

Quasi-One-Dimensional Flow for Use in Real-Time Facility Simulations

A Thesis Presented for
the Master of Science
Degree
The University of Tennessee, Knoxville

Brett Matthew Boylston
December 2011

Acknowledgments

I would like to thank the individuals and parties involved in helping me complete my thesis. None of this would have been possible without the work I perform at the Arnold Engineering Development Center. The opportunities I have had planted the seed for the work reported herein. A special thanks is in order for Dr. Greg Denny, who when the code became out of hand, suggested stepping back and taking a simpler approach. Lastly, I would like to thank my advisor, Dr. Trevor Moulden, for his patients and guidance throughout all of this. Thank you.

Abstract

Simulations have been, and continue to play, an important role at the Arnold Engineering Development Center as an aid in control system development and operator training. These models were just simple lumped-parameter methods. Since their initial inception, over ten years ago, little work has been done to increase the fidelity of the models. The processing power of the computer hardware used by the simulations has increased dramatically during this time and this left an opening for improvements to the models adopted in the simulation. To fill this void a quasi-one-dimensional control volume has been developed to run in real-time. The new control volume accounts for changes in area, transient effects, friction and other minor pressure losses, and localized heat transfer. All of which were previously unaccounted for. This new capability was compared against known analytical solutions and applied to an example flow system that demonstrates the new features. The result is a control volume that can be used in wind tunnel, or in other industrial process, simulations to provide a more realistic model.

Table of Contents

1.0	Introduction	1
1.1	The Existing Lumped-Parameter Method	3
1.2	Improvements to the System	8
2.0	Current Needs and Development	10
3.0	Finite Volume Approach	14
3.1	Conservation of Mass	15
3.2	Conservation of Momentum	16
3.3	Conservation of Energy	16
3.4	The Governing Equations for Quasi-One-Dimensional Flow	18
3.4.1	Heat Transfer	24
3.4.2	Friction and Other Minor Losses	28
3.5	Discretization	32
3.6	Heun's Method for Integration of the Time Derivatives	36
4.0	Implementation in Simulink®	39
4.1	How to Use the New Control Volume in Simulink®	40
4.2	Interfacing with the Quasi-One-Dimensional Control Volume	43
4.2.1	The Main Screen	45
4.2.2	The Geometry Screen	47
4.2.3	The Losses Screen	50
4.2.4	The Heat Transfer Screen	52
4.2.5	The Initial Conditions Screen	54

4.3	Organization of the Code	54
4.3.1	Initialization	56
4.3.2	Integration	59
4.3.3	Time-rate of Change of Mass Flow	59
4.3.4	Time-rate of Change of Mass and Energy	62
4.3.5	Time-rate of Change of Ducting Temperature	62
4.3.6	The Darcy Friction Factor	65
4.3.7	Heat Transfer Between the Fluid and Duct Wall	65
5.0	Results and Comparison with Theory	67
5.1	Flow with Area Change	68
5.2	Flow with Momentum Losses	81
5.3	Flow with Heat Transfer	87
5.4	Transient Results	94
5.5	Real-time Results	99
6.0	Practical Example of a Flow Network	108
6.1	Tee Junction Block	114
6.2	Analysis and Results of Example Network	116
6.3	Real-time Results of Example Network	125
6.4	Comparison Between the Quasi-One-Dimensional Code and the Lumped Parameter Method	125
7.0	Conclusion	131
	References	133

Vita	136
------------	-----

List of Tables

Table 6.1 – Cold Leg Inputs	111
Table 6.2 – Hot Leg Inputs	111
Table 6.3 – Warm Leg Inputs	112
Table 6.4 – Cold Leg Diameters	112
Table 6.5 – Hot Leg Diameters	113

List of Figures

Figure 1.1 – Rough Schematic of AEDC’s Four-foot Transonic Wind Tunnel	5
Figure 1.2 – AEDC’s Four-foot Transonic Wind Tunnel Modeled in Simulink®	5
Figure 1.3 – Overall System Diagram of a Simulation	7
Figure 2.1 – Representation of New Capabilities	12
Figure 3.1 – Control Volume about Point I	19
Figure 3.2 – Staggered Grid	21
Figure 3.3 – Control Volume Used to Develop the Partial Differential Equation for Momentum	21
Figure 3.4 – Pressure Forces Acting on the Control Volume	23
Figure 3.5 – The Modes of Heat Transfer for a Typical Piece of Ducting	25
Figure 3.6 – Friction Analysis on a Control Volume	30
Figure 3.7 – Discretization of the Quasi-One-Dimensional Control Volume	33
Figure 4.1 – Top Level of the New Quasi-One-Dimensional Control Volume in Simulink®	41
Figure 4.2 – Top Level of the New Quasi-One-Dimensional Control Volume with Valves Attached	41
Figure 4.3 – The Subsystem that Makes Up the Quasi-One-Dimensional Control Volume	42
Figure 4.4 – The Main Screen of the Quasi-One-Dimensional Control Volume Block	44
Figure 4.5 – Diagram Showing How the Grid Spacing is Defined	46
Figure 4.6 – The Geometry Screen of the Quasi-One-Dimensional Control Volume Block	48
Figure 4.7 – Example Input File to be Read in to Define the Diameter Distribution of the Duct	49
Figure 4.8 – The Losses Screen of the Quasi-One-Dimensional Control Volume Block	51
Figure 4.9 – The Heat Transfer Screen of the Quasi-One-Dimensional Control Volume Block	53
Figure 4.10 – The Initial Conditions Screen of the Quasi-One-Dimensional Control Volume Block	55

Figure 4.11 – Section of Ducting that Forms a Conical Frustum	57
Figure 4.12 – mdlOutputs Routine of OneDCV	60
Figure 4.13 – Calculation of the Derivatives at the Cell Faces	61
Figure 4.14 – Calculation of the Derivatives at the Cell Centers	63
Figure 4.15 – Calculation of the Time Rate of Change of the Duct Temperature	64
Figure 4.16 – Calculation of the Darcy Friction Factor	66
Figure 4.17 – Calculation of the Duct Heat Flux to the Flow	66
Figure 5.1 – Geometry Used for the Varying Area Comparison	69
Figure 5.2 – Simulation Diagram	71
Figure 5.3 – Mach Number Comparison with Isentropic Flow Theory for Incompressible Flow	72
Figure 5.4 – Static Pressure Comparison with Isentropic Flow Theory for Incompressible Flow	72
Figure 5.5 – Static Temperature Comparison with Isentropic Flow Theory for Incompressible Flow	73
Figure 5.6 – Total Pressure Variation for Incompressible Flow	75
Figure 5.7 – Total Temperature Variation for Incompressible Flow	75
Figure 5.8 – Static Density Variation for Incompressible Flow	76
Figure 5.9 – Mach Number Comparison with Isentropic Flow Theory for Compressible Flow	76
Figure 5.10 – Static Pressure Comparison with Isentropic Flow Theory for Compressible Flow	78
Figure 5.11 – Static Temperature Comparison with Isentropic Flow Theory for Compressible Flow	78
Figure 5.12 – Mach Number Comparison with Isentropic Flow Theory for Mach 0.9 Flow	79
Figure 5.13 – Mach Number Percent Error for Varying Grid Sizes	79
Figure 5.14 – Static Pressure Percent Error for Varying Grid Sizes	80
Figure 5.15 – Static Temperature Percent Error for Varying Grid Sizes	80
Figure 5.16 – Mach Number Comparison with Fanno Flow Theory	83

Figure 5.17 – Static Pressure Comparison with Fanno Flow Theory	83
Figure 5.18 – Static Temperature Comparison with Fanno Flow Theory	84
Figure 5.19 – Total Temperature Comparison with Fanno Flow Theory	84
Figure 5.20 – Computed Friction Values versus Empirical Data	86
Figure 5.21 – Static Pressure Profile with a Minor Loss Coefficient	86
Figure 5.22 – Mach Number Comparison with Rayleigh Flow Theory	89
Figure 5.23 – Static Pressure Comparison with Rayleigh Flow Theory	89
Figure 5.24 – Static Temperature Comparison with Rayleigh Flow Theory	90
Figure 5.25 – Total Pressure Comparison with Rayleigh Flow Theory	92
Figure 5.26 – Total Temperature Comparison with Rayleigh Flow Theory	92
Figure 5.27 – Duct Temperature as a Function of Time	93
Figure 5.28 – Duct Temperature as a Function of Time with Wind Effects	93
Figure 5.29 – Duct and Fluid Temperature as a Function of Time	95
Figure 5.30 – Steady-State Profiles of Duct and Fluid Temperature	95
Figure 5.31 – Pressure Wave at 10 meters Downstream	97
Figure 5.32 – Pressure Wave at 40 meters Downstream	97
Figure 5.33 – Task Execution Times for a Sample Time of 0.001 seconds with Varying Grid Sizes	100
Figure 5.34 – Task Execution Times for a Sample Time of 0.0005 seconds with Varying Grid Sizes	102
Figure 5.35 – Task Execution Times for a Sample Time of 0.0001 seconds with Varying Grid Sizes	102
Figure 5.36 – Mach Number Comparison with Isentropic Flow Theory for Lower Grid Densities	104
Figure 5.37 – Static Pressure Comparison with Isentropic Flow Theory for Lower Grid Densities	104
Figure 5.38 – Mach Number Comparison with Fanno Flow Theory for Lower Grid Densities	105
Figure 5.39 – Static Pressure Comparison with Fanno Flow Theory for Lower Grid Densities	105

Figure 5.40 – Mach Number Comparison with Rayleigh Flow Theory for Lower Grid Densities	107
Figure 5.41 – Static Pressure Comparison with Rayleigh Flow Theory for Lower Grid Densities	107
Figure 6.1 – Example Flow Network	109
Figure 6.2 – Simulink® Diagram of Example Ducting System	109
Figure 6.3 – Junction Header with Indices	115
Figure 6.4 – Velocity Profiles of the Cold Leg with Time	117
Figure 6.5 – Velocity Profiles of the Warm Leg with Time	117
Figure 6.6 – Velocity Profiles of the Hot Leg with Time	118
Figure 6.7 – Time History of the Mass Flow Exiting Each Leg	120
Figure 6.8 – Total Pressure Profiles of the Cold Leg with Time	120
Figure 6.9 – Total Pressure Profiles of the Warm Leg with Time	121
Figure 6.10 – Total Pressure Profiles of the Hot Leg with Time	121
Figure 6.11 – Total Temperature Profiles of the Cold Leg with Time	123
Figure 6.12 – Total Temperature Profiles of the Hot Leg with Time	123
Figure 6.13 – Total Temperature Profiles of the Warm Leg with Time	124
Figure 6.14 – Time History of the Total Temperature Exiting Each Leg	124
Figure 6.15 – Task Execution Time of the Sample Model	126
Figure 6.16 – Time History of the Exit Total Pressures Comparing the Lumped-Parameter Model	128
Figure 6.17 – Time History of the Exit Total Temperatures Comparing the Lumped-Parameter Model	128
Figure 6.18 – Time History of the Exit Mass Flows Comparing the Lumped-Parameter Model	129

List of Symbols/Nomenclature

4T	Four-foot transonic wind tunnel
A	Surface area normal to the flow
A	Cross sectional area
AEDC	Arnold Engineering Development Center
A_{in}	Duct internal wetted perimeter
A_{out}	Duct external wetted perimeter
API	Application programming interface
B	Generic fluid body
c	specific heat of ducting material
c	Speed of sound
c_p	Specific heat at constant pressure
CPU	Central processing unit
c_v	specific heat at constant volume
D	Domain occupied by a fluid body
D	Diameter
d_1	Inside duct diameter
d_2	Outside duct diameter
dA	Infinitesimal small change in area
dE	Infinitesimal small change in total energy
de	Infinitesimal small change in energy
D_H	Hydraulic diameter

dL	Infinitesimal small change in length
dN	Infinitesimal small change in an extensive fluid property
dp	Infinitesimal small change in pressure
dt	Infinitesimal small change in time
$d\mathbf{V}$	Infinitesimal small change in the velocity vector
dx	Infinitesimal small change in distance
E	Total energy of the fluid
e	Sum of the internal, kinetic and potential energies
$\mathbf{F}$	Body force acting on the fluid
f	Friction factor
F_f	Frictional force
F_m	Minor loss force
f_n	Natural frequency
g	Acceleration of gravity
h	Enthalpy
h_{in}	Internal duct heat transfer coefficient
HMI	Human machine interface
h_{minor_loss}	Minor head loss
h_{out}	External duct heat transfer coefficient
HSDI	High speed digital interface
I/O	Input/output
k	Thermal conductivity
K	Minor loss coefficient

L	Distance along the pipe span
M	Mass of fluid
M	Mach number
$\dot{m}$	Mass flow
N	Generic extensive fluid property
n	Constant used in the Dittus-Boelter correlation
Nu_D	Nusselt number
$\mathbf{p}$	local linear momentum of the fluid
p	Pressure
P	Internal duct wetted perimeter
PC	Personal computer
p_{junc}	Fluid pressure inside a junction
PLC	Programmable logic controller
PPC	PowerPC
Pr	Prandtl number
P_t	Fluid total pressure
Q	Net heat transfer rate
q	Heat flux
q_{cond}	Heat transfer due to conduction
$q_{ex_{conv}}$	Heat transfer due to external convection
q_{flow}	Heat transfer rate between the duct wall and fluid
$q_{in_{conv}}$	Heat transfer due to internal convection
q_{rad}	Heat transfer due to radiation

q_{sun}	Heat transfer from the sun
R	Specific gas constant
Ra	Rayleigh number
Re	Reynolds number
RPM	Rotations per minute
t	Time
T	Temperature
T_{amb}	Temperature of the ambient surroundings
T_{duct}	Temperature of the ducting wall
TET	Task execution time
T_f	Temperature of the fluid
T_t	Fluid total temperature
u	Internal energy of the fluid
$\mathbf{V}$	Velocity vector of the fluid
∇	Volume
V	Velocity of the fluid
W	Work done by a system
x	Horizontal distance
y	Dependent variable
z	Height
α	Aborptivity
$\Delta\tau$	Duct thickness

ε	Wall surface roughness
η	Generic intensive fluid property
γ	Ratio of specific heats
γ	Specific weight
ν	Specific volume
π	Mathematical constant
ρ	Density
ρ_t	Fluid total density
σ	Stefan-Boltzman constant
τ_w	Wall shear-stress

Subscripts/Superscripts

1	Upstream Property
2	Downstream Property
i	Index to denote cell face
I	Index to denote cell center
n	Time interval
R	Number of pipelines entering a junction
S	Number of pipelines leaving a junction

1. Introduction

Simulations of all kinds are playing an increasing role in today's world, especially in the aerospace and defense industries. Simulations have moved beyond the classical role of aircraft simulators for pilots. They are now being used in the design phase of many products or in the troubleshooting of existing hardware. The Arnold Engineering Development Center (AEDC) has been using models of its testing facilities to aid in control system development for a number of years. AEDC is one of the world's largest aerospace testing complexes that houses 58 wind tunnels, propulsion test cells, space chambers, and hyperballistic ranges. Each test facility is specific to providing the Air Force and Department of Defense with unique capabilities to test and evaluate hardware systems that are under development [1]. Many of these test facilities are equipped with highly sophisticated control systems for automation and operation. For a number of years, AEDC has been using models of these facilities to allow the control systems to be developed and optimized. Because of their complexity and the high cost of plant hardware and operation, many hours of validation are required before a new control system, or upgrades to an existing system, can be installed in the field for the first time. In addition, plant operators need training on a new system, or refresher training on an existing system, to hone their skills. Both validating and training in the field can be costly and has the potential to cause damage to the system. It is because of these circumstances that facility simulations have been used to check out a control system and for operator training. In the past operator training was done through documentation and direct on the job training [2]. But this process lacked methods to adequately train operators for emergency situations that would involve putting a facility in an unsafe mode of operation with the potential to cause catastrophic damage to plant equipment.

Facility upgrades always involve some time when a facility is taken down so that new hardware can be installed or software changes can be made. Clearly, testing cannot continue while a facility is taken offline. To minimize the impact on test schedules, upgrades were usually done between tests. However, properly checking out a new system could take weeks, so either a thorough check out could not be performed or test schedules were compromised. If errors were found during a system check out, the new system would be removed and the old one put back in place so that operations could continue while the error was corrected offline. Then during the next available opportunity the new system would be put back in place and check outs would resume. The process could repeat itself if multiple errors were found. Overall, this method of upgrading was inefficient and costly in terms of manpower, resources, and lost test time [3]. A new method, referred to as facility simulation, was needed to streamline this process. In 2000, AEDC began an aggressive approach to reduce the amount of time a test facility is taken off line to upgrade both control software and hardware.

To reduce the amount of time it takes to check out a system in the field, there needed to exist a method that allows for a much more comprehensive review offline. Control engineers have been using simplified models for years to test out basic control logic, but nothing has existed prior to the year 2000 to be able to test a control system against dynamic or transient data or to perform a full system check out with multiple subsystems. A program at AEDC was initiated to create real-time facility models which are able to interface, using a common input/output (I/O) path, with actual plant hardware [3]. The simulations receive the same commands that a control system might send to a valve, cooler, heater, compressor, etc. and it provides sensor output to the control system. This could consist of feedback of valve positions or temperatures and pressures found throughout the ducting. This provided a realistic environment for the control system to exercise all its logic before being fielded for the first time. Built alongside a simulation is a human machine interface (HMI) screen called a trainer screen. An HMI is the

typical method in which a user interacts with a machine; whether it is through inputs to manipulate the system or a way to allow the system to indicate effects of the user's manipulation [3]. HMI's are normally used to interface with a process control system that commands one or multiple pieces of equipment, such as: valves, compressors, and heat exchangers. However, parameters can be passed to and from the model through an Ethernet connection. This allows a user to start and stop a model, set initial conditions, or alter the system behavior in some controlled way. The user is also able to place the model in abnormal conditions to simulate certain types of failures. The four most common hardware items that may experience a variety of failure modes are valves, analog sensors such as pressure transducers and thermocouples, and discrete sensors like limit switches, for example. The simulated failure of individual valves, sensors, and switches is used to verify that the control system correctly detects and reacts to warning, danger, and fault conditions [4]. This minimizes the chance for errors to exist and it also provides a high confidence level in the control system. That is, the control system will not cause damage to plant hardware when it is activated for the first time. The simulations also provide a means for training facility-operators offline, an avenue that was previously unavailable. Simulations are also used to familiarize operators with new control screens that are built concurrently with a new control system. To model the physics necessary to provide realistic transient behavior for both checking out a control system and operator training, models written in Matlab/Simulink® are composed of two basic components: control volumes and flow devices [2].

1.1 The Existing Lumped-Parameter Method

Simulink® is an extension of Matlab® that allows engineers to interactively build models as block diagrams. These diagrams look very similar to those drawn with pencil and paper. Through the use of

customizable block libraries, models can be built to simulate complex linear or non-linear systems with either continuous or discrete time-steps. Simulink® also provides built-in solvers and integrators freeing the engineer to focus on the system being modeled [5,6]. The block diagrams that make up a facility model usually represent either a mass flow device or a control volume. Examples of flow devices are valves, compressors, and heat exchangers. These flow devices are connected to networks of ducting or test sections, which are commonly referred to as control volumes. The mass flow devices calculate mass and energy flow rates based on upstream and downstream pressures and provide the necessary boundary conditions for the control volumes. The control volumes use the rates calculated by the mass flow devices to obtain updated values of pressure and temperature using a lumped-parameter method [4]. This method assumes instantaneous mixing and that flow properties are uniform throughout a control volume [7].

Figure 1.1 provides a rough schematic of the four-foot transonic (4T) wind tunnel at AEDC used for aerodynamic testing. The main loop of the tunnel consists of the compressor labeled “A” and a primary cooler used for providing the air flow through the test section. Mach number is controlled by setting the compressor’s RPM and its inlet guide vanes or through suction from the plenum. When using suction to set Mach number, V5 is used for a coarse control and a smaller valve, V5a, is used for a fine control. Pressure in the test cell is controlled by multiple valves and depending upon the test conditions being required several different combinations are used. Typically valves V2, V2a, and V6 are held fixed, regulating air flow into the circuit, while valves V1 and V3 are used to control the pressure [8]. The flow diagram of 4T can be modeled in Simulink® using the lumped-parameter control volume method. Each control volume is defined by the boundaries that the mass flow devices naturally create. For example, a volume can be created between compressor A and the cooler and valve V1. The full model diagram is given in Figure 1.2. Each control volume keeps track of the amount of mass and

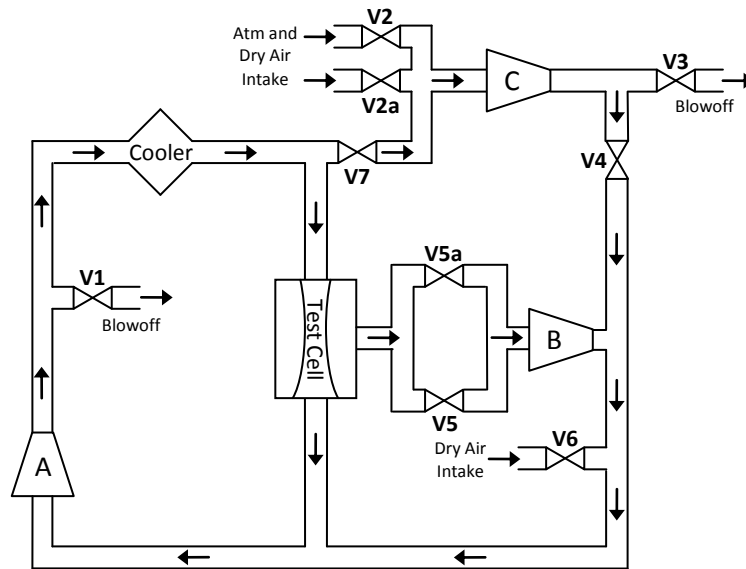


Figure 1.1 – Rough Schematic of AEDC's Four-foot Transonic Wind Tunnel
(see text for notation)

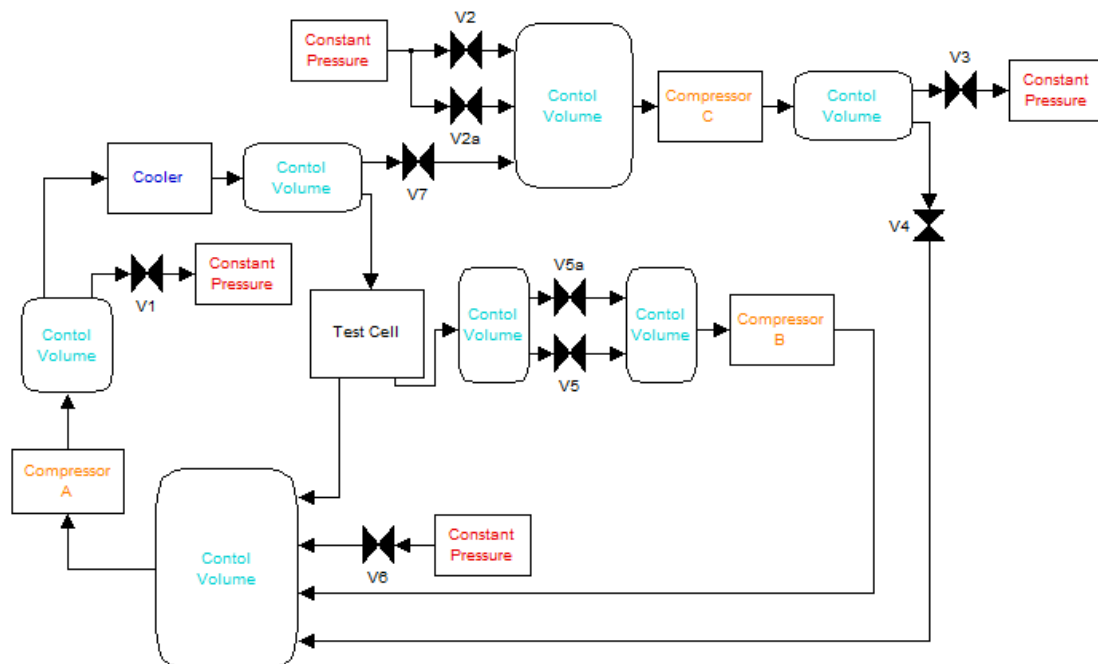


Figure 1.2 - AEDC's Four-foot Transonic Wind Tunnel Modeled in Simulink®
(Lumped-Parameter Model)

energy flowing in and out of that volume by summing up their rates and then integrating over time [4]. Hence at any given time-step the total mass and energy is known in each control volume. Momentum is accounted for by using “pseudo” pressures and temperatures [7]. Once a facility model has been built in Simulink® it is converted into C code using another MathWorks® product called Real-Time Workshop® and can be run on a variety of real-time operating systems. Originally the models were made to run on PowerPC’s® (PPC’s) that ran the VxWorks® operating system [3]. Figure 1.3 shows the structure of the systems that make up a simulation. The simulation, trainer screens, and programmable logic controllers (PLC’s) all interface with each other through a High Speed Digital Interface (HSDI). The HSDI runs on a desktop PC that has a reflective memory card that is connected to adjacent cards in the simulation and PLC’s. Reflective memory cards are physical network cards that replicate information at high speeds between other connected cards. The HSDI runs a converter which converts engineering units used by the simulation, such as lb_m , ft/s , lb_f/ft^2 , etc., to “counts” which are used by the PLC’s [2]. It is through this transformed data that the PLC’s are able to send commands to the simulation and receive feedback just like it would in the field. The trainer screen sends and receives I/O through Ethernet connected to the HSDI. Data traveling from the trainer screens to the simulation is transferred from the computer hosting the screens via Ethernet to reflective memory by the HSDI. If data from the simulation needs to be passed to the screens this process is done in reverse. Information exchange between the trainer screen and the simulation is thus affected.

The first model built using this architecture was found to consume 92% of the CPU’s bandwidth. The PPC’s only had a 300 MHz processor and this posed a problem in later simulations that were much larger. They would not be able to complete all of their calculations in the required time-step so models were forced to be split onto multiple PPC’s [9]. Since then, newer technologies have become available. Standard desktop PC’s have become faster and cheaper and MathWorks® has taken advantage of this

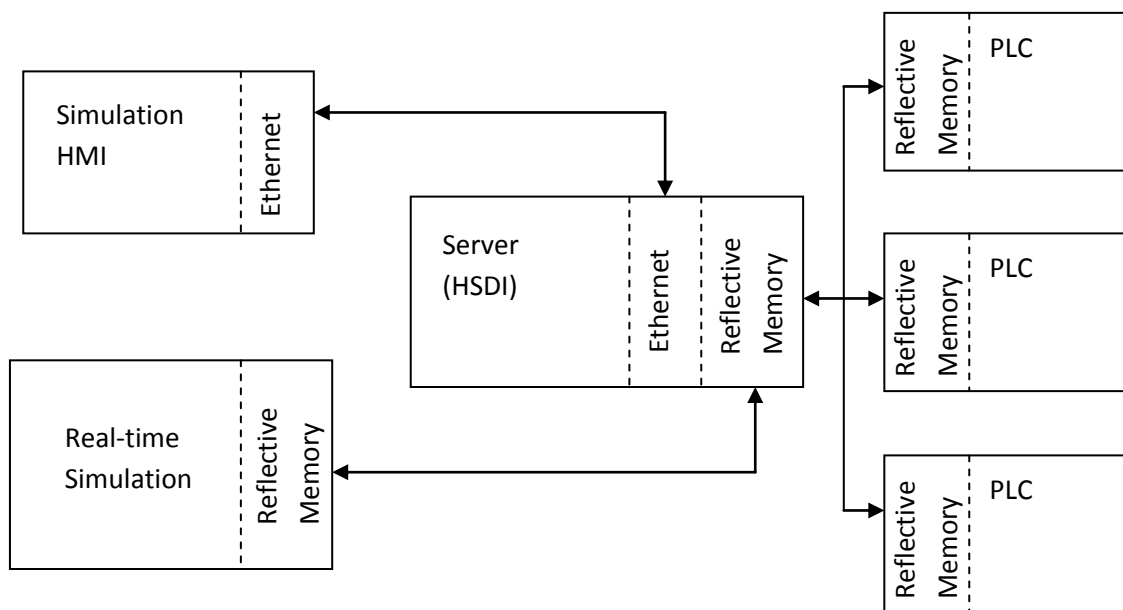


Figure 1.3 - Overall System Diagram of a Simulation

through their product called xPC Target® [2]. It is a real-time operating system that can run on a standard desktop PC. This allows for the models to be executed on much faster machines and reduces the number of machines required to run a given simulation.

The hardware that the real-time models run on is much faster than required by the current needs of the lumped-parameter method. Extra CPU processing power now exists that wasn't available before. The previous computer hardware placed a limitation on the size of the models and the physics that could be simulated because their processors were running at full capacity. This begs the question: what do to with all that extra processing power now available? One answer is to increase the fidelity of the simulations by incorporating spatial effects and solving for the momentum equation in addition to continuity and energy equations. The lumped-parameter method is essentially a zero dimensional control volume that ignores momentum and only solves for continuity and energy.

1.2 Improvements to the System

A new type of control volume has been built that solves all three of the equations of motion and includes new features such as: changes in cross sectional area, friction, minor losses due to bends and other flow obstructions, and localized heat transfer effects. With the new system it is possible to account for real physical phenomena, such as flow transients, that was impractical to simulate before. Section 2 provides more details of the lumped-parameter's deficiencies and capabilities that the improved method will bring. The rest of this thesis focuses on the derivation of the quasi-one-dimensional flow equations and how they are solved. Additional information is also provided detailing how the new method is implemented in Simulink® and interfaces to the user. The last two sections focus on results comparing computational solutions to known analytical problems and providing real-

time performance capabilities. Finally, an example facility problem that demonstrates the ability of the code to simulate a system of ducting is presented. This example compares the new solutions to those from the lumped-parameter method. Concluding remarks will then be given summarizing the results as well as possible future additions.

2. Current Needs and Development

The lumped-parameter control volume method discussed in section 1 has proven to work surprisingly well despite being of low-fidelity. However, this method does require considerable “tuning.” In order to tune a model there must be sufficient facility data available to adequately describe loss coefficients and efficiencies. Because the control volumes don’t account for spatial effects, any losses such as friction, bends, or sudden changes in area are not incorporated directly into the calculations. Instead these effects are built into flow devices as loss coefficients. The loss coefficient at a valve, for example, is usually taken as a constant value. However, in reality the loss coefficient is dependent upon specific conditions. It has been suggested that these types of parameters could be a function of facility conditions (pressure, temperature, flow area, Mach number, etc.) and used in a lookup table or expressed as an equation. These values would be determined empirically through testing or through the use of computational fluid dynamics. Both of these options can be quite costly, in particular for the testing option which could require a lot of time to gather the required data. This kind of model tuning is often ignored because of time and/or budget constraints and, at best, offers a model that can be used to test control logic and roughly find process control gains [7]. Final gain tuning has to be performed with the control system in the field and connected to the facility.

Another drawback to lumped-parameter modeling is the assumption that all flow properties are uniform within a defined boundary. By implication all the associated mixing processes are assumed to take place instantaneously. If Figure 1.1 is used as an example, the pressure, temperature, and density just downstream of valve six are assumed to be the same everywhere and changes in those properties are assumed to happen instantaneously. This is potentially a problem if a control engineer is designing a system to control the pressure upstream of the compressor using valve 6. The pressure feedback

experienced at the compressor inlet is not an accurate representation of the flow field and process lag in the real system is not included in the simulation. This is another reason why only rough gain tuning can be achieved using lumped-parameter control volumes.

Operator training can also suffer from the lumped-parameter simplifications. An operator who can tell the difference between a simulation and the real-world facility may be less inclined to use the simulation. Such an operator may not trust the simulation and may deviate from the normal procedure to achieve a certain condition or operating point. The impact could be greater on young or inexperienced operators who may not know how the facility is supposed to behave. If they used a poor representation of the facility for training, it is possible they could get a false sense of how fast or slow the real facility is supposed to react to their commands when in operation.

A new type of control volume has been developed, as discussed herein, to alleviate some of the shortcomings of the lumped-parameter method. The new method is a quasi-one-dimensional finite volume method using central differencing on a staggered grid. Since all three equations of motion are solved, the addition of a momentum equation allows for losses to be included in the calculations. The effects of area change, friction, and the minor losses (such as those resulting from bends in the pipework) are now accounted for directly. Heat transfer is also included at a local level rather than being lumped in globally. The equations are solved explicitly to allow for the control volumes to run in real-time while maintaining time accurate results within the confines of the quasi-one-dimensional model. Figure 2.1 illustrates the new capabilities that can be performed for a given length of ducting. The next section goes into detail of the equations of motion and algorithms used to solve these equations for this type of problem. This is followed by a section explaining how the new control volume

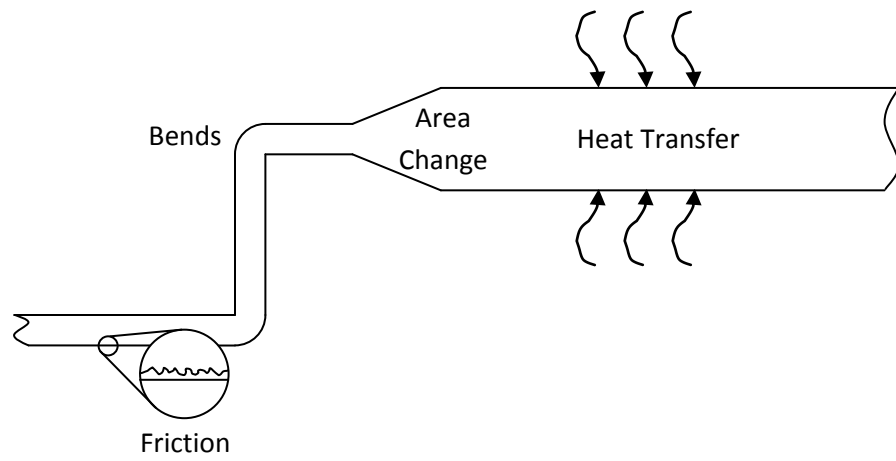


Figure 2.1 - Representation of New Capabilities

is used within Simulink®. Section 5 provides steady-state and transient results as well as real-time analysis of the new method. The last section is an example using all of the new capabilities of the control volume simulating a flow network that might be experienced in the real-world. Concluding remarks will then be given summarizing the results reported herein.

3. Finite Volume Approach

Looking back at Figures 1.1 and 1.2 in section 1, the system of ducting associated with the real facility has been reduced to several control volumes and mass flow devices. These volumes use the existing lumped-parameter method to solve for the conditions of the fluid at any given instance in time. The simplifications of the method assume that all fluid properties are uniform for a given volume. That is fluid properties do not vary spatially. This has several drawbacks described in section 2 and in order to bring higher fidelity simulations to meet new possible demands it is necessary to create a new type of control volume that brings dimension and depth through new modeling capabilities. It is the goal of this section to derive a new set of equations that can be incorporated into the existing methodology of drag and drop modeling and, at the same time, accept the boundary conditions set by the mass flow devices.

There are several approaches to writing the equations of motion of a fluid. One follows a fixed mass of fluid particles as it moves throughout a flow field such as on a streamline. The other considers a fixed volume in space and relates changes in mass, momentum, and energy taking place within it from fluid moving in and out of its boundaries. It is more convenient to describe the dynamics of fluid problems using the latter method and it is similar in convention to the lumped parameter control volume approach described in section 1 and should allow the existing boundaries (mass flow devices) to be used.

Let N be a generic extensive property of fluid body B and ρ the fluid density. D is the domain occupied by B so that $N = \int_D \rho dV$: N can be determined from equation 3.1, the Reynolds transport theorem [10], which is a three-dimensional generalization of the Leibniz integral rule.

$$\frac{dN}{dt} = \iiint_{c.v.} \left[\frac{\partial(\rho\eta)}{\partial t} + \nabla \cdot (\mathbf{V}\rho\eta) \right] dV \quad \text{Eq. 3.1}$$

An alternate form of the equation can be obtained by applying the Gauss theorem to the last term in the equation. It states that the outward flux of a vector field through a closed surface is equal to the volume integral of the divergence on the region inside the surface. Now there is:

$$\frac{dN}{dt} = \frac{\partial}{\partial t} \iiint_{c.v.} \eta \rho dV + \iint_s \eta (\rho \mathbf{V} \cdot d\mathbf{A}) \quad \text{Eq. 3.2}$$

with equation 3.1 equivalent to equation 3.2. The latter equation states that the rate of change of an extensive property N , for a system is equal to the time rate of change of N with the control volume and the rate at which N , the product of $\eta \rho \mathbf{V}$, is being transported through the surface to and from the surrounding region.

3.1 Conservation of Mass

The material derivative of a closed system is zero. If we let N in equation 3.2 equal the mass M , and since η would then refer to the material properties in “total mass per unit mass” and so is equal to one. This transforms equation 3.2 into the well known continuity equation:

$$0 = \frac{\partial}{\partial t} \iiint_{c.v.} \rho dV + \iint_s \rho \mathbf{V} \cdot d\mathbf{A} \quad \text{Eq. 3.3}$$

The time rate of change of mass in the system must equal the rate of mass influx into the control volume. The second term on the right hand side is positive for outflow and negative for inflow.

