Development and Testing of the Universal Aerosol Conditioner

by

Kerry Chen

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Mechanical Engineering
University of Alberta

Abstract

A general-purpose aerosol conditioning device called the Universal Aerosol Conditioner has been designed and tested. The device can condition the aerosol in multiple ways. It can dilute the aerosol with particle-free air, dilute only the gas-phase component of the aerosol without diluting the particle concentration, dilute all components of the aerosol, or denude the aerosol by removing material from the particle phase. The UAC can also humidify or dehumidify an aerosol by either using vapour or dry air respectively. The UAC accomplishes these processes by bringing the aerosol into contact with sheath air and allowing enough time for the gas-phase of the aerosol to diffuse into the sheath flow. Both flows are kept in the laminar regime and are controlled. A theoretical model was derived to assess the theoretical performance of the UAC and was numerically solved. From the model it was determined that two parameters dictated the rate of diffusion between the two flows: the Péclet number and the ratio of sheath-to-aerosol flow rate. A prototype was designed, built, and experimentally validated. The prototype was used to measure the particle penetration efficiency and the gas dilution factor at various particle sizes and flow conditions. The results showed that at low aerosol and sheath flow combinations, the prototype behaved closely to the theoretical model but diverged from the theory once the sheath flows were increased due to mixing between the two flows.

Table of Contents

1 Introduction and Background **1**
1.1 Dilution . . . 1
1.2 Denuding . . . 6
1.3 Humidification/Dehumidification . . . 9
1.4 Motivation . . . 10
1.5 Overview . . . 11

2 Universal Aerosol Conditioner **12**
2.1 UAC Operational Modes . . . 14
2.1.1 Gas dilution without particle dilution . . . 14
2.1.2 Combined particle and gas dilution . . . 15
2.1.3 Denuding . . . 16
2.1.4 Humidification . . . 17

3 Theoretical Modelling of the Universal Aerosol Conditioner **18**
3.1 Diffusion Model . . . 18
3.2 Numerical Modelling of the Dimensionless Convection - Diffusion Equation . . . 25
3.2.1 Numerical Model using the Crank-Nicolson Method . . . 25
3.2.2 Numerical Solution Using MATLAB's bvp4c Solver . . . 29
3.2.3 Code Execution Performance Comparison Between the Two Numerical Methods . . . 30
3.3 Theoretical Performance of the UAC . . . 32
3.3.1 Visualizing Diffusion in the UAC . . . 32
3.3.2 Particle Penetration Performance . . . 34
3.3.3 Gas Dilution Performance . . . 35
3.4 Summary . . . 36

4 Experimental Validation of the Universal Air Conditioner **37**
4.1 Prototype Design . . . 37
4.2 Experimental Setup . . . 39
4.2.1 Particle Penetration Testing . . . 39
4.2.2 Experimental Procedure for Particle Testing . . . 43
4.2.3 Gas Dilution Testing . . . 44
4.2.4 Experimental Procedure for Gas Dilution Testing . . . 44

4.3 Experimental Results and Discussion 45
4.3.1 Correcting for Unaccounted Diffusional Losses 45
4.3.2 Particle Penetration Results 48
4.3.3 Gas Dilution Results 54
4.4 Summary 56

5 Conclusions **57**
5.1 Theoretical Conclusions 58
5.2 Experimental Conclusions 58
5.3 Future Work 60

References **61**

A Analytical Solution to the Particle-Phase Partial Differential Equation **66**
A.1 Particle-Phase Analytical Solution 67

B MATLAB Implementation of the Crank-Nicolson Finite Difference Method **72**

C MATLAB Implementation of the Implicit Euler Finite Difference Method **79**

D Controlling the UAC with an Arduino Microcontroller **86**

E Drawing Package for the UAC **103**

List of Tables

3.1 Comparison between Crank-Nicolson and Euler-Implicit (bvp4c) methods 31

4.1 Lengths for each sampling line segment 46

A.1 Graetz problem eigenvalues and coefficients using the Kummer's function solution. 69

List of Figures

1.1 An example of a dilution tunnel. 3
1.2 Giever's boundary layer diluter (Giever, 1974). 4
1.3 Operating principle of ejection diluters (Dekati, 2017). 4
1.4 Rotary diluter schematic (TSI Incorporated, 2015). 5
1.5 Diffusion denuder (Cheng, 2011). 7
1.6 Hagino's counter-flow denuder (Hagino, 2016). 8

2.1 A preliminary schematic of the UAC 13
2.2 Simplified cross-sectional view of the dilution chamber 13
2.3 Sheath and aerosol flows for gas dilution without aerosol dilution. . . 15
2.4 Sheath and aerosol flows for combined particle and gas dilution. . . . 16
2.5 Sheath and aerosol flows for denuding configuration. 16

3.1 Aerosol outermost streamlines as they enter and exit the dilution chamber 20
3.2 Region that was modelled 22
3.3 Mesh used for Crank-Nicolson discretization 26
3.4 Concentration maps of the particle and gas-phases of the aerosol at different Pe 33
3.5 Theoretical particle penetration at different Pe and sheath-to-aerosol ratios 34
3.6 Theoretical gas dilution factor at different Pe and sheath-to-aerosol ratios 35

4.1 Schematic of the updated UAC prototype dilution chamber. Not to scale. 38
4.2 Experimental setups for testing the particle penetration through the Universal Aerosol Conditioner (UAC). Particles were classified using an Aerodynamic Aerosol Conditioner (AAC) and a Differential Mobility Analyzer (DMA). Particle concentrations were measured using a Condensation Particle Counter (CPC). 40
4.3 Experimental setup for testing the Universal Aerosol Conditioner's (UAC) dilution performance. A Condensation Particle Counter (CPC) was used as a suction pump for the sampling line. 44
4.4 Schematic of the Sampling Line 46
4.5 Experimental η compared to the theoretical model. Flow ratio = 10 and Q_a = 0.3 LPM. Corrected for additional diffusional losses in the sampling line. 49

4.6 Experimental η compared to the theoretical model. Flow ratio = 8.67 and Q_a = 1.5 LPM. Corrected for additional diffusional losses in the sampling line. 51
4.7 Particle Penetration for d_B = 1050 nm as a function of Q_{sh} with Q_a = 1.5 LPM. 52
4.8 Particle Penetration for d_B = 1050 nm as a function of Q_{sh} with Q_a = 0.3 LPM. 53
4.9 Experimental CO_2 DF compared to the theoretical model at various Q_{sh} with Q_a = 0.3 LPM. 55

D.1 The UAC's Graphical User Interface 102

Nomenclature

Acronyms

AAC	Aerodynamic Aerosol Classifier
CHF	Confluent Hypergeometric Function
CO_2	Carbon Dioxide
CPC	Condensation Particle Counter
DMA	Differential Mobility Analyzer
GUI	Graphical User Interface
HF	High Flow
I^2C	Inter-Integrated Circuit
LF	Low Flow
LPM	Liters per Minute
PDE	Partial Differential Equation
PID	Proportional-Integral-Derivative
RH	Relative Humidity
SPI	Serial Peripheral Interface
SOV	Separation of Variables

Symbols

α	Level of significance
β_n	Eigenvalues
η	Particle penetration efficiency
μ	Dynamic viscosity
ν	Dimensionless particle or gas concentration
ν_0	Dimensionless initial concentration
ρ	Dimensionless radius
ρ_a	Dimensionless aerosol bounding radius
ρ_0	Standard density of a particle (1 g/cm^3)
ρ_e	Effective density of a particle
φ	Bend angle
τ	Particle relaxation time
ξ	Dimensionless parameter
ζ	Dimensionless length
A_n	Coefficients for the Graetz solution
B_{90}	Bend in Sampling Line
C_c	Cunningham correction factor
c	Particle migration velocity
c_i	confidence interval
D	Diffusion coefficient
DF	Dilution factor
d	Diameter
d_a	Aerodynamic diameter of a particle
d_B	Mobility diameter of a particle

d_{p}	diameter of a particle
h	Step size in ρ
k	Step size in ζ
L	Dilution tube length
L_{s}	Tube length of sampling line segment
p	Number of samples
Pe	Péclet number
P_x	Precision uncertainty
N	Degrees of freedom
n	Particle or gas concentration
n_0	Initial particle or gas concentration
$n_{\mathrm{CO_2,ambient}}$	Ambient CO_2 concentration
$n_{\mathrm{CO_2,UAC}}$	CO_2 concentration downstream of the UAC
n_{upstream}	Particle concentration upstream of the UAC
$n_{\mathrm{downstream}}$	Particle concentration upstream of the UAC
Q	Volumetric flow rate
Q_{a}	Aerosol volumetric flow rate
Q_{exh}	Exhaust volumetric flow rate
Q_{s}	Sample volumetric flow rate
Q_{sh}	Sheath volumetric flow rate
Q_{t}	Total volumetric flow rate
R	Dilution tube inner radius
R_n	Dimensionless Radial Eigenfunction
r	Radius/radial direction

r_{a}	Aerosol bounding radius
S_x	Standard deviation of the sample
t	Time
$t_{\alpha/2,N}$	t-score
u	Velocity
u_z	Velocity in the axial direction
u_{avg}	Average velocity
Stk	Stokes number
z	Axial direction

Chapter 1

Introduction and Background

Interest in aerosol research has increased in recent years due to the realization that aerosols have a significant effect on global climate change (Boucher et al., 2013; Fiore et al., 2015) and have been linked to health issues in humans (Anderson et al., 2012; Kim et al., 2015). As such, it is important to be able to quantify the physical properties of aerosols and how they are formed to better understand the effects they have on global climate change and health issues in humans.

An aerosol is defined as a solid or liquid particle that is suspended in a gas. The particles in typical aerosols of interest are on the order of 1 nm to 10 microns in diameter. Aerosol instrumentation and measurement is an important part of aerosol research. Most aerosols need to be conditioned in some way before measurement. Typical aerosol conditioning processes include dilution, denuding, and humidification.

1.1 Dilution

Dilution is a process where additional gas (typically air) is added to the aerosol to reduce the concentration of particles and gas-phase species. Dilution is used for three purposes in aerosol research. Firstly, dilution is used to reduce the particle concentration in the aerosol. This is necessary in cases where the particle concentration of the aerosol is too high to be measured (*i.e.*, above the range of the instrument), or

when it is necessary to avoid premature fouling of the measurement instrument. In these cases, the aerosol is diluted before it is sampled into the measurement device. Secondly, dilution is used to dilute the concentration of the gas-phase components in the aerosol to prevent condensation in the aerosol instrument. For example, water vapour produced from a combustion process will condense in sampling lines or in the instruments when the temperature of the aerosol reaches room temperature. Thirdly, an aerosol may be diluted to reduce the rate of coagulation of the particles. Dilution can be achieved using a variety of different methods.

Dilution tunnels are an example of a relatively simple dilution device. Dilution tunnels can vary in design but they operate on one simple principle: they mix the aerosol flow with a dilution gas in an elongated tube or tunnel, *i.e.* the two flows are mixed into one. Typically, air is used as the dilution gas but other gases may be used as well. The most basic dilution tunnel consists of a single tube where the dilution gas and aerosol are drawn in simultaneously. An orifice is sometimes located downstream of the aerosol injection point to help induce turbulent flow. An example of a basic dilution tunnel is shown in Figure 1.1. Examples of this dilution tunnel implementation can be found in Ström et al. (1995a), Ström et al. (1995b), and Yamasaki et al. (1992). The dilution system described by Yamasaki et al. is a pressure monitored controlled feedback dilution tunnel. Pressure loss is monitored in the dilution tunnel. When this pressure loss fluctuates, the aerosol flow rate will vary accordingly. To keep the aerosol to dilution air ratio constant, additional pressurized dilution air is injected through nozzles into the dilution tunnel into the region just downstream of the aerosol inlet pipe.

Some tunnel designs are more elaborate. For example, instead of a single tube, the tunnel may consist of two coaxial cylinders where the aerosol flows through one cylinder while dilution air is brought through the other. Typically, the innermost tube is made of mesh or screen material that allows a portion of the dilution air from the

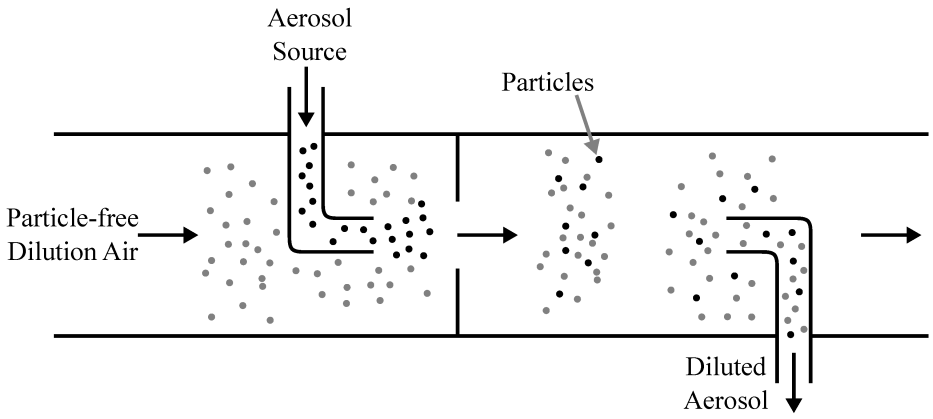


Figure 1.1: An example of a dilution tunnel.

outer annular chamber to flow radially into and combine with the inner aerosol flow. The dilution air and aerosol flows usually run in a counter-flow arrangement which promotes mixing and hence, increases the rate of dilution. Examples of these diluters can be seen in Graze Jr (1991) and Graze Jr (2009). In Graze Jr (2009), a tapered inner cylinder is used; by tapering the cylinder, the dilution ratio varies along the axis of the cylinder due to the dilution air entering the aerosol stream nonuniformly.

Giever (1974) investigated what he calls a boundary layer diluter that is similar to a multi-cylinder counter-flow dilution tunnel design. The fundamental difference between the boundary layer diluter and similar dilution tunnels is that flow within the entire diluter is kept in the laminar regime to prevent excessive particle losses. A schematic of his design is shown in Figure 1.2. Dilution air is brought into the dilution chamber through the use of a perforated tube (the bottom dilution air tube in the figure). Within the tunnel, the sample gas travels through a larger tube made of a mesh screen. The dilution air that enters the tunnel forms a boundary layer around the mesh screen tube which helps prevent particles from escaping the sample line or depositing onto the tube itself, and at the same time dilutes the water content in the sample gas.

Ejector diluters (or eductors) are also used to dilute the aerosol sample. An

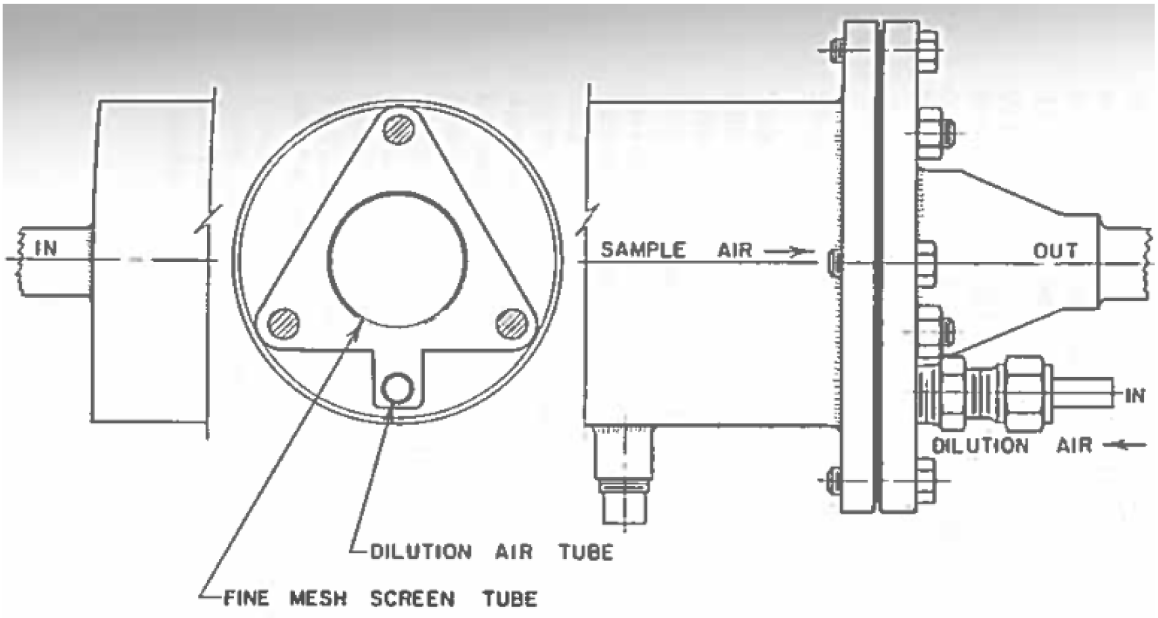


Figure 1.2: Giever's boundary layer diluter (Giever, 1974).

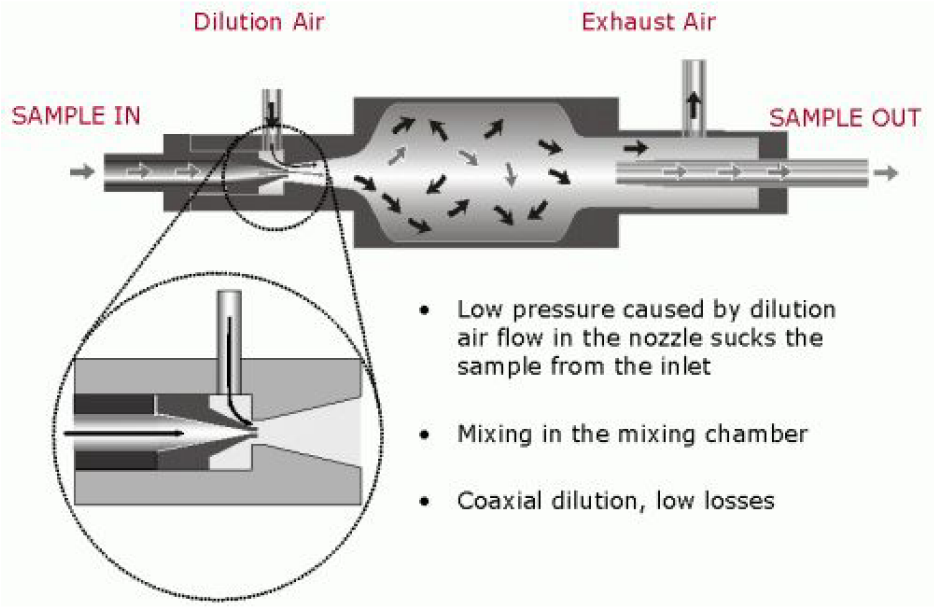


Figure 1.3: Operating principle of ejection diluters (Dekati, 2017).

