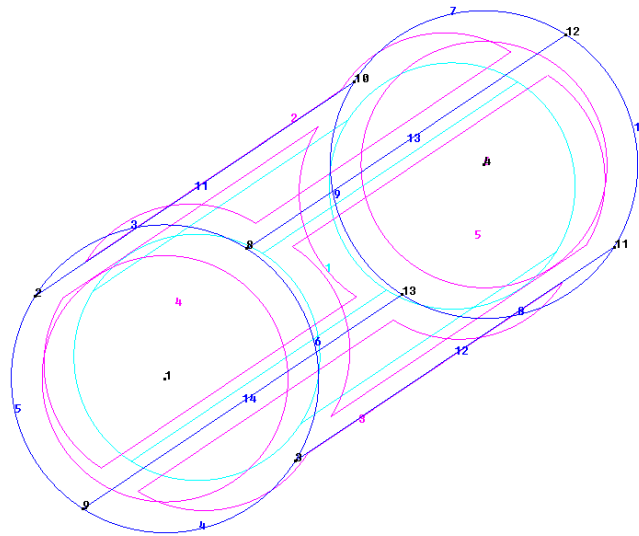


Figure 4.34 Volume and surfaces resulting from the extrusion of the inlet

4.2.1.4 Creating the Curved Section of the Elbow

1. Turn all labels on. Adopt an isometric perspective view and zoom in to clearly see the circular surface opposite the inlet surface (in other words, the outlet of the horizontal section), as well as points 4, 5 and 6.

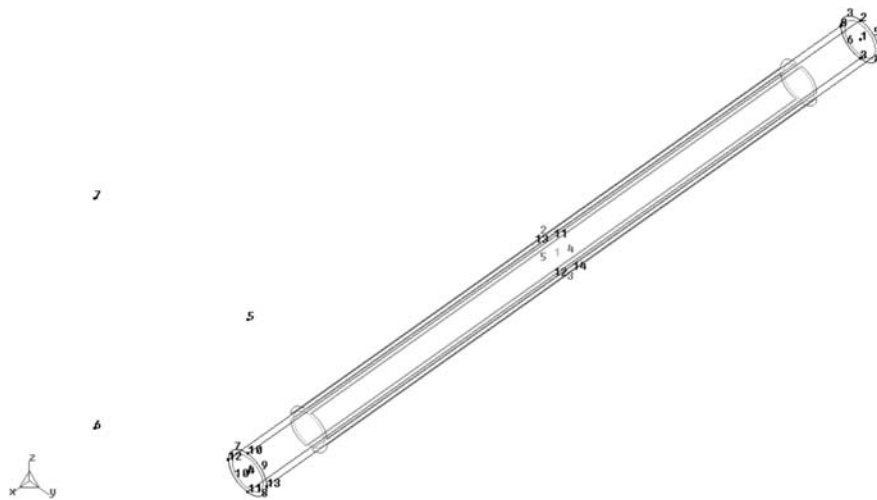


Figure 4.35 Perspective view of the horizontal section

We are going to extrude and rotate the surface of the horizontal section opposite the inlet section in order to create the curved volume. The axis of the rotation is along the y-direction and passes through point 5.

2. Use UTILITIES/COPY to open the *Copy* dialog box. Select Surfaces in the Entities type field and select Rotation in the Transformation field.
3. Enter -90 in the Angle (of rotation) field. To define the axis of rotation, click in the First point field, then click on point 5. Click in the Second point field, and then click on point 5. Note that after each operation, the coordinates of that point appear in the corresponding window in the dialog box.
4. At this stage, the axis of rotation is a vector of length 0. Click in the y-field of the Second point and replace 0 with 1 to specify an axis along the y-direction.
5. Select Volumes in the Do extrude field, and then click .
6. Click on the circular surface opposite the inlet surface of the horizontal volume. This surface is labeled 6 (we may need to turn off point labels to see it clearly), and results from the translation of the inlet surface from point 1 to point 4. Then click to complete the circular extrusion (Figure 4.36).

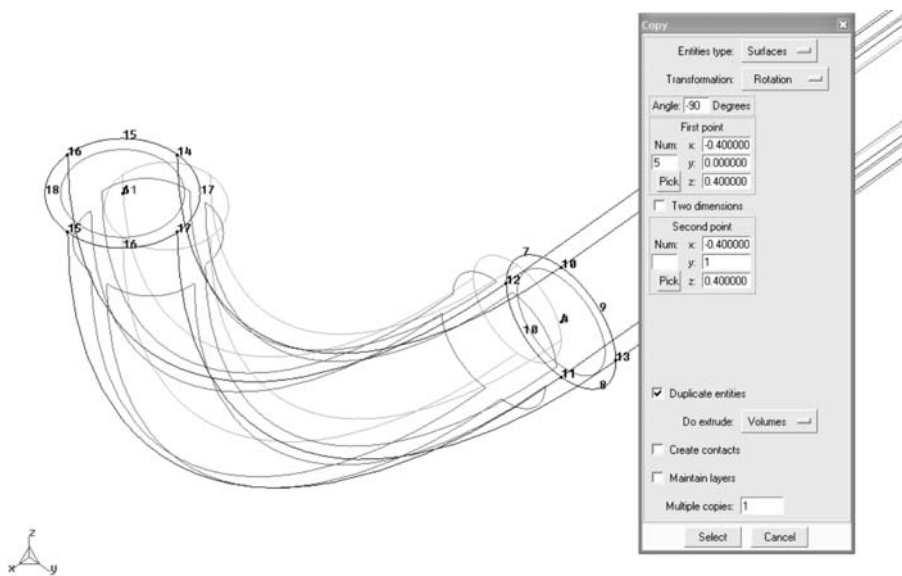


Figure 4.36 *Creating the volume of the curved section*

4.2.1.5 Creating the Vertical Section of the Elbow

1. Turn all labels on. Adopt an isometric perspective view and zoom in to see the outlet of the curved section, as well as points 6 and 7 (Figure 4.37).

7

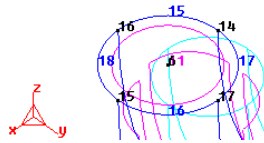


Figure 4.37 Outlet of the curved section of the elbow

2. Use UTILITIES/COPY to open the *Copy* dialog box. Select Surfaces in the Entities type field, and select Translation in the Transformation field.
3. Click in the First point field, then click on point 6, located at the center of the circular surface representing the outlet of the curved section. Click in the Second point field, and then click on point 7.
4. Select Volumes in the Do extrude field, and then click . Our cursor symbol changes to a square, indicating that we must now select the surface(s) to be extruded.
5. Click on the outlet surface of the curved section, and press <Esc> to complete the extrusion (Figure 4.38).

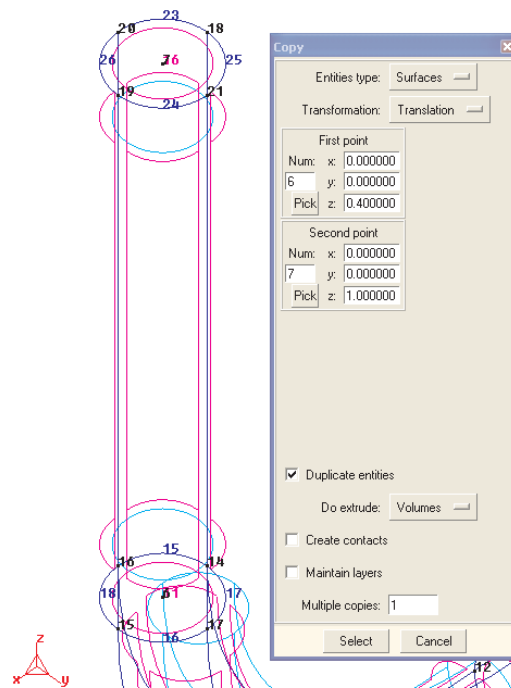


Figure 4.38 *Creation of the vertical volume*

6. Close the *Copy* dialog box. Label all entities to see the 4 newly formed surfaces (shown in pink) and the volume (shown in light blue and labeled as volume 3). Use `FILE/SAVE AS` to save our model as “Elbow.gid.”

4.2.2 Part 2: Fluid Flow Model Setup

In this section, we will define the CFD problem, including boundary and initial conditions, and additional parameters needed for the CFD part of the calculation.