3.2 Conservation of Momentum

Newton's second law of motion states that the change in velocity of a body of fixed mass M is directly proportional to the magnitude of the forces applied to the body and inversely proportional to its mass. This can be represented by the following equation:

$$\sum \mathbf{F} = M \frac{d\mathbf{V}}{dt} \quad \text{Eq. 3.4}$$

Where $\sum \mathbf{F}$ is the sum of all the body forces and $d\mathbf{V}/dt$ is the time rate of change of velocity (that is, the acceleration). Equation 3.4 can be rewritten in terms of the linear momentum of the system:

$$\sum \mathbf{F} = \frac{d\mathbf{p}}{dt} \quad \text{Eq. 3.5}$$

where $\mathbf{p} = M\mathbf{V}$ is the local linear momentum. We can apply equation 3.2 to this set of variables by substituting $\mathbf{p}$ for N and, since $\mathbf{V}$ can be expressed as $\mathbf{p}$ per unit mass, equation 3.2 becomes:

$$\sum \mathbf{F} = \frac{\partial}{\partial t} \iiint_{c.v.} \rho \mathbf{V} dV + \iint_s \mathbf{V} (\rho \mathbf{V} \cdot d\mathbf{A}) \quad \text{Eq. 3.6}$$

The left hand side of the equation represents the summation of all the body forces acting on the control volume; this could be a pressure force, the fluid viscous forces or gravity. The right hand side represents the rate of change of linear momentum within the control volume plus the net rate of efflux of linear momentum from the control volume.

3.3 Conservation of Energy

The first law of thermodynamics states that energy must be conserved. The change in total energy ΔE , typically expressed as the net heat transferred into the system Q , minus the net work done

by the system W :

$$Q - W = \Delta E \quad \text{Eq. 3.7}$$

Equation 3.7 can be written to express the change in energy at any given instant in time:

$$\frac{\delta Q}{dt} - \frac{\delta W}{dt} = \frac{dE}{dt} \quad \text{Eq. 3.8}$$

The terms $\delta Q/dt$ and $\delta W/dt$ represent the instantaneous rates of heat and work transfer between the system and its surroundings. The energy in the system is composed of three terms: the internal energy, kinetic energy, and potential energy and is represented by equation 3.9:

$$e = u + \frac{V^2}{2} + gz \quad \text{Eq. 3.9}$$

Equations 3.8 and 3.9 are substituted into equation 3.2 to give the result:

$$\frac{d}{dt}(Q - W) = \frac{\partial}{\partial t} \iiint_{c.v.} e \rho dV + \iint_S e(\rho \mathbf{V} \cdot d\mathbf{A}) \quad \text{Eq. 3.10}$$

The work W can be divided into two parts: the work required to transport mass across the boundaries and all other forms of work. The flow work done by the system per unit mass is p/ρ and can be combined with the internal energy term to form the thermodynamic property enthalpy h :

$$h = u + \frac{p}{\rho} \quad \text{Eq. 3.11}$$

This equation in conjunction with 3.10 can be used to write the energy equation as:

$$\frac{d}{dt}(Q - W) = \frac{\partial}{\partial t} \iiint_{c.v.} e \rho dV + \iint_S \left(h + \frac{V^2}{2} + gz \right) (\rho \mathbf{V} \cdot d\mathbf{A}) \quad \text{Eq. 3.12}$$

3.4 The Governing Equations for Quasi-One-Dimensional Flow

The equations of motion that have been defined thus far are not very useful, in their current form, for the present application. These integral statements of the governing equations need to be applied to a control volume with a quasi-one-dimensional assumption. Figure 3.1 shows a control volume about point I. There are only two boundaries, A_i and A_{i+1} , that the flow is able to pass. Control volumes use upper case indices while flow boundaries, A , use lower case indices. If it is assumed that the properties of the fluid are known at each face, then equations 3.3, 3.6 and 3.12 can be applied. Further assumptions are made to the control volume to help simplify the equations: the volume remains fixed and does not vary with time and neither do the areas A_i and A_{i+1} . The first term on the right hand side of equation 3.3 simplifies to:

$$\frac{\partial}{\partial t} \iiint_{c.v.} \rho dV \approx V_I \left(\frac{\partial \rho}{\partial t} \right)_I \quad \text{Eq. 3.13}$$

since the volume remains fixed and only density changes with time. The second term is composed of the mass flux at the two faces. As above, outflow was defined to be positive and inflow negative. Hence:

$$\iint_S \rho \mathbf{V} \cdot d\mathbf{A} \approx (\rho VA)_{i+1} - (\rho VA)_i \quad \text{Eq. 3.14}$$

Combining equations 3.13 and 3.14, the time rate of change of mass in the control volume becomes:

$$V_I \left(\frac{\partial \rho}{\partial t} \right)_I \approx (\rho VA)_i - (\rho VA)_{i+1} \quad \text{Eq. 3.15}$$

The same method is applied to the energy equation:

$$V_I \left(\frac{\partial(\rho e)}{\partial t} \right)_I \approx (\rho VA)_i \left(h + \frac{v^2}{2} + gz \right)_i - (\rho VA)_{i+1} \left(h + \frac{v^2}{2} + gz \right)_{i+1} + \frac{d}{dt} (Q - W)_I \quad \text{Eq. 3.16}$$

The equation of energy can be further simplified by assuming that there will be no external work done

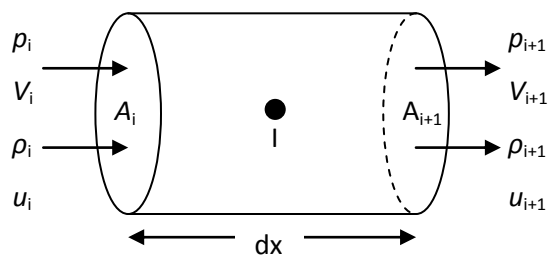


Figure 3.1 - Control Volume about Point I

on the control volume and that any potential energy is negligible because the density of a gaseous fluid is small and pipe diameters are not large. Equation 3.16 then becomes:

$$\dot{V}_I \left(\frac{\partial(\rho e)}{\partial t} \right)_I \approx (\rho V A)_i \left(h + \frac{v^2}{2} \right)_i - (\rho V A)_{i+1} \left(h + \frac{v^2}{2} \right)_{i+1} + \left(\frac{dQ}{dt} \right)_I \quad \text{Eq. 3.17}$$

The fluxes that have been defined thus far are consistent with the existing boundary conditions, from section 1, imposed by a mass flow device that is: mass flow and an energy flow are prescribed. However, those devices do not provide any information that can be used for a momentum flux. Furthermore, the way the equations are written lends itself to the adoption of central differencing. This poses a stability problem when trying to solve the equations. Both of these issues can be resolved by the use of a staggered grid [11]. Instead of keeping track of momentum at the cell centers, it will be calculated at the cell faces. The situation is shown in Figure 3.2. Equation 3.6 is written in the integral form of the momentum equation and calculates the time rate of change of momentum. However, the two terms on the right hand side of equation 3.15 are the mass flows on either face of the control volume. And the first two terms on the right hand side of equation 3.16 are the energy fluxes into and out of the control volume. It would be convenient for the momentum equation to directly solve for the mass flows at each face. By rewriting equation 3.6 as a partial differential equation, the time rate of change of mass flow can be calculated. This will provide the necessary quantities used in the right hand sides of equations 3.15 and 3.16.

Figure 3.3 shows the control volume being considered to develop the momentum equation. The shaded region chosen is a segment of an expansion with an infinitesimal thickness dx . The quasi-one-dimensional assumption demands that the pressure, velocity, density, and internal energy are uniform over the areas A and $A + dA$. Equation 3.6 can be written to account for the pressure forces on the control volume,

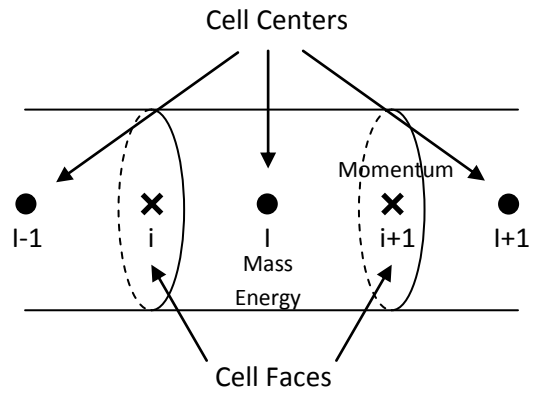


Figure 3.2 - Staggered Grid

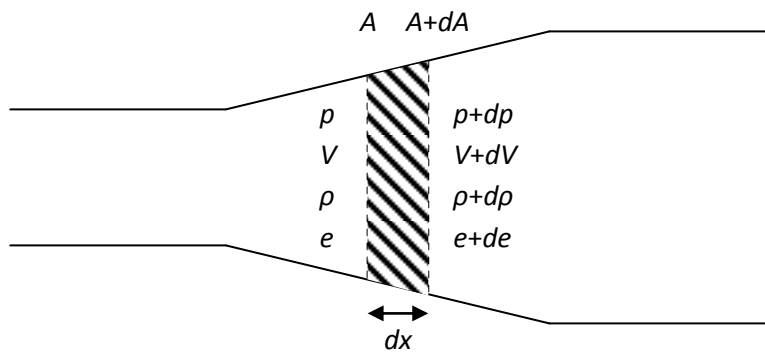


Figure 3.3 - Control Volume Used to Develop the Partial Differential Equation for Momentum

$$\frac{\partial}{\partial t} \iiint_{c.v.} \rho \mathbf{V} dV + \iint_S \mathbf{V}(\rho \mathbf{V} \cdot d\mathbf{A}) = - \iint_S (pd\mathbf{A})_x \quad \text{Eq. 3.18}$$

where the quantity $(pd\mathbf{A})_x$ denotes the x component of the vector $pd\mathbf{A}$. Using the control volume in Figure 3.3, the volume integral in equation 3.18 becomes,

$$\frac{\partial}{\partial t} \iiint_{c.v.} \rho \mathbf{V} dV = \frac{\partial}{\partial t} (\rho V A dx) \quad \text{Eq. 3.19}$$

where $A dx$ is the approximate volume of the control volume. The surface integral in equation 3.18 becomes:

$$\iint_S \mathbf{V}(\rho \mathbf{V} \cdot d\mathbf{A}) = -\rho V^2 A + (\rho + d\rho)(V + dV)^2 (A + dA) \quad \text{Eq. 3.20}$$

Figure 3.4 can be used to evaluate the pressure forces on the right hand side of equation 3.18 [11]. The pressure forces are shown with respect to x on all four sides of the control volume. The x component of $pd\mathbf{A}_1$ is $-pA$ and the force $pd\mathbf{A}_4$ is given by $(p+dp)(A+dA)$. The forces $(pd\mathbf{A}_2$ and $pd\mathbf{A}_3$) acting on the top and bottom of the control volume can be expressed as their inclined areas projected perpendicular to the x direction, $dA/2$. Each of the forces acting in the x direction then become $-p(dA/2)$. The sum of all the pressures forces can be expressed as

$$\iint_S (pd\mathbf{A})_x = -pA + (p + dp)(A + dA) - 2p \frac{dA}{2} \quad \text{Eq. 3.21}$$

Substituting equations 3.19, 3.20, and 3.21 into equation 3.18, we have

$$\frac{\partial}{\partial t} (\rho V A dx) - \rho V^2 A + (\rho + d\rho)(V + dV)^2 (A + dA) = pA - (p + dp)(A + dA) + 2pdA \quad \text{Eq. 3.22}$$

Expanding the products, cancelling like terms, and ignoring the products of differentials to only first order, equation 3.22 reduces to the approximate equality:

$$\frac{\partial}{\partial t} (\rho V A dx) + d(\rho V^2 A) \approx -A dp \quad \text{Eq. 3.23}$$

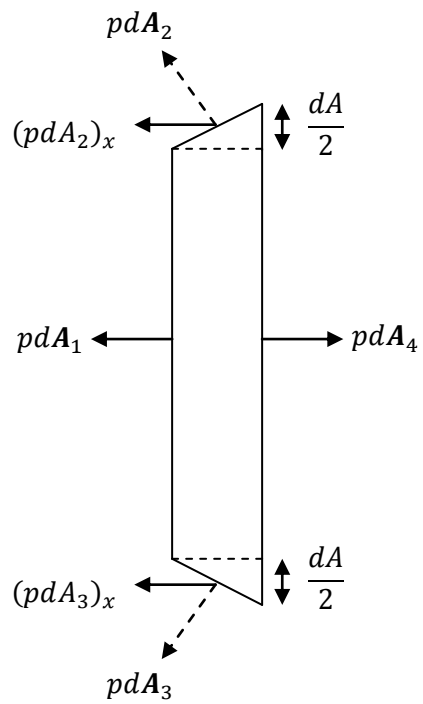


Figure 3.4 - Pressure Forces Acting on the Control Volume

Equation 3.23 can be divided by dx and if the limit of dx as it approaches zero is taken, the partial differential equation:

$$\frac{\partial(\rho VA)}{\partial t} + \frac{\partial(\rho V^2 A)}{\partial x} = -A \frac{\partial p}{\partial x} \quad \text{Eq. 3.24}$$

is obtained, which is the momentum equation appropriate for quasi-one-dimensional flow, written in the conservation form.

3.4.1 Heat Transfer

Unwanted heat transfer is a regular occurring phenomenon in ground testing facilities. Energy is often purposefully added to, or taken away from, the flow so that facility conditions match with conditions that the test article would experience in flight. However, plant ducting is often non-insulated and can be exposed to atmospheric conditions. The large metal mass of the ducting can act as a heat sink or source for the internal flow. The controls engineer needs to know how the heat transfer alters the temperature of the flow before it reaches the test cell. With accurate modeling, the engineer can better tune process gains and even see if existing plant equipment, such as a cooler, is sufficient to compensate for the unwanted heat transfer.

The equations will be written with the assumption that the temperature of the ducting is warmer than both the internal flow and its external surroundings. Figure 3.5 shows the various modes of heat transfer that can exist. On the outside of the ducting there is the solar flux for the sun, which is represented by the wavy-dashed lines, radiation between the material of the ducting and the ambient conditions is represented by the wavy-solid lines. If the outside conditions are windy then forced convection can exist to aid in the amount of heat transferred to or from the ducting. If the outside air is

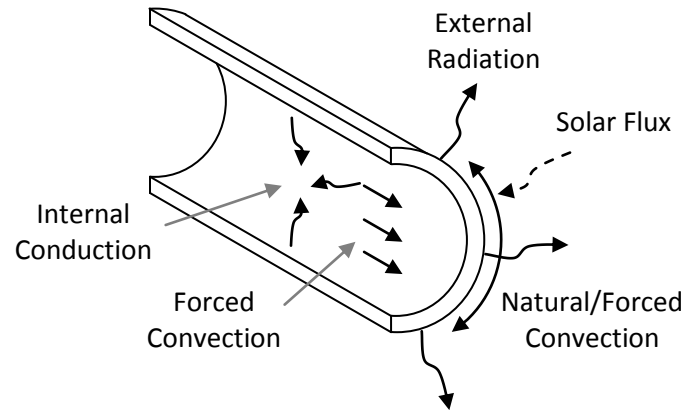


Figure 3.5 - The Modes of Heat Transfer for a Typical Piece of Ducting

quiescent, then convection can still be present but now takes the form of natural convection. Both types of heat transfer are represented by the curved arrow going around the ducting in Figure 3.5. Inside the ducting, the two modes of heat transfer being considered are conduction from the ducting wall to the fluid and forced convection of the moving flow, which are the wavy and straight solid lines, respectively.

The change in energy stored in the ducting can be described by equation 3.25 where ρ is the density of the material, V is the volume, and c is the specific heat of the ducting material. The left hand side of the equation is the change in energy of the ducting with time and is equal to the sum of the heat transfer due to the solar flux, the external radiation and the convection. Finally, the internal conduction and forced convection are included.

$$\frac{dE}{dt} = \rho V c \frac{\partial T}{\partial t} \quad \text{Eq. 3.25}$$

The amount of energy transferred from the sun is given by the solar flux times the surface area of the ducting. This amount of energy is then adjusted to take into account absorption by carbon dioxide and water vapor in the atmosphere. The amount of energy transferred by forced or natural convection is given by equation 3.26. The heat transfer coefficient is defined as h_{out} and is given by equation 3.27, A_{out} is the wetted surface area, T_{duct} and T_{amb} are the temperatures of the duct and the ambient air, respectively. In equation 3.27 Nu_D is the Nusselt number, k is the thermal conductivity of air, and d_2 is the outside diameter of the duct. Churchill and Bernstein [12] established a relation to describe the Nusselt number of a cylinder in cross flow and this is given by equation 3.28a. Here Re is the Reynolds number and Pr is the Prandtl number. The Nusselt number for natural convection around an isothermal cylinder is given by Churchill and Chu [12] and is defined by equation 3.28b where Pr is the Prandtl number of air and Ra is the Rayleigh number. In both equations, the properties are

evaluated at the film temperature. The film temperature is the average of the skin and ambient temperatures.

$$q_{ex_conv} = h_{out} A_{out} (T_{duct} - T_{amb}) \quad \text{Eq. 3.26}$$

$$h_{out} = Nu_D \frac{k}{d_2} \quad \text{Eq. 3.27}$$

$$Nu_D = \frac{0.62 Re^{1/2} Pr^{1/3}}{\left(1 + \left(\frac{0.4}{Pr}\right)^{2/3}\right)^{1/4}} \left[1 + \left(\frac{Re}{282,000}\right)^{5/8}\right]^{4/5} \quad \text{Eq. 3.28a}$$

$$Nu_D = \left\{0.60 + \frac{0.387 Ra_D^{1/6}}{[1 + (0.559/Pr)^{9/16}]^{8/27}}\right\}^2 \quad \text{Eq. 3.28b}$$

The equations for the internal forced convection are similar, but with the Nusselt number defined differently. For laminar flow, equation 3.29 is used to define the Nusselt number while equation 3.30, developed by Dittus and Boelter [12], is used for turbulent flow. The exponent n is equal to 0.4 when the fluid is being heated and 0.3 when the fluid is being cooled. In equation 3.31, the heat transfer rate for forced convection, A_{in} is the surface area inside the duct and T_f is the temperature of the fluid. The heat transfer coefficient can be found using equation 3.32.

$$Nu_D = \frac{48}{11}, \text{for laminar flow} \quad \text{Eq. 3.29}$$

$$Nu_D = 0.023 Re_D^{4/5} Pr^n, \text{for turbulent flow} \quad \text{Eq. 3.30}$$

$$q_c = h_{in} A_{in} (T_{duct} - T_f) \quad \text{Eq. 3.31}$$

$$h_{in} = Nu_D \frac{k}{d_2} \quad \text{Eq. 3.32}$$

The heat transfer given by radiation is defined by equation 3.33 where σ is the Stefan-Boltzman constant, α is the absorptivity, A_{out} is the surface area of the outside duct, and T_{amb} is the temperature

of the ambient surroundings for the external radiation.

$$q_{rad} = \sigma \alpha A_{out} (T_{duct}^4 - T_{amb}^4) \quad \text{Eq. 3.33}$$

The amount of energy transferred due to conduction is given by equation 3.34, where k is the thermal conductivity, d_2 is the outside duct diameter and d_1 is the inside diameter.

$$q_{cond} = -2k\pi\Delta x \frac{T_{duct} - T_f}{\ln \frac{d_2}{d_1}} \quad \text{Eq. 3.34}$$

Therefore, at any given instant in time the temperature change in the ducting can be calculated from:

$$\begin{aligned} \frac{\partial T}{\partial t} = \frac{1}{\rho V c} \left[-\sigma \alpha A_{out} (T_{duct}^4 - T_{amb}^4) + q_{sun} - h_{out} A_{out} (T_{duct} - T_{amb}) - h_{in} A_{in} \times \right. \\ \left. (T_{duct} - T_f) - 2k\pi\Delta x \frac{T_{duct} - T_f}{\ln \frac{d_2}{d_1}} \right] \quad \text{Eq. 3.35} \end{aligned}$$

The amount of heat transfer into the flow is given by the energy taken out from the ducting by the radiation and the forced convection:

$$q_{flow} = h_{in} A_{in} (T_{duct} - T_f) + 2k\pi\Delta x \frac{T_{duct} - T_f}{\ln \frac{d_2}{d_1}} \quad \text{Eq. 3.36}$$

This equation can be substituted for the last term on the right hand side of equation 3.17.

3.4.2 Friction and Other Minor Losses

Including the momentum equation affords the opportunity to incorporate frictional losses in the system and to account for pipe bends and pipe intersections at tee junctions. These items add a head loss and these losses are commonly referred to as “minor losses.” Both types of losses were not a part of the existing lump-parameter control volume approach described in section 1. Pressure losses were

instead lumped at a flow device, through a loss coefficient, to restrict the mass flow into or out of a control volume. This coefficient was typically held constant and did not vary with flow conditions; by adding these losses into the momentum equation they become spatial effects and are a function of the local properties of the fluid. This is much more representative of a flow network in a test facility (or any other process facility). Figure 3.6 is used in the aid of defining the change in momentum due to friction [10]. If we take an infinitesimal control volume and draw the force due to friction as going against the direction of flow, we can define the frictional force F_f as:

$$F_f = -\tau_w(PdL) \cos \theta \quad \text{Eq. 3.37}$$

where τ_w is the shear stress along the wall, P is the wetted perimeter, and dL is the length that the frictional force is acting over. The cosine of the angle of the wall was taken to get the x component of the force. Equation 3.37 can be rewritten by noting that $dx = dL \cos \theta$. Solving for dL and substituting into equation 3.37, the force due to friction becomes:

$$F_f = -\tau_w P dx \quad \text{Eq. 3.38}$$

The friction factor f is related to the average shear stress at the wall to the dynamic pressure by the following equation:

$$f = \frac{4\tau_w}{\rho V^2/2} \quad \text{Eq. 3.39}$$

This is known as the Darcy-Weisbach friction factor. The frictional force can be equated in terms of a hydraulic diameter since not all tubes and flow channels are circular:

$$D_H = \frac{4A}{P} \quad \text{Eq. 3.40}$$

where A is the cross sectional area and P is the wetted perimeter. Equations 4.38 and 4.39 can be substituted into equation 3.38 to yield [10]

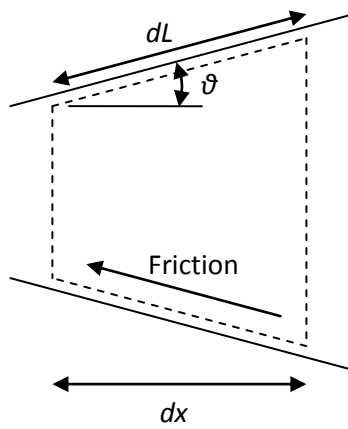


Figure 3.6 - Friction Analysis on a Control Volume

$$F_f = -f \frac{\rho V^2}{2} \frac{A}{D_H} dx \quad \text{Eq. 3.41}$$

Equation 3.40 can be included as part of the forces acting on the control volume in equation 3.24.

Performing the next step by dividing by dx we obtain the final form of the force of friction.

$$\frac{F_f}{dx} = -f \frac{\rho V^2}{2} \frac{A}{D_H} \quad \text{Eq. 3.42}$$

For laminar flow the friction factor can be approximated by equation 3.43, where Re is the Reynolds number. This equation is good for Reynolds numbers less than 2300. For fully turbulent flow, Reynolds numbers greater than 4000, the Darcy friction factor is given by the Colebrook-White equation. It is an implicit equation that fits experimental results of laminar and turbulent flow in pipes. However, since it has to be solved implicitly it is not suitable for real-time applications. Several explicit approximations exist for the Colebrook-White equation and one such is given by Swamee and Jain in equation 3.44, where it is a function of ε , the surface roughness height, the pipe diameter, D , and the Reynolds number.

$$f = \frac{64}{Re} \quad \text{Eq. 3.43}$$

$$f = \frac{0.25}{\left[\log \left(\frac{\varepsilon}{3.7D} + \frac{5.74}{Re^{0.9}} \right) \right]^2} \quad \text{Eq. 3.44}$$

Minor losses in pipe and tube systems are often expressed as a head loss, see equation 3.45. In that equation K is the loss coefficient, V is the velocity of the fluid, and g is the acceleration due to gravity. The definition of a pressure head, the height of a column of fluid of specific weight required to give a pressure difference of $(p_2 - p_1)$ and given by equation 3.45, can be used to rewrite the minor loss equation in terms of a pressure difference, equation 3.46.

$$h_{minor_loss} = K \frac{V^2}{2g} \quad \text{Eq. 3.45}$$

$$h_{minor_loss} = \frac{(p_2 - p_1)}{\gamma}, \text{ where } \gamma, \text{ the specific weight} = \rho g \quad \text{Eq. 3.46}$$

$$\Delta p = K \frac{\rho V^2}{2} \quad \text{Eq. 3.47}$$

From the right hand side of equation 3.24, a pressure difference across the cross sectional area of an infinitesimal control volume leads to a change in mass flow rate (or momentum). Borrowing that term, and substituting equation 3.47 for dp , the force due to minor losses is equal to:

$$F_m = -K \frac{\rho V^2 A}{2 dx} \quad \text{Eq. 3.48}$$

Adding equations 3.42 and 3.48 to the right hand side of equation 3.24 we obtain the final form of the momentum equation.

$$\frac{\partial(\rho VA)}{\partial t} + \frac{\partial(\rho V^2 A)}{\partial x} = -A \frac{\partial p}{\partial x} - f \frac{\rho V^2}{2} \frac{A}{D_H} - K \frac{\rho V^2}{2} \frac{A}{dx} \quad \text{Eq. 3.49}$$

3.5 Discretization

To calculate the time rate of change of mass in the control volume, equation 3.15 can be used, where all that is needed is the mass flow at points i and $i + 1$. For the end points that have a boundary condition, $I - 1$ and $I + 1$, the mass flow into or out of the control volumes is calculated by an attached mass flow device. See Figure 3.7 for an example of how the staggered grid is applied to a given section of ducting.

Applying equation 3.15 to the I 'th node, the equation for calculating the time rate of change of mass is obtained, where M is the mass in the control volume.

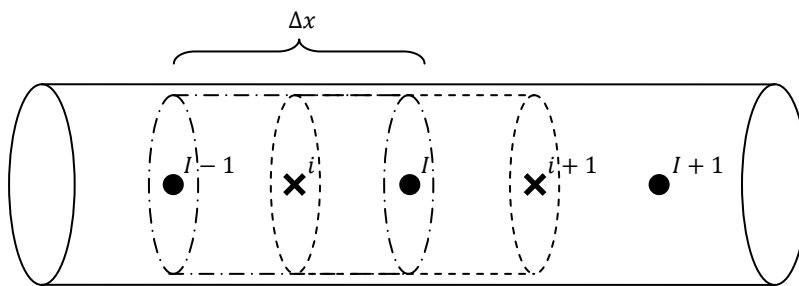


Figure 3.7 – Discretization of the Quasi-One-Dimensional Control Volume

$$\left(\frac{\partial M}{\partial t}\right)_I = (\rho VA)_i - (\rho VA)_{i+1} \quad \text{Eq. 3.50}$$

Applying the momentum equation, equation 3.49, to the i 'th node we obtain the following to calculate the time rate of change of mass flow, where $\dot{m}$ is the mass flow.

$$\left(\frac{\partial \dot{m}}{\partial t}\right)_i = \frac{(\rho VA)_{I-1}|V_{I-1}| - (\rho VA)_I|V_I|}{\Delta x} + \frac{A_i}{\Delta x} (p_{I-1} - p_I) - \frac{1}{2}f(\rho V^2)_i \frac{A_i}{D_H} - \frac{1}{2}K(\rho V^2)_i \frac{A_i}{\Delta x} \quad \text{Eq. 3.51}$$

The first term on the right hand side of equation 3.51 contains velocities that are needed at the cell centers, but because momentum is calculated at the cell faces velocity is not directly known at the cell centers. The velocity is calculated by taking the average mass flow of the cell faces straddling the center and then dividing by density and the mean area, both of which are known at the cell centers. Equations 3.51 and 3.52 calculate the velocities at nodes I and $I - 1$.

$$V_I = \frac{1}{2} \frac{[(\rho VA)_i + (\rho VA)_{i+1}]}{(\rho A)_I} \quad \text{Eq. 3.52}$$

$$V_{I-1} = \frac{1}{2} \frac{[(\rho VA)_{i-1} + (\rho VA)_i]}{(\rho A)_{I-1}} \quad \text{Eq. 3.53}$$

The last two terms on the right hand side of equation 3.50 contain the product of ρV^2 , this can be split into $(\rho V)V$. The product ρV can be calculated by taking the mass flow at the cell face and dividing by its area. Similarly, the velocity can be found by taking the mass flow and dividing by the area and density. Since density is not stored at this location it is calculated by taking the average density from the cell centers on either side of a face. Equations 3.54 and 3.55 are the momentum losses for friction and minor losses, respectively, with their products rewritten in terms of known variables.

$$(F_f)_i = -\frac{1}{2}f \frac{(\rho VA)_i}{A_i} \frac{|(\rho VA)_i|}{0.5(\rho_{I-1} + \rho_I)A_i} \frac{A_i}{D_H} \quad \text{Eq. 3.54}$$

$$(F_m)_i = -\frac{1}{2}K \frac{(\rho VA)_i}{A_i} \frac{|(\rho VA)_i|}{0.5(\rho_{I-1} + \rho_I)A_i} \frac{A_i}{\Delta x} \quad \text{Eq. 3.55}$$

Substituting equations 3.52 through 3.55 into equation 3.51 it is now possible to calculate the time rate of change of mass flow at a cell face.

To obtain the time rate of change in energy, equation 3.17 is applied to figure 3.7. It is repeated below for clarity with the exception that the left hand side of the equation is rewritten in terms of E , the energy in the control volume and is defined as $\rho V(c_v T + 0.5V^2)$. If the enthalpy is written in its expanded form, the internal energy and kinetic energy can be replaced by E/M . This new form is given by equation 3.56.

$$\left(\frac{\partial(E)}{\partial t}\right)_I = (\rho VA)_i \left(h + \frac{v^2}{2}\right)_i - (\rho VA)_{i+1} \left(h + \frac{v^2}{2}\right)_{i+1} + \left(\frac{dQ}{dt}\right)_I \quad (\text{Eq. 3.17})$$

$$\left(\frac{\partial(E)}{\partial t}\right)_I = (\rho VA)_i \left(\frac{E}{M} + \frac{p}{\rho}\right)_i - (\rho VA)_{i+1} \left(\frac{E}{M} + \frac{p}{\rho}\right)_{i+1} + \left(\frac{dQ}{dt}\right)_I \quad \text{Eq. 3.56}$$

The flux terms contain information about the flow not directly known at the cell faces. The mass flow, $(\rho VA)_i$, is known but the energy terms are not. The energy can be averaged by the two adjacent cell centers, similarly to the way that density was calculated at the cell faces. Equation 3.56 can be rewritten as:

$$\left(\frac{\partial(E)}{\partial t}\right)_I = (\rho VA)_i \frac{1}{2} \left[\left(\frac{E}{M} + \frac{p}{\rho}\right)_{I-1} + \left(\frac{E}{M} + \frac{p}{\rho}\right)_I \right] - (\rho VA)_{i+1} \frac{1}{2} \left[\left(\frac{E}{M} + \frac{p}{\rho}\right)_I + \left(\frac{E}{M} + \frac{p}{\rho}\right)_{I+1} \right] + \left(\frac{dQ}{dt}\right)_I \quad \text{Eq. 3.57}$$

In equations 3.51 and 3.57 the pressure terms have not been defined at this point and one more equation is needed before it can be calculated. The equation of state can be used to determine the pressure if the temperature and specific volume of a substance are known. If intermolecular forces and the volume of the molecules are ignored, the equation of state can be modeled by the ideal gas law,

$$p = \rho RT \quad \text{Eq. 3.58}$$

where R is the specific gas constant. Equation 3.58 is a reasonable assumption if the temperature of the gas is significantly higher than its critical temperature and if the pressure is significantly lower than its

critical pressure. This is a valid assumption for most duct flow problems where the gas is not rarefied and it does not approach its saturation region. Equations 3.49, 3.50, 3.56 and 3.57 can now be used to solve for the time rates of change of mass, mass flow and energy. At this point it is helpful to deconstruct the mass, mass flow, and energy terms to calculate the primitive variables ρ , T , and V .

$$\rho = \frac{M}{V} \quad \text{Eq. 3.59}$$

$$V = \frac{\dot{m}}{\rho A} \quad \text{Eq. 3.60}$$

$$T = \frac{\gamma-1}{R} \left(\frac{E}{M} - \frac{1}{2} V^2 \right) \quad \text{Eq. 3.61}$$

Once the temperature and density are known, the ideal gas law can be used to calculate pressure. Equations 3.58 through 3.61 are used to calculate the primitive variables at any instant in time.

3.6 Heun's Method for Integration of the Time Derivatives

The time derivatives of mass (Eq. 3.50), mass flow (Eq. 3.51) and energy (Eq. 3.56), are integrated using Heun's method [13]. Heun's method is a standard method for solving ordinary differential equations (ODE) with a given initial value. The method is second order in time and is equivalent to the second order Runge-Kutta method. The general formula for Heun's method is given below:

$$y'(t) = f(t, y(t)) \quad \text{Eq. 3.62a}$$

$$\tilde{y}^{n+1} = y^n + \Delta t f(t^n, y^n) \quad \text{Eq. 3.62b}$$

$$y^{n+1} = y^n + \frac{\Delta t}{2} [f(t^n, y^n) + f(t^{n+1}, \tilde{y}^{n+1})] \quad \text{Eq. 3.62c}$$

where y is the dependent variable and the time, t , is the independent variable, Δt is defined as the stepsize and n is the time interval. Equation 3.62b uses Euler's method to find a preliminary value $\tilde{y}$ at

the next time-step, $n + 1$. The derivative of $\tilde{y}$ is then found and used in equation 3.62c, in conjunction with the derivative of y , to calculate an average change in y over the interval $[t^n, t^{n+1}]$. This information is then used to compute y at time $n + 1$. Taking the average slope over the time interval, Δt , has an advantage over Euler's method because Euler only uses the slope at the end of t^n and unless $y(t)$ is essentially a straight line, the use of the slope evaluated only at one time step may not be an accurate representation of the change in y with time.

The equations of motion that have been defined are a system of coupled first-order equations. Heun's method, and other initial value solvers, can be easily extended to solve this type of system. If the ODE used to describe Heun's method is extended to include multiple state variables we would get equation 3.63.

$$\begin{aligned} y_1' &= \frac{dy_1}{dt} = f_1(t, y_1, y_2, \dots, y_k), & y_1(t_0) &= y_{01} \\ y_2' &= \frac{dy_2}{dt} = f_2(t, y_1, y_2, \dots, y_k), & y_2(t_0) &= y_{02} \\ &\vdots & &\vdots \\ y_k' &= \frac{dy_k}{dt} = f_k(t, y_1, y_2, \dots, y_k), & y_k(t_0) &= y_{0k} \end{aligned} \quad \text{Eq. 3.63}$$

The rates at which y_1 , y_2 , or any y_k change at time t depend on any combination of the state variables. Furthermore, at any given t , we can describe the rate at which each state variable changes when the system is in a particular state. Equation 3.63 can be rewritten in vector notation if we let

$$\mathbf{y} = \mathbf{y}(t) = [y_1 \quad y_2 \quad \dots \quad y_n]^T \quad \text{Eq. 3.64}$$

then equation 3.63 can be rewritten as:

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(t_0) = \mathbf{y}_0 \quad \text{Eq. 3.65}$$

where $\mathbf{y}'$, $\mathbf{f}(t, \mathbf{y})$, and $\mathbf{y}_0$ are defined by

$$\mathbf{y}' = \begin{bmatrix} y_1' \\ y_2' \\ \vdots \\ y_k' \end{bmatrix}, \quad \mathbf{f}(t, \mathbf{y}) = \begin{bmatrix} f_1(t, \mathbf{y}) \\ f_2(t, \mathbf{y}) \\ \vdots \\ f_k(t, \mathbf{y}) \end{bmatrix}, \quad \text{and} \quad \mathbf{y}_0 = \begin{bmatrix} y_{01} \\ y_{02} \\ \vdots \\ y_{0k} \end{bmatrix} \quad \text{Eq. 3.66}$$

Equation 3.66 can now be substituted into equations 3.62b and 3.66c to give

$$\tilde{\mathbf{y}}^{n+1} = \mathbf{y}^n + \Delta t \mathbf{f}(t^n, \mathbf{y}^n) \quad \text{Eq. 3.67a}$$

$$\mathbf{y}^{n+1} = \mathbf{y}^n + \frac{\Delta t}{2} [\mathbf{f}(t^n, \mathbf{y}^n) + \mathbf{f}(t^{n+1}, \tilde{\mathbf{y}}^{n+1})] \quad \text{Eq. 3.67b}$$

Heun's method for integration can now be used to solve a system of ODEs.