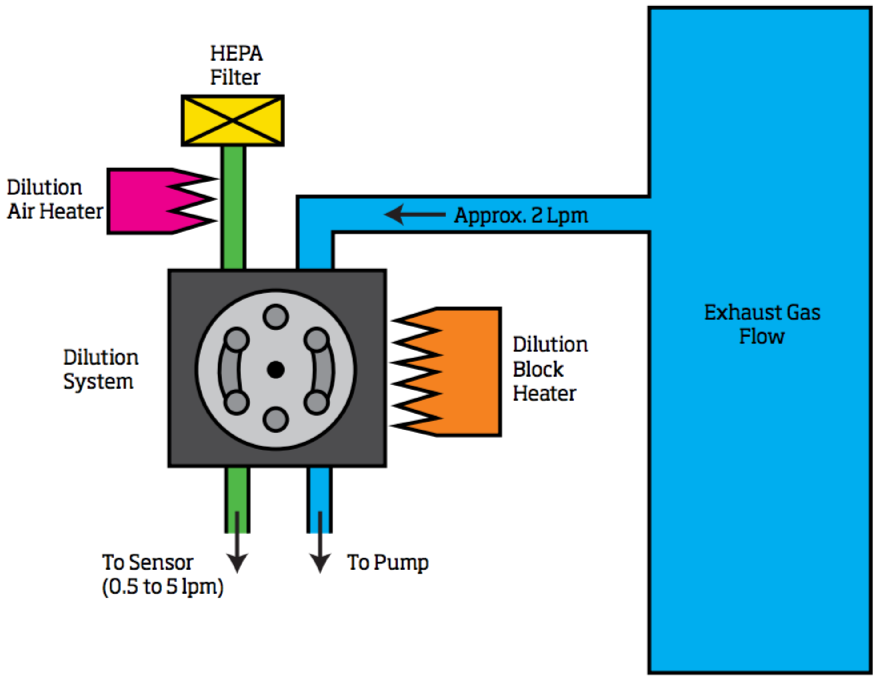


Figure 1.4: Rotary diluter schematic (TSI Incorporated, 2015).

example of an ejector diluter is shown in Figure 1.3. As seen in Figure 1.3, the aerosol stream that is to be measured is drawn through a nozzle that exits into a larger chamber. High flow, pressurized dilution air enters the diluter and flows around the nozzle. This causes a pressure drop which draws in a sample through the ejector nozzle into the mixing chamber which is then diluted as it mixes with the dilution air. This type of diluter relies on turbulent flow to dilute the aerosol sample. External pumping mechanisms are not required for the diluter to operate. Ejector dilutors are commonly used in conjunction with dilution tunnels. Examples of ejector diluters are Dekati's Diluter and High Pressure Diluter DEED-300. Additional examples can be found in Gehner et al. (2006) and Mikkaanen and Koskinen (2010).

Another class of diluters are rotating disk diluters (Bergmann and Pongratz, 2013). Figure 1.4 shows the operating principle for this dilution method. As shown in Figure 1.4, this type of dilution system uses a rotating disk and has two separate gas channels: one for the diluted measurement line, and one for the raw sampling line. As the disk

spins, a portion of the aerosol from the raw gas channel is captured within the cavities present in the disk and is then transferred to the measurement channel where it is mixed with particle-free dilution air. Due to the size and number of the cavities, the rotational speed of the disk, and the flowrate of the dilution gas, rotary disk diluters can achieve very high dilution ratios. However, due to impaction losses caused by the rotating disk, this dilution method does not work well for larger particle sizes (particle diameters greater than 1 micron). TSI Incorporated's Model 37920A Rotating Disk Thermodiluter is an example of a rotating disk diluter that is sold commercially.

It is important to note that these dilution methods cannot be used to independently reduce the particle concentration or the gas-phase concentration; both phases see their concentrations reduced simultaneously. This trait can be undesirable, as in some applications is it beneficial to only reduce the gas-phase concentration of the aerosol. For example, when measuring the exhaust from an automobile that has low particle emissions, it is desirable to reduce the concentration of the gas-phase to prevent water condensation (water vapour is a product of the combustion process in the engine). Since the particle emissions are low, it would be beneficial to keep the particle concentration at its initial concentration to improve measurement sensitivity.

1.2 Denuding

Denuding is the process of removing volatile material from the aerosol; either a volatile coating on a solid particle or volatile material existing as individual particles. This is commonly done in applications where only the solid (non-volatile) material of an aerosol is desired to be measured. Typical applications are in automotive applications where only the solid particle concentration is regulated, as in European automotive regulations, or in atmospheric science applications where researchers are interesting in measuring the particle properties of the solid particulate (e.g. "soot").

Commercial denuders include thermodenuders which consist of a heated tube (to evaporate volatile particle-phase material) and is often followed by activated carbon or a cold region to adsorb or condense the evaporated material (Burtscher et al., 2001; Wehner et al., 2002). Catalytic denuders are also used, where the heated aerosol is passed through a catalytic converter to oxidize organic material (Swanson and Kittelson, 2010), thus turning the organic material into water and carbon dioxide. Catalytic denuders have a problem of very high particle losses at small particle sizes.

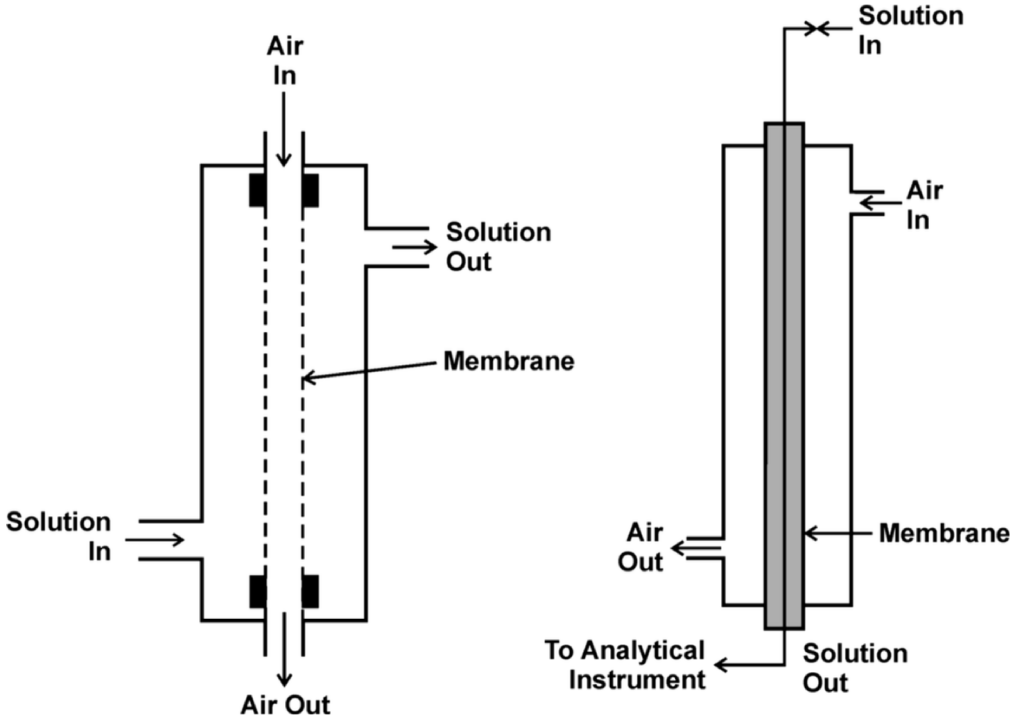


Figure 1.5: Diffusion denuder (Cheng, 2011).

Diffusion denuders are also used to denude aerosols (Kulkarni et al., 2011a) where the aerosol and dilution air are separated by a semi-permeable barrier. An example of a diffusion denuder is shown in Figure 1.5. Depending on the membrane material, certain gases can be selectively removed or introduced into the sample line through diffusion. The differences in concentrations (whether it be differences in concentrations between the particle and gas species) between the two different flows drives the

diffusion process. Gases from the aerosol will diffuse towards the dilution air, while gases in the dilution gas will diffuse towards the aerosol. The diffusion denuder does in a sense, dilute certain gas-phase species of the aerosol.

Hagino (2016) has experimented with counter-flow diffusion denuders that use concentric cylinders with two T-shaped joints to remove trace gases from the aerosol stream. The Hagino denuder comprises of two concentric cylinders where the sample aerosol is transported through the inner tube while the dilution gas moves through the annular region of the denuder. The inner tube is made from a porous membrane. Due to the differences in concentrations between the sample and purge gas streams, the porous tube allows gas species (species that have molecular sizes smaller than the pores in the membrane) to diffuse across to the stream where a lower concentration is present. The counter-flow arrangement is made possible through the use of the porous tube which in turn makes this denuding system very compact. Particle losses are minimal when an appropriate membrane is used.

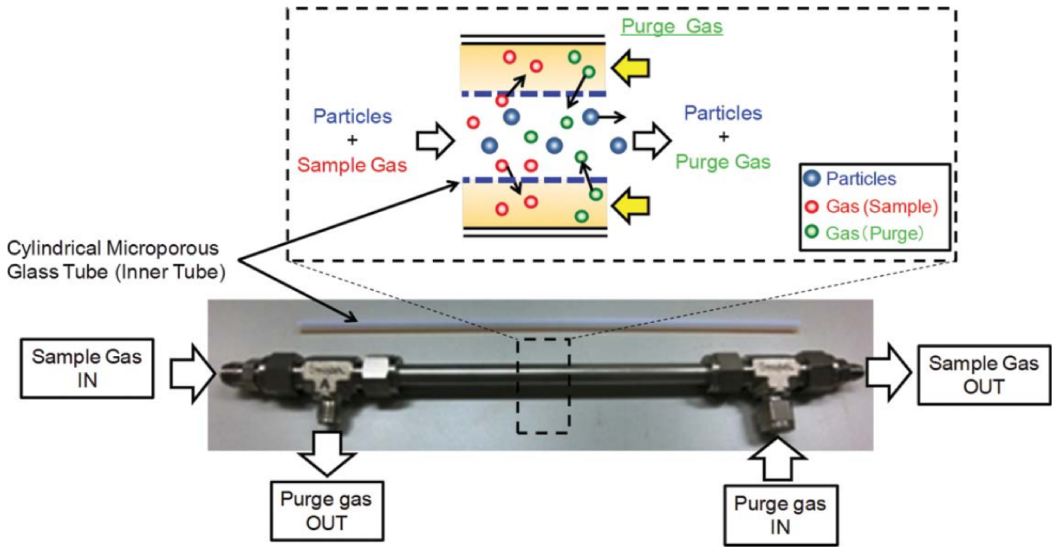


Figure 1.6: Hagino's counter-flow denuder (Hagino, 2016).

A similar system is described in Bryne and Griffith (1977) although their system is intended for use in gas analysis rather than for particles. Like Hagino's denuder, a diffusional membrane is used to dilute the sample gas with dilution air. Gases that diffuse from the sample flow into the diluent is then sent to a gas analyzer for analysis.

1.3 Humidification/Dehumidification

In some aerosol science applications, it is required to humidify or dry the aerosol sample. This is typically done using silica gel dryers or Nafion dryer/humidifiers. An example of a silica-based dryer that is sold commercially is TSI's Diffusion Dryer 3062, in which an aerosol flows in a channel separated from silica gel by a wire screen. Nafion humidifiers and dryers are essentially diffusion denuders that are re-purposed for the removal of water vapour exclusively (Smith and Burke, 2012) where the Nafion material is used as the membrane.

McMurry and Stolzenburg (1989) briefly described what they call a relative humidity (RH) conditioner that was used to adjust the humidity of an aerosol sample. The RH conditioner, which at a high level is similar to a diffusion denuder using parallel flow instead of counter-flow, consists of an inlet and outlet tube that are concentric to a larger outer tube. Aerosol enters and exits the inlet and outlet tubes respectively while humidified or dry sheath air (depending on the desired conditioning) enters and flows through the annular region between the aerosol flow and outer tube wall. The aerosol and sheath flows meet inside the larger tube with no membrane separating the two flows. The aerosol flow rate in the RH conditioner is kept lower than the sheath flow rate. Flow within the conditioner is entirely within the laminar regime to prevent mixing between the aerosol and sheath flows. Like the diffusion denuder, the changes in RH are achieved through the diffusion of water vapour from the aerosol flow to the dilution flow. Aerosol diffusion is negligible in the RH conditioner, meaning that the

RH can be changed without significantly diluting the aerosol.

1.4 Motivation

Many different conditioning techniques and devices have been examined throughout this chapter. While many of these devices function exceptionally well at the tasks they are designed for, in a laboratory environment it can be cumbersome having to use many different instruments just to condition the aerosol prior to measurement. This becomes an even larger issue when these devices operate at different flow rates, thus requiring additional instrumentation to allow the devices to function with each other.

Ideally these conditioning devices and principles can be combined into a single device. A couple of devices examined in this chapter are indeed capable of conditioning aerosols beyond their intended scope. Hagino's denuder for instance, can be used be used to change the RH of the aerosol by using dry or wet purge gas. Particles could theoretically be diluted as well, though in practice would be impractical as it would require the membrane to be changed to one that would allow a larger portion of the particles to pass through.

The RH conditioner described by McMurry and Stolzenburg (1989) can also be extended to other applications by controlling the flows in and out of the conditioner. By controlling the sheath flow entering and exiting the device, many different operational modes can be achieved. Dilution of gases beyond water vapour can be done without affecting the particle concentration and if desired, the particle concentration can be diluted as well. Denuding can also be accomplished by heating the sheath flow entering the conditioner. As no membrane is required for any of these functions to work, this RH conditioner serves as an excellent base for a multi-functional conditioning instrument.

1.5 Overview

This thesis will expand upon the RH conditioner design by McMurry and Stolzenburg. McMurry and Stolzenburg (1989) does not provide a theoretical model showing how the device functions, thus in this thesis a theoretical model will be developed. This thesis is comprised of five chapters. Chapter 2 will introduce the multi-functional aerosol conditioning device. Chapter 3 will discuss the model used to determine the theoretical performance of the conditioner while Chapter 4 is dedicated to the experimental validation of the model. To conclude, Chapter 5 will provide a summary of the conclusions drawn from this thesis and lists potential work that should be considered in the future.

Chapter 2

Universal Aerosol Conditioner

The purpose of the Universal Aerosol Conditioner (UAC) is to accomplish diluting, denuding and drying/humidifying in one device with the additional application of being able to dilute the gas-phase of the aerosol without diluting the concentration of the particle-phase. A preliminary schematic of the UAC is shown in Figure 2.1.

The UAC consists of an inlet and outlet tube which are concentric to an outer tube as shown in Figure 2.2. The aerosol enters and exits the UAC through the sample inlet and outlet tube, respectively. The flow rate of the aerosol flow is controlled by an aerosol instrument, or some other device, downstream of the UAC. Concentric to the aerosol flow is the shealth flow (which arrives via the manifold closest to the sample inlet in Figure 2.1) which is filtered to remove any particles. An annulus of fine mesh screen near the inlet and outlet of the UAC, as seen in Figure 2.2, is used to create a pressure drop in the sheath flow; this pressure drops ensures that the sheath flow enters the dilution chamber uniformly. The flowrate of the sheath flow is controlled by a blower and mass flowmeter. The blower speed is controlled in a feedback system by using the mass flowmeter signal. Both the aerosol and shealth flows are laminar to prevent mixing of the flows.

Once the aerosol enters the UAC, the particles will move along the axis of the outer tube as they are carried along with the flow. By Fick's law the gas in the aerosol

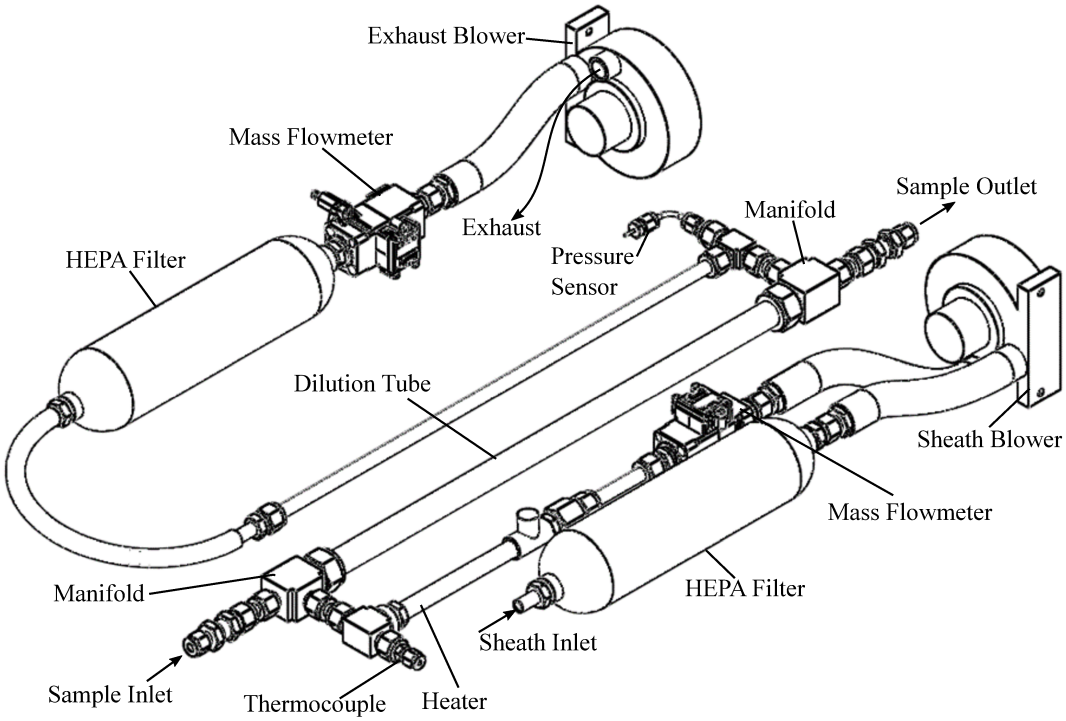


Figure 2.1: A preliminary schematic of the UAC

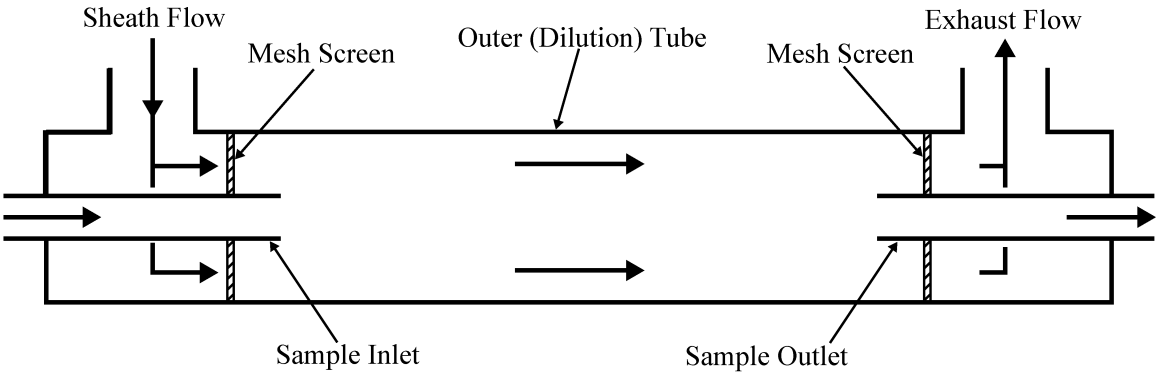


Figure 2.2: Simplified cross-sectional view of the dilution chamber

and the gas in the sheath flow will diffuse along a concentration gradient (i.e. in the radial direction). Therefore, sheath gas will diffuse into the aerosol flow and aerosol gas will diffuse into the sheath flow. If the length of the outer tube is sufficiently long, then the gas concentration will be uniform in the radial direction near the end of the outer tube. At the same time, particles in the aerosol stream will also diffuse outwards towards the sheath flow. However, since the diffusion coefficient of particles is several orders of magnitude lower than the gas diffusion coefficient, the particles will remain in the region near the axis of the outer tube. The particles will then exit with the aerosol outlet flow and the remaining flow will exit in the exhaust flow which is also controlled with a blower and mass flowmeter. A thermocouple and pressure sensor in Figure 2.1 are used to calculate the densities of the gases and volumetric flow rates within the UAC.