4.2.2.1 Material Definition

1. If our model is not open, start GiD. Using `FILE/OPEN`, read in the “Elbow.gid” project folder. Use `DATA/PROBLEM TYPE`, and select `CFD` from the drop-down menu to specify that we are setting up a CFD problem. Use `VIEW/ROTATE/ISOMETRIC` followed by `VIEW/ZOOM/IN` to see all of the details.
2. Use `VIEW/LABEL/OFF` followed by `VIEW/LABEL/ALL IN`, and select the 4th entry (the cube in the drop-down menu) to label all volumes.
3. Select `DATA/MATERIALS` to open the *Materials* dialog box. Select `AIR` as the material in the uppermost field of the dialog box, click `ASSIGN` and select `Volumes` in the drop-down menu. The cursor changes to a square symbol, indicating that we must now select a volume to which the material “air” will be assigned.
4. Click on all 3 volumes (drawn in light blue), or click on the upper-left corner, hold the left mouse button down and drag the cursor to the lower-right corner in order to select the volumes to which the air material is to be assigned. Press `<Esc>` or click `FINISH` to complete the material assignment.
5. To verify that the correct material has been assigned to the correct volume, while the *Materials* dialog box is still open, click `DRAW` and select `All materials` to see the materials of the model, as shown in [Figure 4.39](#).

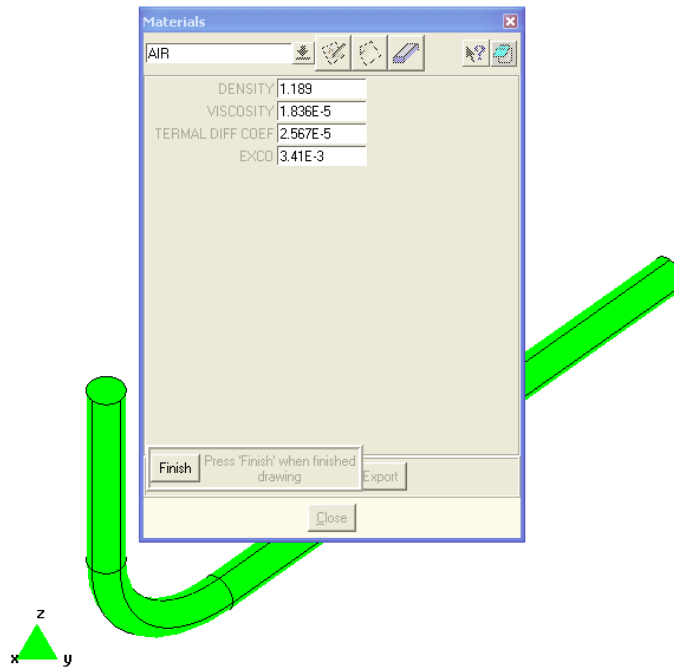


Figure 4.39 A representation of each volume and its material assignment in the model

6. Click

4.2.2.2 Problem Data

1. Use DATA/PROBLEM DATA to open the *Problem Data* dialog box. Select the General tab.
2. Enter Elbow in the TITLE field, select EXPLICIT in the Type of problem field, select NO in the Turb (Turbulence) model field, enter 1 in the Sweep field, enter 0.2 in the Relax factor P field, and enter 0.0 in the Scheme(0.0-1.0) field. Click

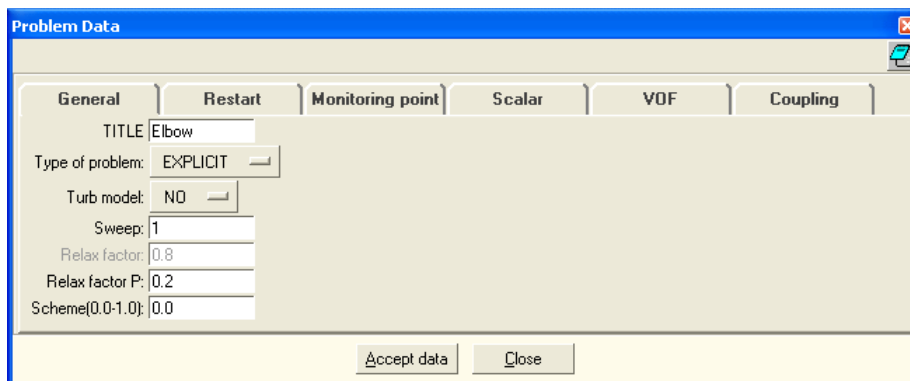


Figure 4.40 The General tab of the Problem Data dialog box

The fields in the Restart, Monitoring point, Scalar and VOF tabs should be left at their default values, as shown in [Figure 4.41](#).

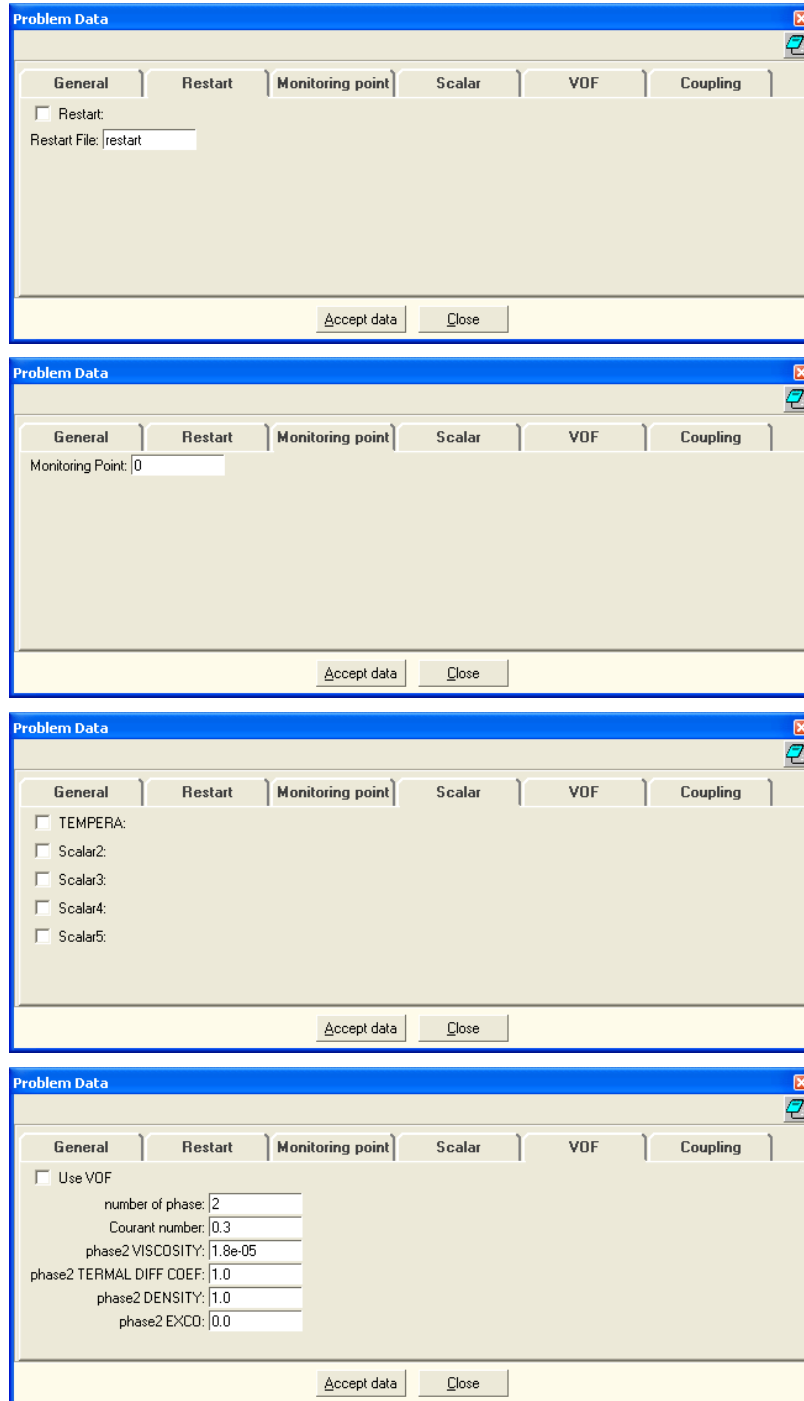


Figure 4.41 The Restart, Monitoring point, Scalar and VOF tabs

3. In the Coupling tab, check the Use Coupling field. For Coupling Type, select the third entry (i.e., CFD [unsteady] <-> FINAS [dynamic]).
4. Clicking on any of the Time Table buttons opens 3 fields. Enter 0 for start, 1 for end, and 0.001 for step (Figure 4.42). Click , followed by , to close the *Problem Data* dialog box.