4. Implementation in Simulink®

The existing lumped parameter control volumes were written using Matlab®'s Simulink® in a block diagram format. However, the equations needed to solve the quasi-one-dimensional flow problem discussed above are complicated enough that it would be quite cumbersome to code them in the same manner. As part of Simulink®, Matlab® provides a way for Simulink® blocks to be written in a computer language such as MATLAB®, C, C++, Ada and FORTRAN and are called S-functions (system-functions). An S-function is then compiled as a MEX-file (MATLAB executable) which allows the S-function to be treated as a dynamically linked subroutine that the MATLAB interpreter can automatically load and execute. S-functions are built using special API commands that enable the user to interact with the Simulink® engine [6]. The S-function can then be placed inside a simulation and treated like any other Simulink® block. From a user's point of view, it looks and acts like any other block.

Keeping in mind that facility models are run in real-time, the S-function used to solve the quasi-one-dimensional equations were written in C. That particular language was chosen for two primary reasons. First and foremost, models that are compiled to run on Matlab's xPC Target operating system are first automatically converted into C code then compiled as an executable. Writing S-functions in C makes linking the code during the compile process seamless. Some of the languages are not compatible when trying to compile a real-time model. For example an S-function written in MATLAB, would produce an error if introduced. The other languages noted above could have been chosen but extra steps would have been needed to link the S-function to the automatically generated C code during the compile process. The second reason for choosing C is the speed of execution gained if a model is run on the desktop (that is, using the Simulink® engine to run a model instead of compiling it). S-functions

written in a compiled language, such as C, execute much faster than S-functions written using an interpreted language such as MATLAB.

The subsections that follow describe how a user would connect the new quasi-one-dimensional control volume to mass flow devices and how to use the user interface to setup geometry and initial conditions. Examples that demonstrate the new control volume will use valves as the mass flow devices. The last subsection will provide flow diagrams of the code that demonstrate how the equations are solved.

4.1 How to Use the New Control Volume in Simulink®

The first step of using the new control volume is to place it in the Simulink® model. Figure 4.1 shows the new control volume placed in an empty simulation. The new block can be copied from an already existing model or placed from a library of custom Simulink® blocks. From the figure, the block accepts one input from an upstream mass flow device and has one output to a downstream mass flow device. Figure 4.2 shows the control volume block connected to two valves. In this example, the current view of the control volume block is only a top level view of the subsystem named “Duct.” The subsystem can be opened up to reveal what is underneath.

Figure 4.3 displays the contents of the “Duct” subsystem. The big block in the center labeled “Control Volume S-function” is the S-function that contains the C code written to solve the quasi-one-dimensional flow problem. The center of the block contains the name of the MEX-file that is called during model execution. “OneDCV” stands for One-Dimensional Control Volume. This particular block

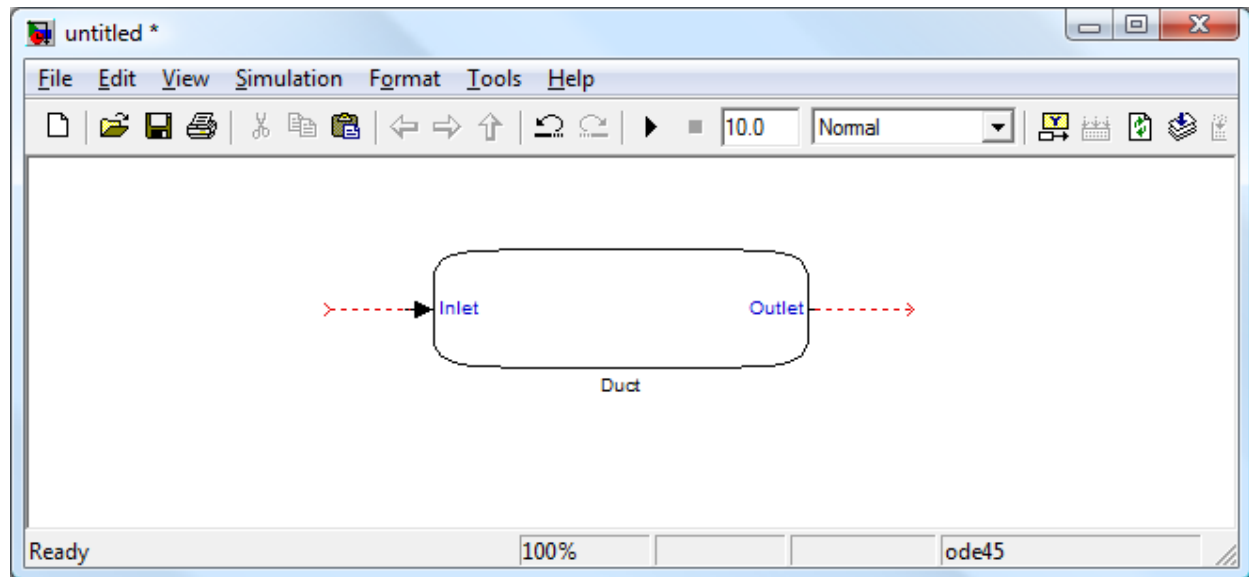


Figure 4.1 - Top Level of the New Quasi-One-Dimensional Control Volume in Simulink®

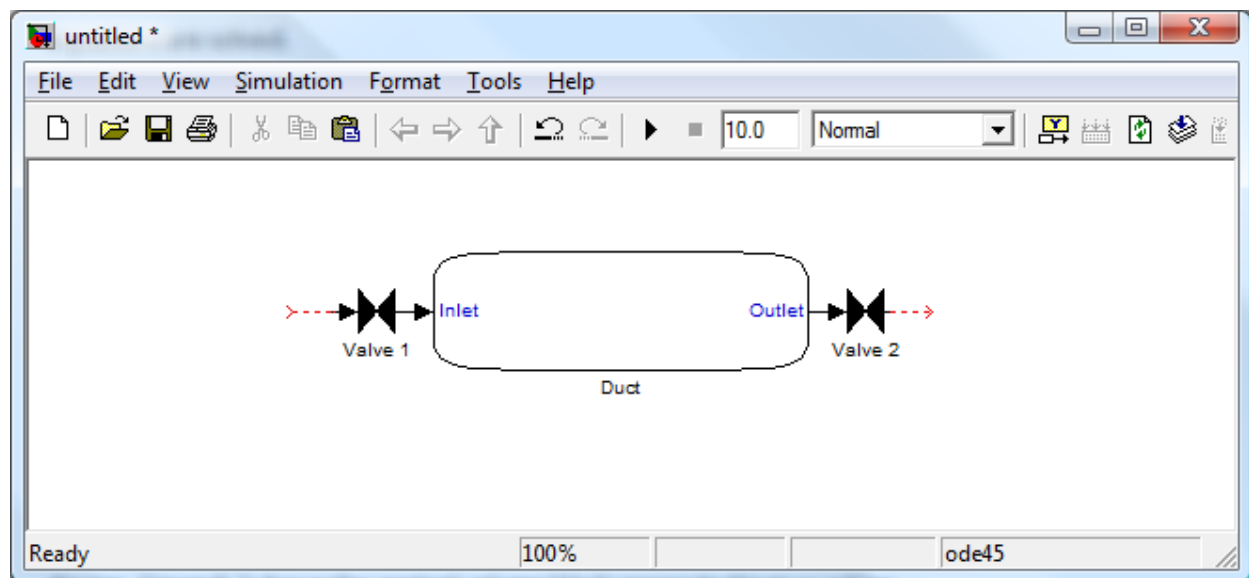


Figure 4.2 - Top Level of the New Quasi-One-Dimensional Control Volume with Valves Attached

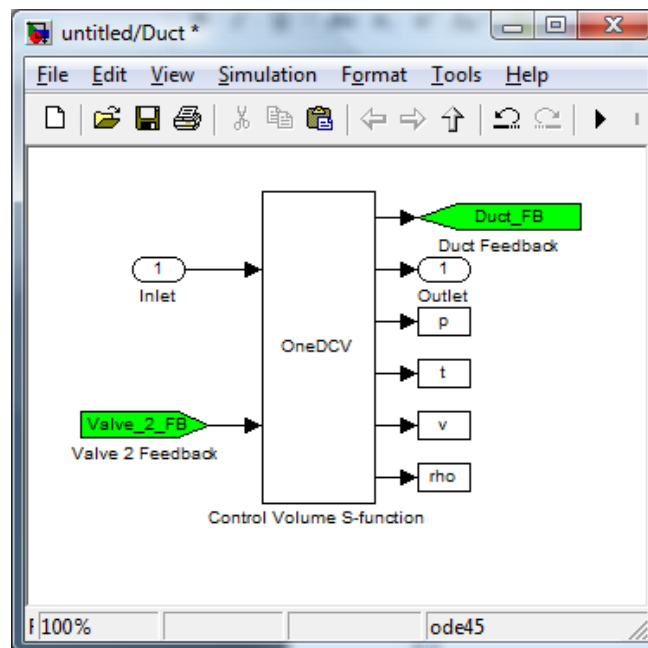


Figure 4.3 - The Subsystem that Makes Up the Quasi-One-Dimensional Control Volume

has two inputs and six outputs. The first input, labeled “Inlet”, are the inputs from Valve 1. The parameters being passed through are a mass flow rate and an energy flow rate. The second input to the S-function is the feedback from the downstream valve, Valve 2. It is also providing a mass flow rate and an energy flow rate. These are the mass and energy fluxes entering the duct and exiting the duct respectively. The first output is the feedback that goes to Valve 1. It provides the conditions of the duct at the inlet, such as pressure and temperature, just downstream of Valve 1. The second output, labeled “Outlet”, goes to Valve 2 and provides the conditions at the duct exit. The next four outputs are to blocks that are called “To Workspace” blocks. They provide a means to get data out of the simulation and into the Matlab workspace. From the workspace the data can be manipulated for analysis or used to make plots. Static pressures and temperatures along with velocities and densities are saved. They are used to visualize pressure, temperature, velocity and density distributions along the length of the duct. These blocks are not needed for anything other than reporting and can be removed from the “Duct” subsystem. By removing these blocks the overall simulation would be able to run faster improving execution time. When compiling the simulation for real-time execution these blocks would be ignored since xPC Target does not support “To Workspace” blocks.

4.2 Interfacing with the Quasi-One-Dimensional Control Volume

Once the control volume block has been placed in a simulation it must be configured for the situation at hand. Each control volume can have its own unique geometry, initial conditions, and internal and external factors acting on the fluid. By double-clicking on the “Duct” block in Figures 4.1 or 4.2 a setup screen will open and it will look like Figure 4.4. The setup screen is split into five major

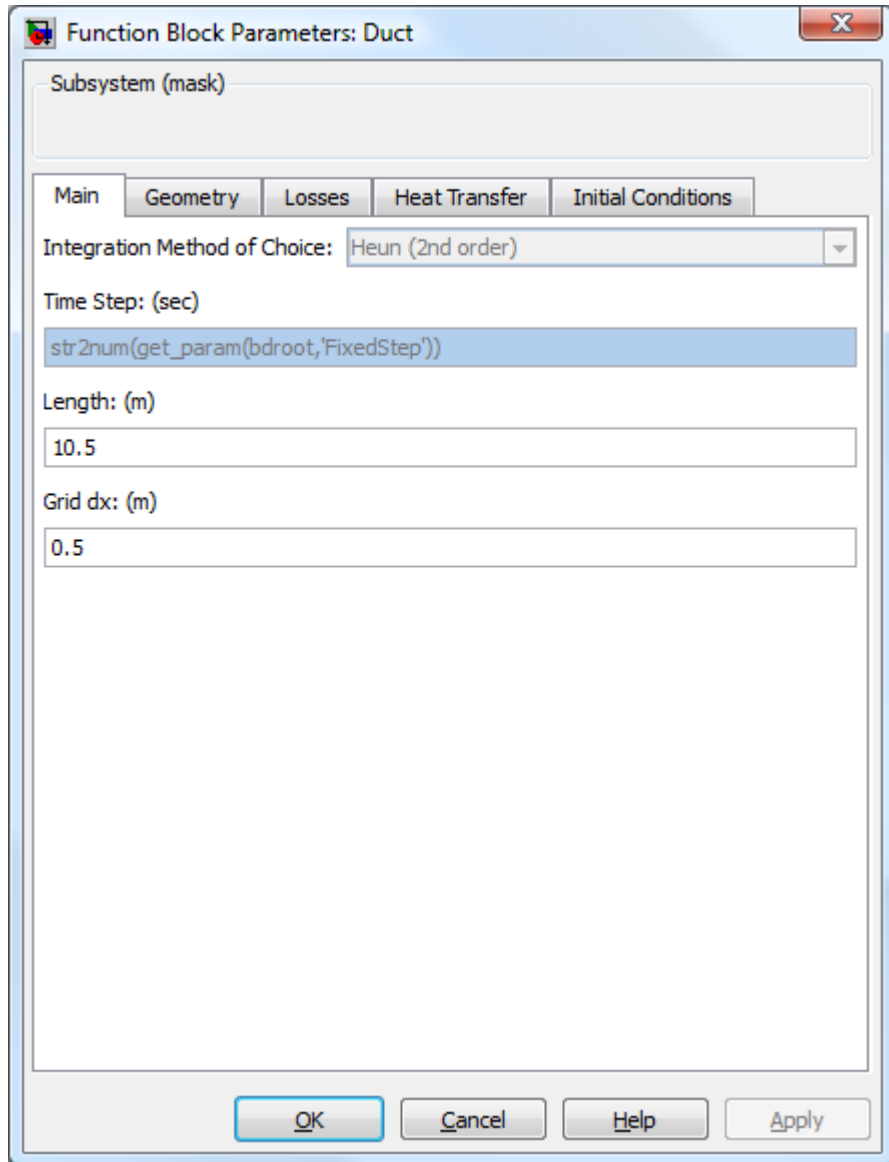


Figure 4.4 - The Main Screen of the Quasi-One-Dimensional Control Volume Block

sections. The first screen to appear is the Main screen. It allows the user to enter some basic information about the duct and simulation. The next screen, Geometry, is where basic geometrical information about the duct is entered. The next two screens contain optional information that can be turned off or on with a couple of check boxes. These screens allow the user to control how the code handles momentum losses and heat transfer. The last screen is where the user enters the initial states of the control volume.

4.2.1 The Main Screen

The first screen displayed when a user double-clicks the “Duct” control volume is the “Main” screen. There are four fields displaying information two of which are user inputs. The first is a disabled drop down menu. This area is reserved for future use. It is intended to allow the option of choosing different methods of integration. Currently, the only method implemented is Heun’s method but others could be added to include first, third or forth order schemes. The next field holds the time-step of the model. This field is intentionally locked because the time-step used by the program cannot differ from the time-step used by Simulink®. The command written in the text box will automatically determine the time-step used by the simulation. This will also benefit the user if he or she wishes to change the time-step. The control volume box will update itself without the user having to manually change this value. This is a nice feature if multiple control volumes are used throughout a model.

The first parameter to be entered is the overall length of the duct in meters. The last parameter on this screen is the grid spacing also in meters. This value defines the locations of all the cell faces. Figure 4.5 is used to illustrate how the spacing is set up. The solid circles represent cell centers

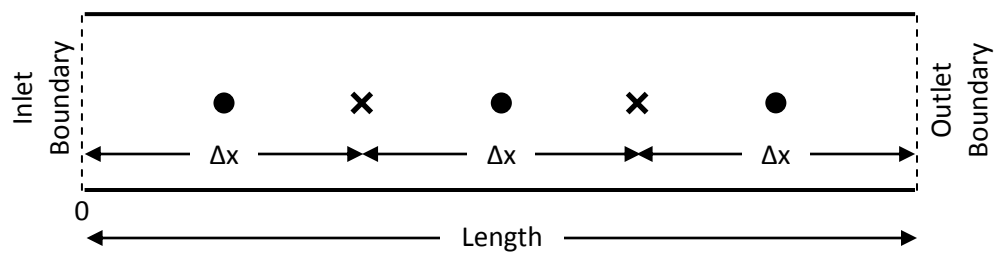


Figure 4.5 – Diagram Showing How the Grid Spacing is Defined

and the X's indicate a cell face. The inlet and outlet boundary conditions match where a cell face would belong and they can be thought of as phantom cells. The inlet is given as location $x = 0$ and the outlet is a distance defined by the length field. The number of cell centers is the length divided by the grid spacing and the number of faces is one plus the number of centers. This computation is automatically done by the program prior to execution.

4.2.2 The Geometry Screen

The next set up screen is used to define the duct geometry. Figure 4.6 shows what this screen looks like. The first option is a pull down menu so that the user can select how the duct cross sectional diameter is to be read in. There are three choices: the diameter is constant for the entire length of the duct and is given as a constant value, the diameter can be defined by an equation as a function of x , or values of the diameter can be input from a file. The next field is where the user enters the constant diameter, an equation, or the name of a file. The cell faces are used as the reference point for defining the duct's diameter and start at $x = \Delta x$ and ends at $x = duct\ length - \Delta x$. Care must be taken when using a file for the diameters of the cell faces because their number must equal the number of cell faces calculated by the program. The interface screen will check for consistency and will generate an error if the two do not match. The convention for using a file is to enter the name of the file followed by .txt. Figure 4.7 is an example of how a file named diameters.txt is set up. Any other inputs that are read from a file must follow the same format.

For this particular example, there are only six lines in the file so there must be the same number of cell faces. Halfway down the length of the duct the diameter changes from 1 meter to 1.5 meters.

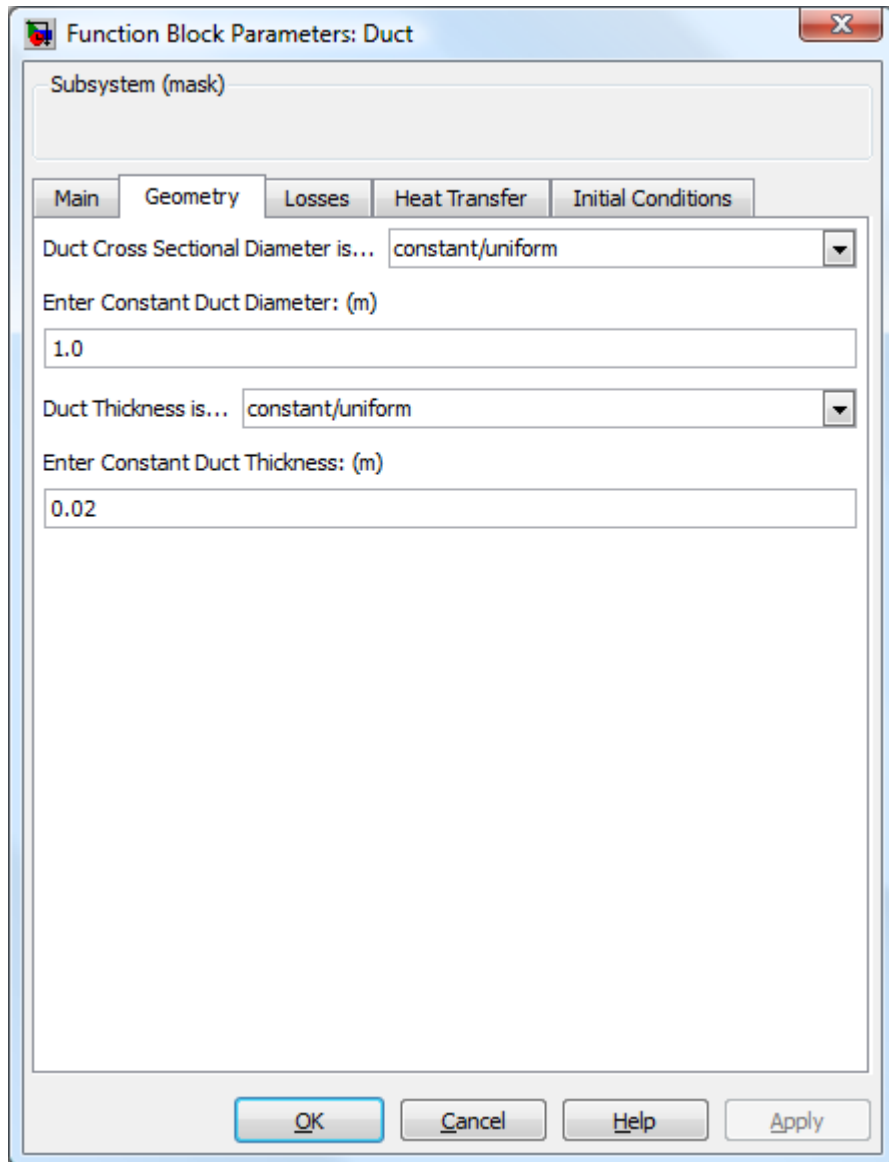


Figure 4.6 - The Geometry Screen of the Quasi-One-Dimensional Control Volume Block

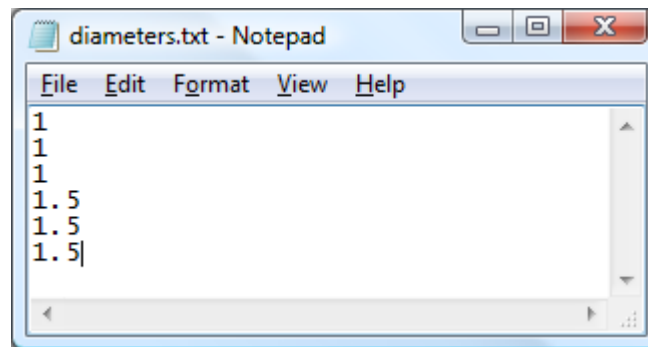


Figure 4.7 - Example Input File to be Read in to Define the Diameter Distribution of the Duct

The transition in-between is assumed to be linear and the area calculated at the cell center is linearly interpolated from the adjacent cell faces. The geometry of the cell in this transition is assumed to be a conical frustum. Discontinuities can exist in the slopes of the geometry; however, the effects of slope discontinuities on the fluid flow are not captured by the quasi-one-dimensional flow equations.

The last two user inputs define the thickness of the duct. It has the same choices as in the specificity of the duct diameter: defined as a constant value, described by an equation, or read from a file. The thickness is only used if heat transfer effects wish to be taken into account and from equation 3.57 we see that this only takes place at a cell center. Thus, thickness is defined at the cell centers rather than the cell faces.

4.2.3 The Losses Screen

The third set up screen, “Losses”, allows for the inclusion of momentum losses due to friction and minor losses such as bends. The first option on this screen is a checkbox that will tell the program to calculate these losses or to exclude them. When checked, the rest of the fields on this screen become active and when unchecked they are disabled. Next, the user can define the surface roughness used by equation 3.44 to calculate the friction factor. It is assumed that the roughness is constant throughout the duct and it is not allowed to vary. The last two fields define K-factor values used in equation 3.48 and their corresponding x locations. Both of these parameters must be given in an array format. Figure 4.8 includes an example of how the user might set this up. The first array has three K-factor values and the second array gives their x locations. K-factors are given at x locations 1, 3, and 7 meters from the duct inlet. The momentum equation used, Eq. 3.51, is defined referenced to the cell faces, so any K-factors values used must be given at those locations. A check is done to determine if the

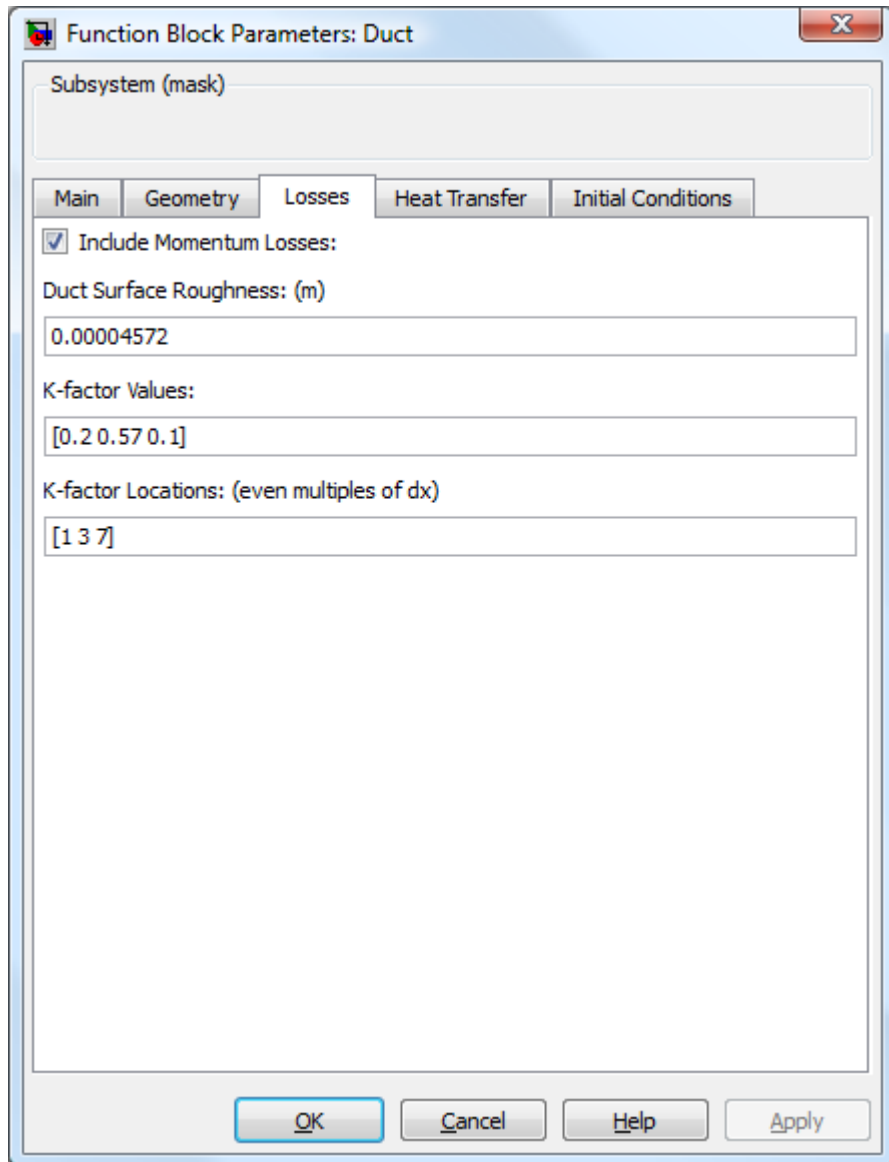


Figure 4.8 - The Losses Screen of the Quasi-One-Dimensional Control Volume Block

locations given are multiples of Δx , if not an error is returned to the user. If no minor losses are to be included then any size array with zeros can be given for the K-factor values.

4.2.4 The Heat Transfer Screen

The “Heat Transfer” screen, Figure 4.9, allows the user to enter in constants and atmospheric conditions used to calculate heat transfer rates to, or from, the ducting. The first option is a checkbox that gives the user the option of turning on, or off, these calculations. When checked the rest of the fields are active and available for input. If left unchecked the rest of this screen is inactive.

The next three inputs are properties of the duct material: the specific heat, density, and emissivity of the internal surface. These values are assumed to be constant throughout the length of the duct and that the emissivity does not vary with the duct temperature. The last three inputs are the solar flux, cross flow velocity, and the ambient air temperature. All three of these parameters have the options of having their values set to a constant or read from a file. If they are being read from a file, the number of lines has to equal the number of cell centers. Having them read from a file enables the user to account for ducting that is partially in shade, covered, or is indoors. The solar flux is used in equation 3.35 to add energy into the ducting. The external cross flow, if non-zero, is used to compute the Reynolds number in equation 4.28a. If this value is zero, then equation 3.28b will be used instead to compute the heat transfer by natural convection. The ambient temperature is used in equations 3.26 and 4.33 to compute the heat transfer due to convection and radiation respectively.

Function Block Parameters: Duct

Subsystem (mask)

Main Geometry Losses Heat Transfer Initial Conditions

☒ Include Heat Transfer:

Specific Heat of Duct Material: (J/kg-K)

500.0

Density of Duct Material: (kg/m³)

8000.0

Emissivity of Duct Surface:

0.585

Solar Flux is... uniform

Enter Uniform Solar Flux (W/m²):

680.0

Cross Flow Velocity is... uniform

Enter Uniform Cross Flow Velocity: (m/sec)

2.0

Ambeint Air Temperature is... uniform

Enter Uniform Ambeint Air Temperature: (K)

288.0

OK Cancel Help Apply

Figure 4.9 - The Heat Transfer Screen of the Quasi-One-Dimensional Control Volume Block

4.2.5 The Initial Conditions Screen

The “Initial Conditions” screen, Figure 4.10, is where the user specifies the initial conditions describing the flow inside the ducting. Here the user enters in the initial density, temperature and velocity of the fluid as well as the initial temperature of the ducting material. Each of these parameters has the ability to be entered in as a uniform value, computed from an equation or read from a file. These values correspond with cell center locations. To compute the initial mass flow at the cell faces, properties such as density and velocity are averaged from the adjoining cell centers. The program will perform checks on the initial values to ensure that the densities and temperatures are all greater than zero to minimize the chance of improper values being used as inputs.

4.3 Organization of the Code

The code is structured into two main routines: the initialization of flow variables and the solver routine which employs Heun’s method to solve for the primitive variables at each time-step. The solver calls other subroutines to compute the derivatives which in turn makes other calls to calculate the heat fluxes and friction factors. The initialization routine takes advantage of the built-in API commands that the Simulink® engine uses to interact with the S-functions. The S-function provides a predefined function called “mdlStart” which is called once prior to the start of model execution. The function can be used to initialize states or perform any calculations that need to take place once. This is where all preprocessing of the data associated with the duct takes place. The other major function is called “mdlOutputs” and this is where the integration of the equations of motion takes place along with computing the outputs that go to the upstream and downstream flow devices.

Function Block Parameters: Duct

Subsystem (mask)

Main Geometry Losses Heat Transfer Initial Conditions

Initial Density is... uniform

Enter Uniform Density: (kg/m³)

1.225

Initial Temperature is... uniform

Enter Uniform Temperature: (K)

288.0

Initial Velocity is... uniform

Enter Uniform Velocity: (m/s)

0.0

Initial Duct Temperature is... uniform

Enter Uniform Duct Temperature: (K)

288.0

OK Cancel Help Apply

Figure 4.10 - The Initial Conditions Screen of the Quasi-One-Dimensional Control Volume Block

4.3.1 Initialization

Initialization happens at the start of model execution in the routine “mdlStart”. Several persistent arrays are created to store values for derivatives, values calculated at intermediate time-steps, information about the geometry and material properties of the duct, and values that describe the initial properties of the fluid. Arrays for the volumes and surfaces areas are the first to be calculated. If a single cell center is examined, Figure 4.11, it forms a conical frustum, assuming that there is a change in area from one side of the cell to the other. Equation 4.1 is used to compute the cell volume and placed at the cell center. If d_i and d_{i+1} denote the diameters of the cell faces and Δx is the grid spacing, then

$$volume = \frac{1}{3}\pi\Delta x \left[\left(\frac{d_i}{2}\right)^2 + \left(\frac{d_i d_{i+1}}{4}\right) + \left(\frac{d_{i+1}}{2}\right)^2 \right] \quad \text{Eq. 4.1}$$

Equation 4.2 is used to define the surface area of the inside of the ducting. The same formula can be used to find the outside surface area, but with twice the duct thickness added to the diameters.

$$surface\ area = \pi \left(\frac{d_i}{2} + \frac{d_{i+1}}{2} \right) \sqrt{\left(\frac{d_i}{2} - \frac{d_{i+1}}{2} \right)^2 + \Delta x} \quad \text{Eq. 4.2}$$

If the diameters are the same, equation 4.1 and 4.2 reduce to the volume and surface area of a cylinder respectively. Next to be computed are the areas of the cell faces and cell centers. Part of the setup outlined in section 4.2.2 is to provide the diameters of the cell faces and this is directly used to compute their areas. The areas of the cell centers are computed indirectly from the average of the two cell faces that encompass a center.

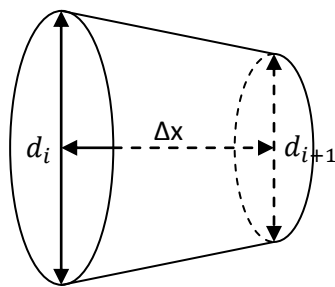


Figure 4.11 – Section of Ducting that Forms a Conical Frustum

The initial mass and energy associated with each cell center can be computed using the volumes calculated from equation 4.1 and from the initial conditions provided at the interface screen in section 4.2.5. Equations 4.3 and 4.4 are used to calculate the initial mass and energy respectively.

$$M_I = (V\rho)_I \quad \text{Eq. 4.3}$$

$$E_I = (V\rho)_I \left(\frac{R}{\gamma-1} T_I + \frac{1}{2} V_I^2 \right) \quad \text{Eq. 4.4}$$

The pressure at each cell face is calculated using the ideal gas law from equation 3.58. The mass flow at each cell face is initialized by taking the average of the product of density and velocity at the surrounding cell centers and multiplying by the cell face area. This is given by equation 4.5.

$$\dot{m}_i = \frac{1}{2} [(\rho V)_{I-1} + (\rho V)_I] A_i \quad \text{Eq. 4.5}$$

Next, the thermal capacitance of the ducting is pre-calculated using equation 4.6 and stored in an array to be used during heat transfer calculations.

$$capacitance = \rho_{duct} c_{duct} \left[\pi \Delta x \left(\frac{1}{2} d_i \Delta \tau + \frac{1}{2} d_{i+1} \Delta \tau + \Delta \tau^2 \right) \right] \quad \text{Eq. 4.6}$$

where ρ_{duct} is the density of the duct material, c_{duct} is the specific heat, and $\Delta \tau$ is the thickness. The expression in brackets is the volume of the duct material at a cell center. The remaining arrays to get initialized are place holders for intermediate calculations and derivatives at major and minor time-steps. These quantities are initialized to zeros.

4.3.2 Integration

The integration of equations 3.35, 3.50, 3.51, and 3.56 take place in the mdlOutputs routine. Figure 4.12 provides a flowchart of how the main routine is organized. It is split into parts to best follow Heun's method. The first set of calculations finds the time derivatives of mass, mass flow, energy, and the temperature of the ducting if heat transfer is included. These values are then integrated according to equation 3.67a to find their preliminary or predicted values. The second grouping of calculations computes their corrected values at time $n + 1$. This is done by calculating the time rate of change of their predicted values and then integrating them using equation 3.67b.

The time rates of change are put into the subroutines "getFaceDerivs" and "getCellDerivs." The subroutine getFaceDerivs returns the time derivative of mass flow at the cell faces and "getCellDerivs" returns the time derivatives of mass and energy at each cell center. They in turn have subroutines to calculate friction factors and the amount of energy transferred between the fluid and the ducting.

4.3.3 Time-rate of Change of Mass Flow

Figure 4.13 is a flowchart of the subroutine "getFaceDerivs" which is called by "mdlOutputs" to calculate the time rate of change of mass flow at the cell faces. Based on the current step of Heun's method, it either uses values of mass flow, pressure, temperature, velocity, and density at the current time-step or it uses their predicted values at time $n + 1$. If the Include "Momentum Losses" checkbox is checked, see Figure 4.8, it will call "getFriction" to compute the Darcy friction factors needed to account for frictional losses. Equations 3.51 through 3.55 are then used to compute the time derivatives of mass flow.