2.1 UAC Operational Modes

The device can be used in four modes of operation: gas dilution, gas and particle dilution, denuding, and humidification.

2.1.1 Gas dilution without particle dilution

This can be accomplished by setting the sheath flow, Q_{sh}, equal to the exhaust flow, Q_{exh}, as seen in Figure 2.3. If the purpose of the gas dilution is to reduce the concentration of a gas component in the aerosol, then the sheath flow should be a gas with a very low concentration of the component to reduce. For example, if the aerosol is saturated with water vapour, then dry air can be supplied to the sheath flow, and the aerosol will be dried. The gas dilution ratio will depend on the ratio of the sheath flow to the aerosol flow. For example, if the sheath flow is 10 times larger than the aerosol flow, then the gas concentration will be reduced by a factor of 10.

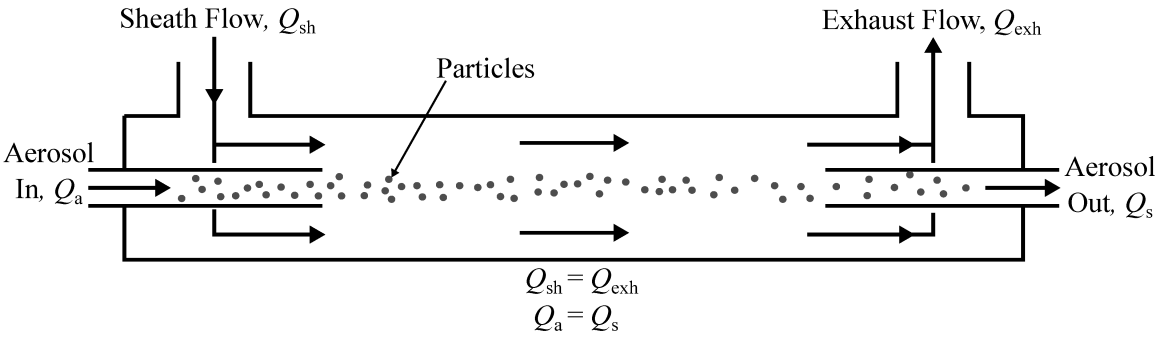


Figure 2.3: Sheath and aerosol flows for gas dilution without aerosol dilution.

2.1.2 Combined particle and gas dilution

Combined dilution can be accomplished by setting the sheath flow less than the aerosol flow, Q_a, but satisfying the condition that the sum of the aerosol outlet, Q_s, and exhaust flows is greater than the sheath flow as shown in Figure 2.4. Given the uncertainty in the mass flowmeters it is expected this would give accurate dilution up to dilution ratios of approximately 10. This has an added benefit of providing a known dilution ratio (i.e. can be calculated by the flow rate measurements), unlike ejector dilutors where the dilution ratio is unknown and needs to be calibrated or measured in some other way.

As a side note, one additional operating mode is setting the sheath flow to a value higher than the sum of the aerosol outlet and exhaust flow. This would result in sheath air exiting the aerosol inlet, aerosol outlet, and exhaust. This can be used to provide particle-free gas to the aerosol instrument to zero it.

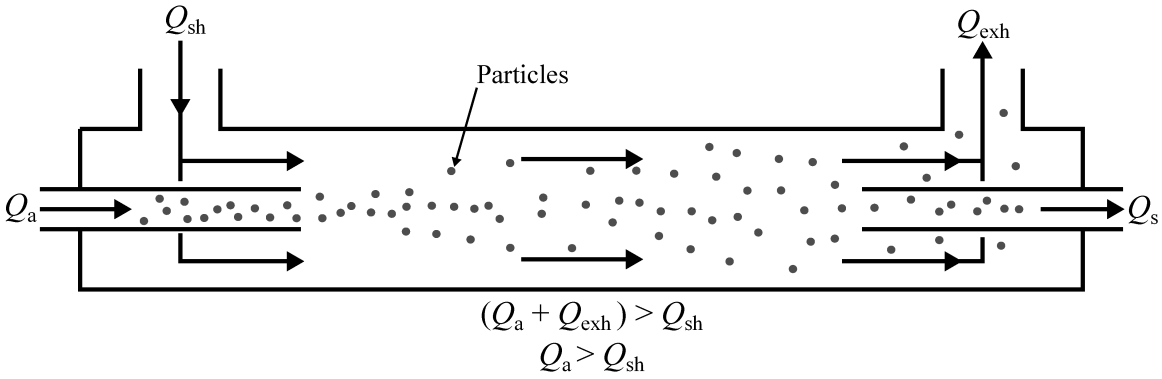


Figure 2.4: Sheath and aerosol flows for combined particle and gas dilution.

2.1.3 Denuding

Denuding can be accomplished by setting the sheath flow equal to the exhaust flow and heating the sheath air (before it enters the outer tube chamber) or the outer tube or both (Figure 2.5). An inline process heater is present to heat the sheath air as shown in Figure 2.1. In this mode, the volatile material in the aerosol evaporates and diffuses into the sheath flow and is exhausted. As in the case of gas dilution without particle dilution, the reduction in concentration is equal to the ratio of sheath to aerosol flow.

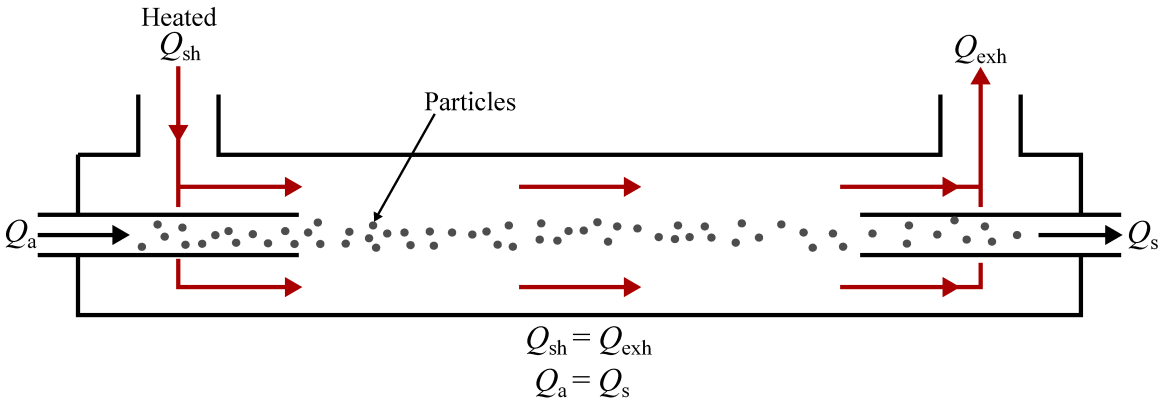


Figure 2.5: Sheath and aerosol flows for denuding configuration.

2.1.4 Humidification

Humidification can be accomplished by setting the sheath flow equal to the exhaust flow and by supplying humidified sheath air into the UAC. In this case, the water vapour in the sheath flow diffuses into the aerosol stream and humidifies it.

Chapter 3

Theoretical Modelling of the Universal Aerosol Conditioner

In this chapter, the main governing model will be developed. The model will be solved using numerical methods. Two methods have been developed and ultimately one will be chosen for use in modelling the prototype.

3.1 Diffusion Model

As diffusion is the primary operating mechanism of the UAC, a model was developed using the convective-diffusion equation as its basis. From the model, the concentration profile of both the particle and gas-phases throughout the UAC's dilution chamber can be found. Two important parameters can be calculated by knowing the concentrations of the particles and gases at the aerosol inlet and outlet sample tubes: the particle penetration efficiency and the dilution factor for the gas-phase. Particle penetration, η, is the ratio of particle flux that exits in the sample flow to the particle flux entering with the aerosol flow. Penetration is important in identifying the concentration of particles that is expected to be retained in the aerosol sample once it passes through the UAC. Keeping the penetration high is vital in ensuring the gas-only dilution operational mode functions properly (remember that this mode would typically be

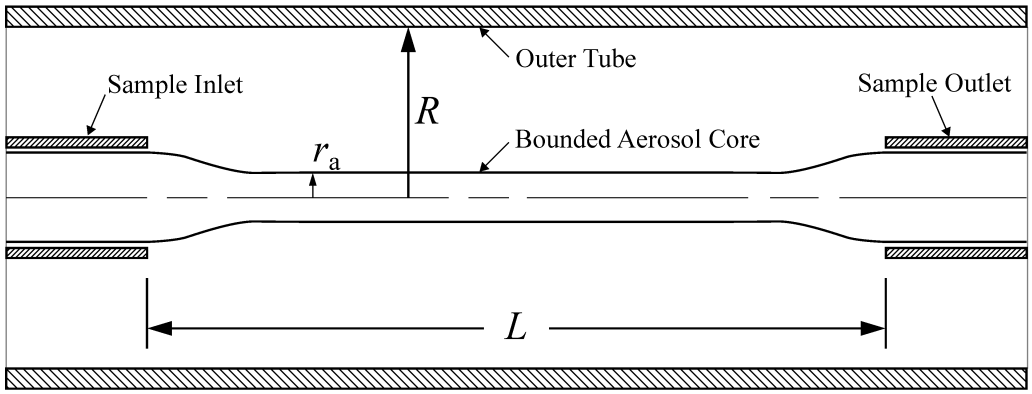
used in situations where the particle concentration entering the UAC is already very low). The dilution factor, DF, is the ratio of gas flux that exits with sample flow to the gas flux that enters with the aerosol flow. The dilution factor shows how much of the undesirable gas species can be diluted in the aerosol.

The convective-diffusion equation (Friedlander, 2000) in the absence of any body forces on the particles is given as:

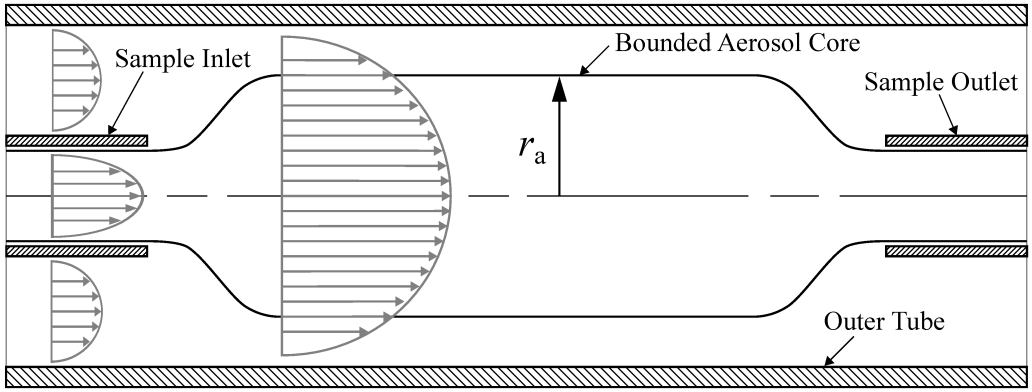
$$\frac{\partial n}{\partial t} + u \cdot \nabla n = D\nabla^2 n \tag{3.1}$$

where n is the particle or gas concentration, u is the gas velocity, and D is the diffusion coefficient of the particle or gas.

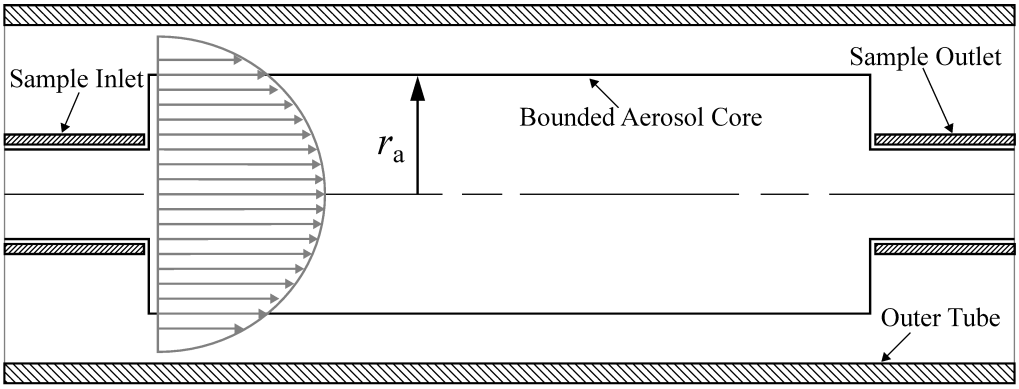
Both the aerosol and sheath flows enter the same dilution tube, where the aerosol enters through the middle while the sheath flow surrounds the aerosol. The flows run parallel to each other, hence no radial mixing between the flows will be assumed. More importantly, as both flows enter the dilution tube the streamlines of both flows will alter such that the flows will reach the same velocity (as mass in the system must be conserved). Figure 3.1 demonstrates how the aerosol streamlines will converge or diverge once they enter the dilution tube. In reality, the outermost streamlines of the aerosol will contract and expand gradually until they reach the bounding radius, r_{a}, as shown in Figures 3.1a and 3.1b respectively. Whether the streamlines will contract or expand is determined by the ratio of sheath-to-aerosol flow rates. Sheath flows that are greater than the aerosol will result in the aerosol to contract relative to the aerosol sample tube's inner diameter and vice versa.



(a) Aerosol streamlines gradually contract.



(b) Aerosol streamlines gradually expand.



(c) How expansion (and contraction) will be modelled.

Figure 3.1: Aerosol outermost streamlines as they enter and exit the dilution chamber

This gradual expansion and contraction is difficult to model. To simplify the model, it will be assumed that the flows entering the dilution chamber will instantly become fully developed as shown in Figure 3.1c. Note that the dilution chamber is the region of the dilution tube with an inner radius, R, and a length of L between the

end and start of the sample inlet and outlet tubes, respectively. To find the bounding radius of the aerosol, r_{a}, the volumetric flow rate must be defined first. Recall that for Poiseuille flow in round ducts the velocity profile, $u(r)$, for laminar flow is given by (Çengel and Cimbala, 2010):

$$u(r) = 2u_{\mathrm{avg}}\left[1 - \left(\frac{r}{R}^2\right)\right] \tag{3.2}$$

where u_{avg} is the average velocity of the gas stream. The volumetric flow rate, Q, in the round duct can be determined by integrating Equation 3.2 across the cross-sectional area of the duct:

$$\begin{aligned} Q &= \int_A u(r)dA \\ &= \int_0^r 2u_{\mathrm{avg}}\left[1 - \left(\frac{r}{R}^2\right)\right] \cdot 2\pi r dr \end{aligned} \tag{3.3}$$

When multiple streams (that are moving in the same direction) flow into the same duct, their average velocities become equal to each other. By equating the average velocities of the aerosol and total flow, Q_{t}, to each other the cross-sectional area of the aerosol core can be found and thus the bounding radius:

$$\frac{Q_{\mathrm{a}}}{\int_0^{r_{\mathrm{a}}} 2u_{\mathrm{avg}}\left[1 - \left(\frac{r}{R}^2\right)\right] \cdot 2\pi r dr} = \frac{Q_{\mathrm{t}}}{\int_0^{R} 2u_{\mathrm{avg}}\left[1 - \left(\frac{r}{R}^2\right)\right] \cdot 2\pi r dr}$$

$$r_{\mathrm{a}} = R\left[1 - \left(1 - \frac{Q_{\mathrm{a}}}{Q_{\mathrm{t}}}\right)^{1/2}\right]^{1/2} \tag{3.4}$$

where $Q_{\mathrm{t}} = Q_{\mathrm{a}} + Q_{\mathrm{sh}}$, Q_{a} is the aerosol volumetric flow rate, and Q_{sh} is the sheath volumetric flow rate.

The derivations of the particle penetration and gas dilution performance were done under several assumptions. Both sample and sheath flows are assumed to be fully

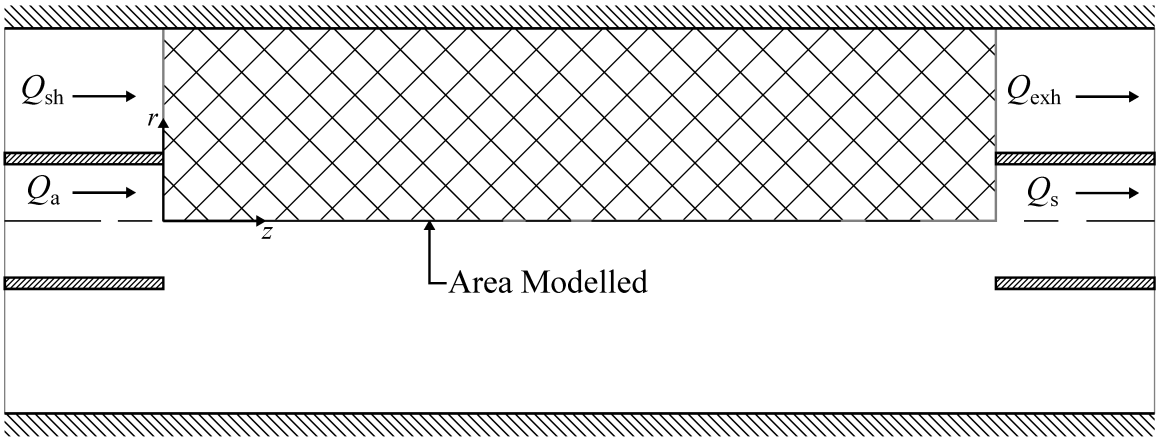


Figure 3.2: Region that was modelled

developed and laminar throughout the domain with no radial velocities. Diffusion in the axial direction is assumed to be negligible due the the relative difference in speed between the axial and radial directions. Particle migration is also assumed to be negligible (*e.g.* gravitational forces are small). The UAC geometry is also axially symmetric, hence only a 2D plane of the tube geometry is considered as shown in Figure 3.2. Note that due to the axial symmetry in the 2D plane, only half of the domain is required to be modelled. Under these conditions, Equation 3.1, using cylindrical coordinates, simplifies to:

$$u_z(r)\frac{\partial n}{\partial z} = D\frac{\partial^2 n}{\partial r^2} + \frac{D}{r}\frac{\partial n}{\partial r} \tag{3.5}$$

where z denotes the axial direction, and r is the radial direction.