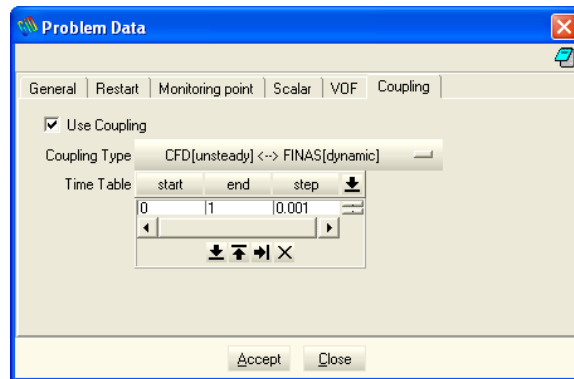


Figure 4.42 *The Problem Data dialog box showing the coupling type and the timestep specification*

4.2.2.3 Interval Data

1. Use `DATA/INTERVAL DATA` to open the *Interval Data* dialog box. In the Control tab, enter 50 in the Number of Steps field, 0.02 in the Time step increment field and 1 in the Step freq. post process field. This will create an output file every 0.02 seconds during the calculation.
2. The fields in the Gravity and Criterion temp tabs should be left at their default values (Figure 4.43).

NOTE: “Start” and “end” in the Coupling tab of the *Problem Data* dialog represent, respectively, the start and end time of coupling between *PFC^{3D}* and CCFD. “Step” represents the timestep between each data exchange. If the coupling between the fluid flow and particles is strong (i.e., if the flow strongly affects particle velocities and distribution, or if particles strongly affect the flow), “step” should remain small.

If “start” is greater than 0, the CCFD calculation will start at time 0 and proceed until coupling begins at time “start.” “Start” should be less than “end,” and “step” should be a whole fraction of (“end” – “step”).

“Number of Steps” in the Control tab of the *Interval Data* dialog represents the number of fluid flow calculation steps, and “Time step increment” represents the duration of each computational step. If the fluid flow computation is explicit and the Time step increment is larger than the one dictated by the CFL criterion, CCFD will perform several inner explicit calculation steps in the course of a Time step increment. For practical purposes, you could choose to complete the CFD calculation in

1 large step. “Step freq. post process” represents the frequency of result output for post processing. By setting Step freq. post process equal to 1, you can effectively control the number of outputs independently using Number of Steps. You should then select an appropriate Time step increment so that the CCFD simulation lasts a total of (Time step increment) \times (Number of Steps).

The CCFD calculation lasts for (Number of Steps) \times (Time step increment). If the CCFD calculation terminates before the end of the coupling (end), the *PFC^{3D}* calculation will wait for data from CCFD. On the other hand, if the coupling ends before the end of the CCFD calculation, the latter will continue without coupling, and will end normally.

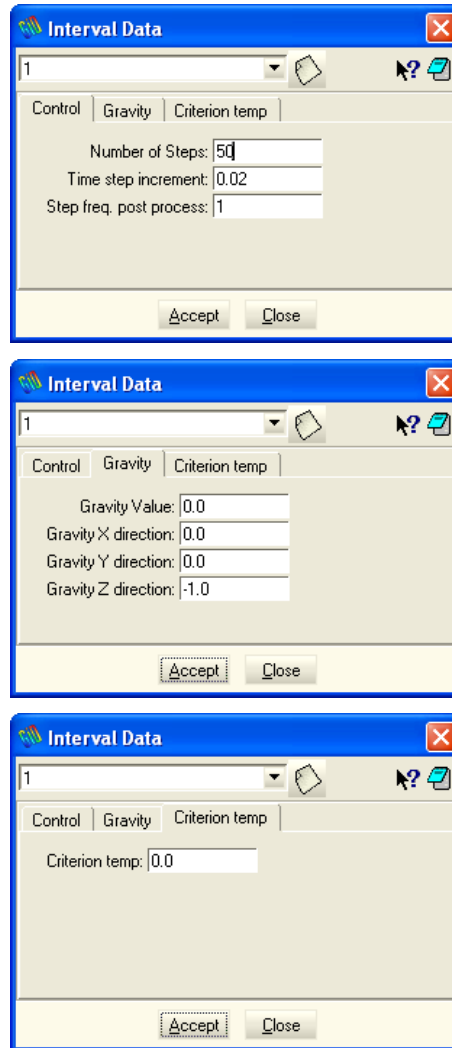


Figure 4.43 *The Control, Gravity and Criterion temp tabs of the Interval Data dialog box*

4.2.2.4 Initial Value Data

1. Use `DATA/CONDITION`, and select `Initial` to open the *Initial* dialog box. In this dialog box, we can specify the initial conditions of the CFD problem. In this problem, we choose to start with the trivial initial condition of 0 velocity everywhere.
2. Click `ASSIGN` to assign the default values appearing in the dialog box to all volumes. Click on volumes 1, 2 and 3, or select them by pressing the left mouse button, dragging the mouse and selecting them all. Note that the volume labels appear in light blue, and turn red when selected. Click `FINISH` to exit.
3. To check where initial conditions have been applied, click `DRAW` and select `Initial-Value` to see this initial value applied to the center of volume 1 (Figure 4.44). Click `FINISH` to close the dialog box.

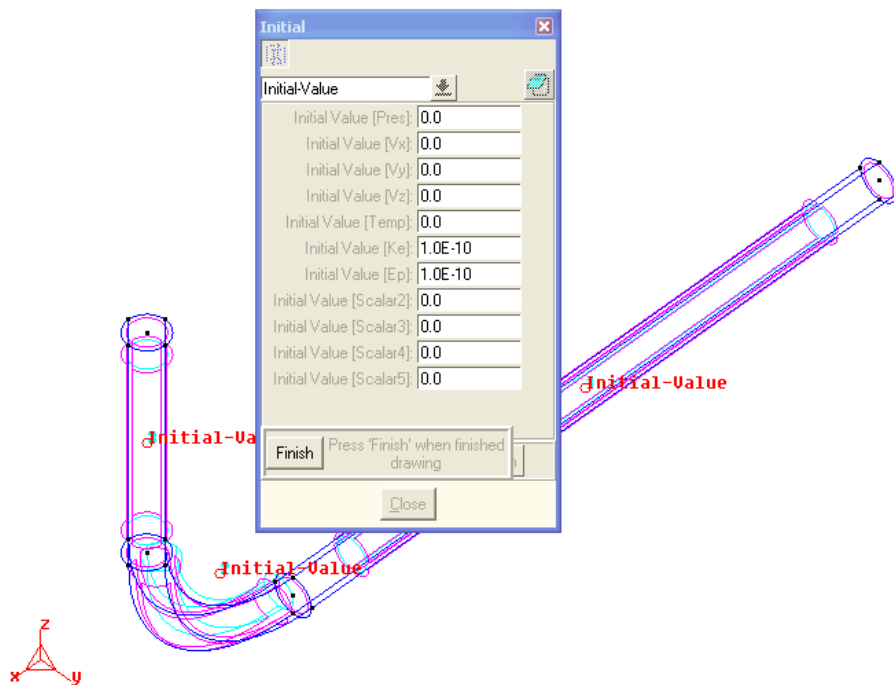


Figure 4.44 Applying initial conditions

4.2.2.5 Boundary Conditions

Air enters the elbow through the horizontal section with a velocity of 18 m/s, and exits from the vertical section.

1. To apply this boundary condition, first turn all labels off with `VIEW/LABELS/ALL OFF`, then turn the surface labels on using `VIEW/LABELS/ALL IN` and select the third item (surfaces) from the drop-down menu.
2. Use `DATA/CONDITION` and select Boundary in the drop-down menu to open the *Boundary* dialog box. In the drop-down menu, select `FIXED-VELOCITY` and enter 18 in the X-Value field, leaving the Y-Value and Z-Value fields at 0.
3. Click `ASSIGN` to select the surfaces to which this boundary condition will be applied.
4. Here, the 18 m/s fixed-velocity boundary condition is applied to the inlet surface. Click on the inlet surface, which is labeled as surface number 1. Press `<Esc>` or click `FINISH` to exit the surface-selection tool.
5. While the drop-down menu still indicates `FIXED-VELOCITY`, enter 0 in the X-Value field, leaving 0 in the Y-Value and Z-Value fields.
6. Click `ASSIGN` to select the surfaces to which the zero velocity (no-slip) boundary condition will be applied. In this example, the fixed-velocity boundary condition is applied to all wall surfaces. Remember that the inlet, curved volume and vertical volume of the elbow each has 4 wall surfaces, for a total of 12 surfaces. These surfaces are 2, 3, 4 and 5 for the inlet volume, 7, 8, 9 and 10 for the curved volume, and 12, 13, 14 and 15 for the vertical volume.
7. We must select the wall surfaces by clicking on each one individually. But make sure that we don't select the inlet (surface 1), outlet (surface 16) or either of the two internal surfaces, 6 and 11 ([Figure 4.45](#)).