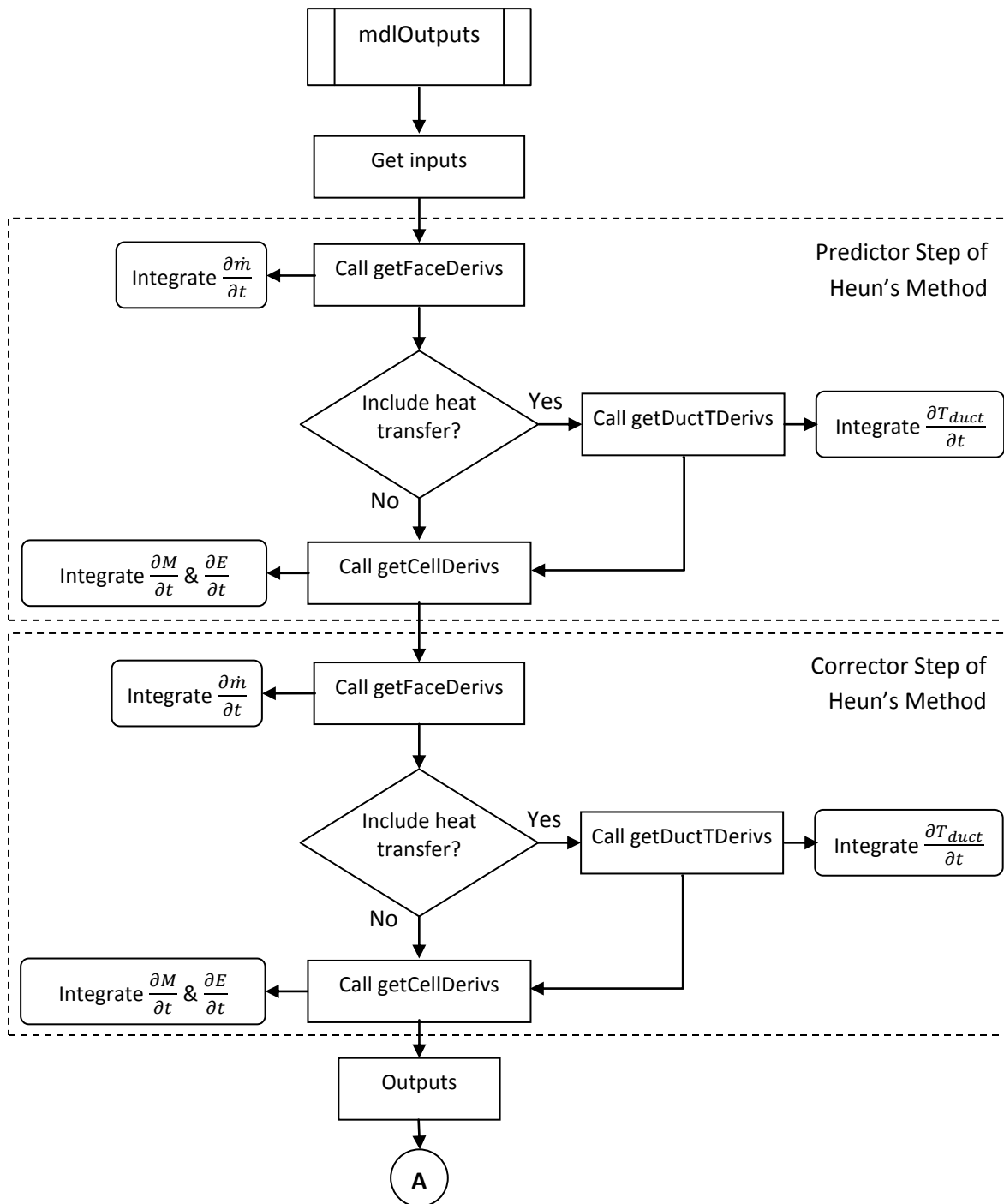


Figure 4.12 - mdlOutputs Routine of OneDCV

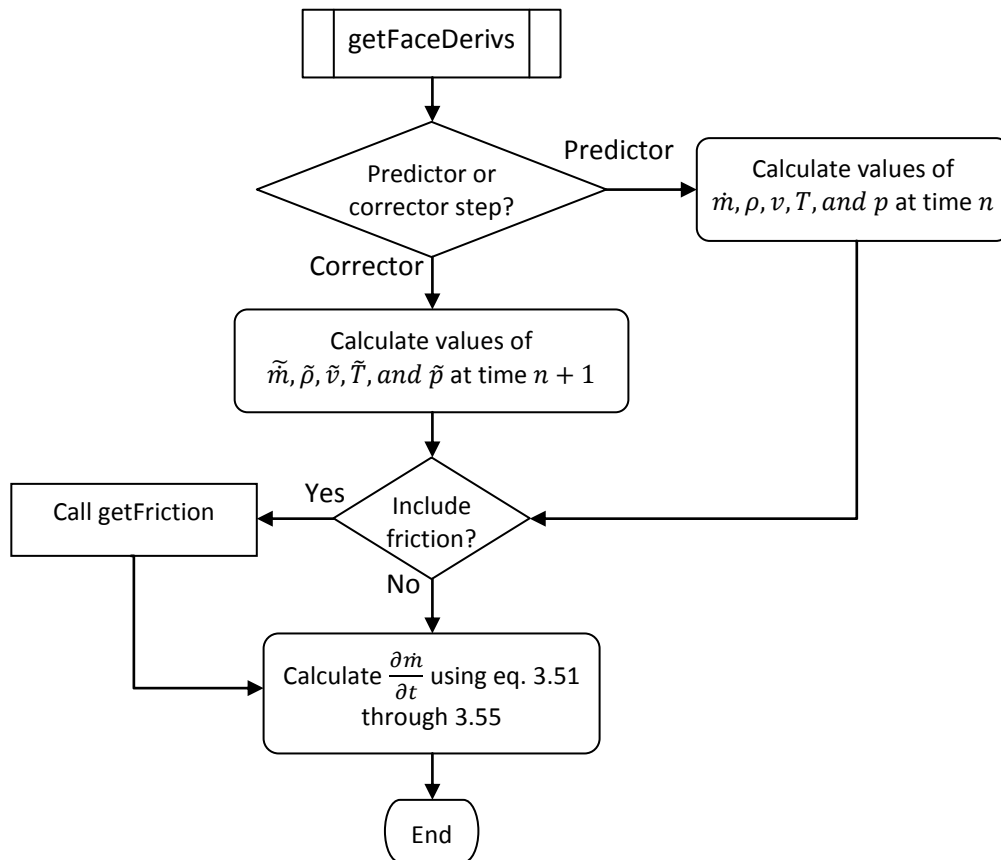


Figure 4.13 - Calculation of the Derivatives at the Cell Faces

4.3.4 Time-rate of Change of Mass and Energy

Time derivatives of mass and energy are calculated by the subroutine “getCellDervis,” see Figure 4.14 for a flowchart outlining the code. At each cell face either current values of mass flow are used or predicted values at time-step $n + 1$ are used. The same is true for pressure, temperature, velocity, density and the ducting temperature at each cell face. If heat transfer effects are included in the calculation, then the subroutine “getHeatXfer” is called to calculate the amount of energy transferred between the fluid and the ducting walls. The time rate of mass is then computed using equation 3.50 and the change in energy is found by equation 3.57.

4.3.5 Time-rate of change of Ducting Temperature

When heat transfer effects are included, “mdlOutputs” will call “getDuctTDerivs” to get the time-rate of change of the ducting temperature at each cell center. See Figure 4.15 for a flowchart depicting the code for this subroutine. Depending on the current step of Heun’s method it either uses current values of mass flow, pressure, temperature, density, and ducting temperature or it uses their predicted values at time-step $n + 1$. If the ambient conditions include an external wind then equation 3.28a is used to compute the Nusselt number to account for cross flow or it will default to equation 3.28b calculating the Nusselt number for natural convection. When calculating the heat transfer coefficient inside the ducting, the program will check to see if the flow is laminar or turbulent based on the Reynolds number. As a general rule of thumb, a fluid is considered to be laminar if the Reynolds number is less than 2,400 and turbulent if it is greater than 4,000. The transition region between laminar and turbulent flow is not well documented and in most applications the Reynolds numbers of interest are much greater than this so the code treats turbulent flow starting at 2,400. If the flow is

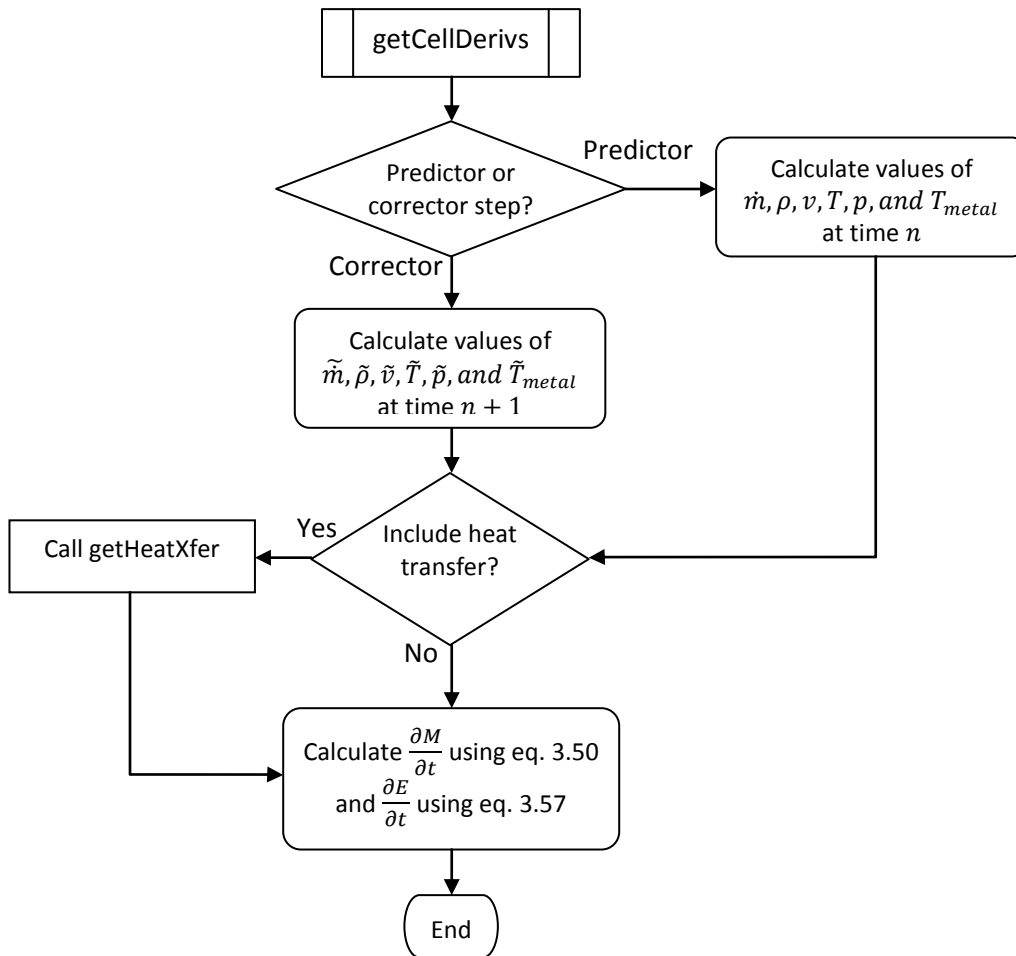


Figure 4.14 - Calculation of the Derivatives at the Cell Centers

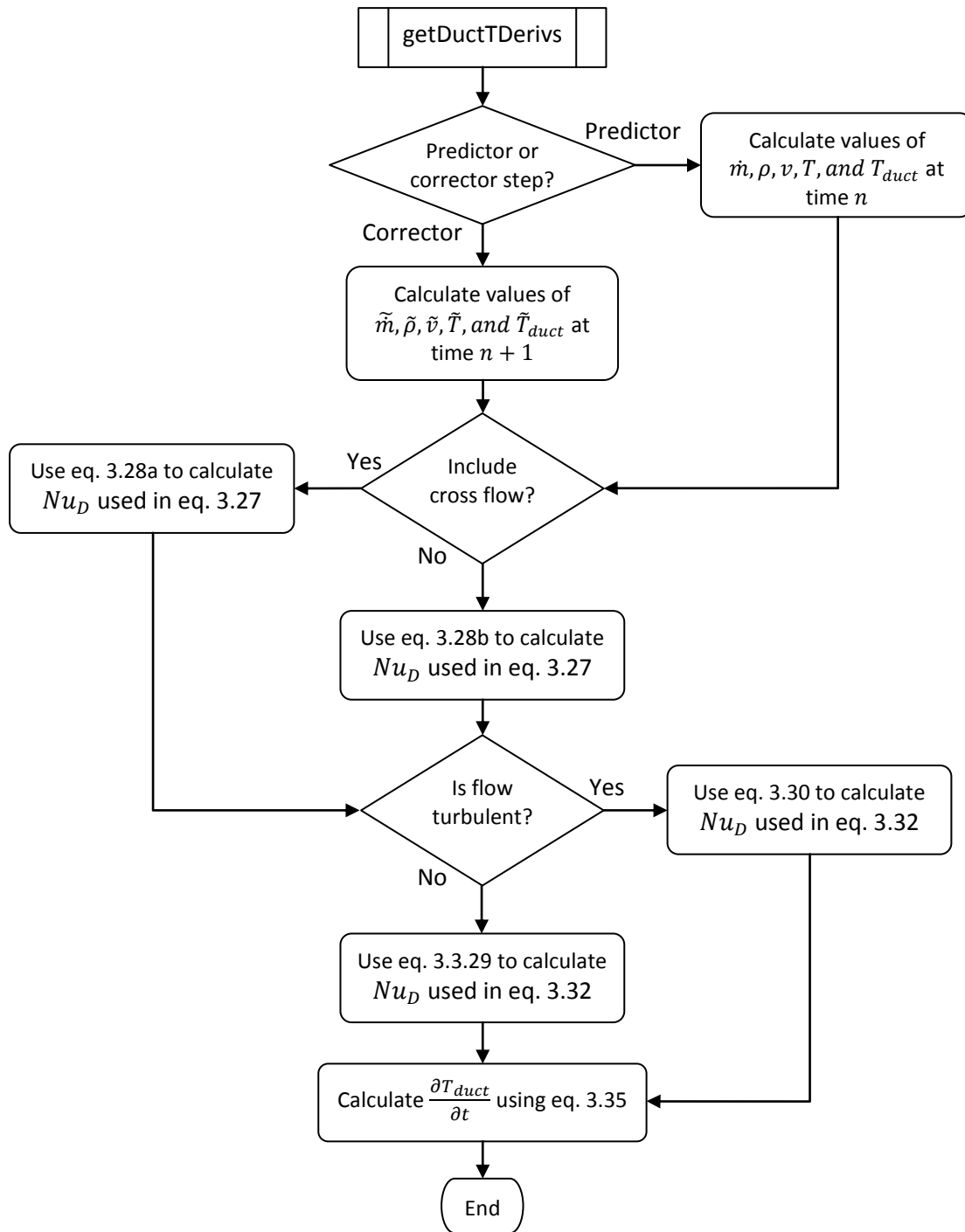


Figure 4.15 - Calculation of the Time Rate of Change of the Duct Temperature

determined to be laminar then equation 3.29 is used to compute the Nusselt number otherwise it uses equation 3.30 for turbulent flow. Lastly, the program uses equation 3.35 to calculate the derivative of ducting temperature with time.

4.3.6 The Darcy Friction Factor

The subroutine “getFriction,” see Figure 4.16, returns the Darcy friction factor when called. It performs a check on the Reynolds number to determine if the flow is laminar or turbulent. If the Reynolds number is above 2,400 then the fluid is assumed to be fully developed turbulent flow and equation 3.43 is used to compute the friction factor. If the flow is laminar it will default to equation 3.42.

4.3.7 Heat Transfer between the Fluid and Duct Wall

The subroutine “getHeatXfer,” see Figure 4.17, calculates the energy flux between the fluid in the duct and the wall. This routine is called by “getCellDerivs” if heat transfer is included in the calculations. It takes into account heat transferred by forced convection and radiation. If the flow is laminar it uses equation 3.29 to compute the Nusselt number and equation 3.30 if it is turbulent. Equation 3.36 is then used to calculate the heat flux.

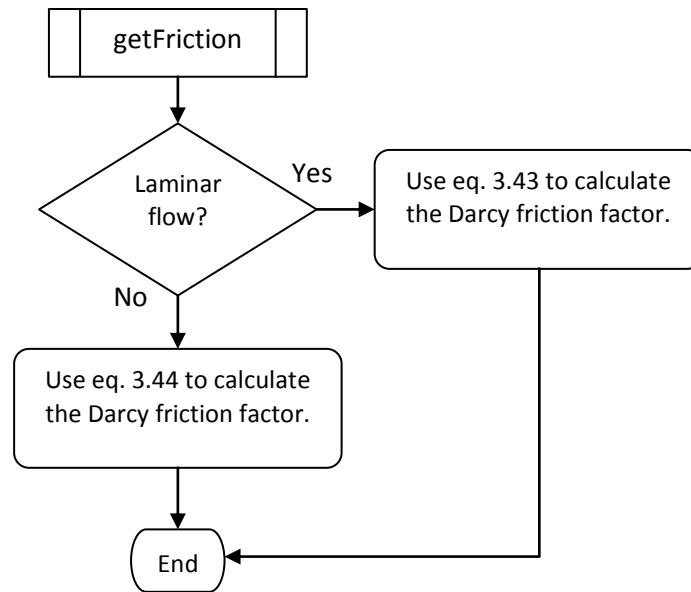


Figure 4.16 - Calculation of the Darcy Friction Factor

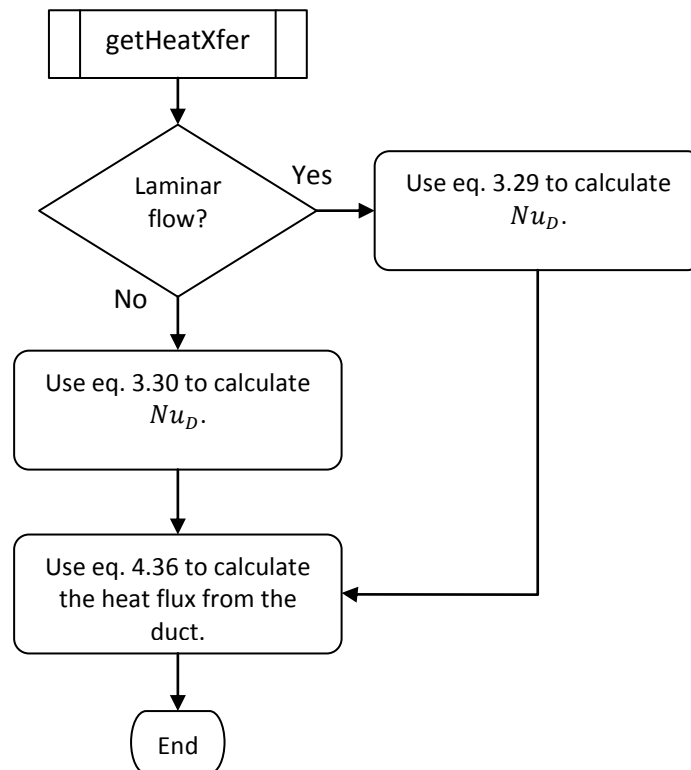


Figure 4.17 - Calculation of the Duct Heat Flux to the Flow

5. Results and Comparison with Theory

The quasi-one-dimensional finite volume code developed herein has many new features that are not part of the existing lumped-parameter control volume. For example the new code is able to account for changes in duct cross sectional area, friction, other minor losses, and localized heat transfer effects. This section details the results of numerical solutions demonstrating these new features and makes comparisons with theory where applicable. Obtaining experimental data for these effects is often difficult due to the cost and the time constraints placed by normal facility operations. Excellent data gathering capabilities exists in the test sections of present day facilities because that is where the greatest interests lie. However, a few sensors are placed in the ducting that provides the airflow to the test section but these primarily provide feedback for the control systems and are not suited for data collection due to low sampling rates.

In this chapter each feature flow is examined (area change, bends etc.). Analytical solutions do not exist that incorporate the effects of friction, heat transfer, and changes in area simultaneously except in rare special cases. However, analytical solutions do exist for each effect in isolation. The steady-state solution for ducting that has changes in cross sectional area can be compared to isentropic flow predictions (if the boundary layer development is neglected). Constant area duct friction can be analyzed with Fanno flow analysis and heat transfer can be treated with Rayleigh flow. To verify that the code properly handles transients, the oscillations of a wave propagating down the length of a duct can be compared to the natural resonance for an open ended tube. Lastly, the code is compiled to be able to run real-time using Matlab's xPC Target and execution times are gathered for varying time-steps to be able to determine limits on the number of cells the code is able to process in real time. These limits can place a constraint on the time-step and grid sizes used so results are also presented using

realistic values for these parameters. The analysis performed is within the subsonic limitation of the code. Due to the staggered grid and the averaging of unknown properties at cell faces and centers the code is unstable for Mach numbers one and greater. This limits the applicability of the code to modeling ducting of a test facility where Mach numbers are less than one. Thus test sections are excluded. However, since the code is designed to run in real-time, modeling super-sonic flows may be unrealistic because of grid sizing and time-step limitations. Very dense grids are needed for shock capturing and due to stability requirements this forces the time-step to become much shorter [11].

5.1 Flow with Area Change

One of the main advantages of the new quasi-one dimensional control volume over the lump-parameter control volume, described in section 1, is the ability to account for changes in flow area. The ducting work that comprises a typical test facility changes area frequently. Unless the only parameters of interests are total properties then the assumption of a constant area is not adequate. If the local Mach number within a section of ducting is in the compressible flow regime then properties like static pressure and temperature will be affected by area changes. Having the mechanism to account for these changes can be advantageous to the controls engineer or designer, especially if local Mach number is of concern.

To test the accuracy with which the quasi-one dimensional code takes into account changes in area, steady-state results are compared to predictions from the isentropic flow equations. The geometry used in this comparison is given in Figure 5.1. The first ten meters of the duct has a constant area of 1 m^2 , and then over the next two meters the area is linearly halved and remains constant for the

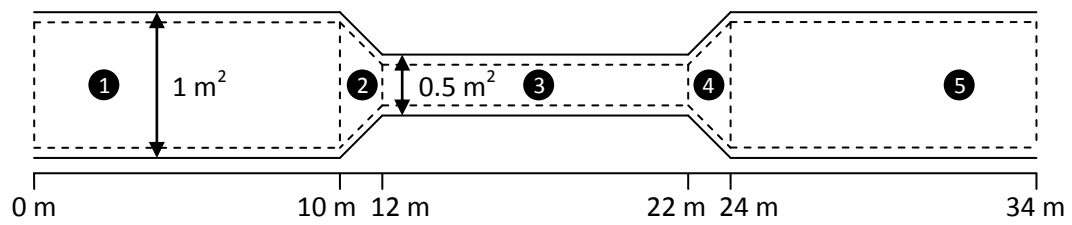


Figure 5.1 - Geometry Used for Varying Area Comparison

next ten meters. It then linearly expands back to the original area. Under steady state conditions the flow properties in the first constant area section, labeled section 1 in the figure, will remain constant. The same is true of the properties in sections 3 and 5. The contraction in section 2 will cause Mach number to increase and static pressure and temperature to decrease, it is assumed that the Mach number will be subsonic before entering the contraction. The expansion in section 4 will cause the Mach number to decrease and the static pressure and temperature to increase. In all five sections the total pressure and temperature should remain constant [14]. Figure 5.2 shows the Simulink® diagram of the model used to simulate adiabatic flow in the geometry described in Figure 5.1. The control volume is placed between two valves that regulate the amount of mass flux and energy flux into and out of the control volume based on the pressure differential across each valve. Upstream of Valve 1 is a constant pressure source and downstream of Valve 2 is a constant pressure sink. Based on these pressures, flow is forced through the control volume. Initially, Valve 1 is closed and Valve 2 is open. Once the simulation starts Valve 1 begins to open at a constant rate until after five seconds it is fully open. Both valves have the same area as the duct inlet and outlet. At time zero there is no flow in the duct; velocity has been uniformly set to zero and the temperature set to 288.0 Kelvin and the density set to 1.225 kg/m³. Using the equation of state, this gives an initial pressure of 100.62 kPa. The pressure sink is set to these same conditions, so that with Valve 2 open there is zero flow in the duct until Valve 1 is opened.

The first case tested demonstrates the capability of the code to perform calculations in the very low speed regime with Mach numbers ranging from 0.026 to 0.052. To achieve such low Mach numbers the upstream pressure source was set to 100.82 kPa. Figures 5.3, 5.4, and 5.5 show how Mach number, static pressure and temperature compare to isentropic flow theory. Equation 5.1 was used to solve for the local Mach number based on the on mass flow $\dot{m}$, the flow area A , and the total pressure P_t , and

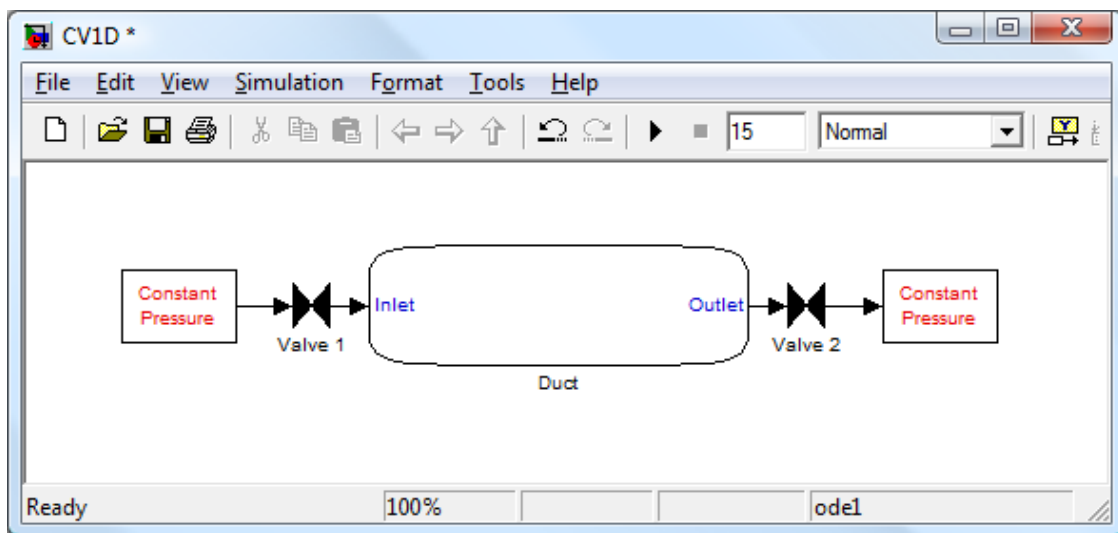


Figure 5.2 - Simulation Diagram

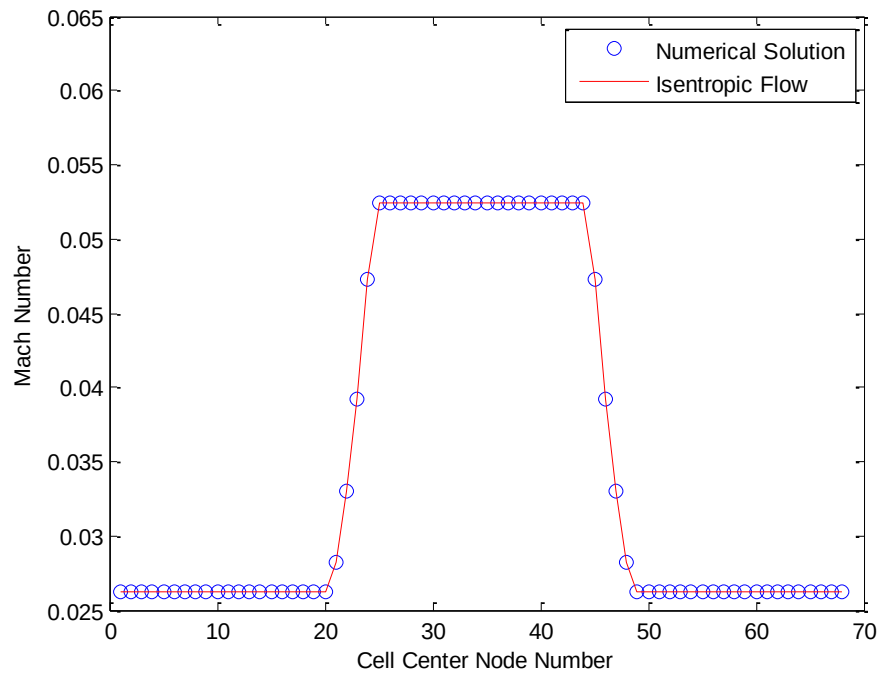


Figure 5.3 – Mach Number Comparison with Isentropic Flow Theory for Incompressible Flow

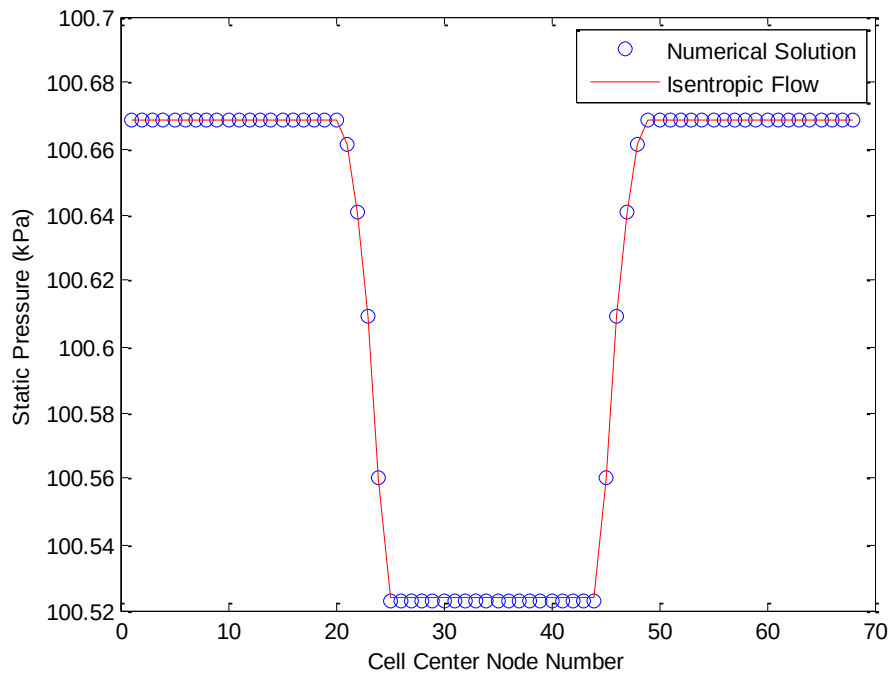


Figure 5.4 – Static Pressure Comparison with Isentropic Flow Theory for Incompressible Flow

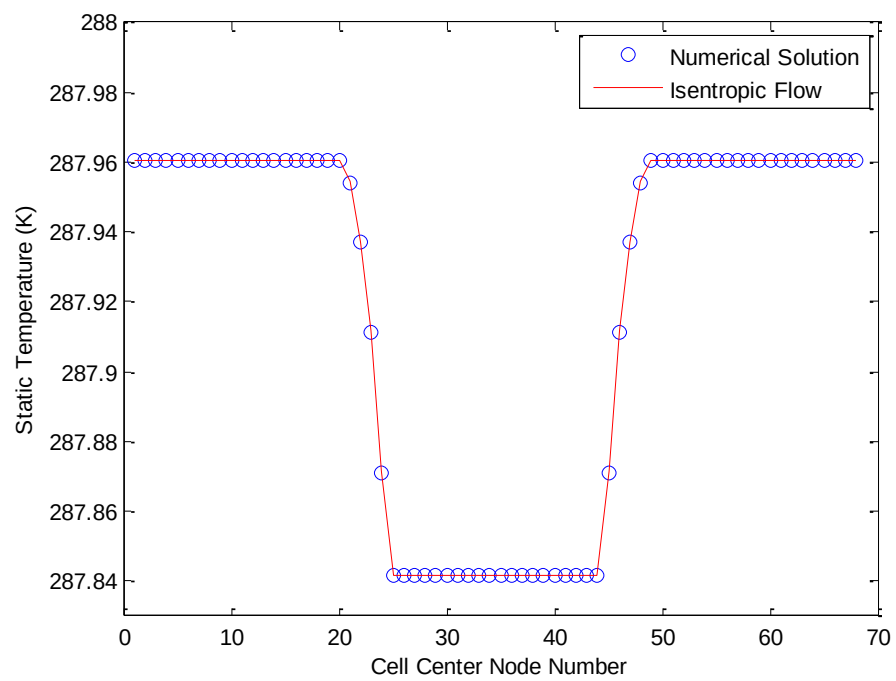


Figure 5.5 – Static Temperature Comparison with Isentropic Flow Theory for Incompressible Flow

total temperature T_t . The specific gas constant is given as R and γ is the ratio of specific heats. Equations 5.2 and 5.3 were used to solve for the static pressure and temperature respectively [10]. These equations were used to compute the theoretical flow properties at each cell center and a percent error was computed from each equation. The Mach number has a maximum error of 0.000838%, the static pressure an error of 0.000898% and the temperature an error of 0.000216%. Total pressure and temperature should remain constant. Figures 5.6 and 5.7 show the values along the length of the duct from the present computation. Total pressure has the highest amount of error with 0.0009% while temperature has an error of 0.0002%. Figure 5.8 displays how density varies along the length of the duct. The solution follows the predicted density quite well with a maximum error of 0.000234%. The appropriate equations are given as:

$$\dot{m} = P_t A M \sqrt{\frac{\gamma}{R T_t} \left(1 + \frac{(\gamma-1)}{2} M^2\right)^{\frac{\gamma+1}{1-\gamma}}} \quad \text{Eq. 5.1}$$

$$P = P_t \left(1 + \frac{\gamma-1}{2} M^2\right)^{\frac{\gamma}{1-\gamma}} \quad \text{Eq. 5.2}$$

$$T = T_t \left(1 + \frac{\gamma-1}{2} M^2\right) \quad \text{Eq. 5.3}$$

$$\rho = \rho_t \left(1 + \frac{\gamma-1}{2} M^2\right)^{\frac{1}{1-\gamma}} \quad \text{Eq. 5.4}$$

A second case was performed with the upstream pressure set to 114.99 kPa. This produced Mach numbers up to 0.48, as can be seen by Figure 5.9. This computation demonstrates the ability of the code to accurately produce results in the compressible flow range as well as in the incompressible flow range. Plots of static pressure and temperature are also created comparing the numerical results

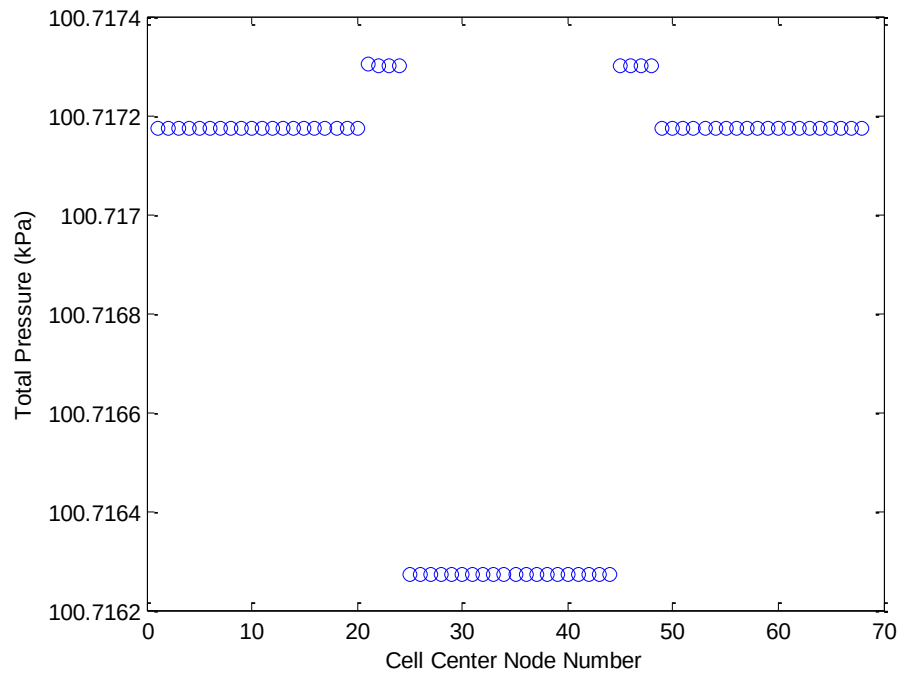


Figure 5.6 – Total Pressure Variation for Incompressible Flow

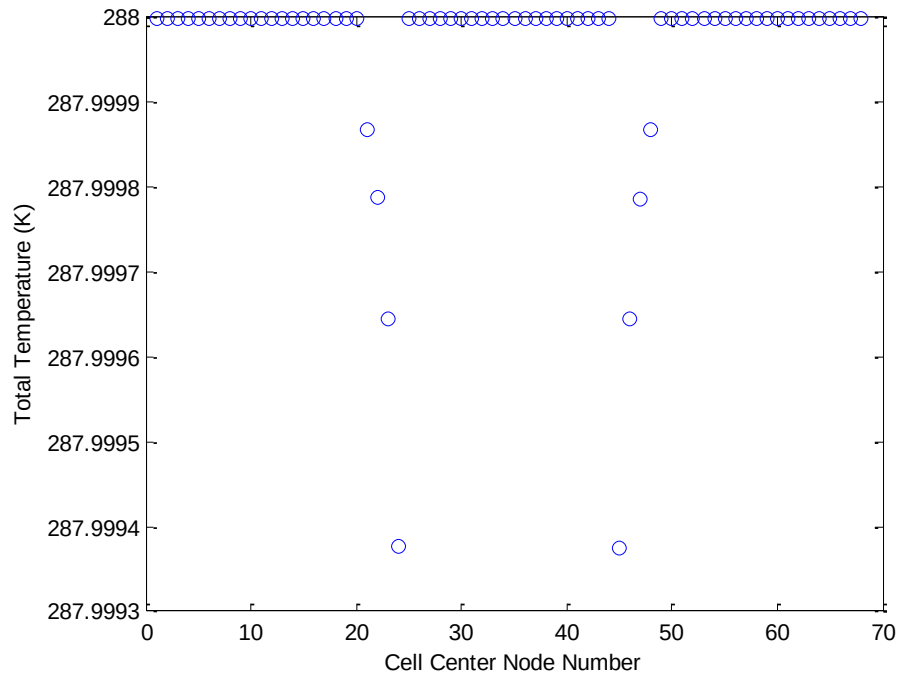


Figure 5.7 – Total Temperature Variation for Incompressible Flow

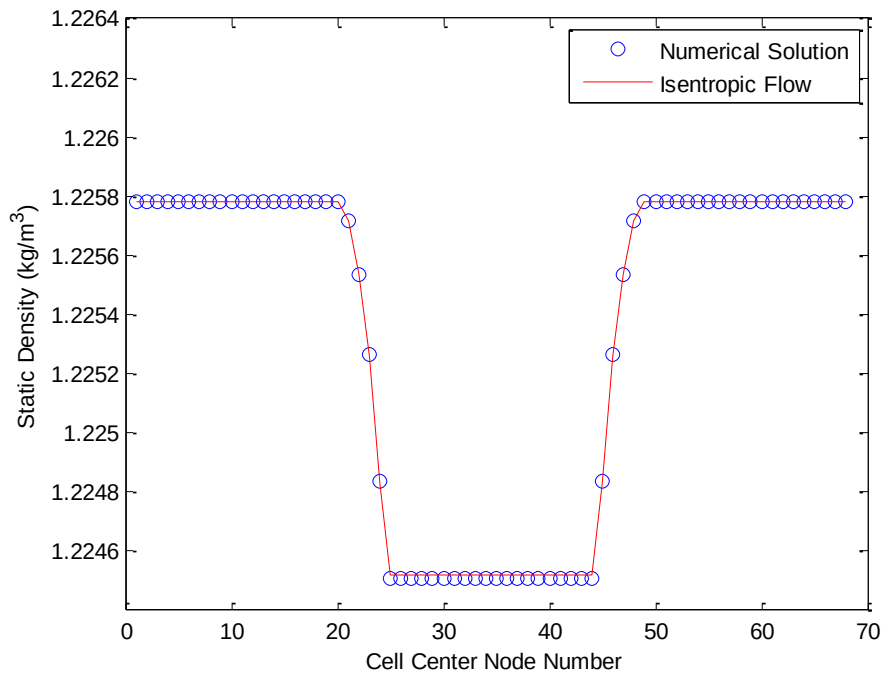


Figure 5.8 – Static Density Variation for Incompressible Flow

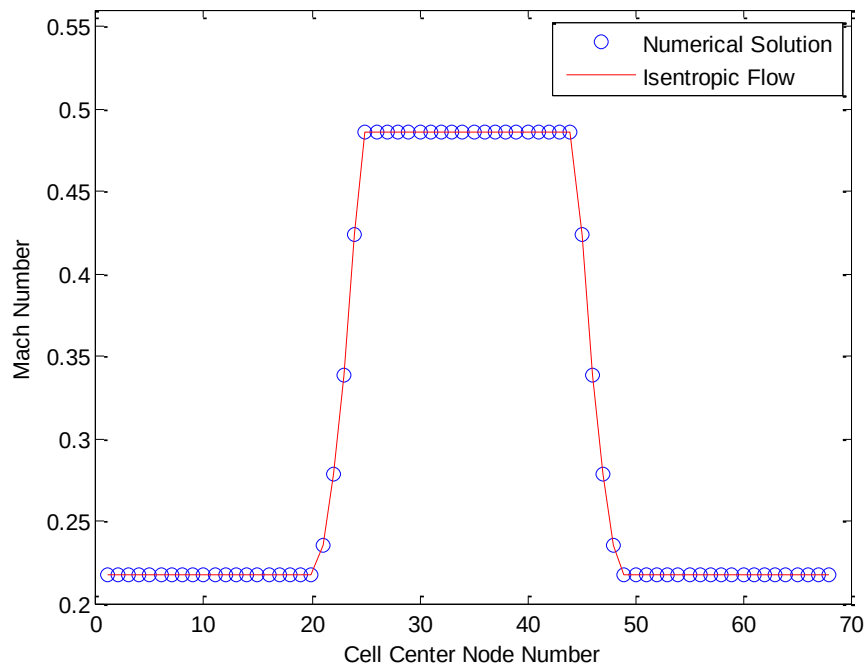


Figure 5.9 – Mach Number Comparison with Isentropic Flow Theory for Compressible Flow

to theory. In Figure 5.10, it can be seen that the static pressure starts to deviate from the isentropic equations in the contraction region and departs from it in the region where the area has been reduced by half. The maximum amount of error seen is 0.15% from what is expected by theory. The static temperature behaves similarly in Figure 5.11 with a maximum error of 0.014%.