Recall that the aerosol sample that is to be conditioned enters through the small central tube, while the sheath flow enters from the annular region formed from the central tube and the outer tube. The entrance concentration profile entering the UAC dilution tube (*i.e.* just past the outlet of the central tube) for both the particle and gas-phases of the combined aerosol sheath flow can be expressed as a step profile:

$$n(r,0) = \begin{cases} n_0 & 0 \leq r \leq r_a \\ 0 & r > r_a \end{cases} \tag{3.5a}$$

where n_0 is the initial concentration. No concentration at the walls and axial symmetry are prescribed for the boundary conditions for the particle phase:

$$n(R, z = 0) = 0 \tag{3.5b}$$

$$\frac{\partial n}{\partial r}(r = 0, z) = 0 \tag{3.5c}$$

Similarly, the boundary conditions prescribed for the gas-phase of Equation 3.1 are no flux across the wall and symmetry about the diluter centerline:

$$\frac{\partial n}{\partial r}(R, z) = 0 \tag{3.5d}$$

$$\frac{\partial n}{\partial r}(r = 0, z) = 0 \tag{3.5e}$$

To evaluate the theoretical performance of the UAC at different scales, Equation 3.5 is written in a dimensionless form using the following dimensionless parameters:

$$\nu = \frac{n}{n_0}$$

$$\rho = \frac{r}{R}$$

$$\zeta = \frac{z}{L}$$

where ν, ρ, and ζ denotes the dimensionless concentration, radial, and axial direction respectively. Substituting the dimensionless parameters into Eq. 3.5, the partial

differential equation (PDE) becomes:

$$\mathrm{Pe}(1-\rho^2)\frac{\partial \nu}{\partial \zeta} = \frac{1}{\rho}\frac{\partial \nu}{\partial \rho} + \frac{\partial^2 \nu}{\partial \rho^2} \tag{3.6}$$

where $\mathrm{Pe} = \dfrac{2Q_{\mathrm{t}}}{\pi D L}$, is the Péclet number. Notice that Pe is independent of the dilution chamber radius. This implies that diffusion in the UAC is completely independent of the radius of the dilution tube.

The dimensionless boundary conditions for the particle-phase PDE are:

$$\nu(1,0) = 0 \tag{3.6a}$$

$$\frac{\partial \nu}{\partial \rho}(0,\zeta) = 0 \tag{3.6b}$$

The corresponding boundary conditions for the gas-phase are:

$$\frac{\partial \nu}{\partial \rho}(1,\zeta) = 0 \tag{3.6c}$$

$$\frac{\partial \nu}{\partial \rho}(0,\zeta) = 0 \tag{3.6d}$$

While the non-dimensionalized initial concentration profile for both PDEs is:

$$\nu_0 = \begin{cases} 1 & 0 \le \rho \le \rho_a, \zeta = 0 \\ 0 & \rho > \rho_a, \zeta = 0 \end{cases} \tag{3.6e}$$

ρ_a, the dimensionless bounding radius of the aerosol core in the dilution tube, can be found by:

$$\rho_{\mathrm{a}} = \left[1 - \left(1 - \frac{Q_{\mathrm{a}}}{Q_{\mathrm{t}}}\right)^{1/2}\right]^{1/2} \tag{3.7}$$

3.2 Numerical Modelling of the Dimensionless Convection - Diffusion Equation

Two separate models were developed to solve the dimensionless partial differential equation: one using Crank-Nicolson discretization and the other using the implicit Euler method. Both methods will be covered in the following sections.

While an analytical solution to the particle-phase of the PDE was found, one could not be found for the gas-phase and hence the solution will not be presented here, rather the derivations of the particle-phase solution can be found in Appendix A.

3.2.1 Numerical Model using the Crank-Nicolson Method

To implement Crank-Nicolson, the domain of the dilution chamber of the UAC was divided into a grid as shown in Figure 3.3. Given that the solution is expected to rapidly change near the entrance of the dilution chamber, a logarithmic mesh was used in the axial direction, ζ, to capture the concentration profile more efficiently. The mesh spacing in the radial direction, ρ, was kept linear.

For Crank-Nicholson discretization, the derivative terms of Eq. 3.6 can be approximated using the following relationships (Smith, 1985):

$$\nu = \frac{1}{2}(\nu_{i,j} + \nu_{i,j+1}) \tag{3.8a}$$

$$\frac{\partial \nu}{\partial \zeta} = \frac{1}{k}(\nu_{i,j+1} - \nu_{i,j}) \tag{3.8b}$$

$$\frac{\partial \nu}{\partial \rho} = \frac{1}{4h}(\nu_{i+1,j} - \nu_{i-1,j}) + \frac{1}{4h}(\nu_{i+1,j+1} - \nu_{i-1,j+1}) \tag{3.8c}$$

$$\frac{\partial^2 \nu}{\partial \rho^2} = \frac{1}{2h^2}(\nu_{i+1,j} - 2\nu_{i,j} + \nu_{i-1,j}) + \frac{1}{2h^2}(\nu_{i+1,j+1} - 2\nu_{i,j+1} + \nu_{i-1,j+1}) \tag{3.8d}$$

where k and h are the step sizes in ζ and ρ respectively. Substituting these relationships into Eq. 3.6, the ν terms that are of similar index can be grouped

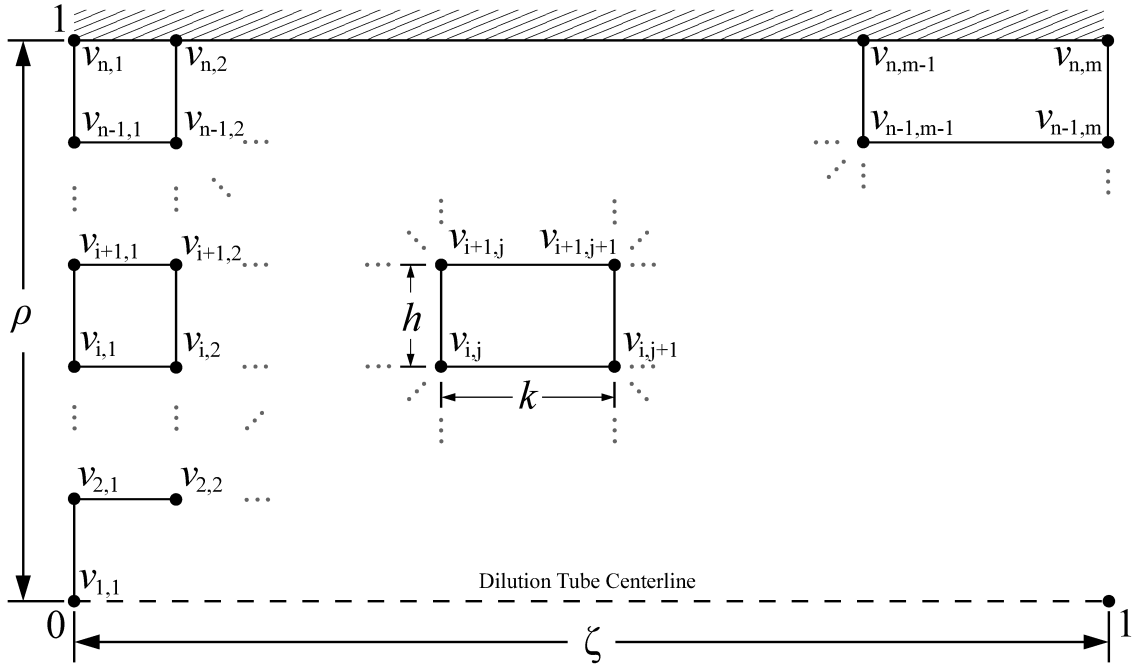


Figure 3.3: Mesh used for Crank-Nicolson discretization

together to form:

$$\begin{aligned}\nu_{i-1,j+1}\Big(\frac{1}{4h\rho}-\frac{1}{2h^2}\Big)+\nu_{i,j+1}\Big(\frac{\mathrm{Pe}(1-\rho^2)}{k}+\frac{1}{h^2}\Big)+\nu_{i+1,j+1}\Big(-\frac{1}{4h\rho}-\frac{1}{2h^2}\Big)\\ =\nu_{i-1,j}\Big(-\frac{1}{4h\rho}+\frac{1}{2h^2}\Big)+\nu_{i,j}\Big(\frac{\mathrm{Pe}(1-\rho^2)}{k}-\frac{1}{h^2}\Big)+\nu_{i+1,j}\Big(\frac{1}{4h\rho}+\frac{1}{2h^2}\Big)\end{aligned} \tag{3.9}$$

For better legibility, Eq. 3.9 can be re-written as:

$$A_i\nu_{i-1,j+1}+B_i\nu_{i,j+1}+C_i\nu_{i+1,j+1}=E_{i,j} \tag{3.10}$$

where

$$A_i = \frac{1}{4h\rho} - \frac{1}{2h^2} \tag{3.10a}$$

$$B_i = \frac{\mathrm{Pe}(1-\rho^2)}{k} + \frac{1}{h^2} \tag{3.10b}$$

$$C_i = -\frac{1}{4h\rho} - \frac{1}{2h^2} \tag{3.10c}$$

$$E_{i,j} = \nu_{i-1,j}\Big(-\frac{1}{4h\rho} + \frac{1}{2h^2}\Big) + \nu_{i,j}\Big(\frac{\mathrm{Pe}}{k} - \frac{1}{h^2}\Big) + \nu_{i+1,j}\Big(\frac{1}{4h\rho} + \frac{1}{2h^2}\Big) \tag{3.10d}$$

and h and k are the step sizes between nodes in ρ and ζ respectively. The boundary conditions were discretized using central-difference schemes. The discretized boundary conditions for the particle-phase are:

$$A_n = \frac{1}{4h} - \frac{1}{2h^2} \tag{3.10e}$$

$$B_1 = \frac{\mathrm{Pe}}{k} + \frac{1}{h^2} \tag{3.10f}$$

$$B_n = \frac{1}{h^2} \tag{3.10g}$$

$$C_1 = -\frac{1}{h^2} \tag{3.10h}$$

$$E_{1,j} = \nu_{1,j}\Big(\frac{\mathrm{Pe}}{k} - \frac{1}{h^2}\Big) + \nu_{2,j}\Big(\frac{1}{h^2}\Big) \tag{3.10i}$$

$$E_{n,j} = 0; \tag{3.10j}$$

For the gas-phase boundary conditions, A_n and $E_{n,j}$ are different due to the no flux condition at the tube wall:

$$A_n = -\frac{1}{h^2} \tag{3.10k}$$

$$E_{n,j} = \nu_{n-1,j}\Big(\frac{1}{h^2}\Big) + \nu_{n,j}\Big(-\frac{1}{h^2}\Big) \tag{3.10l}$$

As a result of using the Crank-Nicolson finite difference method on a parabolic partial differential equation (in this case, the convection diffusion equation), if one were to expand the terms in Eq. 3.10 in matrix form, it would result in a tridiagonal matrix as seen in Equation 3.11. This useful property allows the system of equations to be solved using the Thomas algorithm. The Thomas algorithm is able to solve tridiagonal systems in less operations and consumes less memory (due to not having to store the off-diagonal zeroes in memory) compared to other Gaussian elimination methods (Thomas, 1949).

$$\begin{bmatrix} B_1 & C_1 & 0 & 0 & \dots & 0 & 0 & 0 \\ A_2 & B_2 & C_2 & 0 & \dots & 0 & 0 & 0 \\ 0 & A_3 & B_3 & C_3 & \dots & 0 & 0 & 0 \\ 0 & 0 & A_4 & B_4 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & B_{n-2} & C_{n-2} & 0 \\ 0 & 0 & 0 & 0 & \dots & A_{n-1} & B_{n-1} & C_{n-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & A_n & B_n \end{bmatrix} \begin{bmatrix} \nu_{1,j+1} \\ \nu_{2,j+1} \\ \nu_{3,j+1} \\ \nu_{4,j+1} \\ \vdots \\ \nu_{n-2,j+1} \\ \nu_{n-1,j+1} \\ \nu_{n,j+1} \end{bmatrix} = \begin{bmatrix} E_{1,j} \\ E_{2,j} \\ E_{3,j} \\ E_{4,j} \\ \vdots \\ E_{n-2,j} \\ E_{n-1,j} \\ E_{n,j} \end{bmatrix} \tag{3.11}$$

Note that Eq. 3.11 must be solved for each step in ζ. For each new step, the previous step's solution, $\nu_{i,j+1}$, is substituted into the current $E_{i,j}$. Once the entire set of equations is solved, the particle penetration, η, and gas dilution factor, DF, can be calculated by integrating the flux at the inlet and outlet of the aerosol sample line:

$$\eta = \frac{\int_0^{\rho_a} (1-\rho^2)\nu(\rho,1)\rho d\rho}{\int_0^{\rho_a} (1-\rho^2)\nu(\rho,0)\rho d\rho} \tag{3.12}$$

$$\mathrm{DF} = \frac{\int_0^{\rho_a} (1-\rho^2)\nu(\rho,0)\rho d\rho}{\int_0^{\rho_a} (1-\rho^2)\nu(\rho,1)\rho d\rho} \tag{3.13}$$

The MATLAB implementation can be found in Appendix B.

3.2.2 Numerical Solution Using MATLAB's bvp4c Solver

In the previous section, the particle penetration, gas dilution factor and particle and gas concentration profiles within the UAC were calculated using the Crank-Nicolson finite difference method. To verify and compare the results obtained from that method, an alternative numerical scheme was used. Eq. 3.7 was discretized using the implicit Euler method resulting in:

$$\mathrm{Pe}(1-\rho^2)\Big(\frac{\nu_{i,j+1}-\nu_{i,j}}{h}\Big) = \frac{1}{\rho}\frac{\partial \nu_{i,j+1}}{\partial \rho} + \frac{\partial^2 \nu_{i,j+1}}{\partial \rho^2} \tag{3.14}$$

With the equation discretized, MATLAB's built-in bvp4c solver was used to calculate the concentration profile. Using the bvp4c solver requires Eq. 3.14 to be rewritten as a first order system. To do so, the following change of variable was made:

$$m = \frac{\partial \nu}{\partial \rho} \tag{3.15}$$

Substituting this into Eq. 3.14 and rearranging results in:

$$\frac{\partial m_{i,j+1}}{\partial \rho} = \mathrm{Pe}(1-\rho^2)\frac{\nu_{i,j+1}-\nu_{i,j}}{h} - \frac{m_{i,j+1}}{\rho} \tag{3.16}$$

with particle-phase boundary conditions:

$$m(0,\zeta) = 0 \tag{3.16a}$$

$$\nu(1,\zeta) = 1 \tag{3.16b}$$

and gas-phase boundary conditions:

$$m(0,\zeta) = 0 \tag{3.16c}$$

$$m(1,\zeta) = 0 \tag{3.16d}$$

However, due the bvp4c being a one-dimensional solver, it was necessary to run the solver in a loop to find the concentration profile across the entire domain. Similar to Section 3.2.1, every instance in the loop calculates a single step of the concentration profile by using the previous step's solution as it progresses through ζ. The MATLAB implementation can be seen in Appendix C.

3.2.3 Code Execution Performance Comparison Between the Two Numerical Methods

The two methods were compared to each other based on the DF each method calculated and amount of time it took to reach a solution. Five different mesh sizes were used for each method and the results are summarized in Table 3.1. Particle penetration was not compared due to the implicit Euler's very long convergence times (several orders of magnitude slower than the implicit Euler dilution factor convergence times). Note that Table 3.1 was compiled using a Pe of 0.4377 with an aerosol flow rate of 0.3 LPM and sheath flow rate of 3 LPM. Also, take note that when given enough time

to diffuse, the dilution factor can also be calculated through the following equation:

$$\mathrm{DF} = \frac{\text{Total Flow}}{\text{Initial Flow}} = \frac{Q_{\mathrm{sh}} + Q_{\mathrm{a}}}{Q_{\mathrm{a}}} \tag{3.17}$$