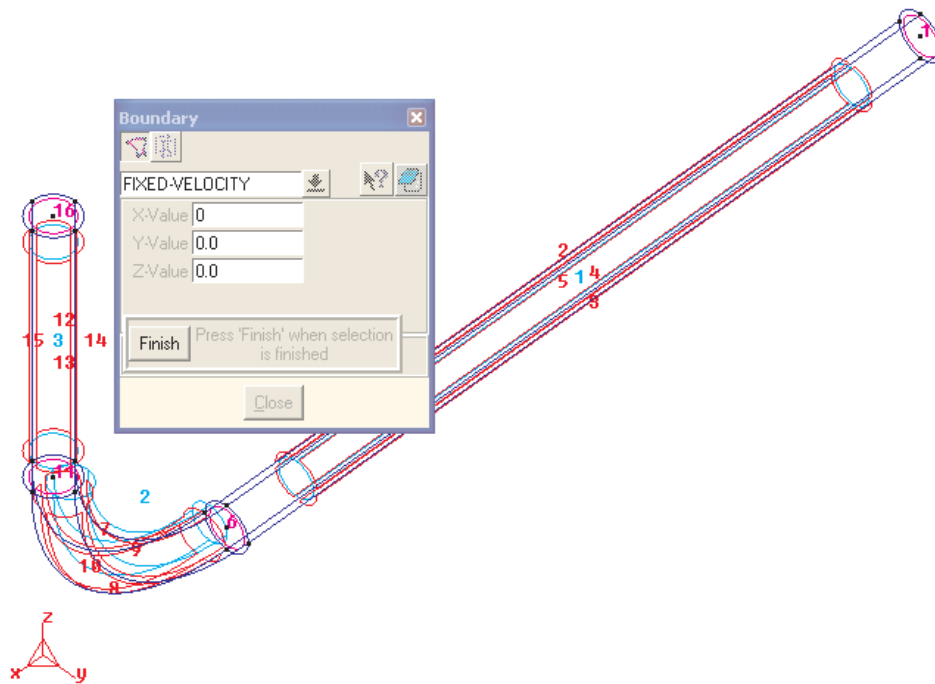


Figure 4.45 Applying the zero velocity wall boundary conditions

If we inadvertently select the wrong surface, either click on that item again to remove it from the selection list, or undo the last operation using the following procedure. Press <Esc> to exit the surface-selection mode. Use UTILITIES/UNDO to open the *Undo* dialog box, and highlight the range of instructions that we want to undo (see Section 4.2.3.1 for further details).

8. Press <Esc> or click to exit the surface-selection tool.
9. In the drop-down menu at the top of the *Boundary* dialog box, select FIXED-PRESSURE, and set its value to 0. Click and click on the outlet surface of the elbow (surface number 16, located at the top of the vertical section), then press <Esc> or click to exit the surface-selection tool.
10. To view all boundary and initial conditions, click and select All conditions (Figure 4.46). Note that a red arrow at the inlet surface illustrates the 18 m/s inflow velocity.

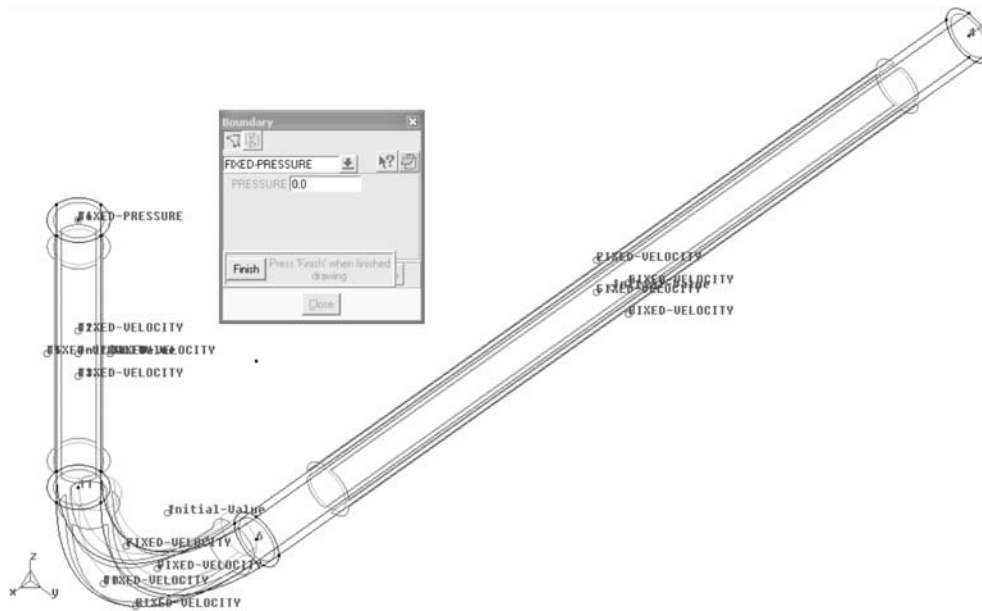


Figure 4.46 Summary of all initial and boundary conditions

11. Click **FINISH** to leave Draw mode.
12. To check the boundary conditions, while the *Boundary* dialog box is still open, rotate the model so that we can see both the inlet and the outlet. Choose the **FIXED-PRESSURE** drop-down menu item, click **DRAW** and select color (Figure 4.47).

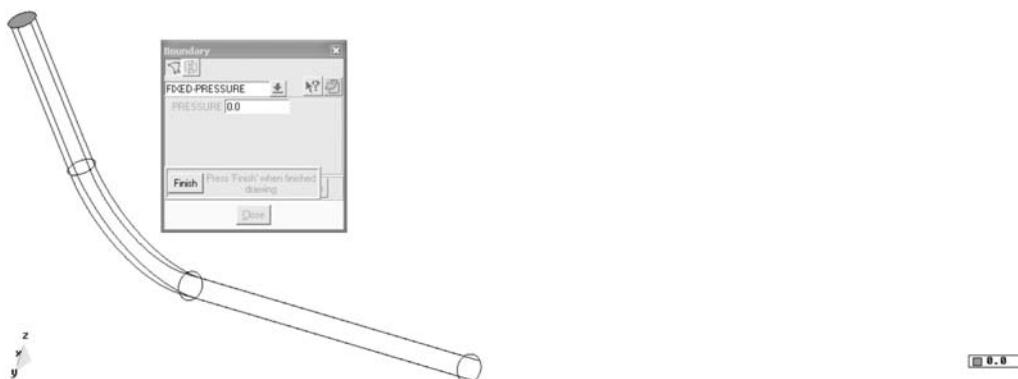


Figure 4.47 Fixed-pressure boundary condition specified at the outlet

13. We can do the same thing with the velocity boundary conditions. Click **FINISH** to leave Draw mode. Select **FIXED-VELOCITY** from the drop-down menu, click **DRAW** and select color (Figure 4.48). Note that the inlet velocity is represented in green, and the 0 velocity (no-slip) boundary condition in yellow.