In both of these test cases, the grid spacing was held constant at 0.5 meters with a time step of 0.0001 seconds. As Mach number is increased, the steady-state numerical results appear to be deviating further from theory. And indeed when the Mach number is further increased to approximately 0.9, the errors in Mach number, static pressure and static temperature increased to 2.96%, 3.25% and 0.76% respectively.

There are several reasons for this. The discretization adopted to solve the equations of motion in section 3 uses a staggered grid. Continuity and energy are kept track of at the cell centers whereas momentum is solved for at the cell faces. This unfortunately means that not all the primitive variables, such as density, temperature, velocity, etc, are known at both the centers and the faces. When a variable is needed at a location where it is not known, it is averaged using information from nearby nodes. If the conditions of the flow are such that strong gradients are produced, this form of averaging can introduce error. This is clearly seen in the example geometry in both the contraction and expansion regions where the velocity is changing rapidly. Figure 5.12 shows the change in Mach number for the Mach 0.9 case and within the span of 2 meters the Mach number has changed by 0.6. One method to overcome this is to use a tighter grid. By using more grid points, these errors are reduced during the averaging process. To test the validity of this, the Mach 0.9 case was run with successively finer grids to see how the error is reduced. Figures 5.13, 5.14, and 5.15 show how the error decays in Mach number, static pressure and static temperature, respectively, with refined grid size. All three plots show the

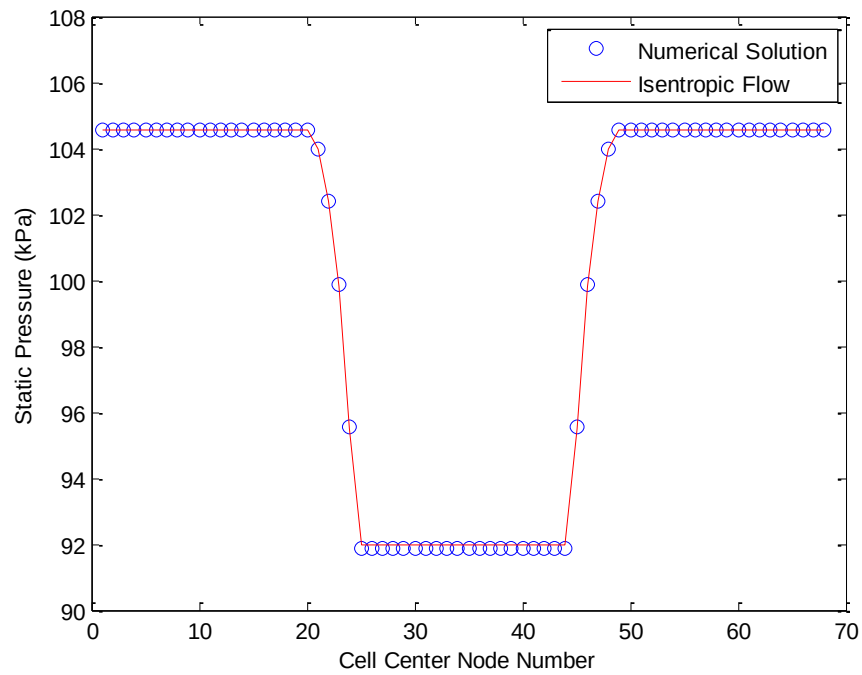


Figure 5.10 – Static Pressure Comparison with Isentropic Flow Theory for Compressible Flow

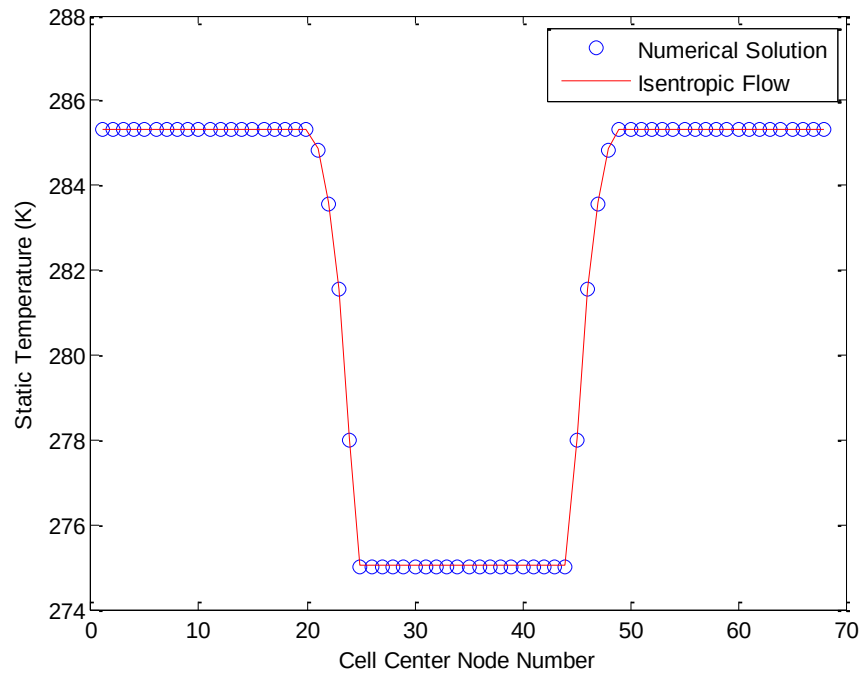


Figure 5.11 – Static Temperature Comparison with Isentropic Flow Theory for Compressible Flow

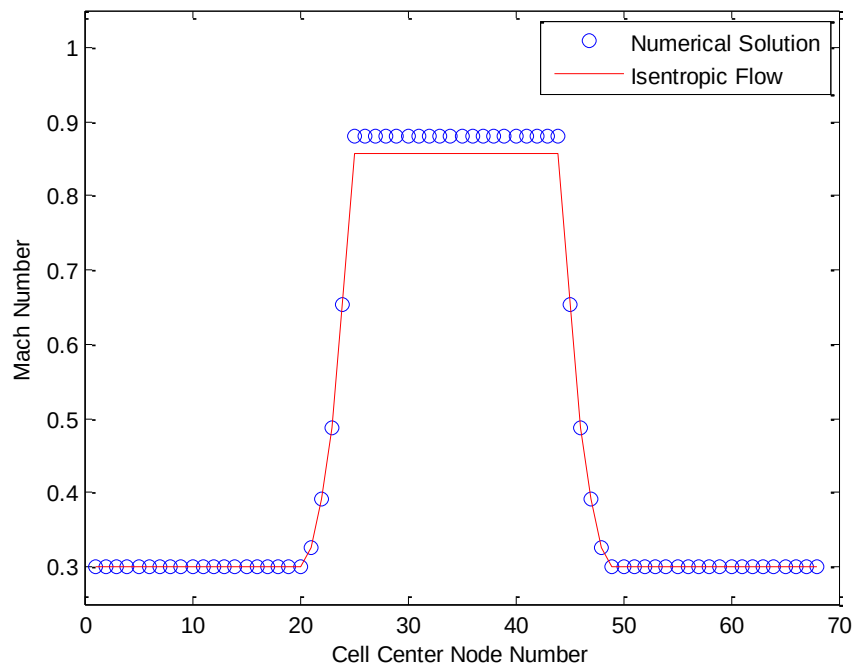


Figure 5.12 – Mach Number Comparison with Isentropic Flow Theory for Mach 0.9 Flow

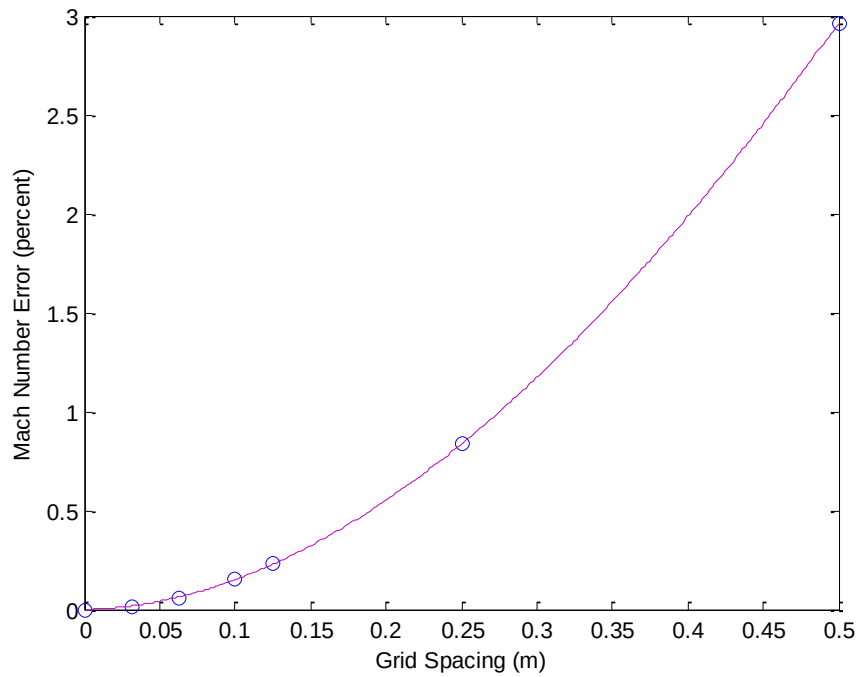


Figure 5.13 – Mach Number Percent Error for Varying Grid Sizes

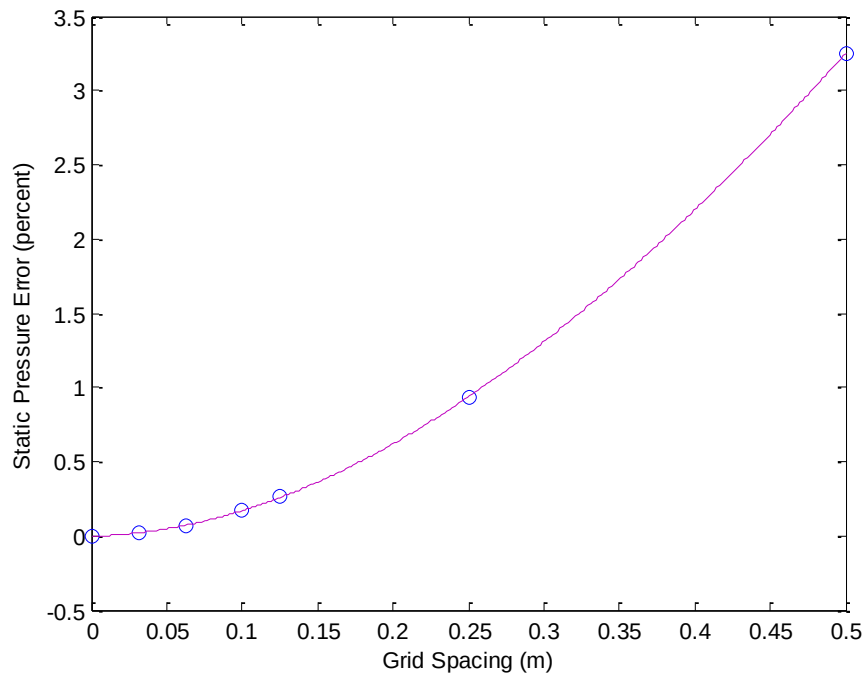


Figure 5.14 – Static Pressure Percent Error for Varying Grid Sizes

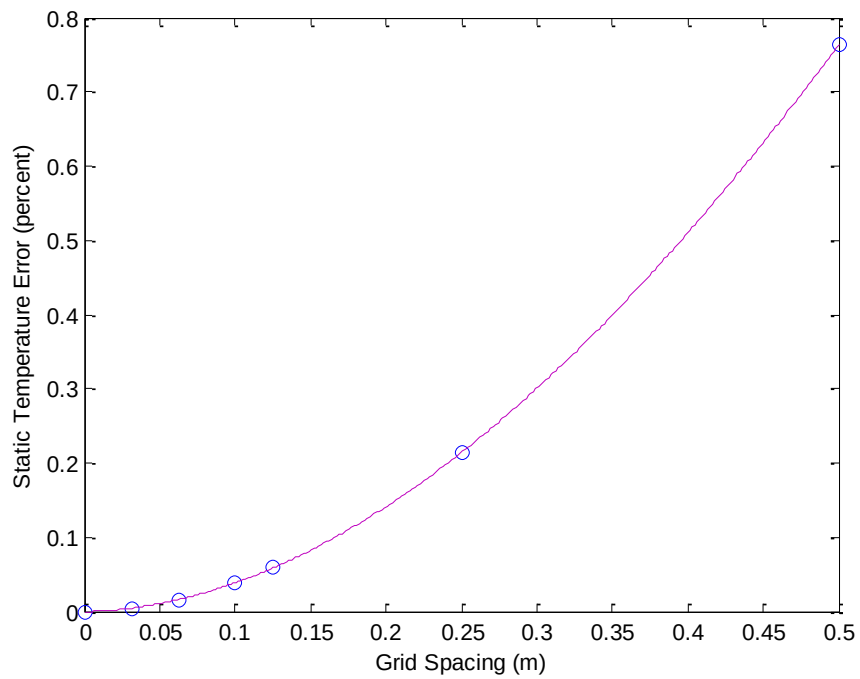


Figure 5.15 – Static Temperature Percent Error for Varying Grid Sizes

same trend of the error being reduced as the cube of grid size.

The steady-state results predicted by the code are in good agreement with the adiabatic isentropic flow equations and the code has been shown to work equally well for very low speeds to high subsonic speeds. In situations where strong velocity gradients exist, significant error can occur but this has been shown to be reduced if finer grid sizes are used. Typical duct Mach numbers are between 0.2 and 0.3, so the Mach 0.9 case that produced a strong gradient is unlikely to be seen in normal operation of the facilities. This will not be true for the test section of a facility where supersonic flow can exist.

5.2 Flow with Momentum Losses

The second advantage of the quasi-one-dimensional control volume method is the inclusion of the momentum equation. This allows for momentum losses such as wall friction and flow separation in duct bends to be directly included in the computation. The previous method lumped any losses at a flow device, like a valve, but these losses were often a constant that did not vary with facility conditions or time [7]. Furthermore, because losses were located at a flow device and not at their physical location, it further detached the simulation from actual facility phenomena. The new quasi-one-dimensional model overcomes these previous short falls.

Two test cases were run to demonstrate the capability of the new code to compute accurate solutions which include frictional losses and a minor pressure loss. The first case assumes a constant duct diameter, a length of two-hundred meters, and a grid spacing of 0.5 meters. The upstream pressure source was set at 114.99 kPa and the downstream pressure sink was set to 100.62 kPa. The initial conditions of the fluid in the duct were the same as those used in section 5.1. The simulation was

run for a significant length of time to allow the solution to reach steady-state. This allowed for the numerical solution to be compared to the Fanno flow equations. In order to fully compare the two solutions, the friction factor was held at a constant value of 0.009572. This value was extrapolated from a table in Crane's handbook [15] that provides friction factors for fully turbulent flow in various pipe diameters. Figures are made comparing Mach number, static pressure and temperature, and total temperature. Equations 5.5, 5.6, and 5.7 from Zucker [10] are used to compute the theoretical results for Mach number, static pressure and temperature respectively. No equation is needed to compute the total temperature because that quantity should remain constant [14].

$$\frac{fL}{D_H} = \frac{1-M_1^2}{\gamma M_1^2} - \frac{1-M_2^2}{\gamma M_2^2} + \frac{\gamma+1}{2\gamma} \ln \frac{M_1^2 [1+(\gamma-1)M_2^2/2]}{M_2^2 [1+(\gamma-1)M_1^2/2]} \quad \text{Eq. 5.5}$$

$$p_2 = p_1 \frac{M_1}{M_2} \sqrt{\frac{1+(\gamma-1)M_1^2/2}{1+(\gamma-1)M_2^2/2}} \quad \text{Eq. 5.6}$$

$$T_2 = T_1 \sqrt{\frac{1+(\gamma-1)M_1^2/2}{1+(\gamma-1)M_2^2/2}} \quad \text{Eq. 5.7}$$

Figures 5.16, 5.17, 5.18, and 5.19 show the results of the steady-state solutions of Mach number, static pressure and temperature, and total temperature, respectively. The initial results show good agreement with known Fanno flow behavior: Mach number increases, both static pressure and temperature decrease (for subsonic flow) and the total temperature remains constant. Looking at the percent error from theory, the results are even better. Mach number has a maximum error of $1.37 \times 10^{-5}\%$, static pressure has a maximum error of $1.38 \times 10^{-5}\%$, static temperature has a maximum error of $1.84 \times 10^{-7}\%$, and total temperature has a maximum error of $6.65 \times 10^{-10}\%$.

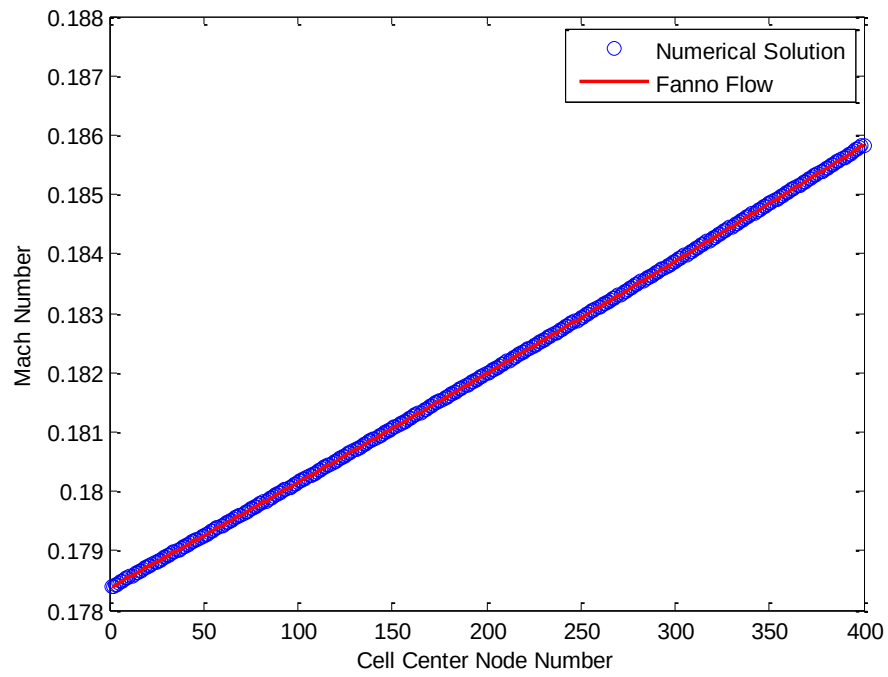


Figure 5.16 – Mach Number Comparison with Fanno Flow Theory

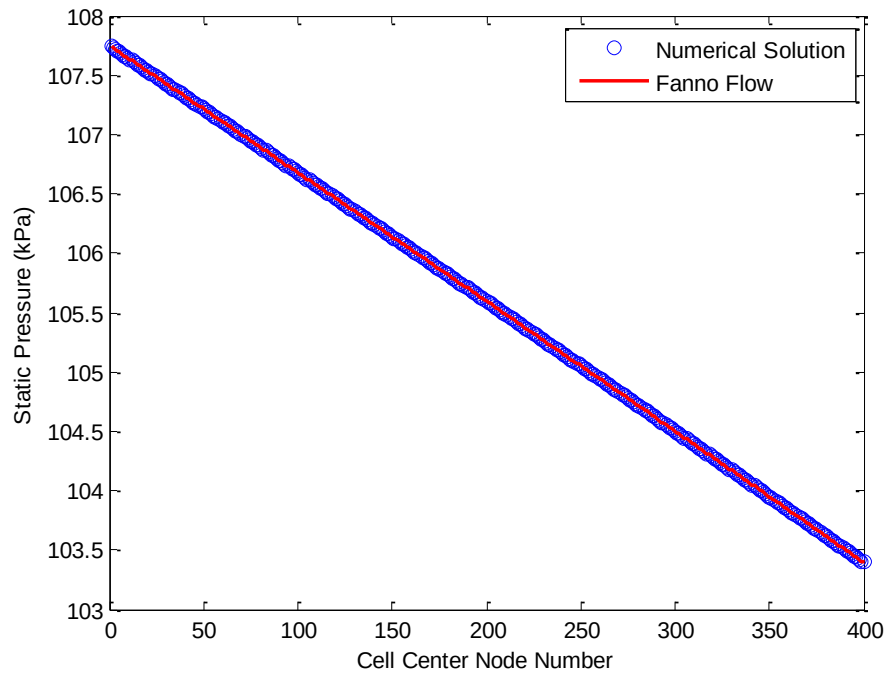


Figure 5.17 – Static Pressure Comparison with Fanno Flow Theory

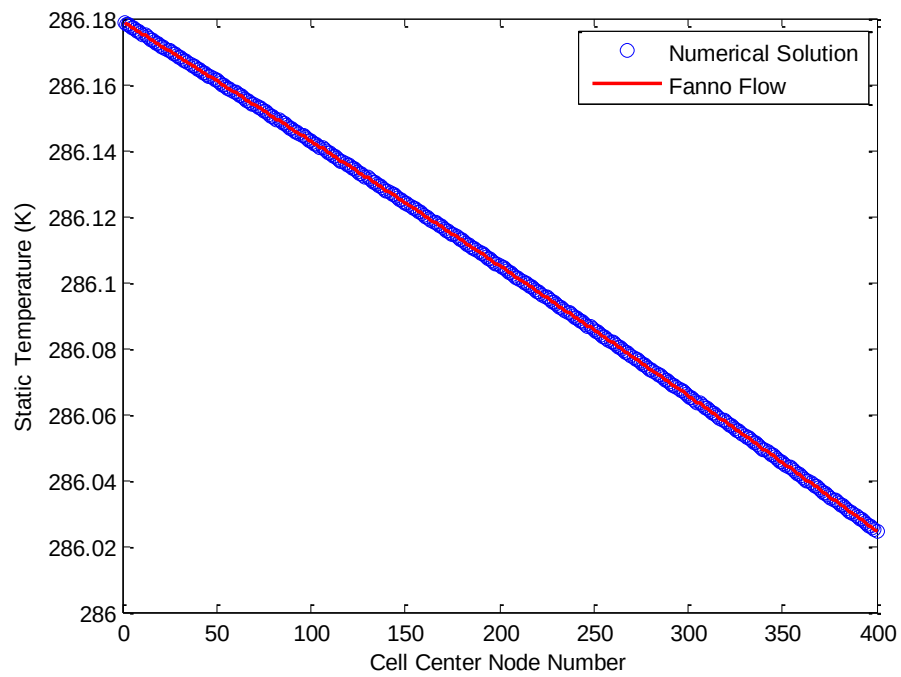


Figure 5.18 – Static Temperature Comparison with Fanno Flow Theory

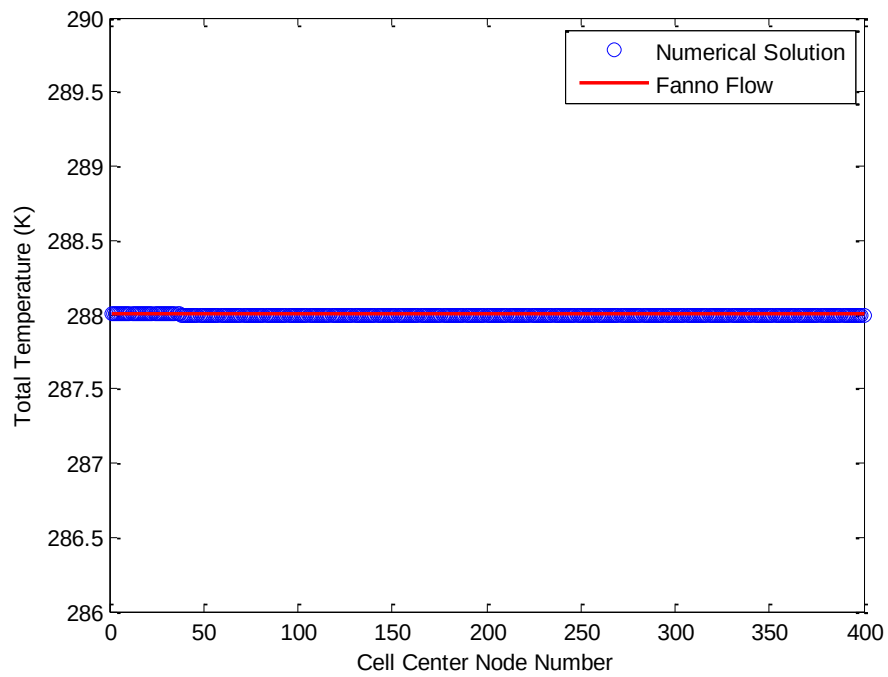


Figure 5.19 – Total Temperature Comparison with Fanno Flow Theory

Under normal circumstances the friction factor would not be held at a constant value but would rather be calculated using the local Reynolds number and surface roughness. Figure 5.20 shows the friction factor computed by equation 3.44 and compared with the one using information from Crane's handbook [15]. A considerable amount of difference exists between the two, 13.9%, but it is unknown what type of roughness was used when Crane collected their data. In addition the data in Crane stops at much smaller pipe diameters than was used in this test case.

The other feature of the new quasi-one-dimensional control volume is the ability to include minor loss coefficients. A simple test case was setup to demonstrate this new ability. It uses the same initial conditions as above but with a ten meter duct length instead of 200 meters. Also, the computation of friction factors is turned off in this example. A single K factor is used to simulate the effects of a 90 degree miter bend at 5 meters downstream of the inlet. Figure 5.21 show the effects on static pressure. As is expected, pressure remains constant up to the bend, because there are no losses up to this point, and at the 5 meter mark pressure drops off and then remains constant for the rest of the duct. The code predicts a pressure drop of 1.8 kPa with a 3.29% error compared to the data in Darcy-Weisbach formula, Equation 5.8, for incompressible flow [16]. In equation 5.8 f is the friction factor, D_H is the hydraulic diameter, L the distance along the pipe span, V the velocity, and ρ the density of the fluid. The minor loss coefficient K can be substituted for fL/D [15].

$$\Delta p = \frac{\rho V^2}{2} \frac{fL}{D_H} \quad \text{Eq. 5.8}$$

The addition of friction and minor losses has been found to be in excellent agreement with theory. By including the momentum equation, which the previous method ignored, spatial effects are now better simulated.

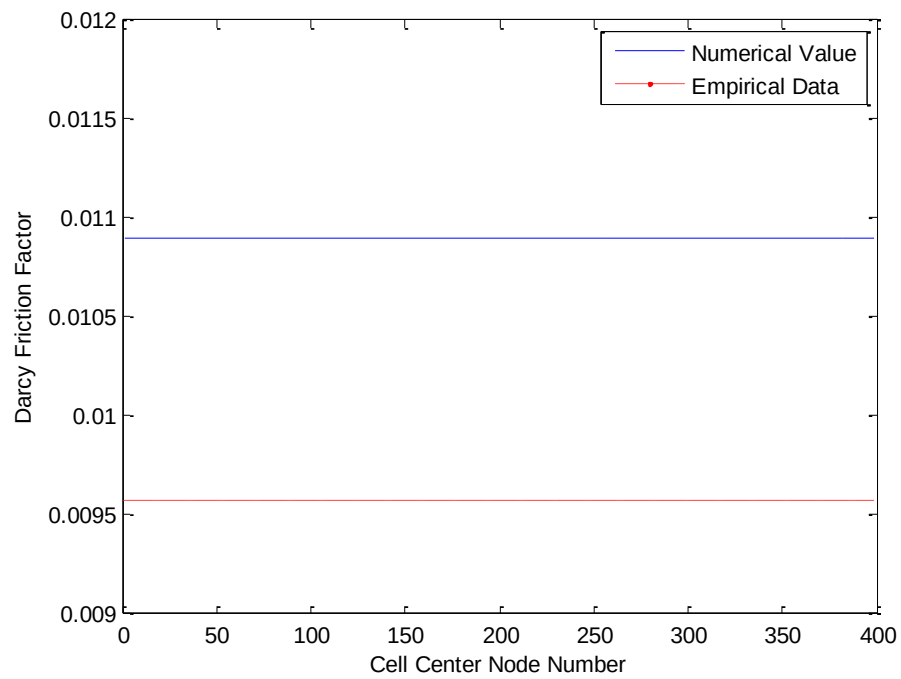


Figure 5.20 – Computed Friction Factors versus Empirical Data

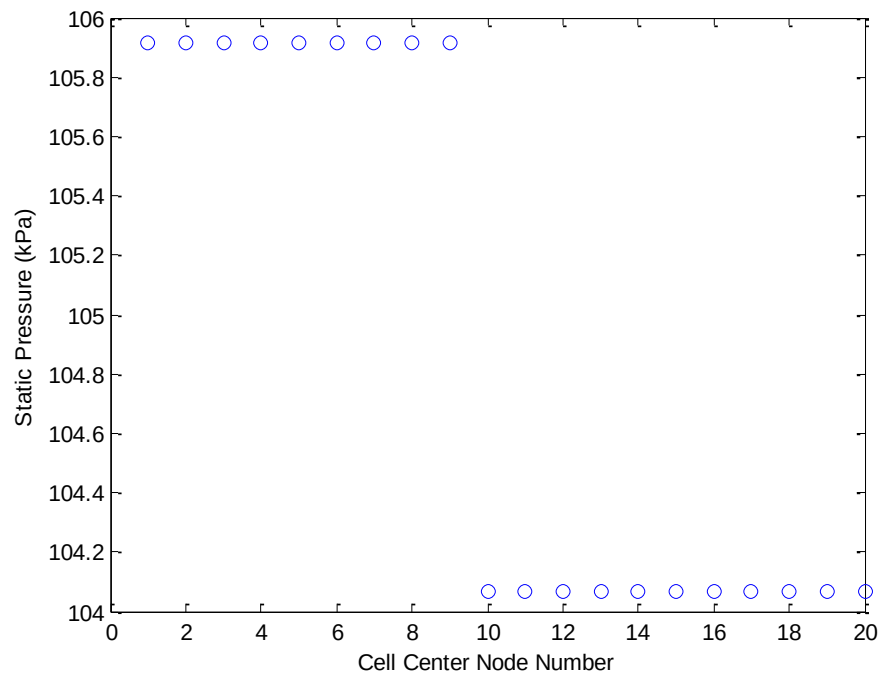


Figure 5.21 – Static Pressure Profile with a Minor Loss Coefficient

5.3 Flow with Heat Transfer

Heat transfer effects can have a significant impact on how a facility operates. Unwanted heat can be introduced to the cold flow in a duct or heat could be lost from the duct to the environment. Both of these effects could cause the facility to no longer be on condition or to never reach a desired temperature set-point. Current modeling methods include these effects in their simulations so it is important to preserve that capability in the new quasi-one-dimensional code. However, unlike the lumped parameter model described in section 1, the quasi-one-dimensional code solves for heat transfer. This affords the advantage of taking into account ducting that might be partially in the shade, pass through buildings, or subjected to non-uniform cross flow velocities, such as wind.

Three test cases were run to demonstrate the ability of the code to properly calculate heat transfer rates and to transfer energy to and from the fluid inside the ducting. The first test case assumes a constant energy flux into the fluid so that steady-state results can be compared to Rayleigh flow. The second calculates heat transfer rates from an initial duct and fluid temperature and allows the system to come to thermal equilibrium with zero flow. The last test case uses the results from the previous test as the initial conditions for the duct and fluid temperatures as a valve is slowly opened introducing fluid at a lower temperature. The simulation is then run until thermal equilibrium is achieved.

The first test case uses the same initial conditions as in the previous momentum loss test cases except that the duct length is twenty meters in length and all friction effects are ignored. The purpose of this test is to check the ability of the code to correctly calculate fluid properties as energy is added to the flow. A uniform, constant heat flux of 200,000 J/s is added at each cell center, except the first cell, so for a grid spacing of 0.5 meters, a total heat flux of 7,800,000 J/s is added at each time step. The reason that the first cell is omitted is to allow its fluid properties to be used as the initial state to

calculate the properties downstream using the Rayleigh flow equations. For subsonic flow, when heat is added to the flow Mach number will increase until the flow becomes choked, assuming enough heat is added, and the total temperature will increase, unlike with Fanno flow where it remained constant. The total pressure will decrease, a phenomenon known as the Rayleigh effect [14]. Equations 5.9 through 5.12 are used to calculate properties of the flow downstream of the first cell center.

$$T_{t2} = T_{t1} \left[1 + \frac{q}{c_p T_{t1}} \right] \quad \text{Eq. 5.9}$$

$$\frac{T_{t2}}{T_{t1}} = \left[\frac{1+\gamma M_1^2}{1+\gamma M_2^2} \right]^2 \left[\frac{M_2}{M_1} \right]^2 \left[\frac{1+(\gamma-1)M_2^2/2}{1+(\gamma-1)M_1^2/2} \right] \quad \text{Eq. 5.10}$$

$$p_2 = p_1 \frac{1+\gamma M_1^2}{1+\gamma M_2^2} \quad \text{Eq. 5.11}$$

$$T_2 = T_1 \left[\frac{1+\gamma M_1^2}{1+\gamma M_2^2} \right]^2 \left[\frac{M_2}{M_1} \right]^2 \quad \text{Eq. 5.12}$$

Equation 5.9 is used to find the total temperature downstream of the first cell center where q is the amount of heat flux added and c_p is the specific heat of the fluid at constant pressure. Once the total temperature is known, equation 5.10 can be used to iteratively solve for the downstream Mach number. Equations 5.11 and 5.12 are then used to calculate the static pressure and static temperature, respectively.