Table 3.1: Comparison between Crank-Nicolson and Euler-Implicit (bvp4c) methods

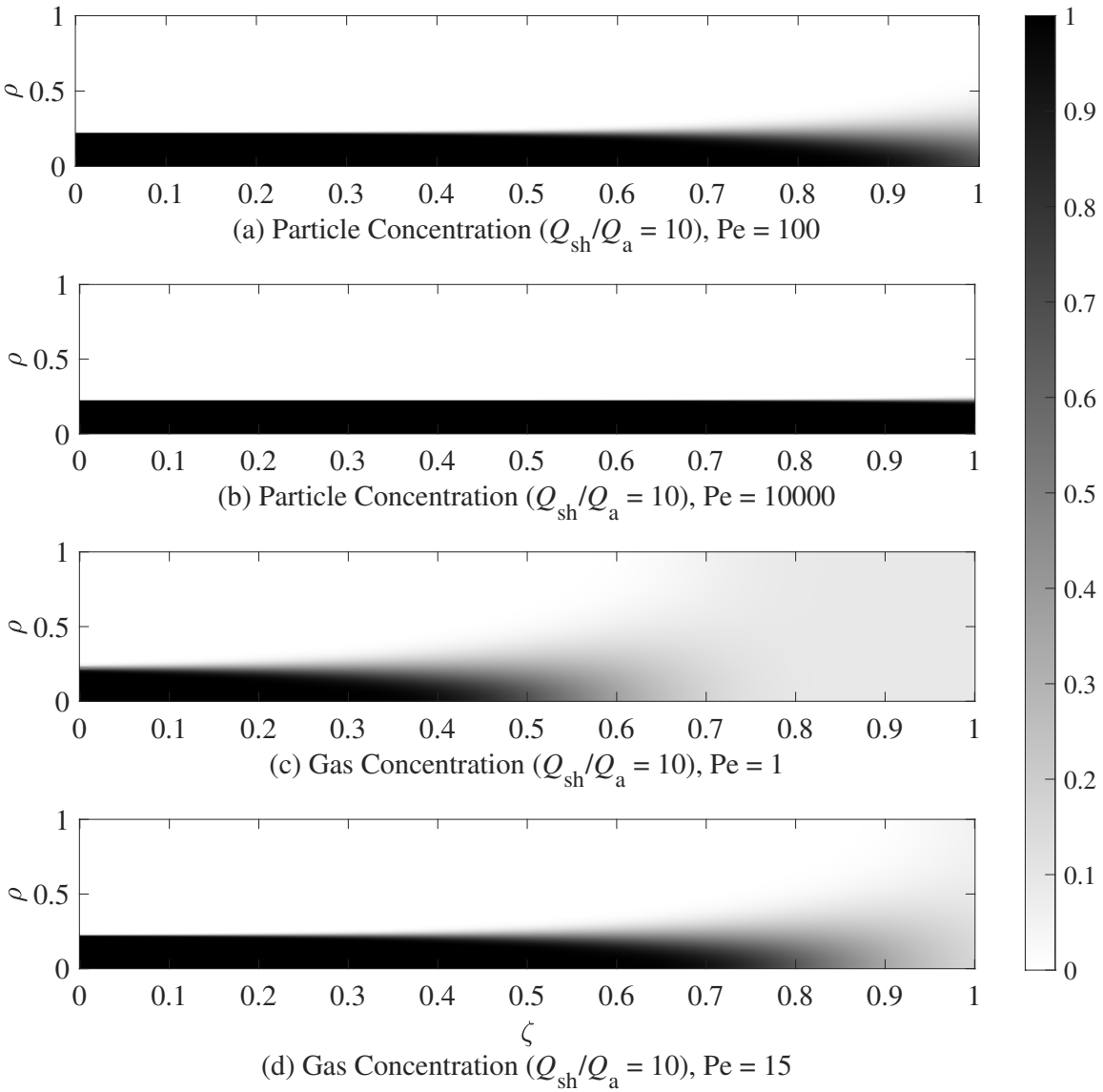
	Crank-Nicolson		Implicit Euler	
Mesh Size (n x m)	DF	Execution Time (s)	DF	Execution Time (s)
200 x 100	10.71	0.0093	10.71	86
400 x 200	10.83	0.0240	10.89	302
600 x 300	10.87	0.0442	10.95	672
800 x 400	10.89	0.0610	10.98	1211
1000 x 500	10.90	0.1026	11.00	1793

For each equivalent mesh size, the Crank-Nicolson method reaches a solution significantly faster than the bvp4c solver. As mesh size increases, the bvp4c solution approaches the theoretical maximum dilution factor for these flow conditions with less nodes than Crank-Nicolson but convergence time as a result suffers (convergence is also influenced by the initial guess that must be provided to the solver before the routine begins). Seeing as the convergence time of the bvp4c solver routine does not warrant the small increase in accuracy using the same number of nodes, the Crank-Nicolson method will be used from hereon in evaluating the theoretical performance of the UAC. Better accuracy can be achieved with Crank-Nicolson just by simply using more nodes while still converging quicker than the bvp4c method. Referring back to Table 3.1, it is apparent that both Crank-Nicolson and the bvp4c routines are mesh independent at a mesh of approximately 400 x 200.

3.3 Theoretical Performance of the UAC

3.3.1 Visualizing Diffusion in the UAC

Diffusion in the dilution tube can be visualized by examining the concentration maps for both phases of the aerosol in Fig. 3.4 over a range of Pe that one might expect in practice. An sheath-to-aerosol ratio of 10 was used to plot these maps as it is near the maximum dilution ratio that could be accurately controlled with such a system. The sheath flow is assumed to contain only diluent gas (*e.g.* it does not contain any of the gas species that are desirable to dilute in the aerosol). In the case of the particle phase, Pe of 100 (Fig. 3.4a) and 10,000 (Fig. 3.4b) were used to illustrate the operating principle of the UAC. At very high Pe, there is minimal particle diffusion – diffusion can only be seen near the tail end of the dilution chamber. With a particle Pe of 100, diffusion at the tail end is more noticeable as a larger portion of the particles have migrated further away from the center of the dilution chamber. For the gas-phase, significant amounts of diffusion are present. However, even at the lower end of gas Pe, diffusion can vary greatly. Despite a relatively small difference between a gas Pe of 1 and 15, Fig. 3.4c shows diffusion is observed much earlier. Thus, by the time the aerosol reaches the end of the dilution chamber, the gas concentration will be uniformly distributed across ρ unlike in Fig. 3.4d. In summary, for optimal gas-only dilution, particle Pe must be kept high while minimizing gas Pe. In general, this is true because the diffusion coefficient for typical particles are orders of magnitude smaller than typical gases.



(a) Particle Concentration (Q_{sh}/Q_a = 10), Pe = 100

(b) Particle Concentration (Q_{sh}/Q_a = 10), Pe = 10000

(c) Gas Concentration (Q_{sh}/Q_a = 10), Pe = 1

(d) Gas Concentration (Q_{sh}/Q_a = 10), Pe = 15

Figure 3.4: Concentration maps of the particle and gas-phases of the aerosol at different Pe

3.3.2 Particle Penetration Performance

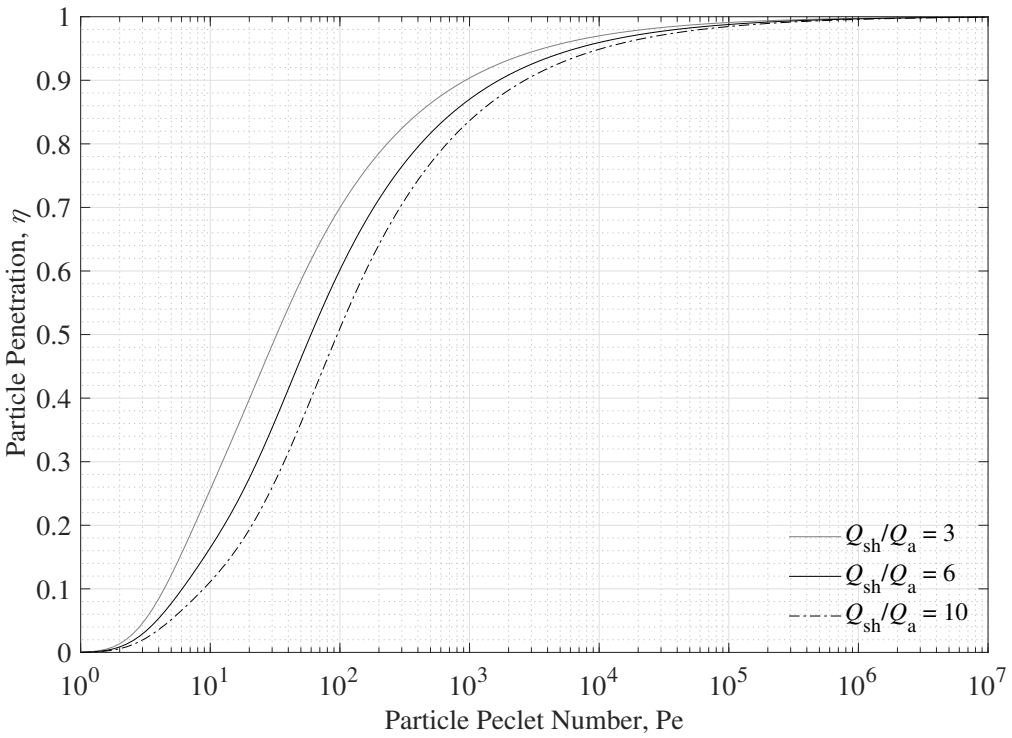


Figure 3.5: Theoretical particle penetration at different Pe and sheath-to-aerosol ratios

Particle penetration at various flow ratios are shown in Fig. 3.5. The particle penetration increases with particle Pe, until it is sufficiently large (Pe of approximately 1000). It is interesting to note that the particle penetration is larger when the flow ratio decreases. Referring back to Eq. 3.7, smaller sheath flows will lead to wider aerosol cores. This change in the initial condition for the model has some effect on the rate of diffusion. In isolation, if maximizing η is desired then smaller sheath-to-aerosol flows should be used but in doing so, gas dilution will decrease as shown in the next section.

3.3.3 Gas Dilution Performance

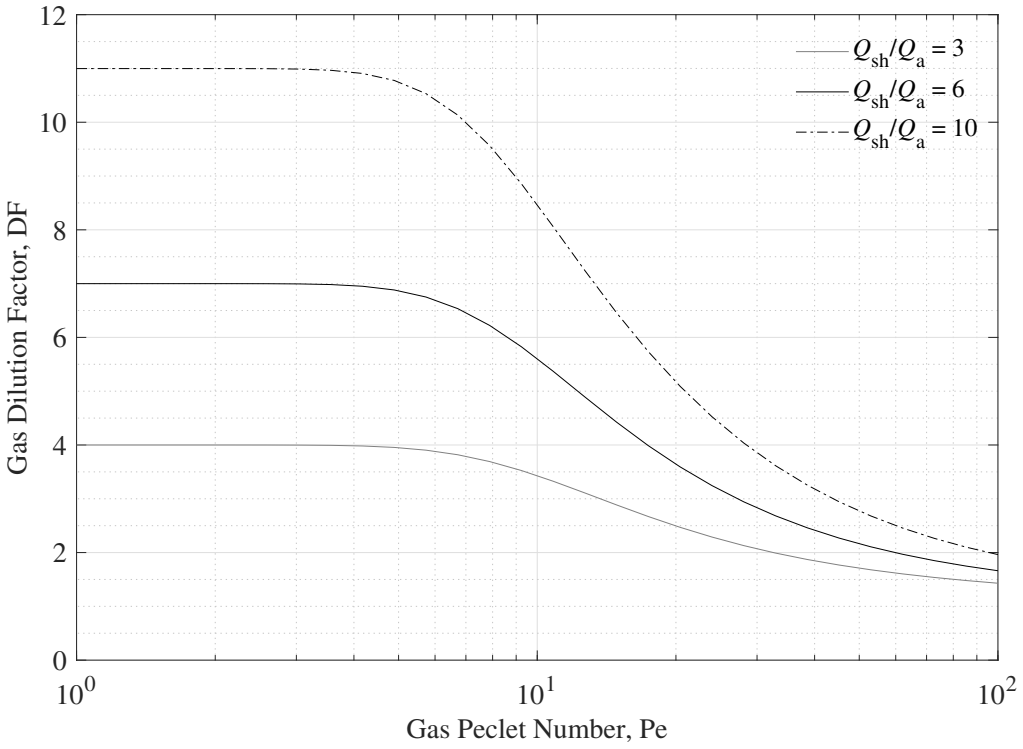


Figure 3.6: Theoretical gas dilution factor at different Pe and sheath-to-aerosol ratios

Gas dilution is significantly reduced once gas Pe is greater than 10. To achieve high gas dilution factors, large flow ratios are needed and Pe must be kept low. Ideally, gas Pe should be kept below 10 as DF is still close to its maximum value. Once Pe is larger than 10, the dilution factor quickly decays until it reaches an asymptote. At high gas Pe (*i.e.* Pe $>$ 100), there is not significant gas diffusion between the two flows because the residence time in the UAC is too short. The conditions that the particle penetration benefits from (short tube length and high volumetric flow rates) have a detrimental effect on the gas dilution.

3.4 Summary

A diffusional model was developed and solved numerically using the Crank-Nicolson method. Conveniently, it was found that both η and DF only depend on Pe and the sheath-to-aerosol flow ratio. Shorter dilution tube lengths and high flow rates increase η as Pe is increased under these conditions. The opposite is true for gas dilution; however, as an increase in Pe (once past a certain threshold) will lead to a decrease in the dilution factor. The dilution factor can be increased by decreasing the gas Pe; Pe in this case can be decreased by lowering the flow rates and by increasing the length of the dilution tube. It is important to note the Pe for the particle- and gas-phases are naturally very different due to the large differences in their diffusion coefficients. Since the gas species typically have diffusion coefficients that are orders of magnitude larger than the typical particle, it is possible to simultaneously have small Pe for the gases and large Pe for the particles. Also, η is influenced less by Pe once Pe becomes sufficiently large (Pe of approximately 1000). Due to this property of the particle-phase, less emphasis can be put on selecting geometry that will maximize η (*e.g.* shorter L) and instead be focused on maximizing DF (longer L). Losing a small portion of particles is acceptable if it will result in significantly improved dilution. L should be selected such that the gas Pe does not exceed 10 to maintain a dilution factor that is greater than 8.

Chapter 4

Experimental Validation of the Universal Air Conditioner

With the theoretical model developed, a prototype of the UAC was built and experimentally tested. This chapter will provide an overview of the design and its experimental performance.

4.1 Prototype Design

A simplified manifold design was shown in Figure 2.2 in Chapter 2. This basic manifold was redesigned and is shown in Fig. 4.1. In this design, the sheath flow enters a longer annular region, concentric to the aerosol sample inlet tube, filled with 1 mm diameter glass beads with fine mesh screens capping the ends prior to meeting the aerosol in the dilution tube. The purpose of the glass beads is to create a pressure drop to distribute the sheath flow evenly. The aerosol sample tubes are 0.2 m in length with an outer diameter (OD), d_1, of 6.35 mm (0.25 inch), and an inner diameter (ID) of 3.56 mm. The dilution tube has an OD, d_2, of 19.05 mm (0.75 inch), an ID of 15.75 mm, and a length of 0.559 m. This dilution tube length will result in a distance, L, of 0.5 m between the sample tubes. A length of 0.5 m between the samples tubes was chosen as this length keeps the overall length of the UAC from being unreasonably

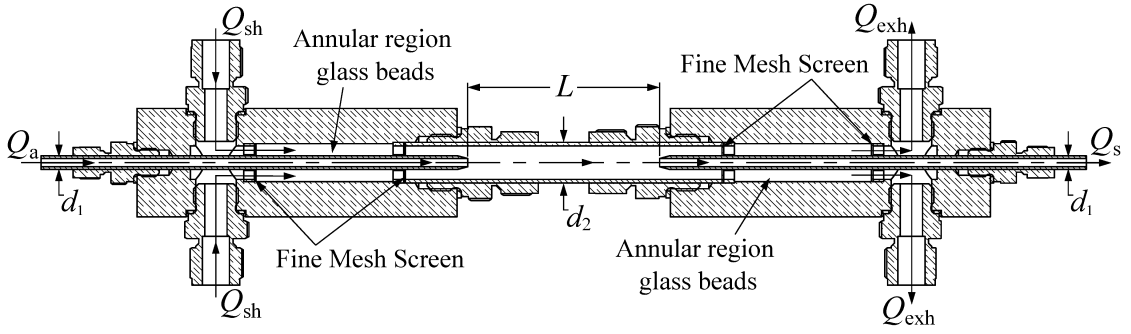


Figure 4.1: Schematic of the updated UAC prototype dilution chamber. Not to scale.

long to use on a work bench.

In addition, a laminar flow meter was integrated into the aerosol sample inlet of the UAC to monitor the actual aerosol volumetric flow rate coming into the UAC. The laminar flow meter is constructed out of two 0.25 inch OD Swagelok tees (SS-400-3) with a 10 cm long tube that has an OD of 3.175 mm (0.125 inch) and an ID of 1.753 mm connecting the tee fittings together. A pressure sensor (Model HSC-DRRN002NG2A3, Honeywell Sensing) is connected to each tee and reads the pressure drop across the length of the 3.175 mm OD tube. The pressure reading is then used to calculate the aerosol flow rate.

An Arduino Uno was programmed to control the sheath and exhaust flows using proportional-integral-derivative (PID) control. The sheath and exhaust flows are each driven by a regenerative blower (Model SE12RE21SA/081780, Ametek Dynamic Fluids Solution). Each blower has its own mass airflow sensor (Model Zephyr HAFUHT0020L3AXT, Honeywell Sensing) to measure the volumetric flow rate through the sheath and exhaust lines. The absolute pressure (Model SSCSRNN1.6BA7A3, Honeywell Sensing) and temperature (Type-K thermocouple, Omega) are also measured and are used to calculate the volumetric flows. Both the sheath and exhaust flows are filtered to keep the mass flow meters clean. The HEPA filters (HEPA Capsule 12144, Pall Laboratory) are located immediately upstream of the mass airflow

sensors. The code written to control the Arduino can be found in Appendix D. Engineering drawings for the UAC can be found in Appendix E.

4.2 Experimental Setup

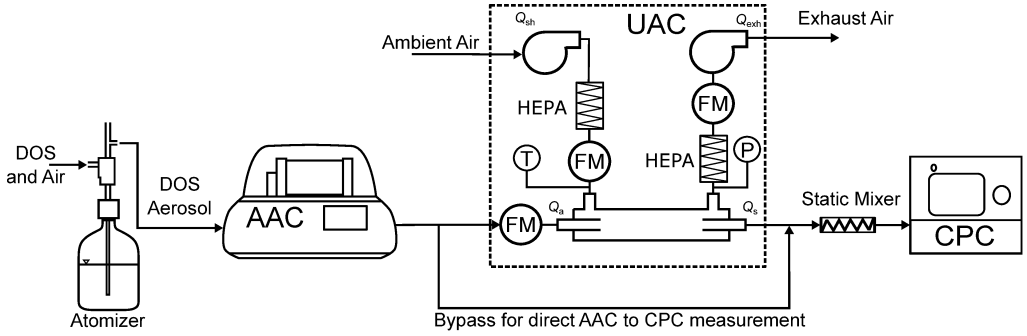
4.2.1 Particle Penetration Testing

Schematics of the particle penetration experimental setup are shown in Figure 4.2. The UAC was set to gas-only dilution mode (*e.g.* $Q_{\text{sh}} = Q_{\text{exh}}$) for all tests.

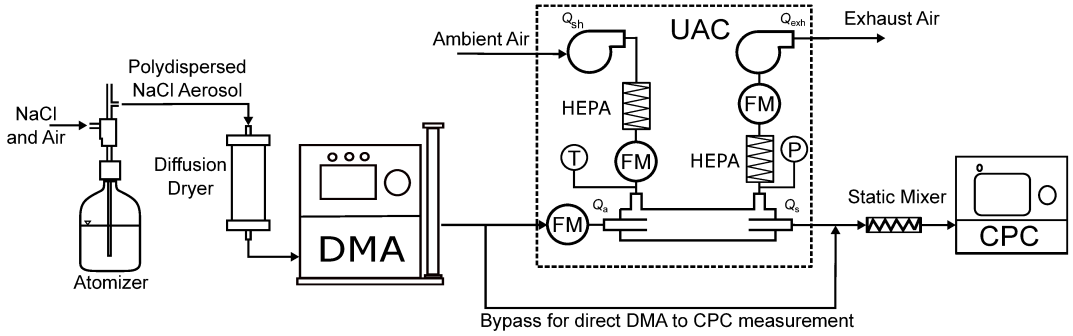
Bis (2-ethylhexyl) sebacate, commonly referred to as "DOS", and sodium chloride (NaCl) particles were used for testing. The particles were generated with an atomizer (Model 3076, TSI Inc.). The NaCl ($\geq$99.0%, Sigma-Aldrich) particles were generated from a solution of distilled water (0.5% mass fraction) while the DOS particles were generated from pure bis (2-ethylhexyl) sebacate (>97.0%, Aldrich Chemistry). The DOS was used to test particle sizes between $\sim$ 50 nm to 1 micron, while the NaCl was used to test particle sizes smaller than 50 nm.