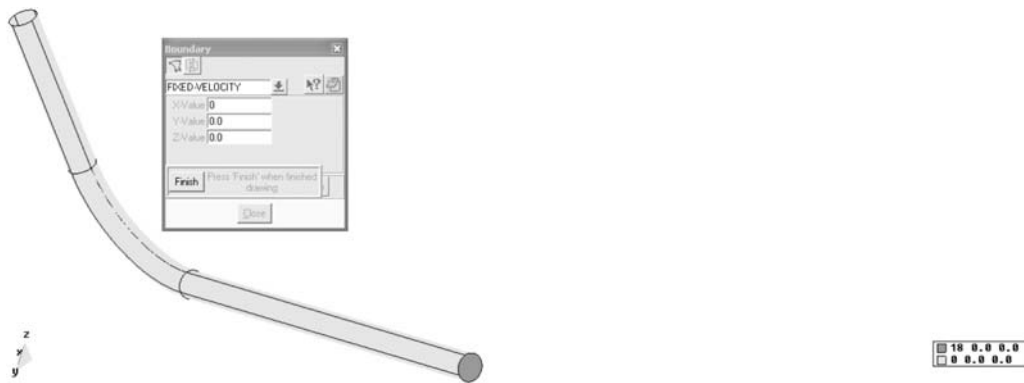


Figure 4.48 Fixed-velocity boundary conditions: 0 in yellow; inlet velocity in green

14. Click to close the dialog box.
15. Save our model as “Elbow.gid.”

4.2.2.6 Mesh Generation

In CCFD, we can use both hexahedral and tetrahedral meshes. In this example, we will create a structured hexahedral mesh.

1. Continue on from the previous section or read in the “Elbow.gid” project file. Use MESH/ELEMENT TYPE/HEXAHEDRA to open an *Information window* dialog box and select the three volumes that are to be meshed with hexahedral elements.
2. Click , and select the 3 volumes constituting the elbow. We can do this by selecting all the volumes at once by pressing the left mouse button and dragging it, or by selecting each volume individually. Note that after the selection of each volume, the message “Added volumes to the selection . . .” appears at the bottom-left of the window.
3. Use VIEW/ROTATE/ISOMETRIC and VIEW/ZOOM/IN to adopt a clear view of our model, enabling us to pick any line in the model. Turn all labels off, and remember that points are represented as black dots, lines are in blue, surfaces are in pink and volumes are in light blue.

A structured mesh is a collection of cube-like volumes, deformed to match a certain shape. This type of mesh is called structured and has a natural orientation along the bottom-top, East-West and South-North directions.

4. Use MESH/STRUCTURED/VOLUMES. The cursor symbol changes and we can read at the bottom-left of the screen that we must now select volumes bound by six faces. Select the volumes by clicking on each individually, or by selecting all 3 at once using the left mouse button and dragging. Then press <Esc>. Exiting volume selection immediately opens an Enter value window.

5. Starting with the inlet (horizontal) volume (volume 1), we only need three numbers to mesh this cylinder as a stretched cube. This dialog box enables us to specify the number of subdivisions in the bottom-top, East-West and South-North directions. Enter 5, then click , and select any of the 4 arcs forming the inlet surface (for instance, line 1). As soon as we click on this line, we will notice that the 3 other lines (the quarter arc at the inlet opposite the one we chose (line 2), and the two corresponding arcs on the surface opposite the inlet surface in volume 1 (lines 5 and 6)) turn red. This indicates that they have been selected automatically, and that the number of subdivisions has been set to 5. We will call the direction corresponding to these edges the East-West direction.
6. While we are still in the line-selection mode, select an arc on the inlet surface adjacent to the one we selected earlier (for instance, line 3), and set the number of subdivisions in this direction to 5 ([Figure 4.49](#)). We will call this direction South-North.

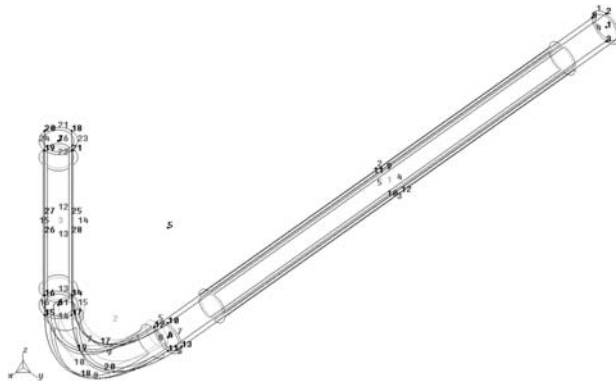


Figure 4.49 *Model after assignment of number of meshing subdivisions in the East-West and South-North directions. Note that the only remaining unassigned (blue) lines are in the bottom-top direction of volume 1.*

7. Press <Esc> to exit line-selection mode. The Enter value window reappears. Enter 40, click , and select any of the 4 remaining horizontal blue lines in volume 1 to assign the number of subdivisions along the bottom-top direction. These 4 lines run along the length of volume 1. Note that, as a result of selecting one line, the 3 remaining bottom-top lines also turn red.
8. Press <Esc> to exit line-selection mode. The Enter value window reappears.

So far, we have completed the subdivision assignment for volume 1. Volumes 2 and 3 are connected to volume 1. As a result, the same East-West and South-North subdivisions run through all 3 volumes, and hold for volumes 2 and 3. Therefore, all we have left to do is assign the bottom-top number of subdivisions for volumes 2 and 3.

9. Enter 20 in the Enter value window, click , and select any of the 4 lines running along the length of volume 2 (line 17, 18, 19 or 20). The other 3 will be automatically selected ([Figure 4.50](#)).

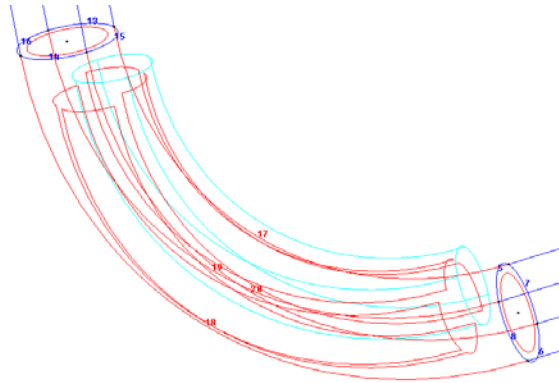


Figure 4.50 *Assigning the number of subdivisions along the curved volume*

10. Finally, to enter the number of subdivisions along the length of the vertical section, press <Esc>. The Enter value window reappears. Enter 20, click and select any of the 4 vertical lines running along the length of the vertical volume (line 17, 18, 19 or 20). The 3 other lines will be automatically selected (Figure 4.51). Press <Esc>, then click when the Enter value window returns.

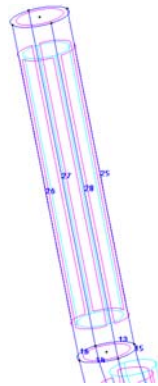


Figure 4.51 *Assigning the number of subdivisions along the vertical volume*

In summary, the elbow will be meshed with 5 elements transversely in each direction and along the flow, with 40 elements in the horizontal section, and 20 elements in the curved and vertical sections.

11. We are now ready to generate the mesh. Use MESH/GENERATE MESH to start the mesh-building. A window appears, reporting the progress of mesh generation, followed by an informational window, reporting that 2000 hexahedral elements and 2916 nodes were created. Click to see the resulting mesh (Figure 4.52).

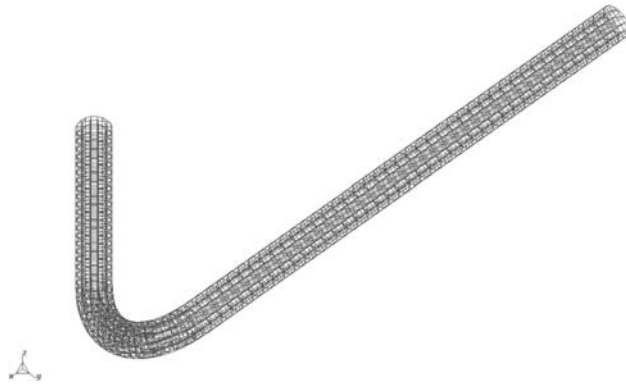


Figure 4.52 *Structured hexahedral mesh of the elbow with 5 subdivisions across in each direction and 80 subdivisions along the pipe*

12. Use `FILE/SAVE` to save the geometry (points, lines, surface and volume), all operating conditions and the mesh of our model in “Elbow.gid.”

One important final note: If at any time we decide to modify any of the boundary or initial conditions, we must regenerate the mesh by selecting `MESH/GENERATE MESH` again. This will maintain the associativity between the boundary/initial conditions and the mesh.

4.2.3 Part 3: Preparation of the Wall Geometry for PFC^{3D}

The PFC^{3D} portion of the computations requires a description of the walls of the domain. The CFD part of the calculation uses the mesh we created earlier. The PFC^{3D} portion of the calculation requires a description of the walls as a formatted STL file, described below.