Figures 5.22 through 5.26 show the comparison of the numerical solution with the analytical solution predicted by equations 5.9 through 5.12. Figure 5.22 displays the Mach number and has a maximum error of $5.15 \times 10^{-6}\%$. Equally impressive are Figures 5.23 and 5.24 which display the static pressure and temperature, their maximum errors being $6.708 \times 10^{-7}\%$ and $8.963 \times 10^{-6}\%$ respectively.

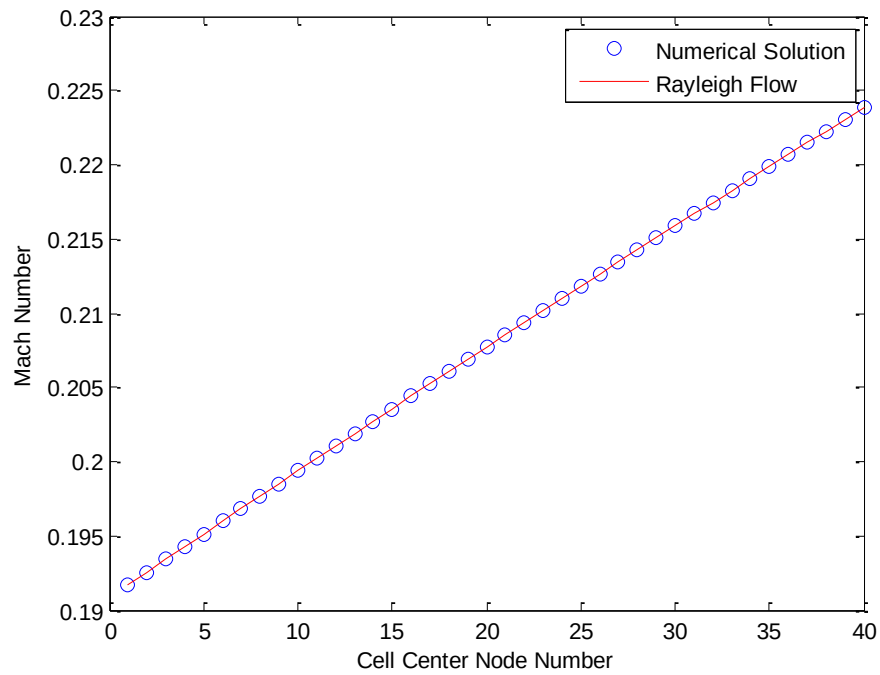


Figure 5.22 – Mach Number Comparison with Rayleigh Flow Theory

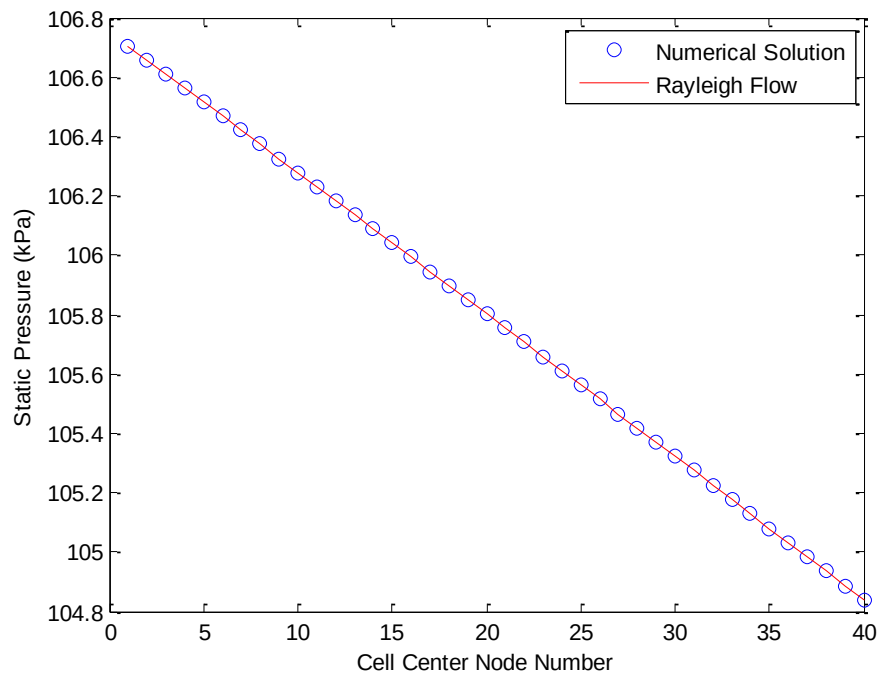


Figure 5.23 – Static Pressure Comparison with Rayleigh Flow Theory

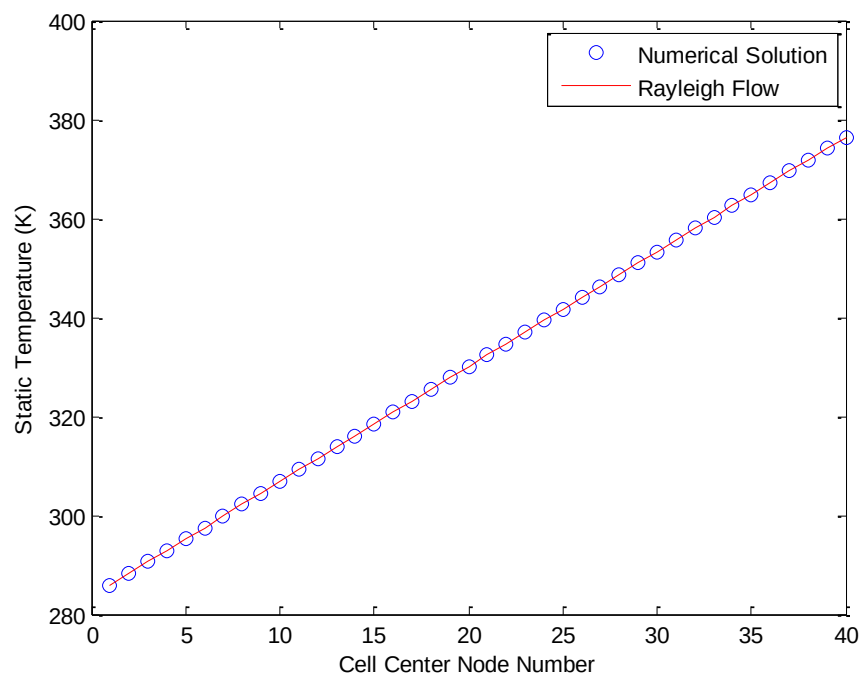


Figure 5.24 – Static Temperature Comparison with Rayleigh Flow Theory

Figures 5.25 and 5.26 show how the total pressure and temperature vary along the length of the duct. Total pressure has a maximum error of $3.156 \times 10^{-7}\%$ and total pressure has an error of $1.181 \times 10^{-12}\%$.

The second test case demonstrates the ability of the code to correctly compute heat transfer rates between the fluid and the duct and also between the fluid and the surrounding environment. Both the fluid and duct temperature start at an initial temperature of 288 K and the ambient surroundings have a constant temperature of 300 K. It is assumed that the duct is fully exposed to the sun with a uniform solar flux of 680 W/m^2 striking one side. It is assumed that no wind is blowing. The thermal properties of the duct were taken as the nominal values for stainless steel. These values along with other material properties are widely available through many different resources. During this test, both the upstream and downstream valves are closed and the simulation is run until thermal equilibrium is reached. Figure 5.27 displays the duct temperature as a function of time. After approximately 450 minutes the heat transfer rates between the duct and surroundings balance out and a final duct temperature of 316.042 K is reached. The temperature inside the duct also reaches 316.042 K at approximately the same rate as the duct temperature. There is a slight difference between the two on the order of 10^{-7} and this is probably due to the fact that equilibrium has not been completely achieved but that this small difference can be assumed to be negligible. While this example may be artificial, it does exercise the capability of the code to calculate heat transfer effects and gives confidence that thermal lag can be simulated.

Wind affects can also be included in the solution. If we assume a constant cross-flow of five m/s we find that the system reaches equilibrium in approximately 250 minutes, half the time it took for the condition without wind, and it achieves a final temperature of 307.09 K, see Figure 5.28. This is much closer to the ambient conditions, as would be expected.

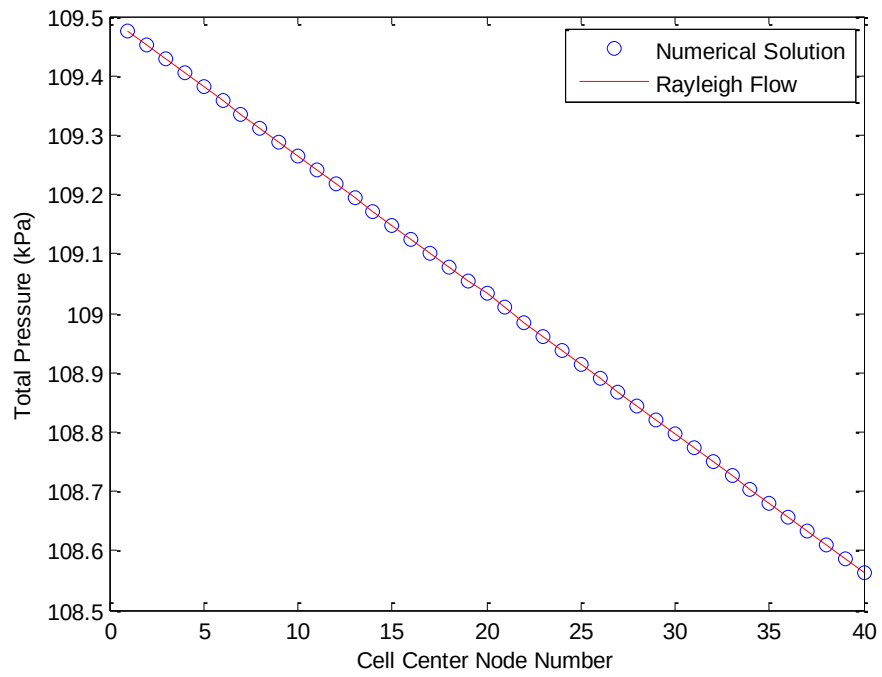


Figure 5.25 – Total Pressure Comparison with Rayleigh Flow Theory

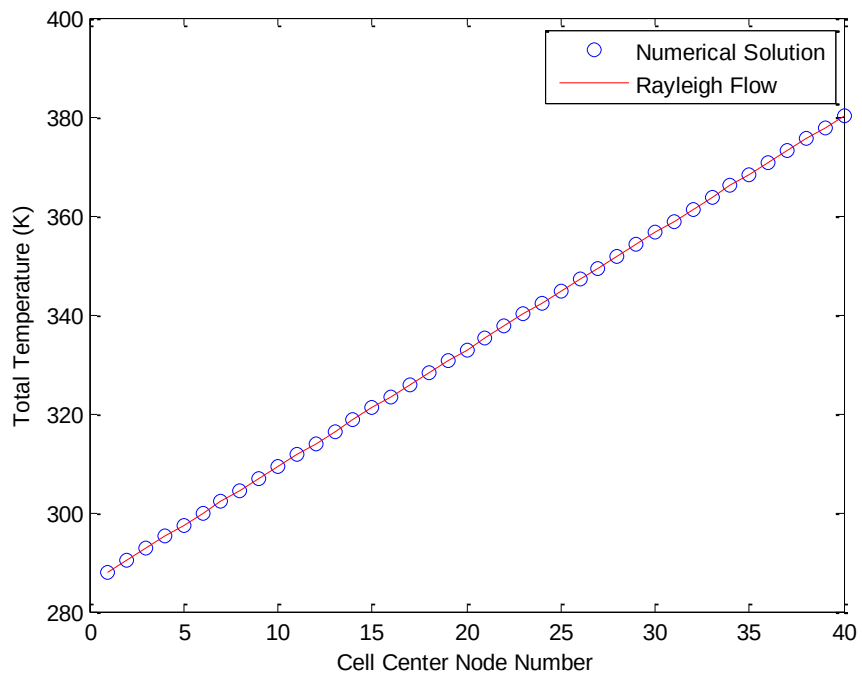


Figure 5.26 – Total Temperature Comparison with Rayleigh Flow Theory

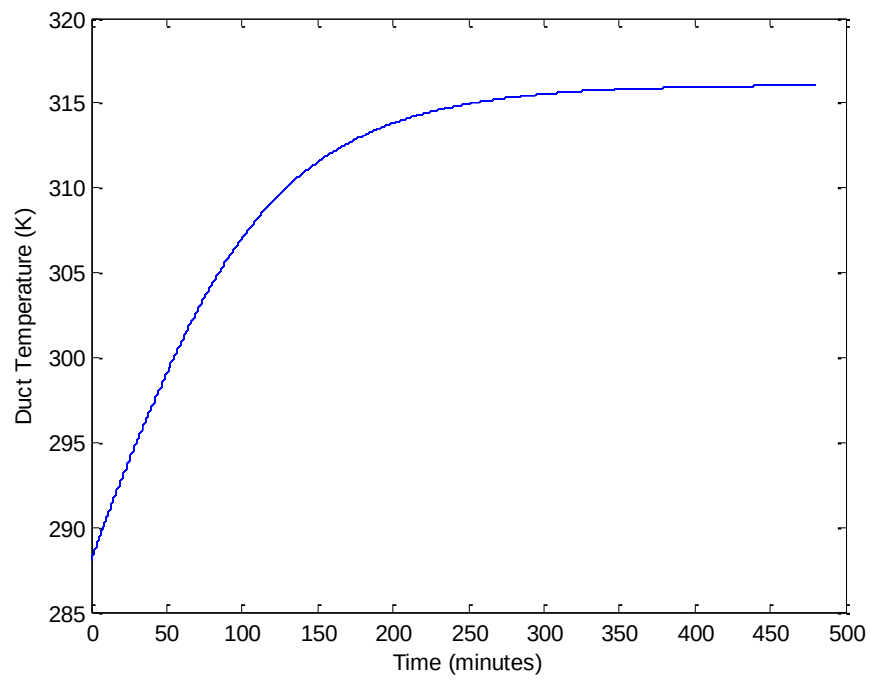


Figure 5.27 – Duct Temperature as a Function of Time

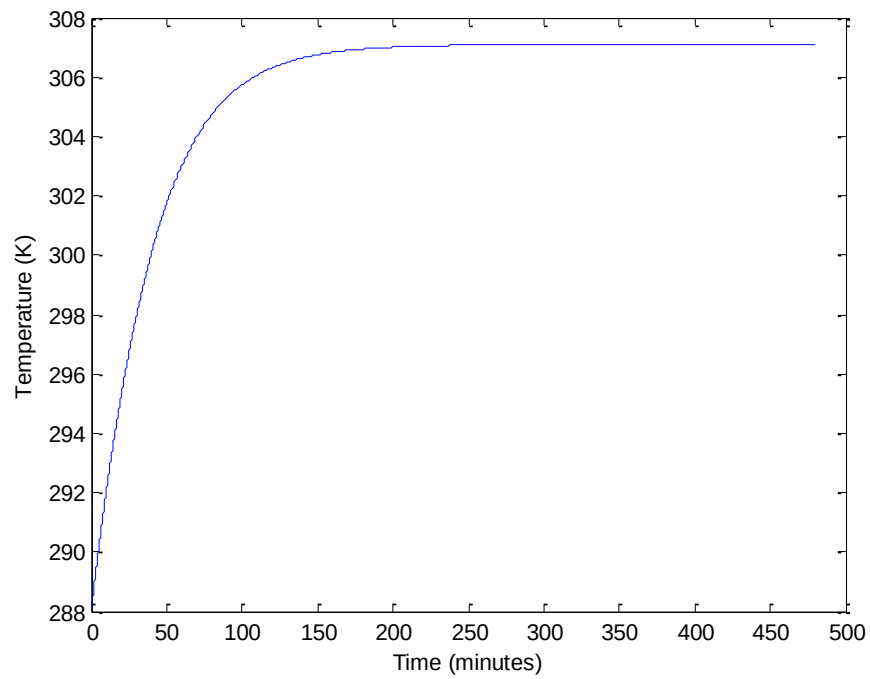


Figure 5.28 – Duct Temperature as a Function of Time with Wind Effects

The last test case uses the steady-state results from the first part of the previous test as the initial temperatures for the fluid inside the duct and the ducting temperature. The only differences are that the downstream valve is initially open and at time zero the upstream valve is closed. When the simulation starts the upstream valve opens and lets in cooler air at 288 K. Figure 5.29 displays the last cell center (node 40) fluid and duct temperature as a function of time. After 350 seconds the system has reached thermal equilibrium. From the figure it appears the both of the temperatures are the same; however, Figure 5.30 displays the steady-state results of both the duct and fluid total temperatures. There is a slight difference of 0.05 K. This feature is particularly useful for simulating and determining thermal lag in a system. This situation often exists in the real-world where facility time is spent waiting for the plant to come on condition due to initial strong thermal gradients between the fluid and ducting wall temperatures.

These three tests have demonstrated the ability of the code to correctly calculate the effects of temperature differences between the fluid and duct wall and between the ducting and its environmental surroundings. The capability is preserved from the lumped-parameter method, but now spatial effects are correctly taken into account.

Section 5.4 Transient Results

An additional improvement, by including spatial effects, over the lumped-parameter method is the ability to capture transient responses. The lumped-parameter model calculates mean properties of the fluid for a given volume with the underlying assumption that the properties are uniform throughout. At any given instant in time the solution is assumed to be at steady-state. For small volumes, or in the

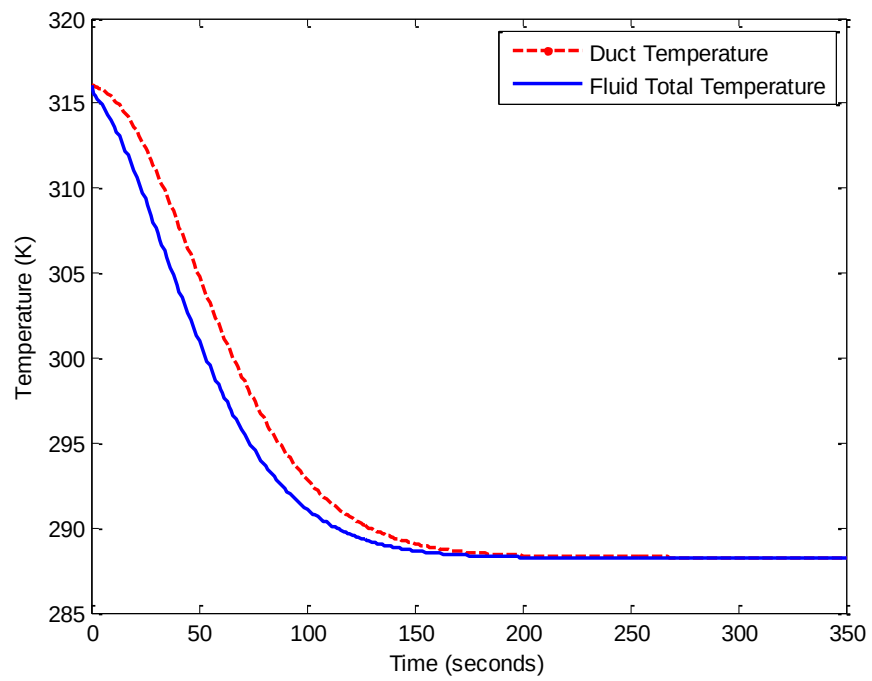


Figure 5.29 – Duct and Fluid Temperature as a Function of Time

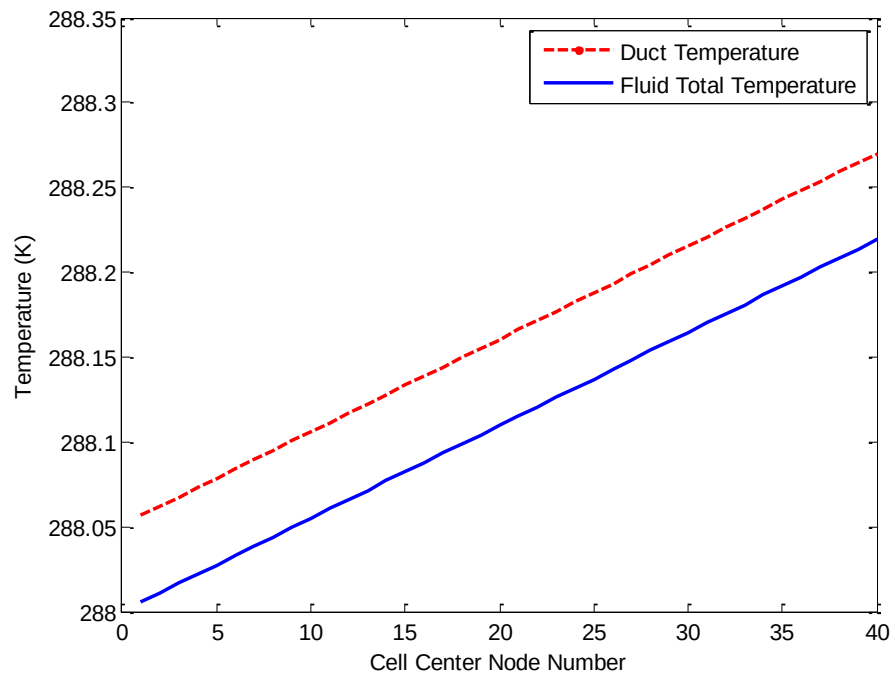


Figure 5.30 – Steady-State Profiles of Duct and Fluid Temperature

incompressible flow regime this is a valid assumption, but the lumped-parameter method misses any process lag that would exist in long ducting or pressure oscillations that might exist from the sudden opening or closing of a valve. The new quasi-one-dimensional control volume solves the unsteady equations of motion so that these effects are inherent in the solution.

To analyze the transient response of the computational solution, the frequency of pressure oscillations can be compared to the fundamental frequency of a sound wave in a tube. Equation 5.13 approximates the resonances occurring for an open ended tube [17].

$$f_n = \frac{nc}{2L} \quad \text{Eq. 5.13}$$

Where n is a positive integer representing the resonance node, L is the length of the tube and c is the speed of sound in air. Assuming a pipe length of fifty meters at an initial temperature of 288 K gives a natural frequency of 3.39 Hz. A model of a constant area, frictionless duct was constructed to simulate a pressure wave travelling along its length. The setup is the same as in Figure 5.2 and uses the same initial conditions given in the second test case in section 5.1 with a time-step of 0.0001 seconds and a grid spacing of 0.5 meters. At the start of the simulation Valve 1 is instantaneously opened to 100 percent to provide the pressure wave. Two time slices are taken of the pressure wave as it travels the length of the duct. One when the peak of the wave reaches the 10 meter mark and another when it reaches 40 meters, both of which can be visualized by Figures 5.31 and 5.32. The peak of the wave reaches ten meters downstream in 0.0306 seconds and travels forty meters in 0.1153 seconds. Knowing the time it took for the wave to travel 30 meters, the speed of the wave was calculated to be 354.19 m/s. This equates to a frequency of 3.54 Hz. Also present in the figures is evidence of the expansion and compression waves propagating through the duct by the sinusoidal variation of pressure with time.

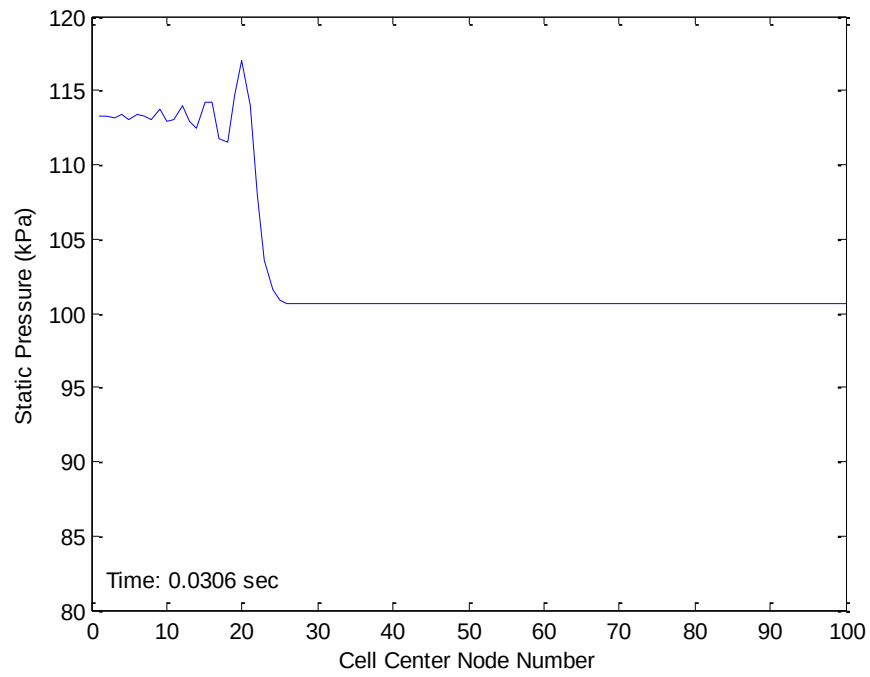


Figure 5.31 – Pressure Wave at 10 meters Downstream

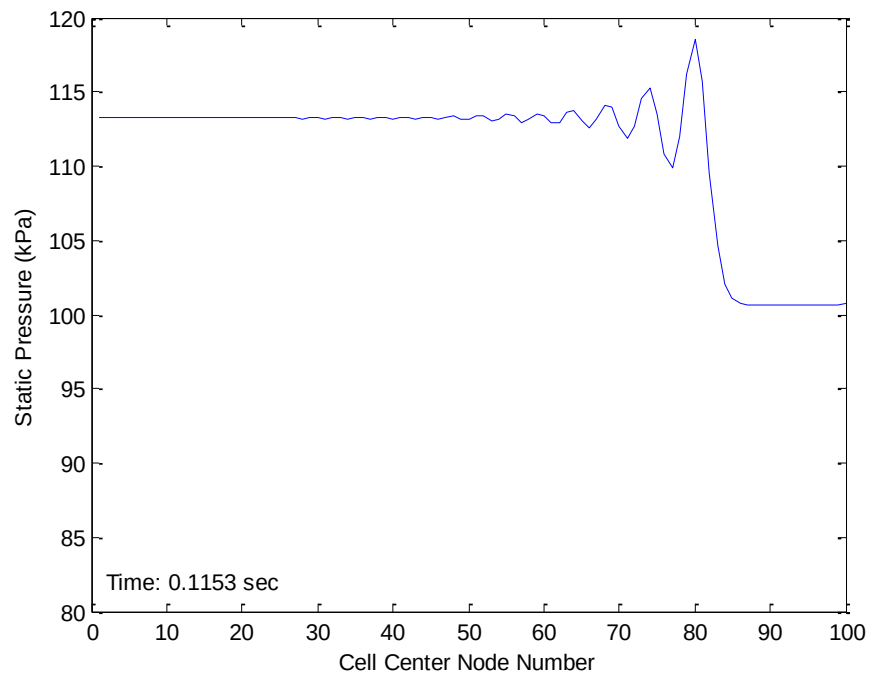


Figure 5.32 – Pressure Wave at 40 meters Downstream

Several more runs were made with grid densities equal to 200, 400 and 800 points to find at what frequency the pressure oscillations converge to. It was found that after 200 points the unsteady simulation predicted a duct frequency of 3.57 Hz. This results in a 5.31% error from the natural frequency given by equation 5.13. This is possibly due to the method chosen for time integration.

The effects of friction and heat transfer on the frequency were also examined and compared with the results obtained from the adiabatic solution with a grid density of 800 points. Adding friction resulted in a frequency 0.001 Hz lower than for the flow without friction; which is not completely unexpected. From Fanno flow theory, friction causes the static temperature to decrease which directly affects the speed of sound [14]. The overall effect of friction on the resonance can be neglected since its effect on the frequency is small. However, the magnitude of the pressure wave does dampen as it travels down the length of the duct which is to be expected because of the pressure loss associated with friction. Heat addition had the opposite effect of friction; it increased the frequency of the pressure oscillations. By directly adding energy to the flow the static temperature was increased. This increased the speed of sound and thus directly influenced the speed at which the wave travelled. Over the length of the pipe, 800,000 J/s of energy was added (1,000 per cell center) and this increased the frequency by 0.01 to 3.58 Hz. Heat addition also had the effect of damping the amplitude of the pressure wave which was also predicted by Rayleigh flow theory [14]. The findings of this analysis demonstrate the capability of the quasi-one dimensional control volume to produce transient results with reasonable agreement between theoretical and physical expectations.

Section 5.5 Real-time Results

The ability of a facility simulation to run in real-time is equally as important as the quality of the results obtained. The whole purpose of creating facility simulations was driven by the need for the controls engineer to check PLC logic prior to implementing the software or hardware changes in the field. The need for real-time results comes from the control hardware connected to the simulations which only has the capacity to run in real-time. Thus, the ability to meet the PLC's clock cycle is a firm requirement [3]. The lumped-parameter models were simple in their design and can run extremely fast on today's computers [2]. The quasi-one-dimensional code has been shown thus far to agree quite well with the theoretical and physical expectations of steady and unsteady duct flow and it will be shown that these capabilities still exist in a real-time environment.

To test the new ability of the code to run in real-time a simple model was created, much like the one in Figure 5.2, which exercises all the routines available. This includes calculating momentum losses and heat transfer rates. The model was compiled with three different time-steps: 0.001, 0.0005, and 0.0001 seconds and the number of cell volumes was incrementally increased for each until the execution time of a time-step was longer than the required sample time. The models were compiled using Matlab version 2009a and Microsoft's Visual Studio 2008 and were run on a dual core 2.6 GHz machine with 4 gigs of ram. The version of xPC Target supported by 2009a does not utilize multiple cores, so the model is actually only executed on a single processor.

Results from a sample time of 0.001 seconds are displayed in Figure 5.33 and the task execution times (TET) are plotted against the number of cell centers which range from 10 to 230. The TET is the measured CPU time to run the model equations and post outputs during each sample interval [6]. A linear fit was applied to the data to find the approximate maximum number of cells that can be

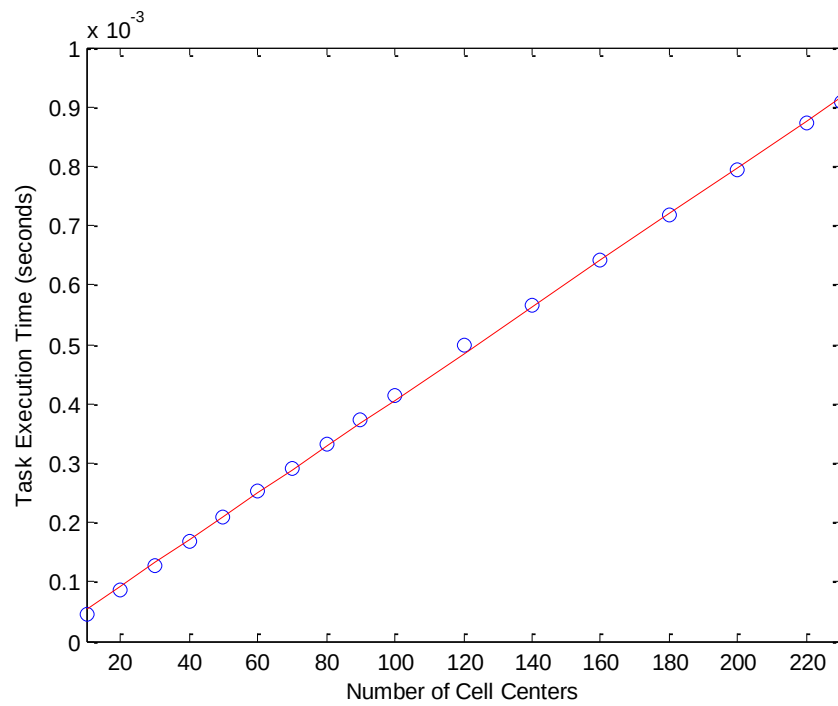


Figure 5.33 – Task Execution Times for a Sample Time of 0.001 seconds with Varying Grid Sizes

computed before the TET is greater than 0.001 seconds. This value was found to be 251 cells. Figures 5.34 and 5.35 show the results from sample times of 0.0005 and 0.0001 seconds, respectively. It was found that for a sample time of 0.0005 seconds 121 cells are possible and a maximum of 23 cells is achievable with a sample time of 0.0001 seconds.

The majority of the results that have been presented thus far have had a sample time of 0.0001 seconds with a grid spacing of 0.5 meters. From the above analysis this sample time is unrealistic because it would only result in a duct length of 11.5 meters. The total length of the ducting in any facility is much longer than this, so it would behoove the modeler to choose a sample time which results in a reasonable number of cells so that the ducting simulated can capture the full geometry while still producing accurate results. Using a larger time step is also going to cause the step size to increase. The analysis in Section 5.1 suggests that decreasing both the temporal and spatial resolutions will have a negative impact on the computational solution where there are changes in area. This begs the question: what kind of errors might exist in a real-time simulation? To answer this, several runs were made using realistic grid spacing and time-step to see what kind of steady-state and transient errors exist.

The grid spacing used for this analysis was two meters with a time-step of 0.001 seconds. Two meters was chosen because it was felt that this would provide enough resolution to maintain a reasonable solution while allowing for longer duct lengths to be modeled. The time-step has a theoretical limit of 251 cells which translates to a possible duct length of 502 meters (or 1,647 feet) for the chosen two meter grid spacing. The geometry in Figure 5.1 is used again here to explore the steady-state error that might exist in a real-time scenario. The upstream pressure is set at 114.99 kPa with a downstream pressure of 100.62 kPa. The ducting's pressure is initially set to 100.62 kPa with an initial velocity of zero. At the start of the simulation the upstream valve begins to slowly open until it reaches

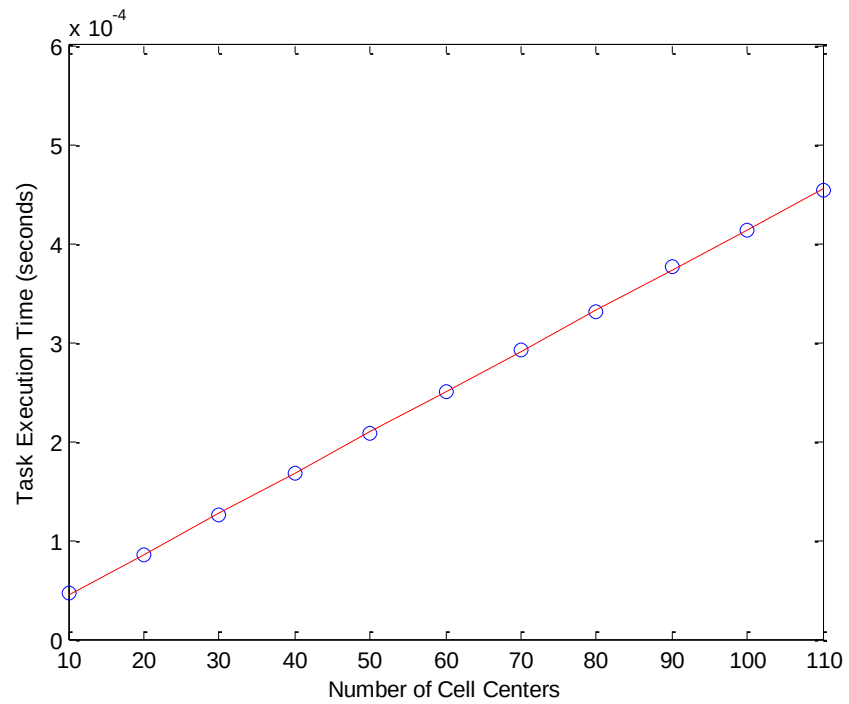


Figure 5.34 – Task Execution Times for a Sample Time of 0.0005 seconds with Varying Grid Sizes

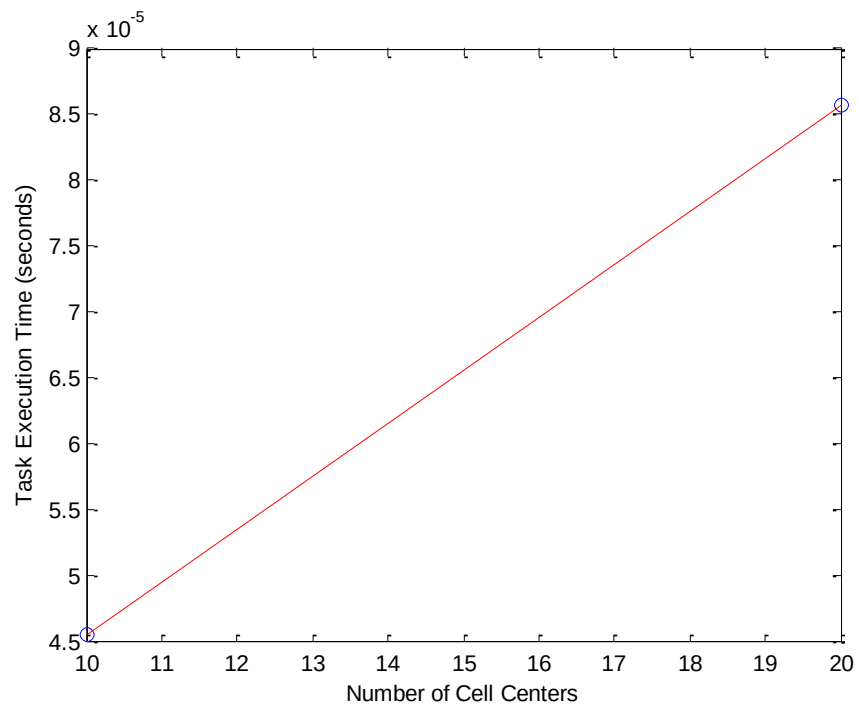


Figure 5.35 – Task Execution Times for a Sample Time of 0.0001 seconds with Varying Grid Sizes

100% open five seconds later. The model is then allowed to run for a sufficiently long time until the transients settle out and a steady-state solution has been reached. Figures 5.36 and 5.37 plot the calculated and theoretical Mach number and pressure profiles, respectively. The figures clearly show there is a discrepancy between the computed results and the analytical solution. This is not unexpected, as the analysis done in Section 5.1 predicts a degraded flow solution for coarse grid sizes and time-steps. The maximum error in Mach number is 1.78% and 1.82% in pressure. The given example has a maximum Mach number of 0.48; however, most ducting is designed for Mach numbers of 0.2 to 0.3. The reason for this is that if the Mach number were much lower than the ducting has been oversized and material wasted. High speed flows can also cause damage to plant equipment and for these reasons duct velocities are designed to be low. The results from section 5.1 also showed that the error is much smaller at low speed flows versus that at higher speeds. One would typically expect errors slightly lower than those presented here. When a case was run with conditions to produce Mach 0.25 flow in the contraction, errors in Mach number were 0.32% and the pressure error was also 0.32%. This would suggest that under normal circumstances errors of less than 0.5% are to be expected.