To measure the UAC's particle penetration efficiency, a monodisperse aerosol was drawn into the UAC. The particle concentration in the aerosol was measured before and after the UAC and particle penetration was then calculated by taking the ratio of particle concentration post-UAC to the concentration pre-UAC. A TSI Ultrafine Condensation Particle Counter 3776 (CPC) was used to measure the particle concentrations.

In a CPC, the aerosol is drawn into a saturation chamber where it is brought into contact with a supersaturated vapour. The vapour will condense onto the particles, forming droplets that are easily detectable using optical methods. These enlarged particles are then drawn into an optical chamber that houses a light source (typically a laser). When a particle is illuminated by the light source, the light is either absorbed or scattered. The light scattered by the particle is detected by a photodetector.



(a) Experimental setup using Bis (2-ethylhexyl) sebacate (DOS) as particle source.



(b) Experimental setup using NaCl as particle source.

Figure 4.2: Experimental setups for testing the particle penetration through the Universal Aerosol Conditioner (UAC). Particles were classified using an Aerodynamic Aerosol Conditioner (AAC) and a Differential Mobility Analyzer (DMA). Particle concentrations were measured using a Condensation Particle Counter (CPC).

The number concentration of the particles is determined from the count rate of the photoelectric pulses generated by the photodetector.

DOS particles were used in the setup depicted in Fig. 4.2a. In this setup, a Cambustion Aerodynamic Aerosol Classifier (AAC) was used to classify the particles by their aerodynamic diameter, d_{a}.

In an AAC (Tavakoli and Olfert, 2013), the aerosol is drawn into an annular classification region formed by two rotating cylinders that are concentric to each other. The aerosol drawn into this region enters through a small gap in the inner cylinder and is carried by a particle-free sheath flow. The rotation of the cylinders generates centrifugal forces on the particles. As the particles travel through the classification region, the centrifugal force causes the particles to move radially towards the outer cylinder. Particles that have a relaxation time within a narrow band exit the classification region through a gap in the outer cylinder and leave the classifier with the sample flow. Particles that have relaxation times that are too high will impact the outer cylinder and adhere to it and those with smaller relaxation times will exit the classifier with the exhaust flow. The aerodynamic diameter can then be calculated from the relaxation time of the particles that exited with the sample flow.

The monodisperse aerosol produced by the AAC is used to determine the particle penetration as a function of particle size. Particle aerodynamic diameters of 100, 250, 500, and 1000 nm were used. Once the monodisperse aerosol exits the AAC, it is sent to either the UAC which then exits to the CPC or is bypassed to the CPC directly. To keep the diffusional losses outside of the UAC the same the bypass line was made equal in length to the UAC-CPC path. Ambient air was used as the sheath gas.

Particle penetration for diameters smaller than 50 nm was tested using the setup shown in Fig. 4.2b. A differential mobility analyzer (Model 3080, TSI Inc.) was used to classify the aerosol instead of the AAC due to the AAC being unable to classify particles smaller than 20 nm. Before the aerosol enters the DMA, it first enters

a neutralizer which exposes the particles to high concentrations of bipolar ions (TSI Incorporated, 2009). The particles leave the neutralizer carrying a known distribution of electrical charges. The charged particles then enter the DMA which is comprised of two concentric cylinders: an outer cylinder that is electrically grounded and a charged central rod. The charged rod is kept at a negative voltage which creates an electric field between the rod and the cylinder walls. The aerosol and sheath flows are introduced from the top of the DMA and travel through the annular region formed by the cylinders. The aerosol flow surrounds the inner sheath core as the flows travel through the region. The electric field causes the charged particles to be attracted towards the charged rod through the sheath flow. The rate at which the particle travels towards the charged rod is determined by the particle's electrical mobility. Particles that have an electrical mobility within a narrow range will exit the DMA with the sample flow through a slit located at the end of the central rod. Particles that have electrical mobility's that are too high will be collected onto the charged rod while the remaining particles will exit the DMA via the exhaust flow.

NaCl particles with sizes of 11 nm (the smallest mobility size for NaCl that could be measured in the experiment) and 25 nm were used. After the aerosol was generated it was dried with a diffusion dryer (Model 3062, TSI Inc.) before entering the DMA. From there, the classified aerosol enters either the UAC or bypass line. Similar to setup (a), the bypass line from the DMA to the CPC was kept the same length as the UAC to CPC line. Ambient air was again used as the sheath gas.

When the aerosol sample exits the UAC, the particles are generally not well-mixed with the gases. Due to some mixing (which will be shown later) between the aerosol and sheath flows, the particles exiting the UAC with the sample flow will be concentrated in a region that is off center from the sample tube centerline. This becomes a problem when the particle concentration is measured with the CPC 3776. If the aerosol is not thoroughly mixed, the CPC will report inaccurate particle concentra-

tions due to the way the CPC internally splits the incoming aerosol sample for particle counting (TSI Incorporated, 2014). Depending on how the particles are orientated in the aerosol sample, the flow can be split in such a way that the CPC will either over-count or under-count the true particle concentration in the aerosol. To avoid this issue, a static mixer (Model 1/4-40-2-6-2, Koflo Corporation) was connected directly to the CPC's sample inlet. As the mixer was used to measure the particle concentration upstream and downstream of the UAC, the particle penetration measurements were not biased due to particle losses in the mixer.

4.2.2 Experimental Procedure for Particle Testing

To conduct the experiments, the desired particle size was first selected on the classifier. The sample flow rate was set to 0.3 LPM while the sheath flow rate was varied. At each sheath flow rate, the particle concentration was first measured upstream of the UAC for a sample period of 30 seconds by connecting the classifier directly to the CPC (with static mixer attached to the CPC's sample inlet) with a fixed length of tubing. After measuring the concentration, the tubing was disconnected from the CPC and then attached to the sample inlet of the UAC. At the same time, the UAC's sample outlet is connected to the CPC. The particle concentration downstream of the UAC is then measured for another period of 30 seconds. This procedure was repeated five times for each sheath flow rate tested (for a total of five samples at each sheath setting). After each sheath flow rate was tested, the sample flow rate was changed to 1.5 LPM, and the testing procedure was repeated. Once testing at 0.3 LPM and 1.5 LPM was completed, the particle size was changed and the entire testing procedure was repeated. The particle penetration is calculated by dividing the particle concentration downstream of the UAC, $n_{\text{downstream}}$, by the concentration upstream of the UAC at each flow combination, n_{upstream}:

$$\eta = \frac{n_{\text{downstream}}}{n_{\text{upstream}}} \tag{4.1}$$

4.2.3 Gas Dilution Testing

The gas dilution setup is shown in Fig. 4.3. A LI-COR LI-840A gas analyzer was used to measure the CO_2 concentration in the sample flow before and after the UAC. Ambient air was used as the sample gas while the sheath gas was dry nitrogen (N_2). The N_2 line was vented before meeting the sheath inlet of the UAC to ensure that the UAC's sheath blower was only drawing in the necessary amount of N_2. A CPC was used as a suction pump for the sample line as the LI-840A lacks an internal pump. The sample flow rate was held constant as the sheath flow was varied.

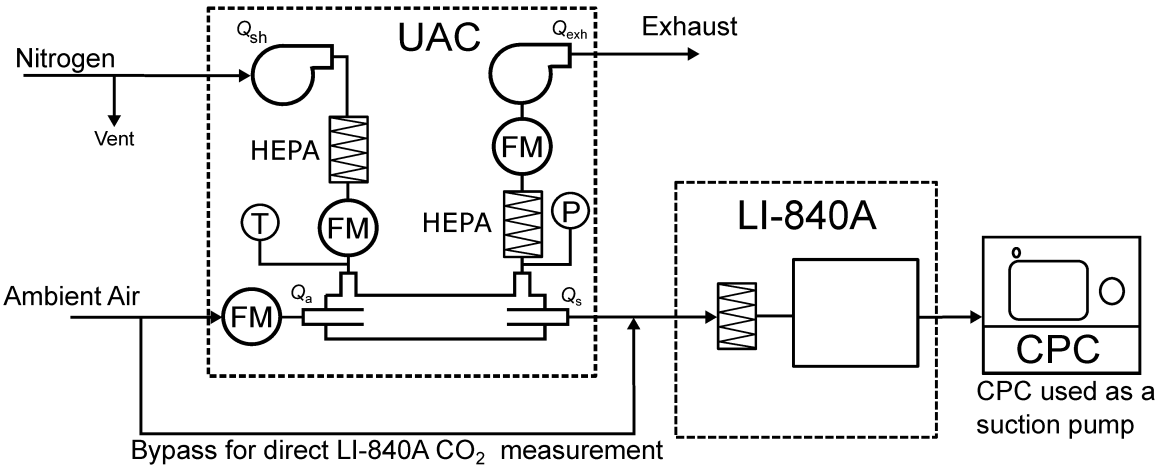


Figure 4.3: Experimental setup for testing the Universal Aerosol Conditioner's (UAC) dilution performance. A Condensation Particle Counter (CPC) was used as a suction pump for the sampling line.

4.2.4 Experimental Procedure for Gas Dilution Testing

The sample flow rate was set to 0.3 LPM while the sheath flow rate was varied. At each sheath flow rate, the CO_2 concentration in ambient air was first measured for

a period of 15 seconds. After the ambient concentration of CO_2 was measured, the sample outlet of the UAC was connected to the LI-840A. The CO_2 concentration was monitored until it decreased to a steady reading. The CO_2 concentration was then measured for 15 seconds. This procedure was repeated five times for each sheath flow rate. The dilution factor was then calculated by dividing the ambient CO_2 concentration by the CO_2 concentration downstream of the UAC:

$$\mathrm{DF} = \frac{n_{\mathrm{CO_2,ambient}}}{n_{\mathrm{CO_2,UAC}}} \tag{4.2}$$

where $n_{\mathrm{CO_2,ambient}}$ is the CO_2 concentration upstream of the UAC and $n_{\mathrm{CO_2,UAC}}$ is the CO_2 concentration downstream of the UAC.

4.3 Experimental Results and Discussion

4.3.1 Correcting for Unaccounted Diffusional Losses

Recall from Fig. 3.2 that only a region between the exit of the sample inlet and entrance of the outlet tubes in the UAC was modelled. This model does not account for the diffusional losses before and after the UAC dilution chamber. A schematic of the sampling line is shown in Fig. 4.4. L_s is the length of the sampling line segment and B_{90} is the 90° bend. The sampling line consists of seven straight tube segments and two 90° bends. The lengths for each segment are shown in Table 4.1. The inner diameter of the sampling line changes with each segment and bend so to calculate the diffusional losses in the sampling line accurately, the loss in each segment must be calculated. The inner diameter of the bends are 4.83 mm.

Gormley and Kennedy (1949) have formulated a relationship that estimates the particle penetration through straight tubing:

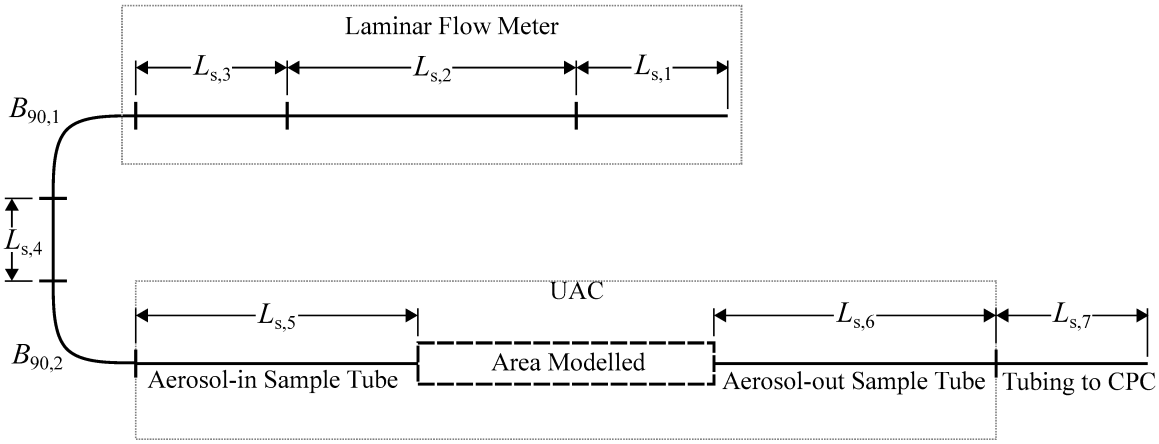


Figure 4.4: Schematic of the Sampling Line

Table 4.1: Lengths for each sampling line segment

Segment	Length (mm)
1	67
2	120
3	67
4	30
5	200
6	200
7	123

$$\eta_{\text{tube,diff}} = 1 - 2.56\xi^{2/3} + 1.2\xi + 0.177\xi^{4/3} \tag{4.3a}$$

for ξ <0.02, and

$$\eta_{\text{tube,diff}} = 0.819e^{-3.657\xi} + 0.097e^{-22.3\xi} + 0.032e^{-57\xi} \tag{4.3b}$$

for ξ >0.02, where

$$\xi = \frac{\pi D L_{\text{s}}}{Q_{\text{a}}} \tag{4.4}$$

where D is the particle diffusion coefficient and L_{s} is the length of the sampling line.

Inertial deposition losses in the sampling line bends can also be accounted for. For laminar flow Brockmann (2011) gives a relation to calculate the particle penetration through a bend:

$$\eta_{\text{bend,inert}} = \left[1 + \left[\frac{\text{Stk}}{0.171}\right]^{0.452\frac{\text{Stk}}{0.171}+2.242}\right]^{-\frac{2}{\pi}\varphi} \tag{4.5}$$

where φ is the angle of the bend in radians and Stk is the Stokes number. The Stokes number is defined as (Brockmann, 2011):

$$\text{Stk} = \frac{\tau u}{d} \tag{4.6}$$

where u is the gas velocity, d is the characteristic length (the diameter of the tube in this case), and τ, the particle relaxation time. τ is calculated using (Kulkarni et al., 2011b):

$$\tau = \frac{\rho_{\text{p}} d_{\text{p}}^2 C_{\text{c}}}{18\mu} \tag{4.7}$$

where ρ_{p} is the density of the particle, d_{p} is the particle diameter, C_{c} is the Cunningham correction factor, and μ is the dynamic viscosity of the aerosol gas. In Eq. 4.7,

particles are assumed to be spherical with a standard density of 1000 kg/m^3. Particles from this point on will be described by their equivalent electrical mobility diameter, d_{B}. Thus the particle effective density, ρ_{e}, and d_{B} will be used in Eq. 4.7 instead. The effective density of a particle is defined as the mass of the particle divided by its mobility-equivalent volume. Aerodynamic diameters can be converted to mobility diameters with the following equation (DeCarlo et al., 2004):

$$d_{\mathrm{B}} = d_{\mathrm{a}} \sqrt{\frac{\rho_0 C_{\mathrm{c}}(d_{\mathrm{a}})}{\rho_e C_{\mathrm{c}}(d_{\mathrm{B}})}} \tag{4.8}$$

where ρ_0 is the standard density of a particle (1 g/cm^3). The overall penetration of particles through the sample line (excluding the UAC) is the product of the particle loss mechanisms in each tube section and bend:

$$\eta_{\mathrm{transport}} = \prod_{\mathrm{diffusion}} \cdot \prod_{\mathrm{inert,bends}} \cdot \eta_{\mathrm{mechanisms}} \tag{4.9}$$

Thus to account for the extra diffusional losses, the theoretical particle penetration from the model can be corrected by using:

$$\eta_{\mathrm{overall}} = \eta_{\mathrm{model}} \cdot \eta_{\mathrm{transport}} \tag{4.10}$$

Note that these equations assume the particle concentration is uniform at the entrance of each region.

4.3.2 Particle Penetration Results

Particle penetration was tested at two different sample flow rates. Aerosol sample flow rates of 0.3 LPM (low flow; LF) and 1.5 LPM (high flow; HF) were used as these are the two flow rates at which the CPC can operate. For each particle size and sample flow rate, the UAC sheath flow rate was varied and the particle concentration

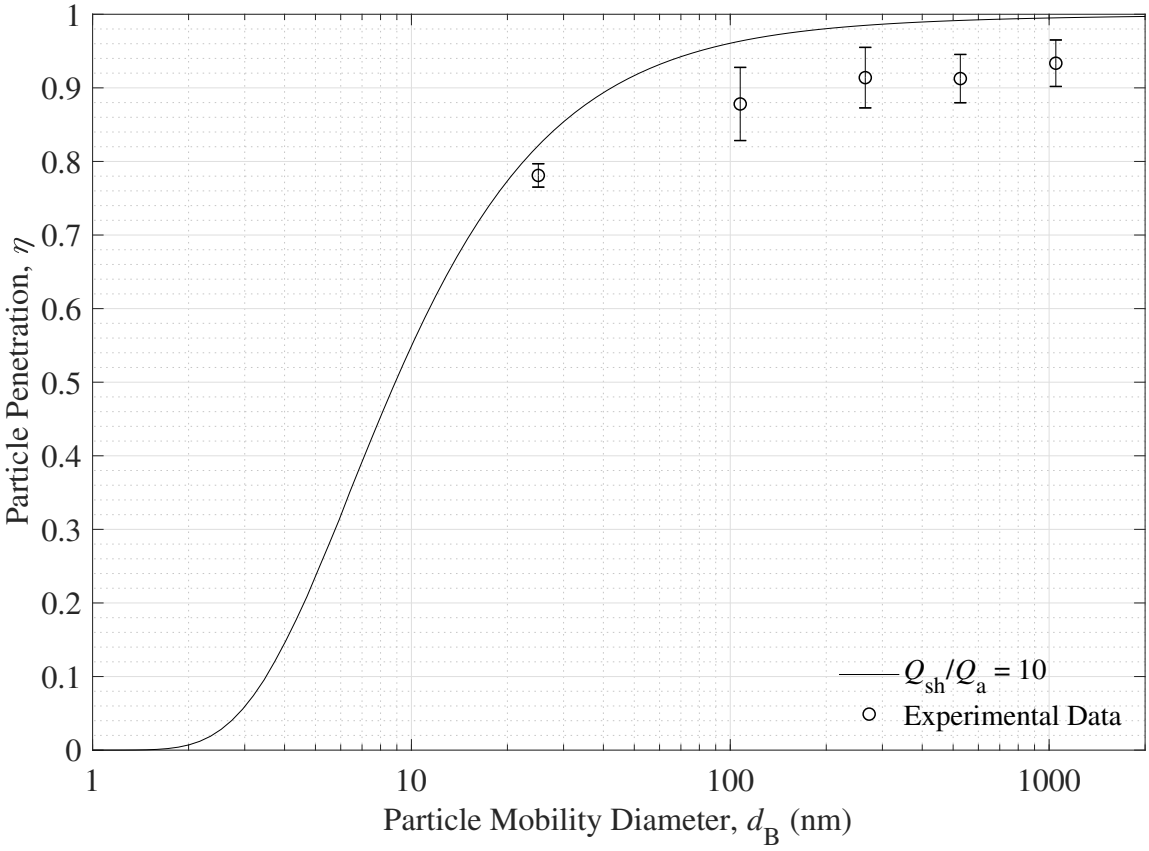


Figure 4.5: Experimental η compared to the theoretical model. Flow ratio = 10 and $Q_a = 0.3$ LPM. Corrected for additional diffusional losses in the sampling line.

was recorded. The maximum sheath flow rate the regenerative blowers could produce was 13 LPM.