4.2.3.1 Model Cleanup

We are going to delete all volumes and internal surfaces, as well as the inlet and outlet surfaces, because all PFC^{3D} needs is the description of the walls along the flow. Beyond this point, we do not save our project because we will need the unaltered “Elbow.gid” project to conduct the computations.

1. Continue on from the previous section or read in the “Elbow.gid” project file. If a mesh is currently displayed on our screen, select GEOMETRY/VIEW GEOMETRY to switch to the geometry view. Use GEOMETRY/DELETE/VOLUME, select all 3 volumes and press <Esc> to delete them.

We also need to delete 4 surfaces: the inlet and outlet surfaces, and the two surfaces located at the interface of volumes 2 and 1, and 2 and 3. The inlet surface is surface 1, and the outlet is surface 16. The internal surfaces are the surfaces at either end of the curved volume: 6 and 11 (Figure 4.53).

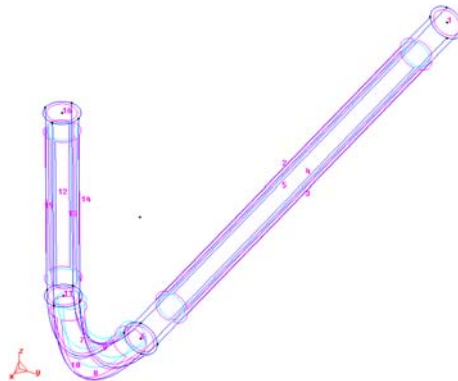


Figure 4.53 Surfaces 1, 6, 11 and 16 are clearly visible in the model. They need to be removed prior to the creation of the STL file.

2. Use GEOMETRY/DELETE/SURFACE and carefully click on the surface at each end of the curved section (6 and 11). Make sure that we do not inadvertently select any of the wall surfaces. VIEW/ZOOM/IN may be used to get closer to the details, if necessary.

If we inadvertently select the wrong surface, either click on that item again to remove it from the selection list, or undo the last operation using the following procedure. Press <Esc> to exit surface-selection mode. This will delete the surface. To un-delete it, proceed as follows. Assume that we deleted surface 11, and that we want to un-delete it.

Select `UTILITIES/UNDO` to open the *Undo* dialog box. Note that this box lists all of the operations performed in GiD since the last save operation.

Click on geometry delete surface in order to highlight everything including and following this statement (Figure 4.54). Then click `UNDO`. The state of our model returns to the state that existed prior to the delete operation, and surface 11 is restored.

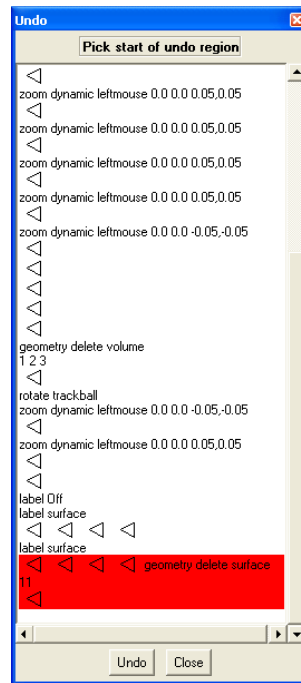


Figure 4.54 *The Undo dialog box*

3. Delete surfaces 6 and 11, located at either end of the curved volume. The remaining surfaces are all outer surfaces. All internal surfaces have been removed.

4.2.3.2 *Creating an STL File Describing the PFC^{3D} Wall Geometry*

We are going to build a triangular mesh of the outer surface of the volume.

1. Select `MESH/RESET MESH DATA` to reset the previous settings of the meshing function. Click `OK` to complete this operation.
2. Adopt an isometric perspective view of the model and zoom in so we can see everything.
3. Select `MESHING/UNSTRUCTURED/SIZES BY CHORDAL ERROR` to open the *Assign sizes by chordal error* dialog box. Chordal error, or offset, is the distance between the meshed surface and the original geometry.

4. Enter 0.001 in the Chordal Error field, set the Maximum meshing size to 1, and leave the Minimum meshing size field set at Automatic (Figure 4.55).

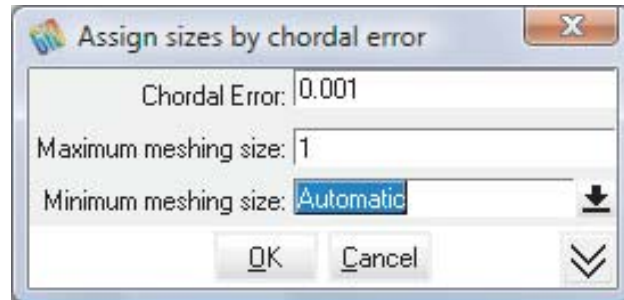


Figure 4.55 Parameters of size assignment by chordal error

5. Click to exit the dialog box. Select MESHING/GENERATE MESH. When the Enter value window opens, leave the default value and click .
6. A dialog box opens, reporting the total number of triangles and nodes created in the mesh. Click to display the mesh (Figure 4.56).

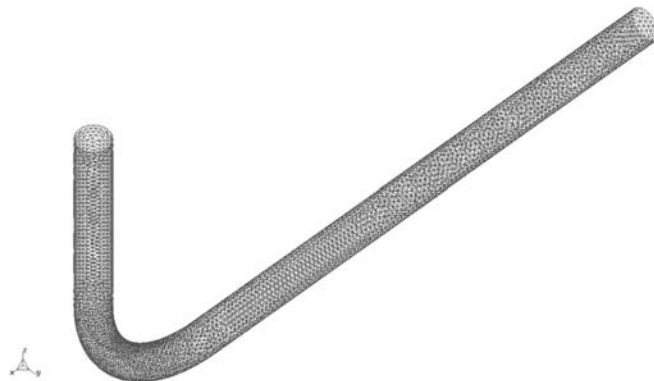


Figure 4.56 Surface mesh of the elbow that will be used for the definition of walls in PFC^{3D}

7. Select FILE/EXPORT/USING TEMPLATES/STL to open the *Output* dialog box. Navigate to the “Elbow.gid” directory, which is our current elbow GiD project directory, and enter “wall.stl” in the File name field, then click Save.
8. Do not save the project.

4.2.4 Part 4: PFC^{3D} Data File Preparation

The PFC^{3D} portion of this simulation requires a data file describing the location, size and properties of particles.

The PFC^{3D} data file provided with this tutorial (“pfc_Elbow.dat”) starts with a call to the **read-StlFile FISH** function, which reads the “wall.stl” file into PFC^{3D}.

4.2.4.1 Data File “pfc_Elbow.dat”

```

set logfile pfccfd.log
set log on ov

; read in pipe surface defined by .stl file.
call readstl.dat
set fn_stl = 'wall.stl'
readStlFile

; set wall properties
def inline_
  kn_w = 1e7
  ks_w = 1e7
end
inline_

wall prop kn kn_w ks ks_w fric 0.3

; In this example the entire surface of the pipe is described by the
; wall.stl file.
;
; Note that for faster performance general walls can be used
; for the straight sections of the pipe. Only the bent segment of pipe
; needs to be defined with an .stl file
; for example:
; wall type cylinder end1 -2 0 0 end2 -0.4 0 0 rad 0.05 0.05 id 1
; wall type cylinder end1 0 0 0.4 end2 0 0 1 rad 0.05 0.05 id 2
; wall kn kn_w ks ks_w fric 0.3 id 1
; wall kn kn_w ks ks_w fric 0.3 id 2

; generate initial ball distribution
def inline2
  ids = 1
  ide = 700
end
inline2

```

```
def fish_filter
; used in generate command to fill a cylindrical space with balls
  tr=fc_arg(0)
  tx=fc_arg(1)
  ty=fc_arg(2)
  tz=fc_arg(3)

  dist = sqrt(tz^2 + ty^2) + tr*1.5
  if dist < 0.05 then
    fish_filter = 0
  else
    fish_filter = 1
  end_if
end

damp default local 0.0
generate x -1.9 -.5 y -0.05 0.05 z -0.05 0.05 rad=0.002,0.003
  id=ids,ide filter fish_filter

; set ball properties
prop dens 2e3 range id ids ide
prop kn 1e7 ks 1e7 range id ids ide
prop fric 0.3 range id ids ide
prop xvel 18 range id ids ide

; configure ccfid and set fluid properties
config ccfid
ccfd set bouy off
ccfd set density 1.189
ccfd set viscosity 1.836e-5
ccfd set porosity poly
ccfd set coupling on
ccfd set interval 10

; define a plot
plot create Ballvel
plot add axes
plot set background white
plot add ccfidvel blue
plot add ball yellow
plot set cent -2.448e-1 0.0 1.449e-1
plot set mag 7.45
plot show
```

```
; catch balls that are flying off the top of the pipe
def remove_top
  j=j+1
  if j = 100 then
    j=0
    bp = ball_head
    loop while bp # null
      bpn=b_next(bp)
      if b_z(bp)> 1.0 then
        oo=b_delete(bp)
      end_if
      bp = bpn
    end_loop
  end_if
end
set fishcall 0 remove_top

ccfd connect

set plot avi size 640 480
movie avi_open file Elbow.avi
movie step 100 1 file Elbow.avi

pause
ccfd slave

movie avi_close file Elbow.avi
```

4.2.5 Part 5: Launching the Simulation

In this section, we will learn to launch *PFC^{3D}* with the CCFD option.