A comparison was also performed for frictional losses. A repeat of the first test case of Section 5.2 was done but with a grid spacing of two meters and a time step of 0.001 seconds. It was found that the numerical results agree quite well with the Fanno flow solution. Figures 5.38 and 5.39 plot the steady-state profile for Mach number and pressure. It was found that Mach number had a maximum error of $1.37 \times 10^{-5}\%$ and pressure had a maximum error of $1.38 \times 10^{-5}\%$. These are the same errors that were reported in section 5.2 so it can be concluded that there is no degradation in the numerical solution for a coarser grid and temporal spacing.

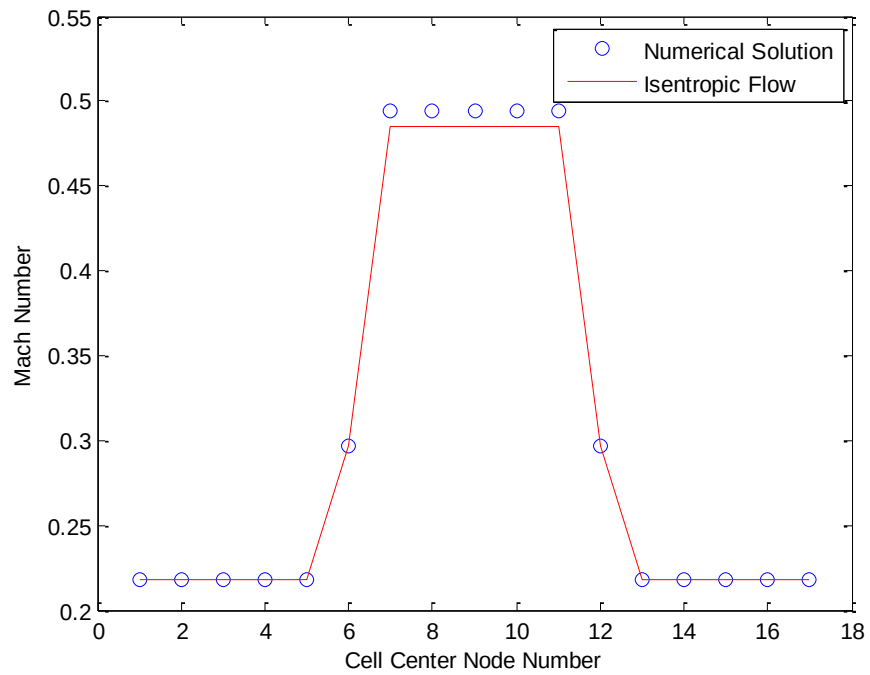


Figure 5.36 – Mach Number Comparison with Isentropic Flow Theory for Lower Grid Densities

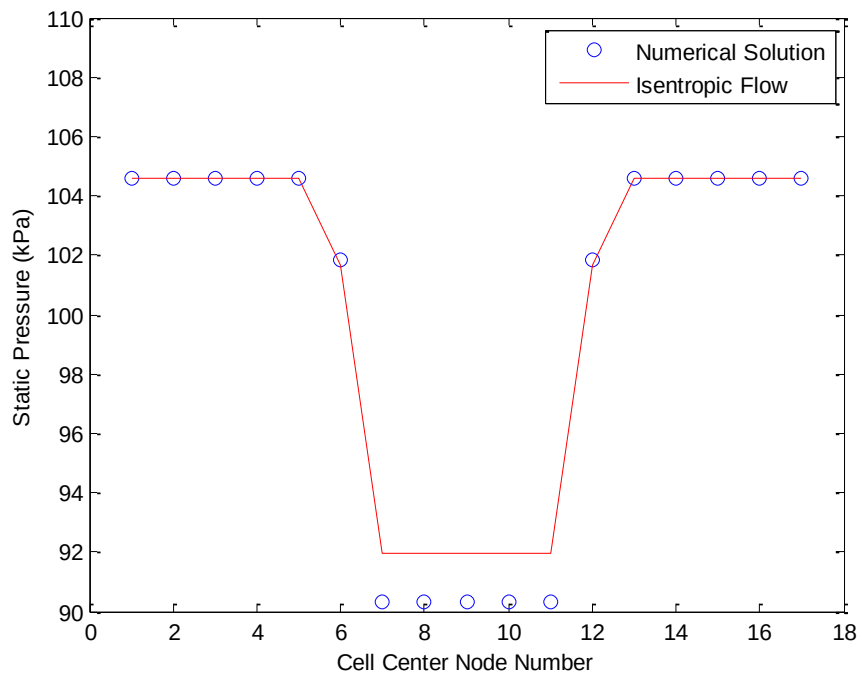


Figure 5.37 – Static Pressure Comparison with Isentropic Flow Theory for Lower Grid Densities

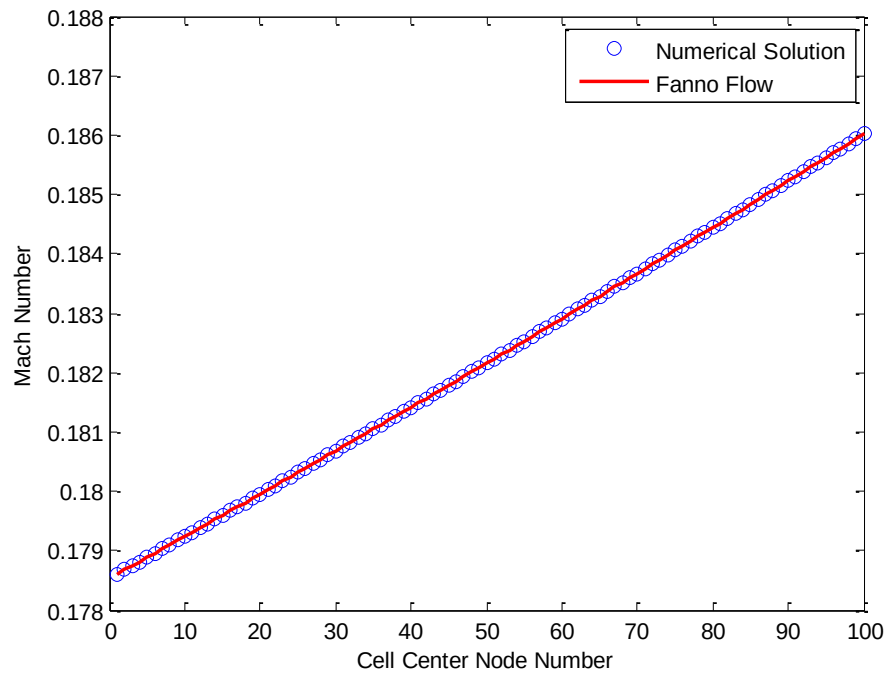


Figure 5.38 – Mach Number Comparison with Fanno Flow Theory for Lower Grid Densities

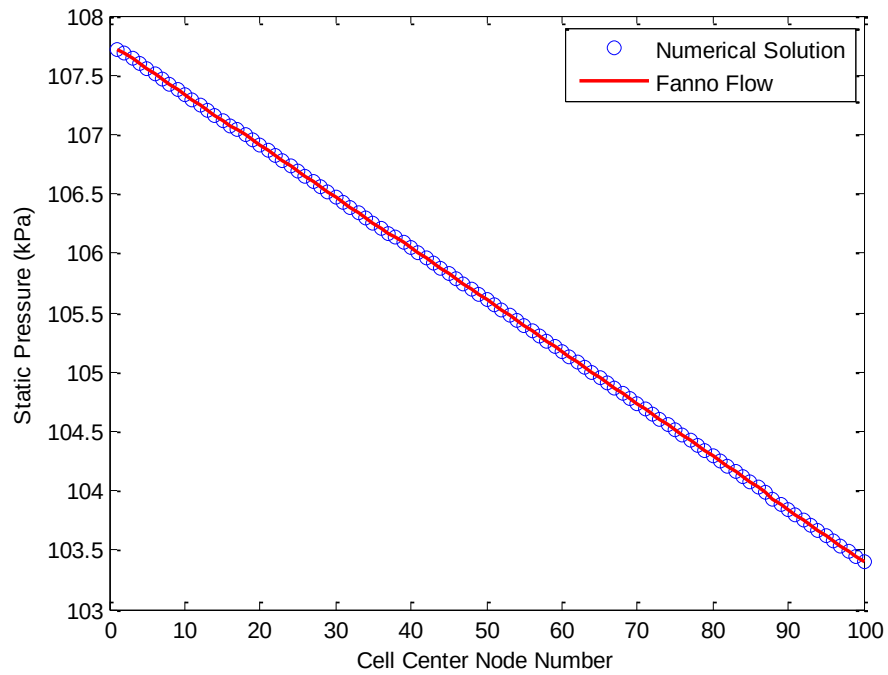


Figure 5.39 – Static Pressure Comparison with Fanno Flow Theory for Lower Grid Densities

To test how a reduced grid size and time-step affects the solution to heat addition, the first test of section 6.3 was repeated. It was found that Mach number had a maximum steady-state error of $5.13 \times 10^{-6}\%$ and pressure had an error of $6.70 \times 10^{-7}\%$. These agree well with the initial estimate of error with a tighter grid. Figures 5.40 and 5.41 show the steady-state profiles of Mach number and pressure as they compare with the Rayleigh solution. It is shown that the grid size and the time-step will have little or no effect on the flow solution under heat transfer effects.

A second transient analysis was also performed with the same reduced grid density and time-step to see their effects on the calculated resonance of the duct. The model and process used for determining the frequency were the same as in Section 5.4. A frequency of 3.41 Hz was calculated which has an error of 0.55% when compared to the theoretical value of 3.39 Hz.

The above analysis has shown that the new quasi-one-dimensional control volume produces accurate results; although, one does have to be mindful of conditions where strong gradients could exist in the geometry. If the grid size chosen is rather coarse, then significant error could exist in areas of rapidly changing velocities. But for a typical duct flow, the errors expected are quite small and the solutions are still more accurate than those given by the lumped parameter method discussed in section 2.

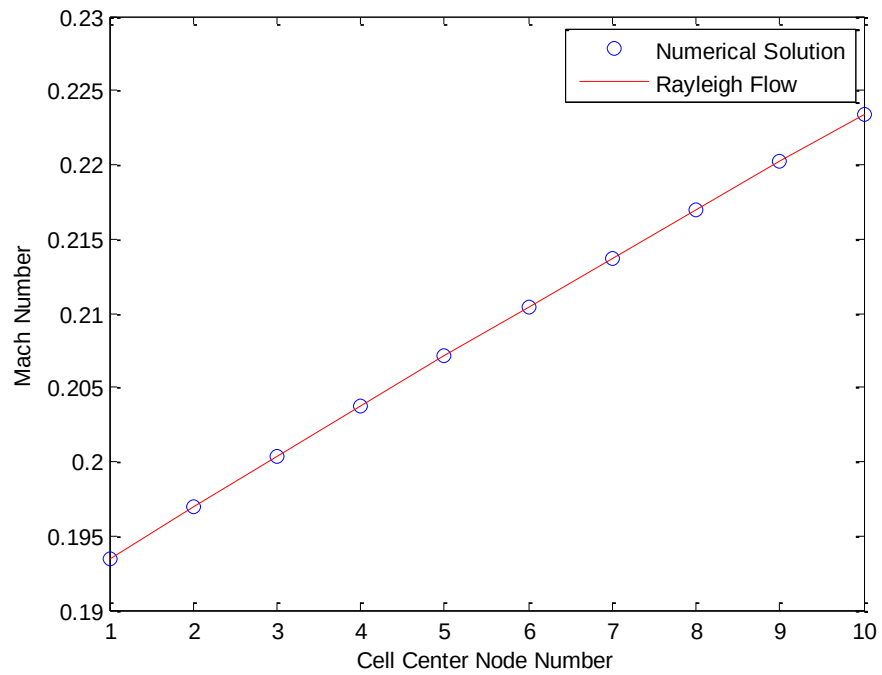


Figure 5.40 – Mach Number Comparison with Rayleigh Flow Theory for Lower Grid Densities

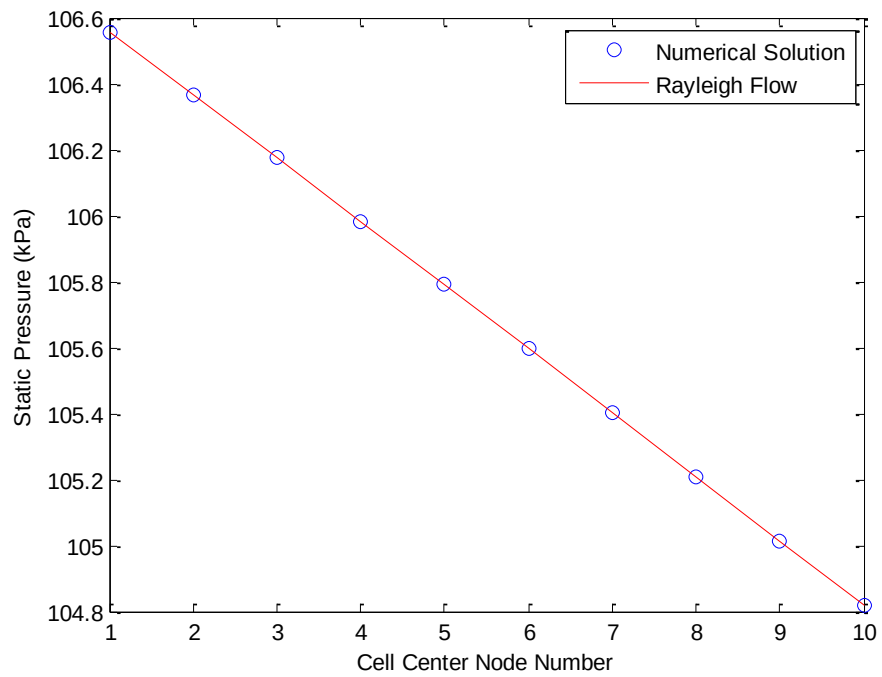


Figure 5.41 – Static Pressure Comparison with Rayleigh Flow Theory for Lower Grid Densities

6. Practical Example of a Flow Network

In section 5 the features of the quasi-one-dimensional control volume were analyzed separately in order to compare steady-state and transient results with theoretical solutions. In this section all the features are exercised in an example to demonstrate them working together to simulate two pieces of ducting intersecting at a tee junction and combining into one single pipe. The simple network could be a representative section of a much larger facility. Figure 6.1 is a diagram of the piping network being simulated. There are two legs of varying temperatures that are combined into a single pipe. The “cold leg” has a pressure source of 137.9 kPa at 249.82 K and the “hot leg” is connected to a pressure source of 137.9 kPa at a temperature of 447.6 K. These two pipes then join at a tee junction before entering a single piece of ducting that is connected to a pressure sink at 100.62 kPa (roughly atmospheric pressure). As shown on the diagram the piping has several 90 degree bends. This simulates a real facility, wherein the piping, for practical reasons, is never straight. For example, the pipe work has to be run around existing structure or loops can be purposefully designed into the system to alleviate stress due to thermal expansion or contraction. These pipe bends also have the effect of introducing pressure losses as discussed above. The older lumped-parameter method, described in section 2, is incapable of modeling the affect of pipe bends.

The purpose of simulating the diagram in Figure 6.1 might to be examine the mass flow rates achievable with the given pressure sources and sinks and the losses throughout the ducting. In addition, these are large differences in temperature. It is also of interest to study the final temperature attainable and to look at how long it takes the system to reach thermal equilibrium.

The pipe work shown in Figure 6.1 is converted into a Simulink model using the method outlined in section 5.1 and this is given by Figure 6.2. The setup information for the “cold,” “hot,” and “warm

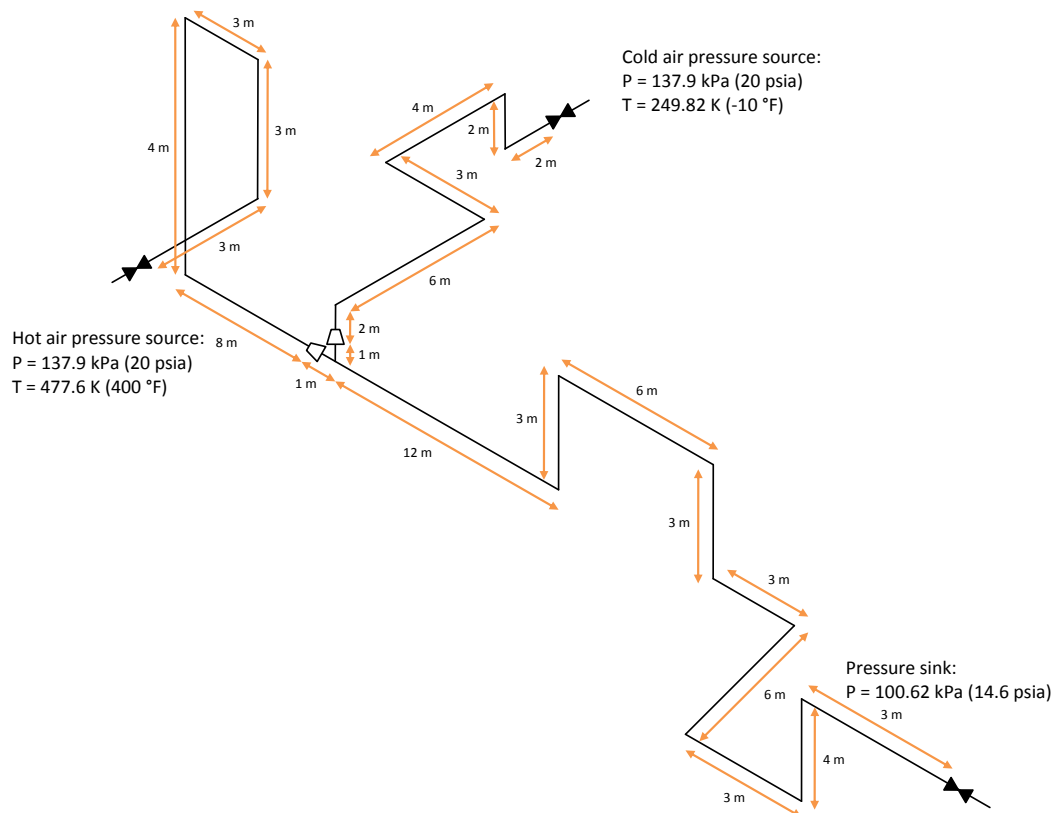


Figure 6.1 – Example Flow Network

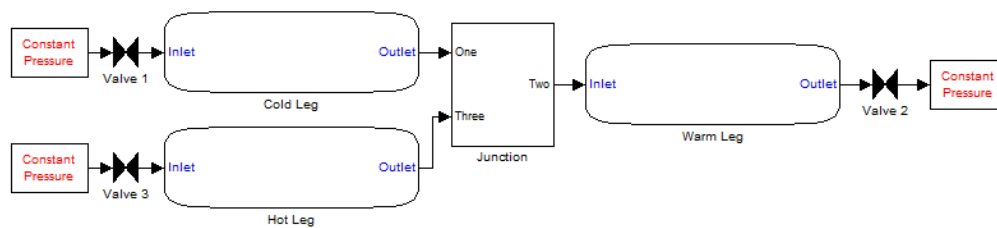


Figure 6.2 – Simulink® Diagram of Example Ducting System

leg” control volumes are given in Tables 6.1, 6.2, and 6.3 respectively. The diameters of the cold and hot legs are read from a file since they do not remain constant through their entire length. The junction that is used has a diameter that is equal to the diameter of the “warm leg” so both the “cold” and “hot legs” have expansions that start two meters upstream of where they connect to the junction. Tables 6.4 and 6.5 give the geometry used in the “cold” and “hot legs” respectively. It is important to note that the two meter Δx values chosen for the grid spacing are not an even multiple of the duct lengths or of the locations of bend placements. This is intentional as a compromise of being able to simulate a given system of ducting in real-time with adequate accuracy. That is preserving enough resolution to obtain meaningful results. The consequence is that the bends modeled do not correspond to their physical locations and at least one of the ducts is slightly longer in the model than in reality. An attempt was made to preserve overall duct lengths by either shortening or lengthening straight duct where applicable. This ensured that the effects of friction and overall heat transfer are well preserved. Both the “cold” and “hot legs” remained the same lengths. The length of the “warm leg” was increased overall by one meter.

The following subsections go over the junction block, which has yet to be introduced, and the results of the analysis. This includes looking at transients, steady-state mass flows, pressure losses, and the effects of heat transfer. Real-time task execution times of this system are also compared with the results obtained in section 5.5 to see how well they agree. Lastly, a simple lumped-sum control volume is created to model the system as shown in Figure 6.1 and its results are compared to the system modeled using the new quasi-one-dimensional control volume.

Table 6.1 – Cold Leg Inputs

Parameter Name	Value
Length: (m)	20
Grid dx: (m)	2
Duct Diameter: (m)	ColdLegDiameters.txt
Duct Thickness: (m)	0.014275
Duct Surface Roughness: (m)	0.00004572
K-factor Values:	[0.24 0.24 0.24 0.24 0.24]
K-factor Locations:	[2 4 8 12 16]
Specific Heat of Duct Material: (J/kg-K)	500
Thermal Conductivity of Duct Material: (W/K-m)	16.2
Density of Duct Material: (kg/m3)	8000
Emissivity of Duct Surface:	0.585
Uniform Solar Flux: (W/m2)	340
Cross Flow Velocity: (m/s)	0
Ambient Air Temperature: (K)	305
Uniform Density: (kg/m3)	1.225
Uniform Temperature: (K)	320.32113
Uniform Velocity: (m/s)	0
Uniform Duct Temperature: (K)	320.321132

Table 6.2 – Hot Leg Inputs

Parameter Name	Value
Length: (m)	22
Grid dx: (m)	2
Duct Diameter: (m)	HotLegDiameters.txt
Duct Thickness: (m)	0.009271
Duct Surface Roughness: (m)	0.00004572
K-factor Values:	[0.28 0.28 0.28 0.28]
K-factor Locations:	[4 8 12 16]
Specific Heat of Duct Material: (J/kg-K)	500
Thermal Conductivity of Duct Material: (W/K-m)	16.2
Density of Duct Material: (kg/m3)	8000
Emissivity of Duct Surface:	0.585
Uniform Solar Flux: (W/m2)	340
Cross Flow Velocity: (m/s)	0
Ambient Air Temperature: (K)	305
Uniform Density: (kg/m3)	1.225
Uniform Temperature: (K)	320.196164
Uniform Velocity: (m/s)	0
Uniform Duct Temperature: (K)	320.1961644

Table 6.3 – Warm Leg Inputs

Parameter Name	Value
Length: (m)	44
Grid dx: (m)	2
Duct Diameter: (m)	0.5747
Duct Thickness: (m)	0.01745
Duct Surface Roughness: (m)	0.00004572
K-factor Values:	[0.24 0.24 0.24 0.24 0.24 0.24 0.24 0.24]
K-factor Locations:	[10 14 18 22 26 32 36 40]
Specific Heat of Duct Material: (J/kg-K)	500
Thermal Conductivity of Duct Material: (W/K-m)	16.2
Density of Duct Material: (kg/m3)	8000
Emissivity of Duct Surface:	0.585
Uniform Solar Flux: (W/m2)	340
Cross Flow Velocity: (m/s)	0
Ambient Air Temperature: (K)	305
Uniform Density: (kg/m3)	1.225
Uniform Temperature: (K)	320.809384
Uniform Velocity: (m/s)	0
Uniform Duct Temperature: (K)	320.809386

Table 6.4 – Cold Leg Diameters

x-Location (m)	Diameter (m)
0	0.42865
2	0.42865
4	0.42865
6	0.42865
8	0.42865
10	0.42865
12	0.42865
14	0.42865
16	0.42865
18	0.42865
20	0.5747

Table 6.5 – Hot Leg Diameters

x-Location (m)	Diameter (m)
0	0.254508
2	0.254508
4	0.254508
6	0.254508
8	0.254508
10	0.254508
12	0.254508
14	0.254508
16	0.254508
18	0.254508
20	0.254508
22	0.5747

6.1 Tee Junction Block

Moving away from a lumped-parameter control volume and into one that has dimension causes an additional problem that must be addressed. How should junctions in the ducting network be treated? A mass flow device that can compute solutions for multiple joining flows is needed. It must satisfy appropriate boundary conditions for the quasi-one-dimensional control volume. An exact, time dependent, solution already exists and a set of equations for its computation is provided in [18]. These equations are hyperbolic in time and hence the method of characteristics can be applied. Assume that the fluid velocity is much lower than the sonic velocity, so that the characteristics are found to be straight and invariant with time. A set of equations with an explicit solution is thus formed [18]. Equations 6.1 and 6.2 calculate the mass flows entering the junction and leaving it, respectively.

$$A^{(r)}\Delta t \left[-\frac{2f^{(r)}}{(A^{(r)})^2 D^{(r)}} v_{K-1,t-1}^{(r)} \dot{m}_{K-1,t-1}^{(r)} \left| \dot{m}_{K-1,t-1}^{(r)} \right| - \frac{1}{c} \frac{p_{junc,t} - p_{K-1,t-1}^{(r)}}{\Delta t} \right] + \dot{m}_{K-1,t-1}^{(r)} = \dot{m}_{K,t}^{(r)} \quad \text{Eq. 6.1}$$

$$A^{(s)}\Delta t \left[-\frac{2f^{(s)}}{(A^{(s)})^2 D^{(s)}} v_{1,t-1}^{(s)} \dot{m}_{1,t-1}^{(s)} \left| \dot{m}_{1,t-1}^{(s)} \right| + \frac{1}{c} \frac{p_{junc,t} - p_{1,t-1}^{(s)}}{\Delta t} \right] + \dot{m}_{1,t-1}^{(s)} = \dot{m}_{0,t}^{(s)} \quad \text{Eq. 6.2}$$

The letter r denotes which pipeline is carrying fluid into the junction and s denotes pipelines carrying fluid away. The index $K - 1$ refers to the properties in the pipeline just upstream of the junction and K are conditions at the junction for flows entering. The index 1 refers to the properties in the pipelines just downstream of the junction for flows leaving and 0 is the index of properties in the pipeline at the junction. Figure 6.3 provides a diagram of the labeling at the junction. The index “ $t - 1$ ” refers to the previous time-step and t is the current time-step. The letter A is the cross sectional area of the pipeline entering or leaving the junction, f is the fanning friction factor, v is the specific volume, $\dot{m}$ is the mass

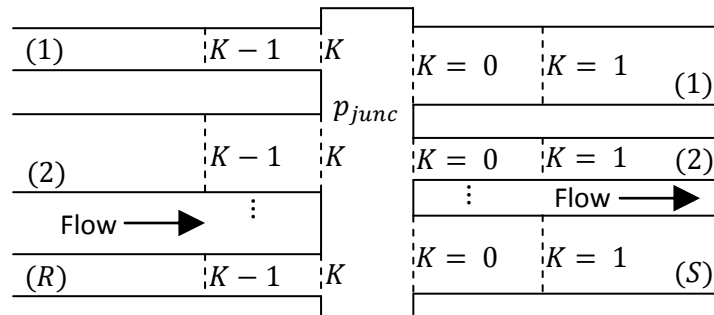


Figure 6.3 – Junction Header with Indices

flow, p is the pressure of the fluid entering or leaving and p_{junc} is the pressure at the junction. To solve for the pressure inside the junction, p_{junc} , equation 6.3 [18] can be used:

$$\begin{aligned}
 p_{junc,t} = & c_{son} \left(\frac{1}{\sum_{r=1}^R A^{(r)} + \sum_{s=1}^S A^{(s)}} \right) \\
 & \times \left[\sum_{r=1}^R \left\{ \dot{m}_{K-1,t-1}^{(r)} + \frac{A^{(r)}}{c} P_{K-1,t-1}^{(r)} - A^{(r)} \Delta t \left[\frac{2f^{(r)}}{(A^{(r)})^2 D^{(r)}} v_{K-1,t-1}^{(r)} \dot{m}_{K-1,t-1}^{(r)} \left| \dot{m}_{K-1,t-1}^{(r)} \right| \right] \right\} \right. \\
 & \left. - \sum_{s=1}^S \left\{ \dot{m}_{1,t-1}^{(s)} - \frac{A^{(s)}}{c} P_{1,t-1}^{(s)} - A^{(s)} \Delta t \left[\frac{2f^{(s)}}{(A^{(s)})^2 D^{(s)}} v_{1,t-1}^{(s)} \dot{m}_{1,t-1}^{(s)} \left| \dot{m}_{1,t-1}^{(s)} \right| \right] \right\} \right]
 \end{aligned} \tag{Eq. 6.3}$$

The letters R and S denote the number of pipelines entering and leaving the junction respectively.

The new quasi-one-dimensional control volume, together with the junction block is better able to simulate facility ducting by preserving more physical characteristics over the lumped-parameter modeling used previously. The next subsections will present some results and directly compare them to the older method.

6.2 Analysis and Results of Example Network

The model depicted in Figure 6.2 was run with a time-step of 0.001 seconds and a grid spacing of two meters. At time zero, all three valves are closed and begin to open after one second. They open at a rate of 20% per second, so at a simulation time of six seconds all the valves are fully open. The model is run for 1000 seconds to allow the system to reach thermal equilibrium. Figures 6.4, 6.5, and 6.6 are time slices of the velocity profiles in the “cold,” “warm,” and “hot” legs respectively. The velocity in the “cold” leg goes through several changes before reaching its steady-state value. Within the first five seconds any oscillations that were created by the valve opening have already damped out and the

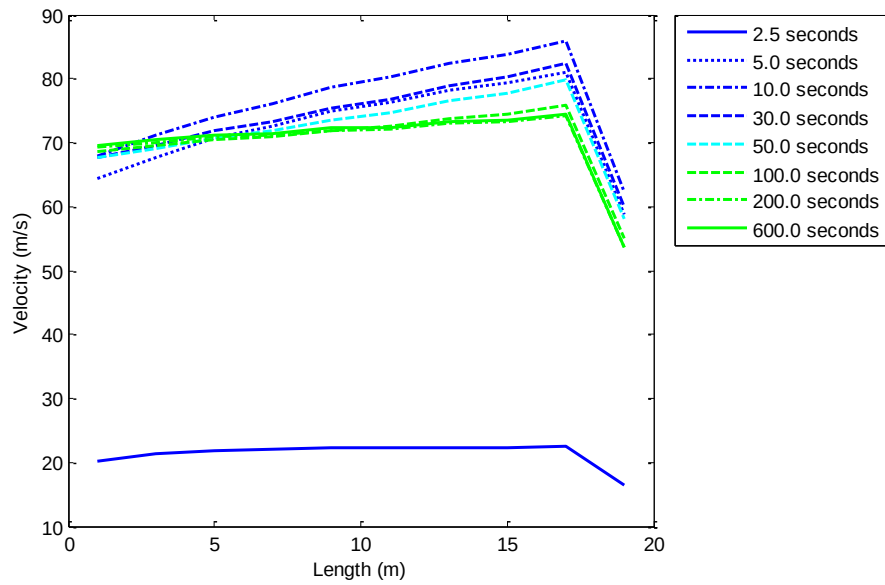


Figure 6.4 – Velocity Profiles of the Cold Leg with Time

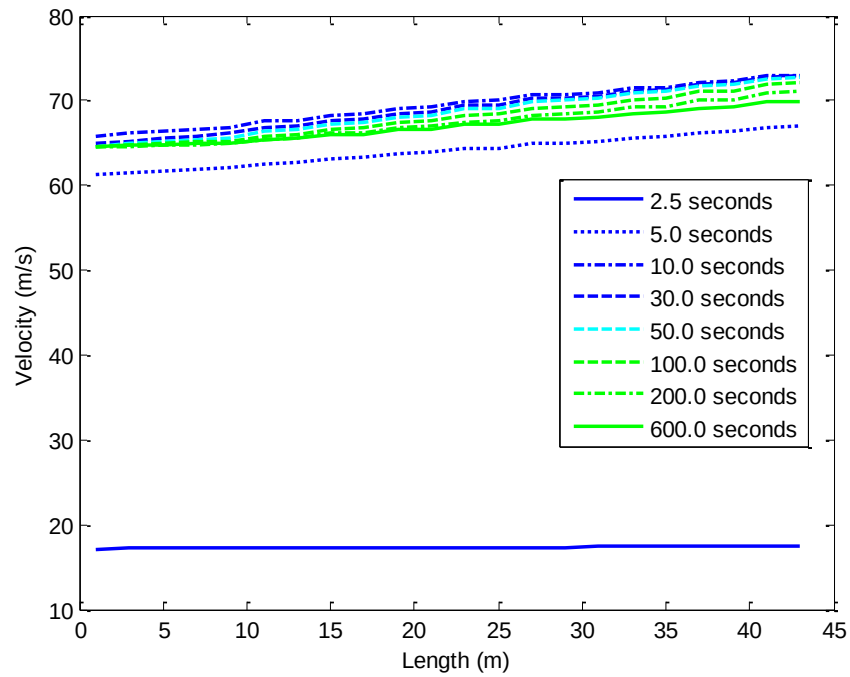


Figure 6.5 – Velocity Profiles of the Warm Leg with Time

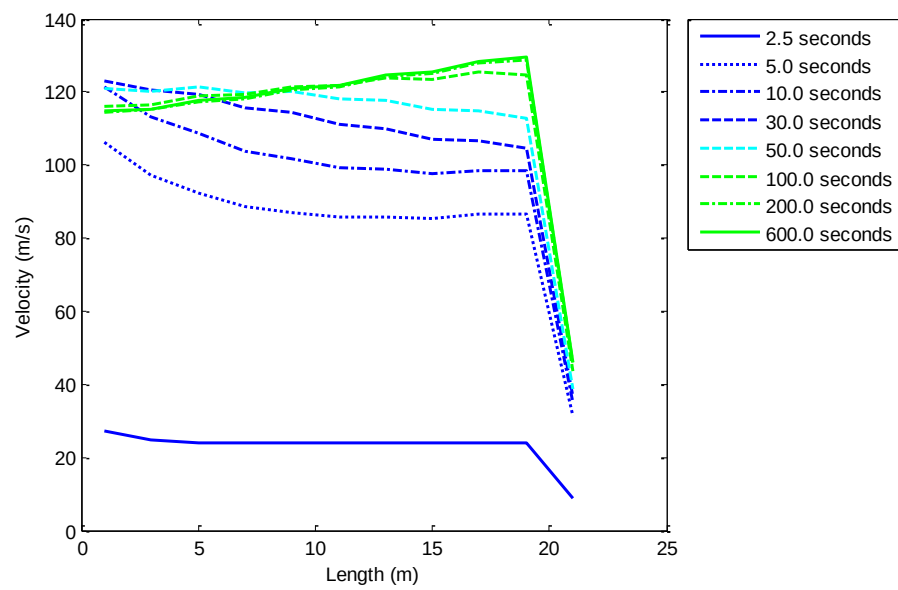


Figure 6.6 – Velocity Profiles of the Hot Leg with Time

changes that take place are mainly due to the system coming into thermal equilibrium. The effects of both friction and energy addition increase the velocity down the length of the duct. As more energy from the wall is transferred it is found that the profile line tends to be lowered by as much as 20 m/s. This is due in part to the density changing as heat is added to the flow. Since the mass flow is relatively constant after the first five seconds, see Figure 6.7, as the density is increased the velocity must be lowered. The velocity profile of the “warm” leg follows the same trends as the “cold” leg with the exception that small changes in velocity can be seen due to the bends in the ducting. These are also present in the other profiles but are more prevalent due to the scaling of the plots. The velocity profile in the “hot” leg goes through more changes. In the first 50 seconds, a lot of energy is taken out of the flow to heat up the ducting walls. This results in steep changes in the flow velocity. Once the wall has come closer to its final temperature, it is found that the profile behaves more like that in the warm and cold legs. However, since energy is leaving the flow, the heat transfer is acting to slow down the fluid. The velocity still increases, however, due to the effects of friction and the minor loss of the bends. In this situation, momentum losses have a larger effect on the velocity than the effects of heat transfer.