Particle penetration over a range of mobility diameters and flow ratios are shown in Figures 4.5 and 4.6. For reference, the particle Pe in Fig. 4.5 spans from ∼4,900 to 8.42 $\times 10^6$ and ∼6,800 to 1.20 $\times 10^7$ in Fig. 4.6. The error bars presented in these figures and those following represent the precision uncertainty, P_x, in the measurement. The precision uncertainty is comprised entirely of random errors that arise when a measurement is taken. These errors are different for each successive measurement. The precision uncertainty can be estimated statistically within a certain amount of confidence. The error bars in these figures were estimated with a confidence interval, c_i, of 95% by using the following (Beckwith et al., 2007):

$$P_{\mathrm{x}} = t_{\alpha/2,N} \frac{S_x}{\sqrt{p}} \tag{4.11}$$

where p is the number of samples and S_x is the standard deviation in the measurements taken. $t_{\alpha/2,N}$ is the t-score from the t-distribution where $\alpha = 1 - c_i$ and $N = p - 1$. α is sometimes referred to as the level of significance while N is the degrees of freedom.

Fig. 4.5 shows good agreement between the experimental data and model when the sample flow rate is set to 0.3 LPM and a flow ratio of 10 is used. In particular, the model seems to predict diffusional losses very well at small particle sizes. At a mobility diameter of 25 nm, there is only a difference of ∼0.04 between the model and experiment. η for mobility diameters larger than 100 nm were 0.88 or better. Note that 11 nm particles could not be measured when Q_{a} is set to 0.3 LPM as the diffusional losses in the sample line prior to the UAC were too high.

The offset between the experimental data and the theoretical penetration at larger particle sizes suggests that at this flow setting, there is a possibility that there is some mixing between the aerosol and sheath flow, meaning the gas streamlines do not follow the trajectories as expected in the model. Examining Fig. 4.6 reinforces the idea that there is some flow instability. In this figure the aerosol flow rate was set to 1.5 LPM and the sheath flow rate was set to 13 LPM. The experimental particle penetration in Fig. 4.6 at all particle sizes tested are significantly lower than predicted. It is interesting to note that the experimental penetrations are quite close to each other in value. Even particles with a large mobility diameter of 1050 nm see large losses once passed through the UAC. On average ∼54% of the particles do not reach the UAC sample outlet. As the particle penetration is only weakly dependent on particle size other loss mechanisms besides diffusion must be prevalent. As mentioned earlier, mixing between the flows would explain why the experimental penetration values are

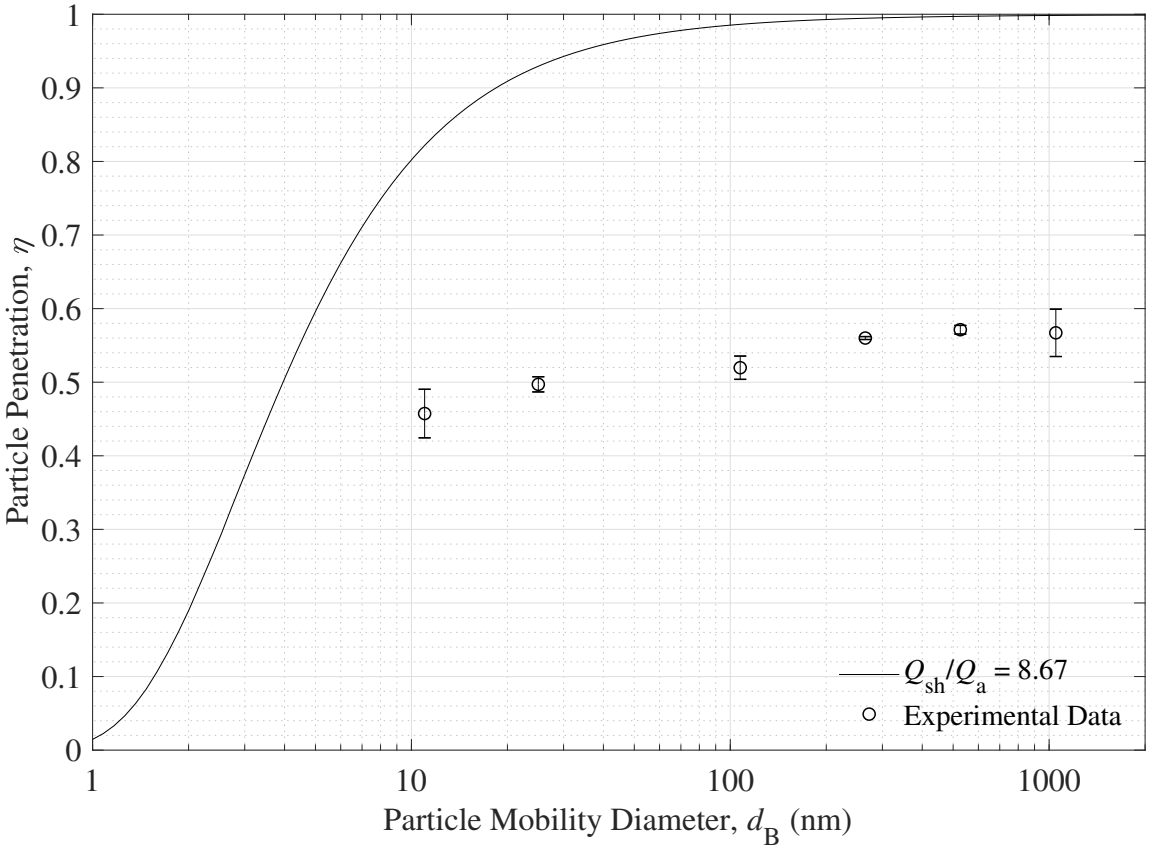


Figure 4.6: Experimental η compared to the theoretical model. Flow ratio = 8.67 and $Q_a = 1.5$ LPM. Corrected for additional diffusional losses in the sampling line.

significantly lower and why the data forms a relatively flat slope.

To investigate if mixing is occurring, η was measured for a particle with a mobility diameter of 1050 nm across several different sheath flow rates as shown in Fig. 4.7. The sample flow rate was set to 1.5 LPM to minimize diffusional losses. The smallest particle Pe presented here is 2.0 $\times 10^6$, meaning little to no particle diffusion is expected as seen by the flat theoretical line. In this figure, the experimental data of η remains constant (with an offset from the theory) for sheath flow rates less than 8 LPM. For sheath flow rates greater than 8 LPM, penetration starts to decrease. The small offset at the lower flow rates suggests mixing is already present at low sheath flow rates and continually gets worse once the sheath flow rate passes a certain threshold. This means that the sheath flow is not evenly distributed in the

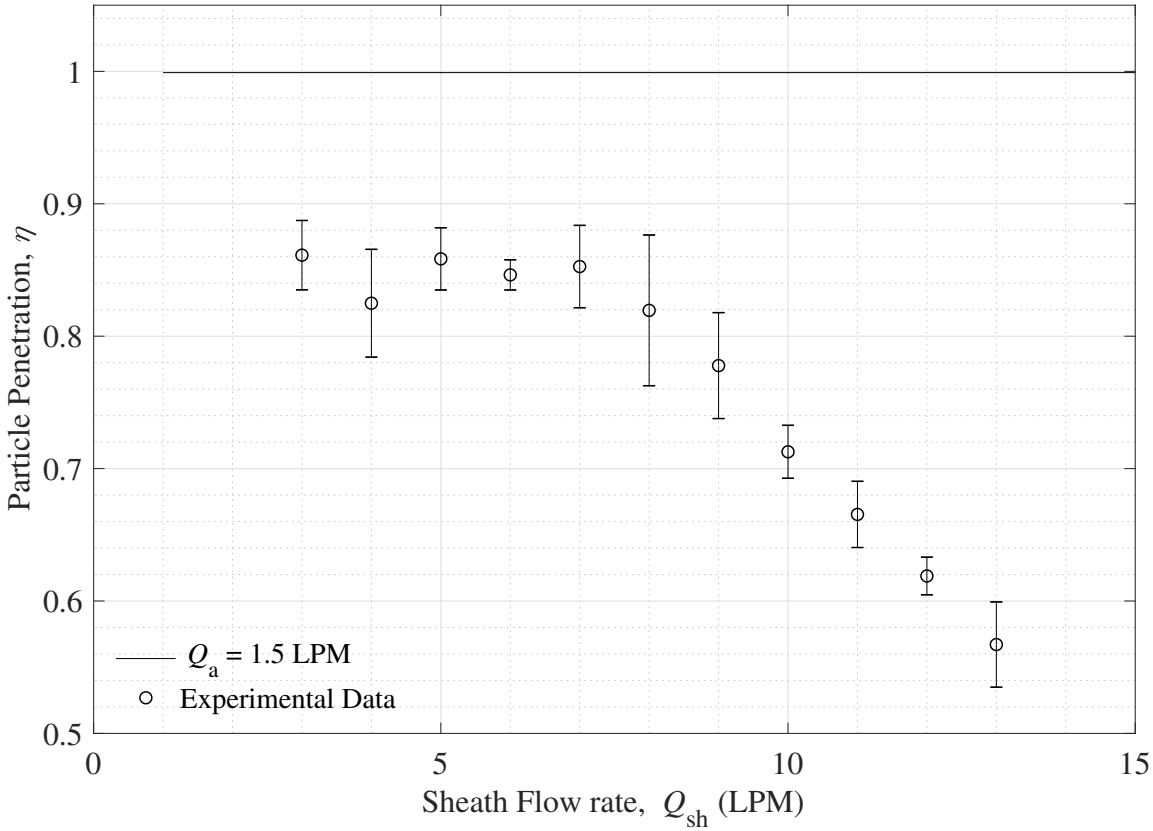


Figure 4.7: Particle Penetration for $d_B = 1050$ nm as a function of Q_{sh} with $Q_a = 1.5$ LPM.

dilution chamber and that the flow has yet to be fully developed as assumed in the model. More evidence of this will be shown in the following section.

For comparison, the penetration with the sample flow set to 0.3 LPM is shown in Fig. 4.8. There are a few interesting things to note in this figure. Firstly, the penetration at sheath flows ranging from 1 to 5 are very high; the particle penetration on average is greater than 0.90 for flow rates in this range. The particle penetration for sheath flows ranging from 3 to 6 LPM are higher when the sample flow rate is at 0.3 LPM than it is at 1.5 LPM. Penetration was observed to have started decreasing earlier at 0.3 LPM than at 1.5 LPM. With sheath flow rates greater than 6 LPM, the penetration decreases rapidly. At a sheath flow of 7 LPM, the particle penetration at 0.3 LPM sample flow is significantly worse than at 1.5 LPM (a difference of $\sim$0.21).

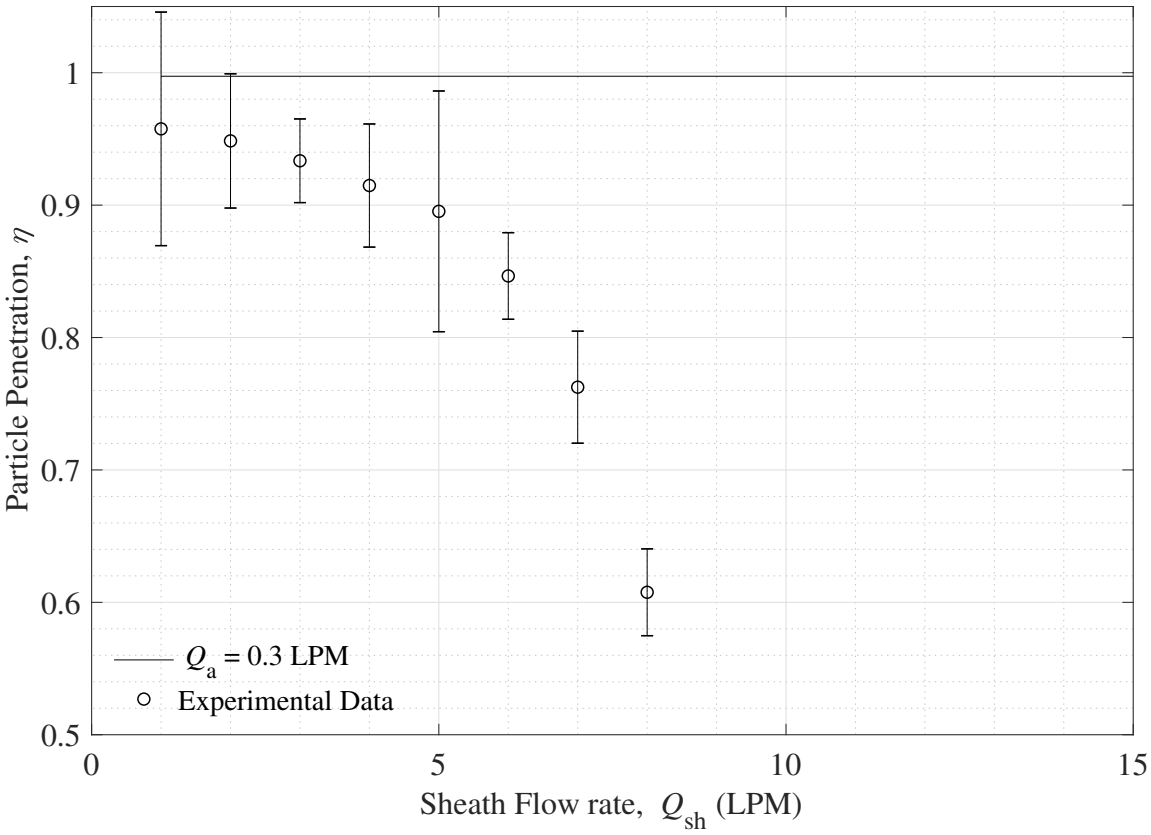


Figure 4.8: Particle Penetration for $d_{\mathrm{B}} = 1050$ nm as a function of Q_{sh} with $Q_{\mathrm{a}} = 0.3$ LPM.

4.3.3 Gas Dilution Results

Gas dilution was only tested with a Q_a of 0.3 LPM because the flow rate of 1.5 LPM exceeded the maximum that the LI-840A is capable of operating at. The results of the dilution test are shown in Fig. 4.9. The expected dilution factor shown in the figure is the dilution factor predicted by the theoretical model while the maximum dilution factor is the largest dilution factor possible when the aerosol and sheath flows are completely mixed. The area in the plot that is bounded by the expected and maximum dilution factor curves are regions where mixing occurs.

At low sheath flows, the maximum and expected dilution factors are the same because the gas has sufficient time to uniformly diffuse across the dilution tube. Once Q_{sh} becomes larger than three, the two curves begin to deviate from each other. This implies that the Pe for CO_2 is too high for the CO_2 in the aerosol to completely diffuse which explains why the expected dilution factor tapers off. For reference, the data points shown in the figure correspond to gas Pe ranging from 1.72 to 13.7. With small sheath flows (Pe ranges from 1.72 to 4.38 for $Q_{sh} = 1$ to 3), the experimental DF aligns very closely with theory. This is interesting to note as not only does this validate the gas-phase portion of the model, but it shows that the offsets seen in the particle penetration results in Fig. 4.5 are actually not due to flow mixing. The offsets from the theoretical penetration are more likely due to the inaccurate assumptions used to model the flow development or it could be due to mechanisms that were not accounted for. The dilution factors measured at high sheath flow rates confirm that the large particle losses seen in the previous section (flow ratio of 8.67, $Q_a = 1.5$ LPM) are due to flow mixing since the DF data points lie in the region where mixing is expected to occur. Dilution factors at these flow rates were larger than expected.

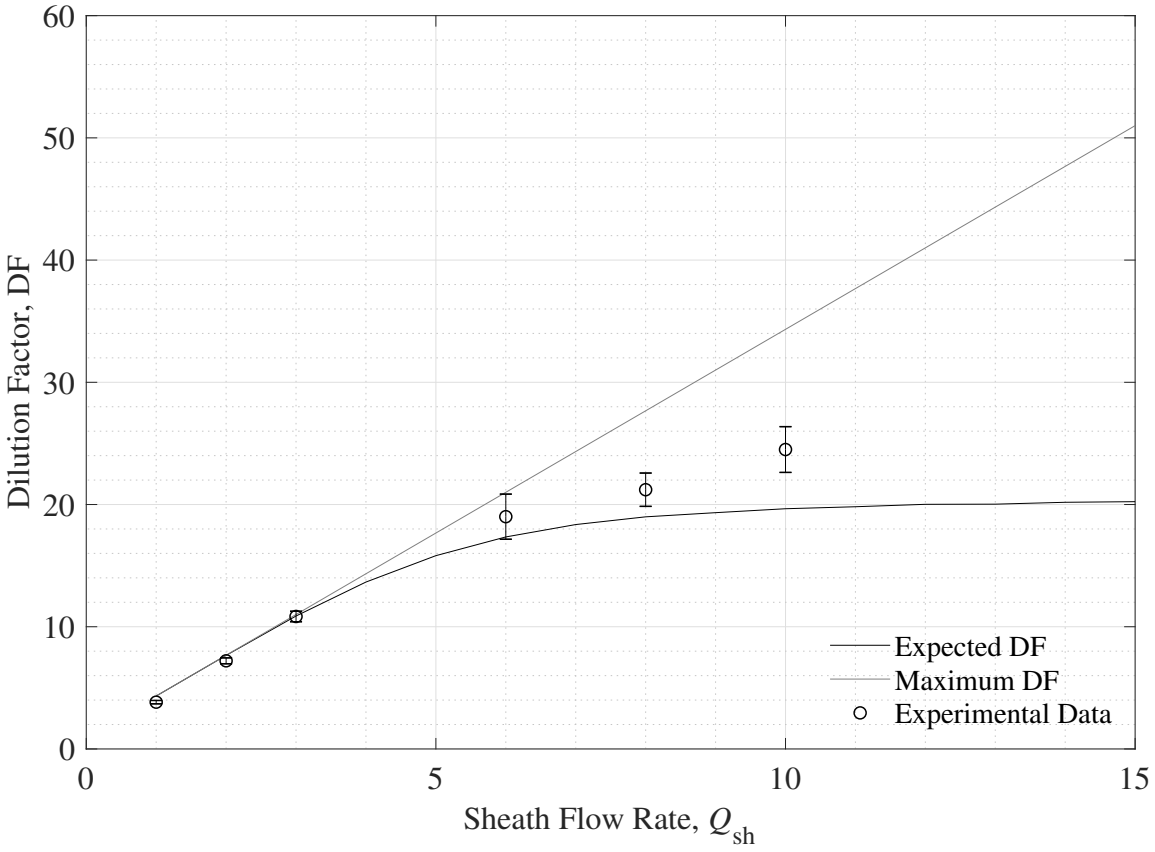


Figure 4.9: Experimental CO_2 DF compared to the theoretical model at various Q_{sh} with $Q_a = 0.3$ LPM.