1. Start GiD and open the “Elbow.gid” project.
2. Using *CALCULATE/CALCULATE*, open the *FINAS-CFD* dialog box. Click , and after a short time the *Open* dialog box opens. This box lists all of the files with extension “.dat” that are located in the “Elbow.gid” project.
3. We must now read the *PFC^{3D}* data file needed for the simulation. Select “pfc_Elbow.dat” and click (Figure 4.57).

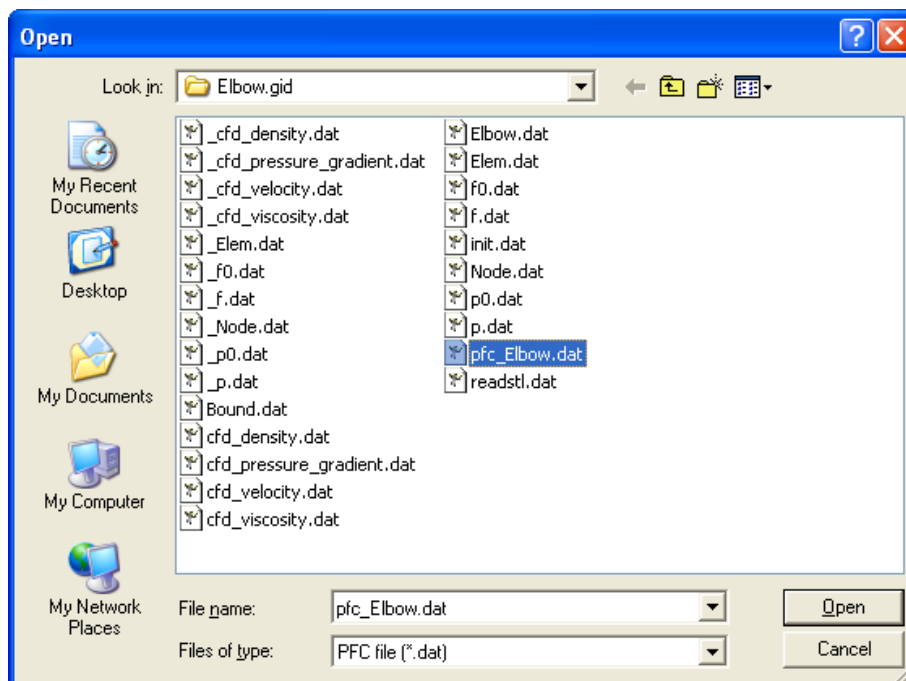


Figure 4.57 Reading the *PFC^{3D}* data file

4. After reading the *PFC^{3D}* data file, a proxy program which manages communications between the *PFC^{3D}* and the CCFD solvers (Figure 4.58) is initiated.

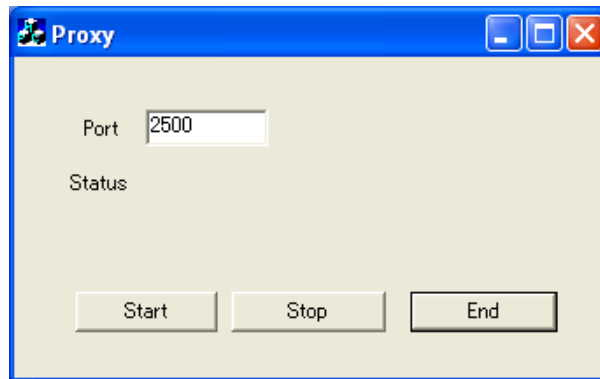


Figure 4.58 Proxy program about to start

5. Clicking launches the coupled PFC^{3D} -CCFD computation. A PFC^{3D} window opens and we notice that several commands are executed, until the word “pause” at the bottom of the PFC^{3D} data file is reached (Figure 4.59).

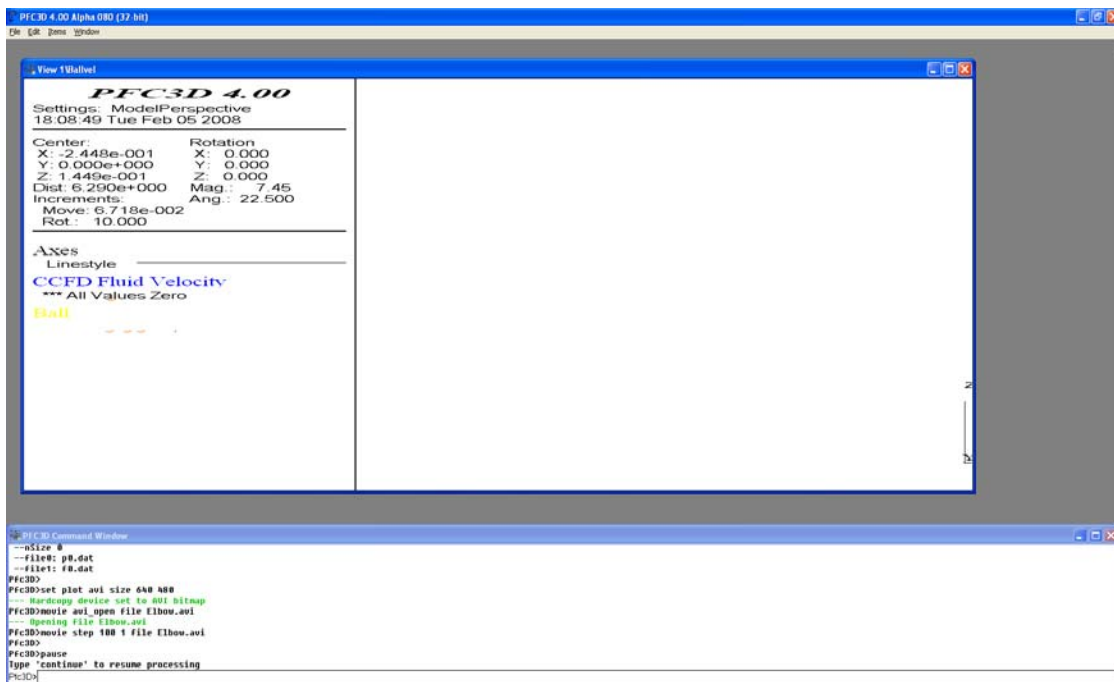


Figure 4.59 PFC^{3D} screen launched by the proxy program

6. Type “continue” in the command window of PFC^{3D} to resume the computation. At pre-set time intervals (as defined earlier in the Coupling tab of the *Problem Data* dialog box – Figure 4.42), PFC^{3D} and CCFD exchange information. Use the *Windows* taskbar to see the CCFD screen displaying the current state of the CFD computation (Figure 4.60).