Figures 6.8, 6.9, and 6.10 display time slices of the total pressure profiles for the “cold,” “warm,” and “hot” legs respectively. Similarly to the velocity profiles, the overall shapes of the profiles are established within the first few seconds. Gradual changes take place over time as energy is added to, or removed from, the flow. In both Figures 6.8 and 6.9, the total pressure decreases as energy is added. In Figure 6.10 the profile rises as energy is transferred out of the fluid. From nodes 10 to 11 there is a sharp increase in total pressure, this is in part due to the expansion slowing down the velocity at the same time that energy is taken out of the flow. If heat transfer wasn’t taking place, the pressure would continue to decrease in the expansion due to the effects of friction. However, in this case heat transfer

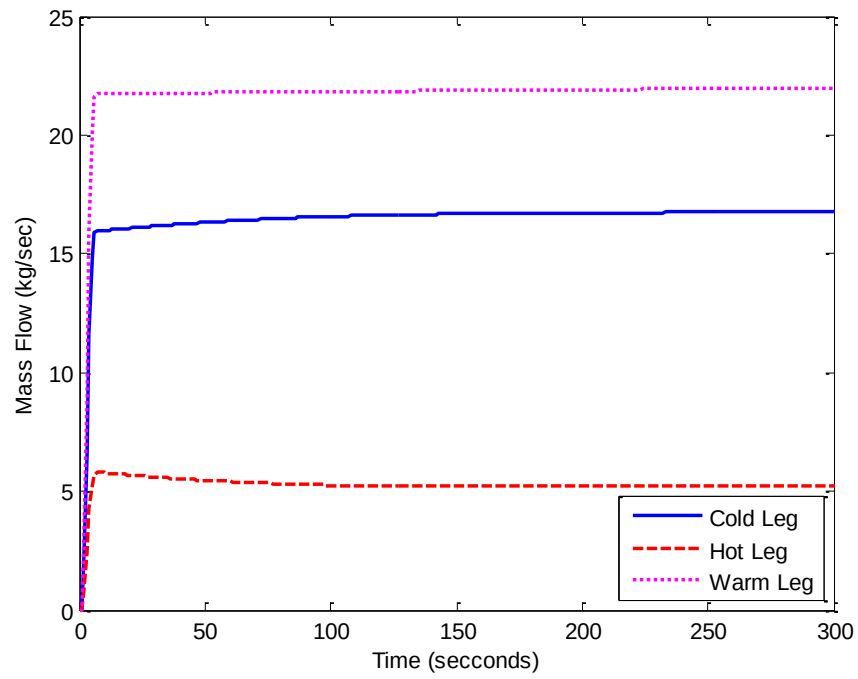


Figure 6.7 – Time History of the Mass Flow Exiting Each Leg

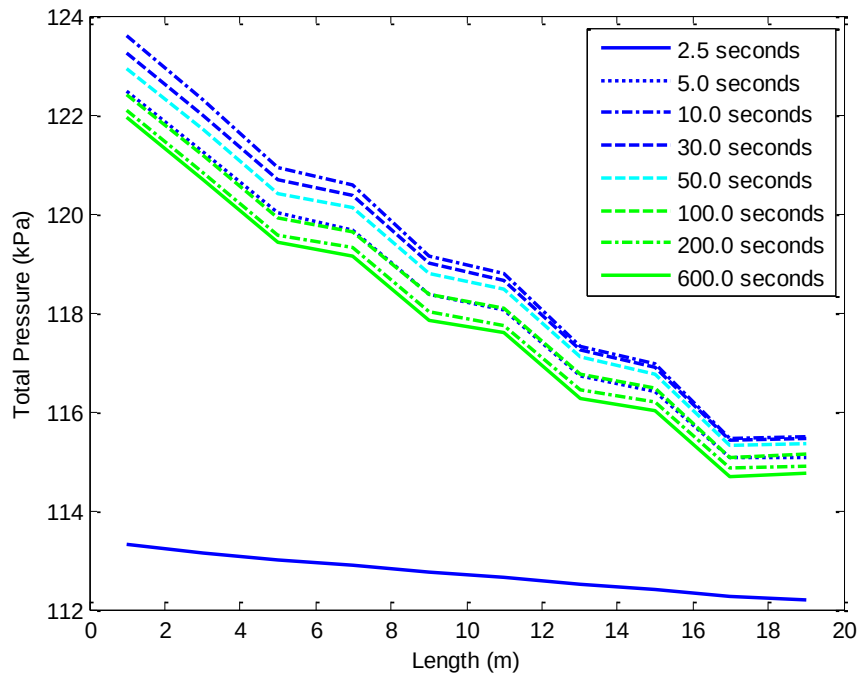


Figure 6.8 – Total Pressure Profiles of the Cold Leg with Time

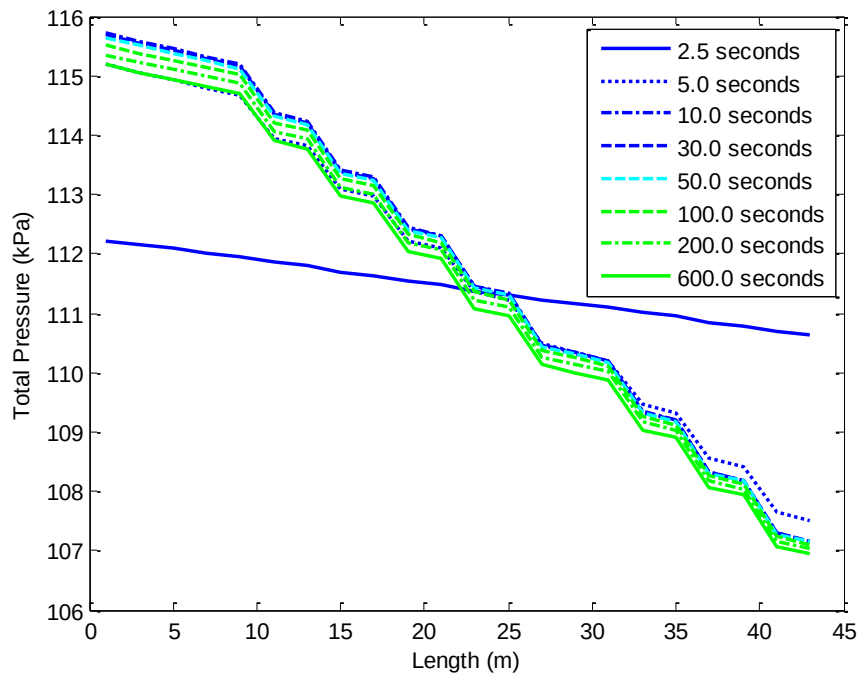


Figure 6.9 – Total Pressure Profiles of the Warm Leg with Time

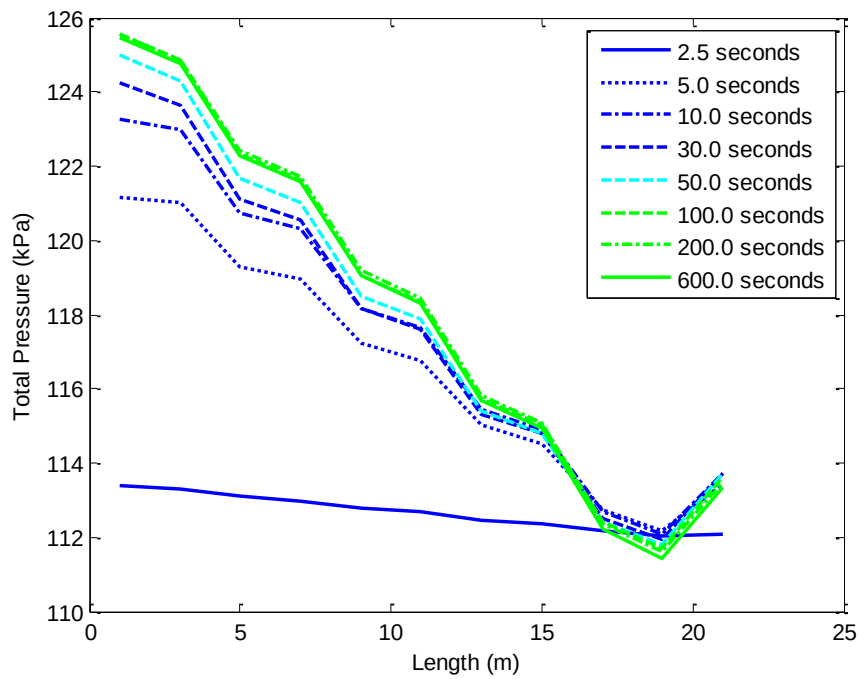


Figure 6.10 Total Pressure Profiles of the Hot Leg with Time

has a larger effect. In all three pressure profile plots, the effects of the bends are noticeable by the sharp decreases in pressure.

Plots of the temperature profiles varying with time are presented in Figures 6.11, 6.12, and 6.13. Figure 6.11 is a series of temperature profiles of the “cold leg.” Until the duct wall begins to approach the temperature of the fluid, energy is transferred into the flow as can be seen by the time slices up to approximately 200 seconds, shortly after this equilibrium has been reached. Figure 6.12 displays the temperature changes that take place inside the “hot leg.” Its temperature transformation is similar to that on Figure 6.11 except that the fluid temperature is increasing with time, and its equilibrium is reached at roughly the same time as for the “cold leg.” The “warm leg,” Figure 6.13, however, reaches a final temperature much later, it’s not until after 500 seconds that its profile stops changing significantly. This is due to the relative temperature and velocity of the fluid entering the duct. There is less of a temperature difference between the fluid and the duct wall, combined with lower velocities the heat transfer rate is lower than in the other legs. Figure 6.14 displays time change of the flow total temperature as it is leaving each of the legs. The time it takes each to reach a steady-state value can be seen in this figure.

The above analysis shows that even with all the features of the quasi-one-dimensional control volume working together, the physics of the fluid behaves as one would expect. The next subsection examines how well the simulation performs in real-time.

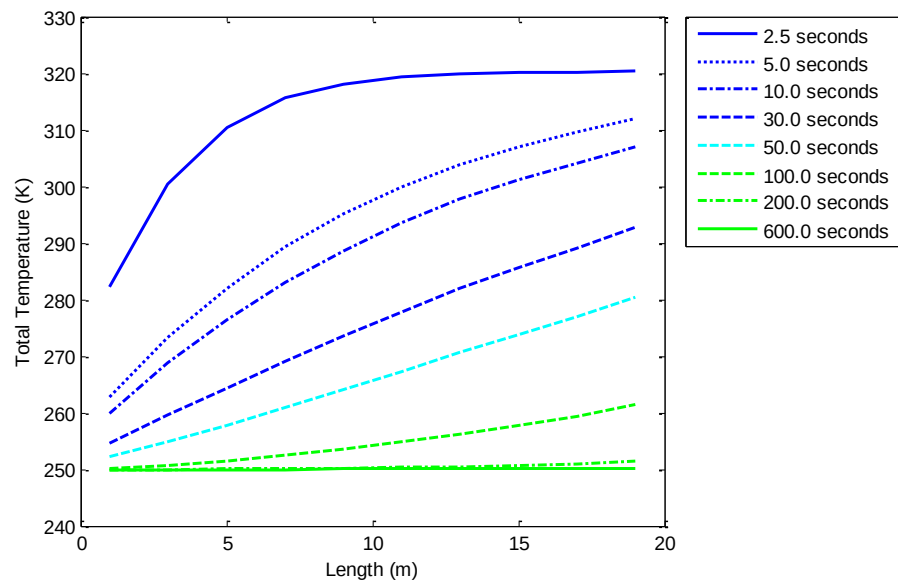


Figure 6.11 – Total Temperature Profiles of the Cold Leg with Time

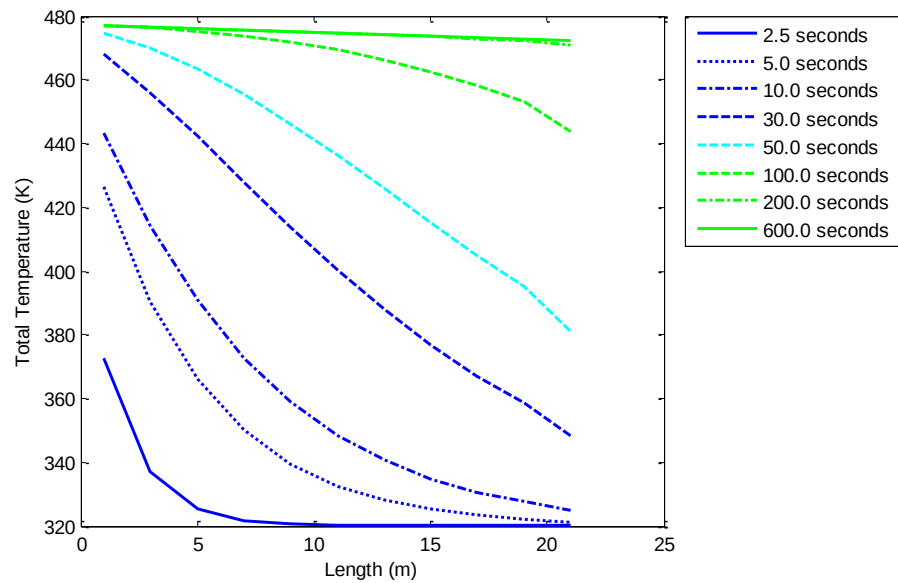


Figure 6.12 Total Temperature Profiles of the Hot Leg with Time

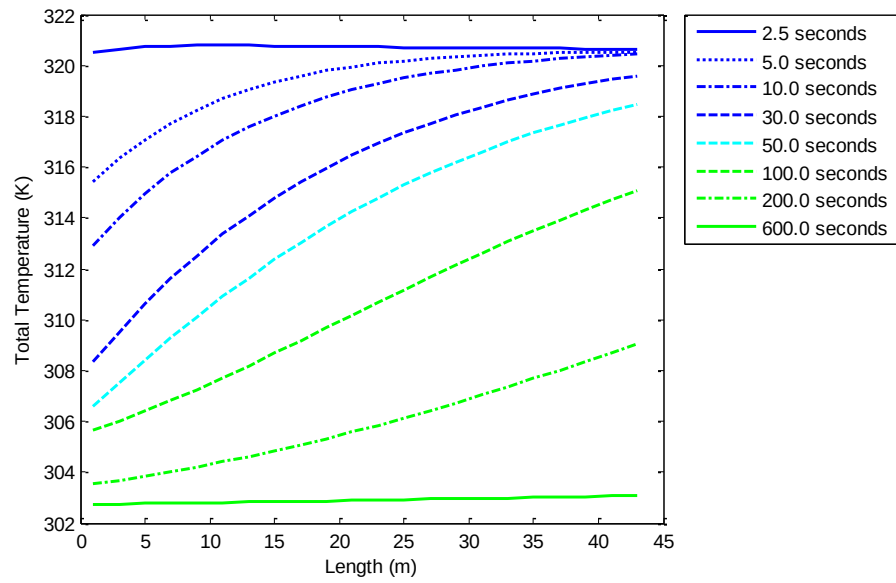


Figure 6.13 Total Temperature Profiles of the Warm Leg with Time

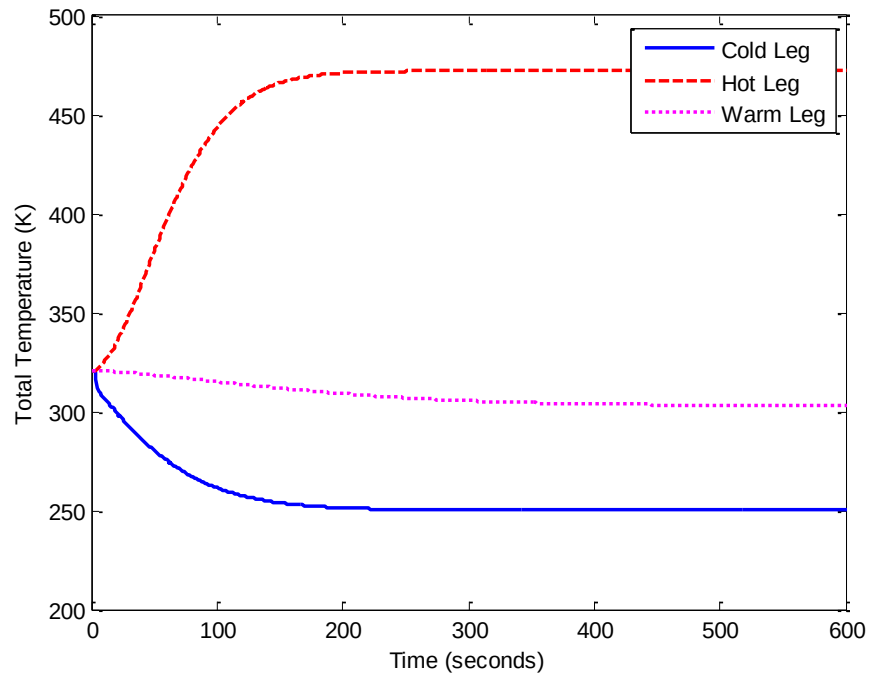


Figure 6.14 Time History of the Total Temperature Exiting Each Leg

6.3 Real-time Results of Example Network

The simulation of the example network given by Figure 6.1 was also run in real-time to gauge its performance. The model was compiled using the method outlined in section 5.5 and run on the same PC as the analysis in that section. Figure 6.15 displays the task execution time of the model as it compares to Figure 5.32. The star on the plot denotes the TET of the current simulation. It is found to be close to the line of the previous data but does not fall exactly on it. This is due to the fact that the example simulation has to perform other computations besides just the control volumes discussed before: several other blocks are now included that were not part of the previous study. These include the junction block and the three valves, as well as the pressure sources and sinks. What the figure does show is that the additional blocks do not add a considerable amount of execution time to the simulation. The additional amount of time required is not unexpected and it may be assumed that if these additional blocks were not present then the TET would be in line with the previous findings. The following subsection compares the results of the quasi-one-dimensional control volume with these from a lumped-parameter control volume analysis.

6.4 Comparison Between the Quasi-One-Dimensional Code and the Lumped-Parameter Method

The network in Figure 6.1 can be converted into a model using the lumped-parameter method described in section 1. The results from this simulation can be compared to the model built in Figure 6.2 using the new quasi-one-dimensional control volume. The single most notable difference in the models is that the three legs of the network are combined into a single control volume. The output of the parameters will be uniform so that profile plots cannot be used for a comparison. Instead the uniform properties will be compared to the output values in each leg. The results from the lumped-parameter

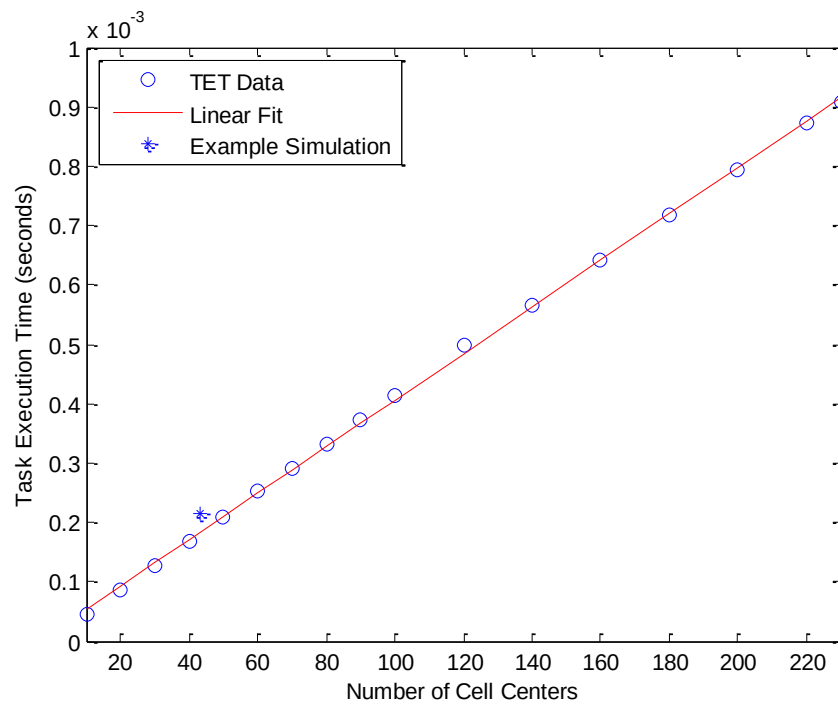


Figure 6.15 Task Execution Time of the Sample Model

control volume should compare closely with the output of the “warm leg” in Figure 6.1. If a comparison of the total pressures in Figure 6.16 is made, it is immediately noticed that the pressure better corresponds with the “cold leg.” However, it only has a difference of 8.01% from the “warm leg.” Also, the settling time appears to be very similar. A major difference does exist in the transient response at the start of the simulation.

The pressure in the lumped-parameter control volume briefly increases before falling back down. The opposite is true of the pressure inside the “warm leg,” it starts with a sharp drop in pressure. Both the “cold” and “hot” legs, however, experience sudden increases in pressure at the start. These results suggest that, depending on where sensors are placed in the facility, a false sense of transient response could be obtained. Considering the temperature plots in Figure 6.17, it is seen that there is a much better correlation between the total temperature in the lumped-parameter control volume and the “warm leg.” They differ by only 2.05%. However, the older method gives no information about what is happening in the other two legs. A sense of the steady-state temperature, or the final temperature of the ducting wall, would not be obtained. Again, depending on where feedback devices are placed, the lumped-parameter method could provide a false sense of facility performance.

One other property to look at is the mass flow through the models. Figure 6.18 provides the mass flow exiting in each leg and the mass flow calculated by the lumped-parameter method. It is evident that the latter method over predicts the mass flow by 53.9%, when compared with the mass flow exiting the “warm leg.” This is a significant difference and its consequences could be far reaching. Besides overstating the mass flow capabilities, it could lead to the inclusion of oversized valves if the model were used to determine design requirements. The over prediction is because the lumped-parameter method assumes a uniform pressure distribution in the control volume. In reality, a pressure

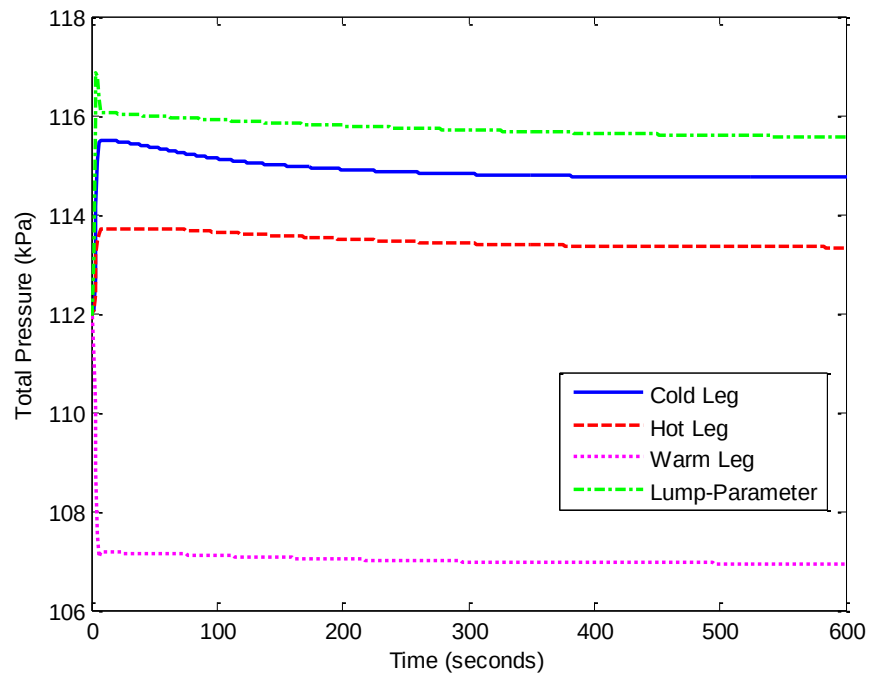


Figure 6.16 Time History of Exit Total Pressures Comparing the Lumped-Parameter Method

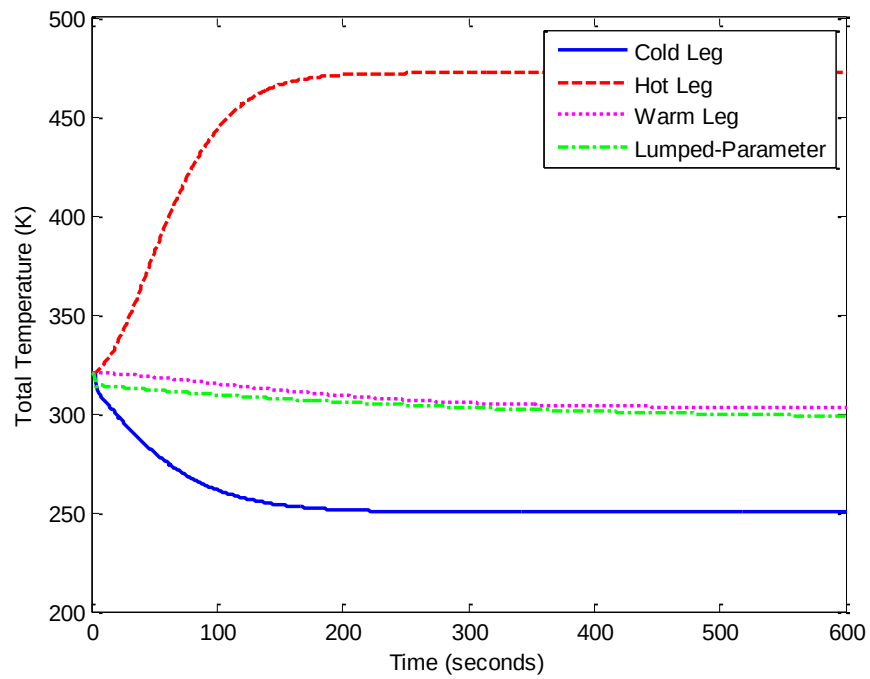


Figure 6.17 Time History of Exit Total Temperatures Comparing the Lumped-Parameter Method

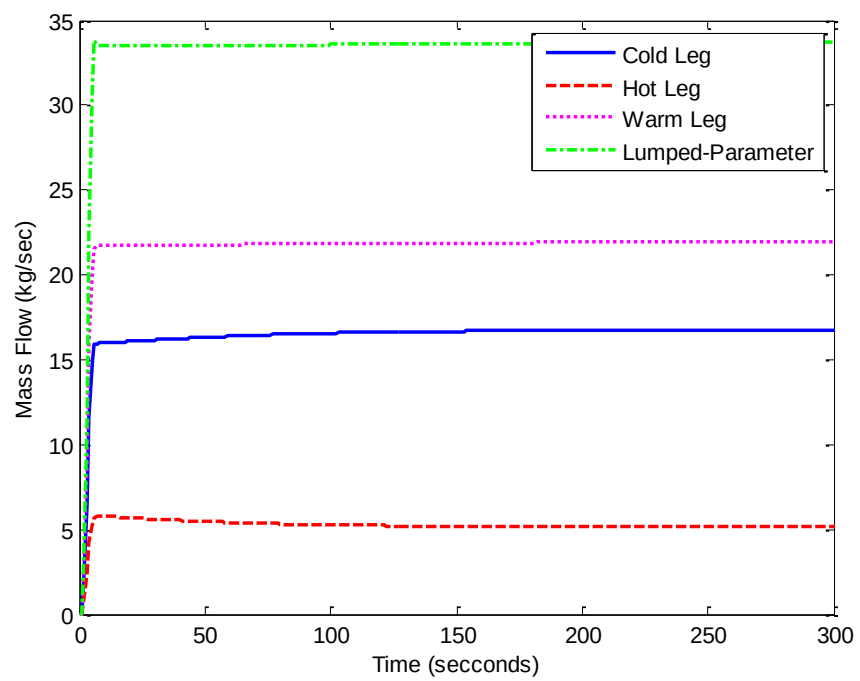


Figure 6.18 Time History of Exit Mass Flows Comparing the Lumped-Parameter Method

gradient exists from high to low and the value that is calculated by this method can be considered an average value. Thus, the mass flow devices, such as valves, encounter a much larger pressure change than would normally be present.

The coarse grid and time-step required to run the quasi-one-dimensional model in real time was still found to be adequate to produce realistic and meaningful results. The system in Figure 6.1 demonstrated the ability of the code to combine several different flow phenomena in a practical example. The results of the new method were then compared to the simple lumped-parameter control volume. It was found that the two different approaches had similar agreement for the steady-state total pressure and temperature. However, the transient response differed significantly as noted above. Since the lumped-parameter method lacks any physical dimension, it should be noted that depending on where physical plant feedback devices are located, results between the real world and a lumped-parameter model could vary. The mass flow estimated by the older method differed by more than 50% from the quasi-one-dimensional approach. This has the biggest potential to cause disagreement between the actual mass flow of the plant and what is predicted by the model. This could lead to a false sense of plant performance or improperly sized equipment. The quasi-one-dimensional control volume is built around spatial effects and its ability to provide accurate results should result in an improved method of modeling.

7. Conclusion

Higher fidelity facility simulations are now possible with the new quasi-one-dimensional control volume that has been presented herein. Several new capabilities exist that were previously unattainable with the lumped-parameter control volume. The code developed for this effort allows for time accurate solutions to ducting problems with friction, minor momentum losses, changes in duct cross sectional area, and localized heat transfer effects. Because the new method uses a discretized grid to solve the equations, flow properties are now solved at a local level rather than a global one. This important difference from the older method allows for the modeling of process lag (transients) and the calculation of streamwise profiles of fluid properties. Section 5 detailed the ability of the quasi-one-dimensional control volume method to accurately calculate steady-state results. Its solutions were compared against known analytical solutions to Fanno, Rayleigh, and isentropic flow problems. The grid sizes and time-steps used only affected the solutions to problems with varying area. This is primarily due to the use of a staggered grid. Since not all flow properties are known at a given location, certain averaging has been incorporated. It was found that for conditions with gradual gradients, this effect only produced moderate errors. However, error grows as the local Mach number approaches Mach one. Since the upper Mach number limit in typical ducting is around 0.2 to 0.3, the errors found at these speeds were deemed acceptable. This, of course, is not true in the test section where the flow may be transonic. The present method does not apply under these conditions.

A direct comparison between the lumped-parameter method and the quasi-one-dimensional control volume has been made. It was found that the latter proved more accurate for simulating transient response, as expected, but an astonishing result was found in the mass flow comparison. The lumped-parameter control volume over predicted the state-steady mass flow by more than 50%. This is

directly due to the uniform pressure that is calculated by this method. More accurate mass flows are obtained when pressure is allowed to vary naturally along the length of the duct. The result is that the mass flow devices upstream and downstream of the volume experience a smaller pressure difference, unlike that computed by the lumped-parameter method. In addition, varying flow properties, such as pressure, temperature, and Mach number, provide a more realistic representation of the facility to the controls engineer or operator in training.

Several opportunities exist for future developments. The single disadvantage of the quasi-one-dimensional control volume is that it is currently limited to only dealing with properties of air. In testing facilities controlling the humidity can be just as important as controlling the pressure and temperature. To accurately simulate the water vapor content additional equations must be added to keep track of the mass fractions and to calculate specific heats of the mixtures. In addition to adding water vapor, it would be beneficial to account for other species. During certain types of rocket testing, nitrogen is often pumped into the ducting to render the gases oxygen deficient. The byproducts of the rocket exhaust, when mixed with proper amounts of oxygen, can become flammable. If properly modeled, a simulation can be used to predict the amount of mixing taking place.

Another area of opportunity would be to create, or explore options for, off the shelf real-time operating systems that take advantage of parallel computing. The quasi-one-dimensional code is designed to run in real-time, but it might be an unrealistic expectation to be able to simulate an entire facility using this method and have it run on a single CPU. Computer hardware technology has continued to progress and it is often difficult to buy a PC with only a single core. It would be worth investing the time and effort to develop parallel computing, in combination with the quasi-one-dimensional control volume, to advance the state of the art in facility simulation.

References

1. United States Air Force, Arnold Engineering Development Center. *Test Facilities Handbook*. 13th ed. Arnold AFB: AEDC, 1992.
2. Thelen, Michael, Steve Salita, and Greg Halliburton. *Enhanced High Speed Digital Interface between Control Systems and Simulated Facility Models*. ISA 53rd International Instrumentation Symposium, Tulsa, OK, April 2007.
3. Montgomery, Peter A., Rick Burdette, and Brian Krupp. *A Real-Time Turbine Engine Facility Model and Simulation for Test Operations Modernization and Integration*. 2000-GT-0576, ASME TurboExpo, Munich, Germany, May 2000.
4. Klepper, Jason, Scott Sisk, Stephen Salita, and Rick Burdette. *Development of a Facility Simulator for Implementation of an Automated Control System*. AIAA 2002-2785, AIAA 22nd Aerodynamics Measurement Technology and Ground Testing Conference, St, Louis, MO, June 2002.
5. Dabney, James B., and Thomas L. Harman. *Mastering Simulink*. Upper Saddle River: Pearson Prentice Hall, 2004.
6. MathWorks. "Simulink – Simulation and Model-Based Design." *Mathworks.com*. MathWorks, n. d. 20 September 2010.
7. Sheeley, J. M., D. A. Sells, and L. B. Bates. *Experiences with Coupling Facility Control Systems with Control Volume Facility Models*. AIAA 2004-932, 42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 2004.
8. Sheeley, J. M., S. Salita, B. Boylston, M. Thelen, and M. Hamby. *Testing of a Model-Based Predictive Control System in a Transonic Aerodynamic Test Facility*. AIAA 2010-1309, 48th AIAA Aerospace Sciences Meeting, Orlando, FL, January 2010.

9. Montgomery, Peter A., Rick Burdette, Larry Wilhite, and Steve Salita. *Modernization of a Turbine Engine Test Facility Utilizing a Real-Time Facility Model and Simulation*. 2001-GT-0573, ASME TurboExpo, New Orleans, LA, June 2001.
10. Zucker, Robert D., and Oscar Biblarz. *Fundamentals of Gas Dynamics*. 2nd ed. Hoboken: John Wiley & Sons, 2002.
11. Anderson, John D. *Computational Fluid Dynamics, The Basics with Applications*. New York: McGraw-Hill, 1995.
12. Bejan, Adrian. *Convection Heat Transfer*. New York: John Wiley & Sons, 1995.
13. Maron, Melvin J., and Robert J. Lopez. *Numerical Analysis: A Practical Approach*. 2nd ed. Belmont: Wadsworth, 1991.
14. John, James E. A. *Gas Dynamics*. Upper Saddle River: Prentice-Hall, 1984.
15. Crane Co. *Flow of Fluids Through Valves, Fittings, and Pipe*. The Woodlands: Crane Co., 1988.
16. Blevins, Robert D. *Applied Fluid Dynamics Handbook*. 3rd ed. Malabar: Krieger Publishing, 2003.
17. Kinsler, Lawrence E., Austin R. Frey, Alan B. Coppens, and James V. Sanders. *Fundamentals of Acoustics*. 4th ed. Hoboken: John Wiley & Sons, 1999.
18. Thomas, Philip. *Simulation of Industrial Processes for Control Engineers*. Woburn: Butterworth-Heinemann, 1999.

Vita

Brett Boylston was born in West Columbia, SC, to the parents of Gil and Sandy Boylston. He has one older brother who now lives in NC with his wife and three children. After attending elementary and middle school in Blythewood, SC, Brett moved to Spring, TX, where he attended Klein Oak High School. Upon graduation he set his sights on the Georgia Institute of Technology where he earned his Bachelor's of Science in Aerospace Engineering in 2004. Promptly afterwards, he joined the Arnold Engineering Development Center near Tullahoma, TN. In 2005 he applied to the University of Tennessee Space Institute to pursue a Masters of Science in Aerospace Engineering. After many years of hard study he graduated in 2011.