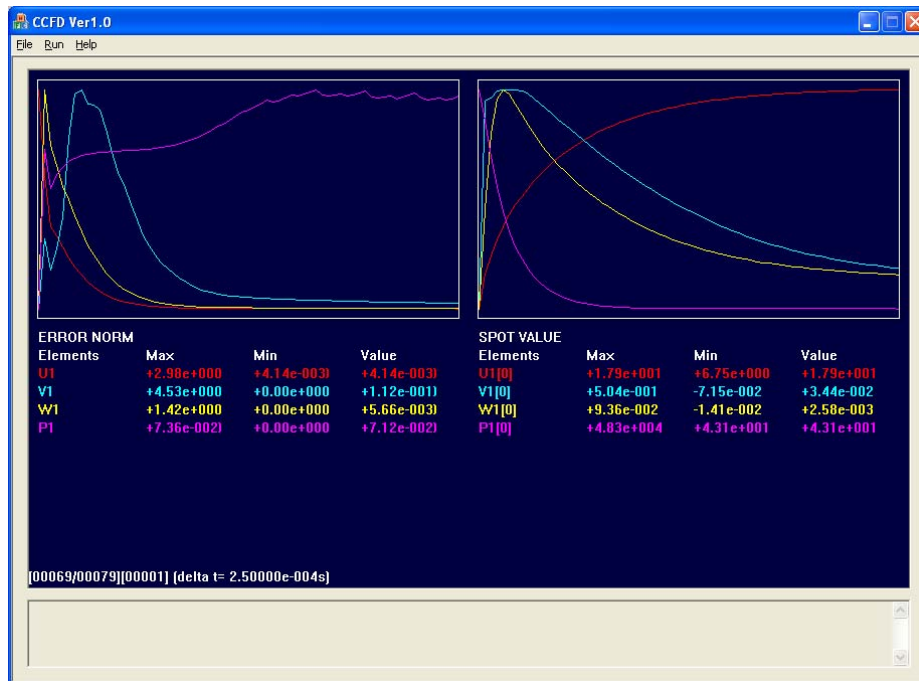


Figure 4.60 CCFD screen running in conjunction with PFC^{3D}

7. The computation will stop after 1 second of physical (problem) time, as specified in the Coupling tab of the *Problem Data* dialog box (Figure 4.42).

4.2.6 Part 6: Visualizing the Results

In this section, we will use GiD to visualize the results.

1. Click in the proxy server window.
2. Click on in the dialog that appears, to invoke post-process mode in GiD.
3. Use VIEW/ROTATE/PLANE XZ and VIEW/ZOOM/FRAME to get a view of the pipe.
4. Use VIEW RESULTS/CONTOUR FILL/PRESSURE to draw contours of the fluid pressure (Figure 4.61).

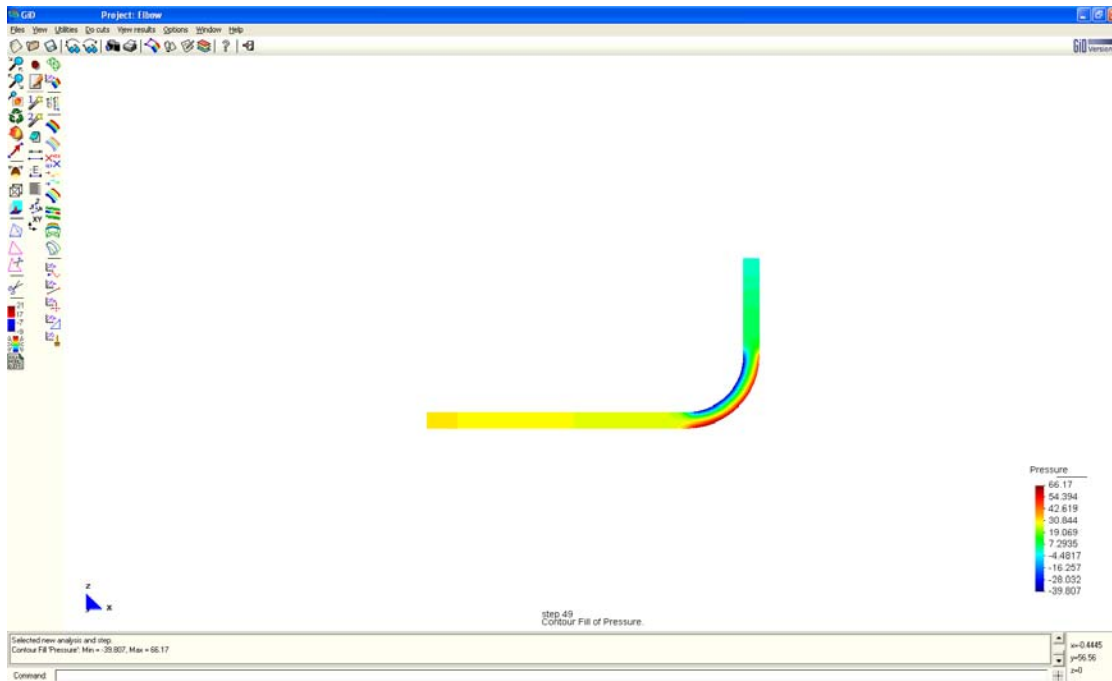


Figure 4.61 *Contours of fluid pressure using GiD in post-process mode*

5. By default, GiD displays the results from the last timestep of the calculation. To view results from earlier steps, use `VIEW RESULTS/DEFAULT ANALYSIS/STEP/FLUID FLOW` and select a step. The steps available for viewing are those specified in [Section 4.2.2.3](#).

The pressure plotted above is the fluid pressure on the surface of the pipe. Next we will generate a cut plane along the length of the pipe to see the results inside the pipe.

1. Use `VIEW RESULTS/NO RESULTS` to clear the plot. Use `VIEW/RENDER/SMOOTH` to get a shaded view of the geometry. Use `VIEW ROTATE/PLANE XY` to obtain a view looking down on the pipe from above. Use `VIEW/ZOOM/IN` to select the zoom tool. Click and drag from the top-left of the pipe to the bottom-right, and release the mouse button. The view should now look like [Figure 4.62](#).



Figure 4.62 *Top-down view of pipe geometry*

2. Select `DO CUTS/CUT PLANE/2 POINTS` to activate the cut-plane mode. Use `<CTRL-A>` to change the mouse pointer from the plus shape to the square shape. Click on the far-left side of the pipe, on the center line; click again on the far-right side of the pipe, also on the center line. Press `<Esc>` to exit cut-plane mode. To create a surface from the cut-plane, use `DO CUTS/CONVERT CUTS TO SURFACE SETS`.
3. Adopt a side view using `VIEW/ROTATE/PLANE XZ`. Open the *Select & Display Style* dialog with `WINDOW/VIEW STYLE`. (Figure 4.63).

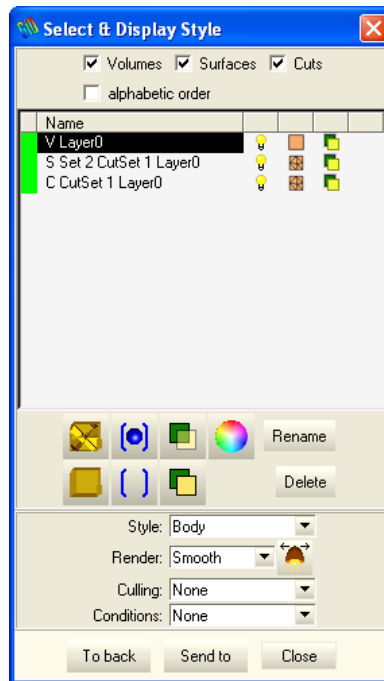


Figure 4.63 GiD *Select & Display Style* dialog

4. Click on the light-bulb icon to the right of V Layer0 to hide the volume set and reveal the surface we created with the cut plane. Close the *Select & Display Style* dialog.
5. Use VIEW RESULTS/DEFAULT ANALYSIS/STEP/FLUID FLOW/3 to select time interval 3. Draw contours of the drag force magnitude on the exposed surface using VIEW RESULTS/CONTOUR FILL/DRAG FORCE/MAG. (Figure 4.64).



Figure 4.64 *Contours of drag force magnitude shown on cut-plane surface*

6. To view an animation of this field, open the *Animate* dialog with `WINDOW/ANIMATE` and click play (Figure 4.65).

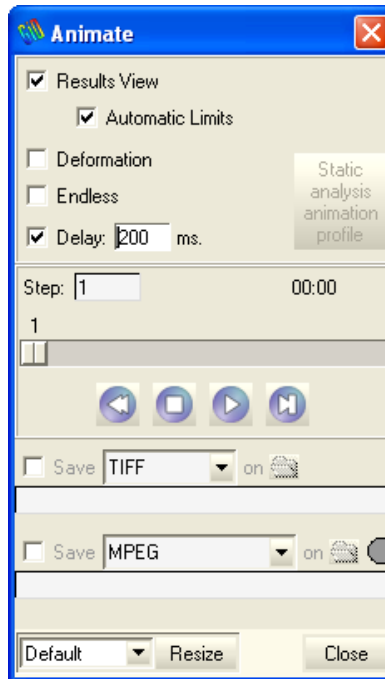


Figure 4.65 GiD Animate dialog

7. For more information about post-processing using GiD, access the GiD documentation via the Help menu.