4.4 Summary

A UAC prototype was developed and tested. The gas-dilution mode of the UAC was tested to validate the theoretical model. This mode has the most novel application as no other aerosol conditioning device currently fulfills the same role. The particle penetration and gas dilution factors at several different mobility diameters and flow combinations were measured with a CPC and LI-840A respectively. The measured penetration at a sample flow of 0.3 LPM and a 10:1 sheath-to-aerosol ratio showed good agreement with the model. An offset was observed on all of the data points at this flow setting which indicates that there are some mechanisms that were not accounted for in the model. The measured penetration for 25 nm in particular showed very good agreement with theory as it was only 5% smaller than predicted. The measured penetration for mobility sizes of 107, 265, 523, and 1050 nm were on average ∼8.4% smaller than the model. The measured penetrations at a sample flow of 1.5 LPM and flow ratio of 8.67 on the other hand deviated significantly from the model. The particle penetration on average was ∼46% smaller than the model for all of the mobility diameters that were tested. Flow mixing between the aerosol and sheath flows was confirmed to be the reason for the large particle losses in the high flow particle tests through a combination of penetration and gas dilution tests.

Experimental results for the CO_2 dilution in the aerosol showed very good agreement with the theoretical model for small sheath flows. With higher sheath flows, the measured dilution factors were much higher than expected. By comparing the measured dilution factor to the what was predicted from the model and the maximum dilution factor possible (when the flows are completely mixed), it was determined that flow mixing was responsible for the higher experimental dilution factors at high sheath flows. Mixing became progressively worse as the sheath flow was increased (once past a sheath flow rate of 3 LPM).

Chapter 5

Conclusions

Almost all of the aerosol conditioning devices currently being used are focused in two key areas: removal of volatiles that may interfere with measurement of the particle properties and dilution of high particle concentration sources. Many particle sources that are of interest for study tend to produce very high particle concentrations. These concentrations are so high that they often must be diluted to much smaller concentrations to allow measurement devices to operate properly. Since so many of these particle sources naturally create many particles, aerosol conditioning devices have almost exclusively focused on diluting all of the components that make up the aerosol. This becomes a problem when an aerosol that has very low particle concentration needs to conditioned.

The UAC was developed to fulfill this niche in aerosol studies. The UAC combines this niche along with the other traditional conditioning techniques to serve as a general-purpose aerosol conditioning device. While there are other conditioning devices that can condition the aerosol without decreasing the particle concentration, they often employ the use of a porous membrane to separate the aerosol from the diluent gas. These membranes often only allow certain species to traverse across them limiting the usefulness of the conditioning device.

5.1 Theoretical Conclusions

A theoretical model was developed for the UAC. Through the development of the model, two key details were found:

1. Diffusion was found to be independent of radius. Once the model was made dimensionless, diffusion was shown to have no dependency on the radius of the dilution chamber. This discovery falls inline with the conclusions found by Gormley and Kennedy (1949). Instead of the radius, diffusion was influenced by the sheath-to-aerosol flow ratio. Large sheath-to-aerosol flow ratios decreased η and increased DF while smaller flow ratios would do the opposite.

2. Both η and the gas DF are functions of Pe. Conveniently, diffusion in the UAC only relies on the sheath-to-aerosol ratio and Pe. Diffusion is much greater at low Pe than at high Pe. This property can be exploited as the typical particle has a much smaller diffusion coefficient than the typical gas species. As a result, for identical flow conditions the particles in the aerosol will have Pe that are orders of magnitude larger than the gas Pe. Thus, it is possible for η and DF to be high simultaneously. Pe can be increased by increasing the volumetric flow rates of the aerosol and sheath flows and by shortening the length between the aerosol inlet and outlet tubes which will increase the particle penetration and decrease the dilution factor. Conversely, Pe can be decreased by decreasing the flow rates or by increasing the length which will lead to higher dilution factors and lower penetration.

5.2 Experimental Conclusions

It was found that the particle penetration and gas dilution performance agreed quite well with the theoretical model when the sample flow was set to 0.3 LPM and an

sheath-to-aerosol ratio of 10:1 was used. The particle penetration on average was >80% for each mobility diameter tested. Dilution factors with sheath flow rates from 1 to 3 LPM were very close to the predicted values from the model. Experimental testing has shown that there may be some mechanisms not accounted for in the model or that some assumptions were incorrect, such as the assumption of fully developed flow entering the dilution chamber. Higher sample flow and larger sheath flow rates showed problematic areas with the current UAC design. Mixing between the aerosol and sheath flows was observed at higher sheath flow rates, resulting in poor particle penetration and higher than expected gas dilution.

If the UAC is kept operating solely in low flow mode, it will function fairly well in the gas-only dilution mode for flow ratios up to 10. Ratios beyond 10 are not guaranteed to work well due the mixing issue that arises at high sheath flows. In its current form, usage with the aerosol flow set to 1.5 LPM is not recommended as there simply isn't enough gas dilution occurring before the particle penetration begins to significantly decrease.

Compared to the denuder developed by Hagino, the UAC does not perform quite as well. The Hagino denuder achieves particle penetrations that are on average greater than 90% (for particle sizes ranging from 20 nm to 300 nm) versus the 80% observed in the UAC. The particle penetrations observed in Hagino's experiments are almost identical to what the theory predicted. The UAC gas dilution performance is quite comparable to the Hagino denuder when they are operated at the same flow rates. At a sample flow rate of 0.3 LPM and a 10:1 sheath-to-aerosol flow rate ratio, the UAC removes ∼90% of the CO_2 from the aerosol while the Hagino denuder removes ∼95%. The Hagino denuder is able to hold a performance advantage over the UAC while also being roughly half the physical size.

5.3 Future Work

Further work can be done to investigate how well the other UAC operational modes function in its current form such as the humidification and combined particle-gas dilution modes. Aerosol drying has been indirectly examined through the dilution tests with CO_2 as the operating principles between the two operational modes are fundamentally identical. Testing humidification could prove to be interesting from a theoretical standpoint since the model would be work in inverse (*e.g.* diffusion from the sheath flow into the aerosol).

There are a few avenues in which the UAC design can be improved to increase η at high sheath flows. Although the UAC is quite long already, the overall length of the device could be increased to allow the sheath flow to become fully developed. Instead of increasing the length, the diameters of the dilution tube and annular regions in the manifold can be increased. While dilution was found to be independent of radius, increasing the diameters of these items will decrease the average velocity of the sheath flow which should aid in distributing the sheath flow evenly (which will reduce the amount of mixing). The sheath flow can also be distributed more evenly by creating a larger pressure drop in the annular region in the manifolds.

References

Abramowitz, M. and Stegun, I. A. (1964). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, ninth dover printing, tenth gpo printing edition.

Anderson, J. O., Thundiyil, J. G., and Stolbach, A. (2012). Clearing the air: A review of the effects of particulate matter air pollution on human health. *Journal of Medical Toxicology*, 8(2):166–175.

Beckwith, T., Marangoni, R., and Lienhard, J. (2007). *Mechanical Measurements*. Pearson Prentice Hall.

Bergmann, A. and Pongratz, H. (2013). Rotating disc diluter for fluid flows. *U.S. Patent 8,435,512.*

Boucher, O., Randall, D., Artaxo, P., Bretherton, C., Feingold, G., Forster, P., Kerminen, V.-M., Kondo, Y., Liao, H., Lohmann, U., Rasch, P., Satheesh, S. K., Sherwood, S., Stevens, B., and Zhang, X. Y. (2013). *Clouds and aerosols*, pages 571–657. Cambridge University Press, Cambridge, UK.

Brockmann, J. E. (2011). Aerosol transport in sampling lines and inlets. In Kulkarni, P., Baron, P. A., and Willeke, K., editors, *Aerosol Meas. Princ. Tech. Appl.*, chapter 6, pages 71, 72, 90–94. John Wiley & Sons, Inc., Hoboken, 3 edition.

Bryne, J. D. and Griffith, P. A. (1977). Gas Analysis Diluter. *U.S. Patent 4,004,882.*

Burtscher, H., Baltensperger, U., Bukowiecki, N., Cohn, P., Hüglin, C., Mohr, M., Matter, U., Nyeki, S., Schmatloch, V., Streit, N., and Weingartner, E. (2001). Separation of volatile and non-volatile aerosol fractions by thermodesorption: Instrumental development and applications. *J. Aerosol Sci.*, 32(4):427–442.

Çengel, Y. A. and Cimbala, J. M. (2010). *Fluid Mechanics: Fundamentals and Applications*, chapter 8, page 344. McGraw-Hill Higher Education, 2 edition.

Cheng, Y.-S. S. (2011). Instruments and Samplers Based on Diffusion Separation. In Kulkarni, P., Baron, P. A., and Willeke, K., editors, *Aerosol Meas. Princ. Tech. Appl.*, chapter 16, pages 365–79. John Wiley & Sons, Inc., Hoboken, 3 edition.

Davis, E. J. (1973). Exact solutions for a class of heat and mass transfer problems. *The Canadian Journal of Chemical Engineering*, 51(5):562–572.

DeCarlo, P. F., Slowik, J. G., Worsnop, D. R., Davidovits, P., and Jimenez, J. L. (2004). Particle morphology and density characterization by combined mobility and aerodynamic diameter measurements. part 1: Theory. *Aerosol Science and Technology*, 38(12):1185–1205.

Dekati (2017). Dekati diluter operating principle. `https://www.dekati.com/products/Aerosol%20Sample%20Conditioning/Dekati%C2%AE%20Diluter`. Web. Accessed on Feb. 20, 2017.

Fiore, A. M., Naik, V., and Leibensperger, E. M. (2015). Air quality and climate connections. *Journal of the Air & Waste Management Association*, 65(6):645–685. PMID: 25976481.

Friedlander, S. K. (2000). *Smoke, Dust, and Haze - Fundamentals of Aerosol Dynamics (2nd Edition)*. Oxford University Press.

Gehner, G. S., Graze Jr, R. R., Iyer, K. G., Koch, K. A., Mallampalli, H. P., and Wagner, J. R. (2006). Method and apparatus for mixing gases. *U.S. Patent 7,100,459.*

Giever, P. M. (1974). An Automatic Monitor for Particulate Matter Emissions.pdf. *Adv. Instrum.*, 29(Part 3):708.

Gormley, P. G. and Kennedy, M. (1949). Diffusion from a stream flowing through a cylindrical tube. *Proceedings of the Royal Irish Academy. Section A: Mathematical and Physical Sciences*, 52:163–169.

Graze Jr, R. R. (1991). Gas sampling device and dilution tunnel used therewith. *U.S. Patent 5,058,440.*

Graze Jr, R. R. (2009). Dilution tunnel. *U.S. Patent 7,549,350.*

Hagino, H. (2016). Laboratory evaluation of nanoparticle penetration efficiency in a cylindrical counter flow denuder for non-specific removal of trace gases. *Aerosol Sci. Technol.*, 0(0):1–8.

Kim, K.-H., Kabir, E., and Kabir, S. (2015). A review on the human health impact of airborne particulate matter. *Environment International*, 74:136 – 143.

Kulkarni, P., Baron, P. A., Sorensen, C. M., and Harper, M. (2011a). Nonspherical Particle Measurement: Shape Factor, Fractals, and Fibers. In Kulkarni, P., Baron, P. A., and Willeke, K., editors, *Aerosol Meas. Princ. Tech. Appl.*, chapter 23, page 510. John Wiley & Sons, Inc., Hoboken, 3 edition.

Kulkarni, P., Baron, P. A., and Willeke, K. (2011b). Particle Migration in External Force Fields. In Kulkarni, P., Baron, P. A., and Willeke, K., editors, *Aerosol Meas. Princ. Tech. Appl.*, chapter 1, page 25. John Wiley & Sons, Inc., Hoboken, 3 edition.

Lauwerier, H. A. (1951). The use of confluent hypergeometric functions in mathematical physics and the solution of an eigenvalue problem. *Applied Scientific Research*, 2(1):184.

McMurry, P. H. and Stolzenburg, M. R. (1989). On the sensitivity of particle size to relative humidity for Los Angeles aerosols. *Atmos. Environ.*, 23(2):497–507.

Mikkaanen, P. and Koskinen, T. (2010). Device and a method for diluting a sample. *U.S. Patent 7,759,131.*

Sideman, S., Luss, D., and Peck, R. E. (1965). Heat transfer in laminar flow in circular and flat conduits with (constant) surface resistance. *Applied Scientific Research, Section A*, 14(1):157–171.

Smith, G. D. (1985). *Numerical Solution of Partial Differential Equations: Finite Difference Methods*. Oxford University Press, 3rd edition.

Smith, T. P. and Burke, D. J. (2012). Membrane gas dryers incorporating rotatable fittings. *U.S. Patent 8,268,055.*

Stolzenburg, M. (1988). *An Ultrafine Aerosol Size Distribution System*. PhD thesis.

Ström, H., Ekdahl, R., and Harbuck, E. S. (1995a). Gas sampling method and dilution tunnel therefor. *U.S. Patent 5,450,749.*

Ström, H., Ekdahl, R., and Harbuck, E. S. (1995b). Gas sampling method and dilution tunnel therefor. *U.S. Patent 5,410,907.*

Swanson, J. and Kittelson, D. (2010). Evaluation of thermal denuder and catalytic stripper methods for solid particle measurements. *J. Aerosol Sci.*, 41(12):1113–1122.

Tavakoli, F. and Olfert, J. S. (2013). An instrument for the classification of aerosols by particle relaxation time: Theoretical models of the aerodynamic aerosol classifier. *Aerosol Science and Technology*, 47(8):916–926.

Thomas, L. H. (1949). Elliptic problems in linear difference equations over a network. *Watson Sci. Comput. Lab. Rept., Columbia University, New York*, 1.

TSI Incorporated (2009). *Series 3080 Electrostatic Classifiers - Operation and Service Manual*. TSI Incorporated, J edition.

TSI Incorporated (2014). *Ultrafine Condensation Particle Counter Model 3776 - Operation and Service Manual*. TSI Incorporated, H edition.

TSI Incorporated (2015). TSI Thermodiluter and Thermal Conditioner. `http://www.tsi.com/uploadedFiles/_Site_Root/Products/Literature/Brochures/379020-30thermodiluters.pdf`. PDF. Accessed on Feb. 20, 2017.

Tyree, C. and Allen, J. (2004). Diffusional particle loss upstream of isokinetic sampling inlets. *Aerosol Science and Technology*, 38(10):1019–1026.

Wehner, B., Philippin, S., and Wiedensohler, A. (2002). Design and calibration of a thermodenuder with an improved heating unit to measure the size-dependent volatile fraction of aerosol particles. *J. Aerosol Sci.*, 33(7):1087–1093.

Yamasaki, A., Takeda, Y., Abe, S., and Fukano, I. (1992). Multiple Flow-dividing Dilution Tunnel System. *U.S. Patent 5,090,258.*

Appendix A

Analytical Solution to the Particle-Phase Partial Differential Equation

In this section, an analytical solution for the partial differential equation posed in Chapter 3 will be presented for the particle-phase. This PDE along with its boundary conditions are analogous to the heat transfer problem originally posed by Graetz in 1883. The Graetz problem has been investigated in aerosol research for use in predicting particle sampling losses due to diffusion. However, these solutions are not applicable to the PDE posed in Chapter 3 because in these solutions particle concentrations are averaged over the entire flow rather than being presented as a function of radial position. Stolzenburg (1988) has already presented an analytical solution for the extended Graetz problem with a piecewise initial condition and has reported the particle penetration as a function of non-dimensional radius. Stolzenburg's solution will be adapted for use with confluent hypergeometric functions (CHF) in this section. CHFs have seen use in Graetz solutions after Lauwerier (1951) first proposed their use due to their superior convergence properties.

A.1 Particle-Phase Analytical Solution

Derivation of the analytical solution will first start with the non-dimensional PDE presented in Eq. 3.6 of Chapter 3. This PDE is an example of a Sturm-Liouville system and can be solved using separation of variables (SOV) and an appropriate variable transformation.

To start, separation of variables is applied to Eq. 3.6 because the equation and its boundary conditions are linear and homogeneous. The form of the general solution is:

$$\nu(\rho,\zeta) = \sum_{n=1}^{\infty} A_n R_n(\rho) \exp\left(\frac{-\beta_n^2}{\mathrm{Pe}}\zeta\right) \tag{A.1}$$

where A_n are the coefficients, β_n are the eigenvalues, and $R_n(\rho)$ are the eigenfunctions of the ordinary differential equation (derived from SOV):

$$\rho\frac{\mathrm{d}^2R_n}{\mathrm{d}\rho^2} + \frac{\mathrm{d}R_n}{\mathrm{d}\rho} + \beta_n^2\rho(1-\rho^2)R_n = 0 \tag{A.2}$$

with boundary conditions:

$$R_n(0) = 1 \tag{A.3}$$

$$R_n(1) = 0 \tag{A.4}$$

$$\frac{\mathrm{d}R_n}{\mathrm{d}\rho}(0) = 0 \tag{A.5}$$

Equation A.2 can be transformed into Kummer's Equation by using two variable transformations (Davis, 1973): $x = \beta_n\rho^2$ and $W(x) = e^{\frac{x}{2}}R_n$. The general form of Kummer's equation (Abramowitz and Stegun, 1964) is:

$$z\frac{\mathrm{d}^2W}{\mathrm{d}z^2} + (b-z)\frac{\mathrm{d}W}{\mathrm{d}z} - aW = 0 \tag{A.6}$$

Once transformed, Eq. A.2 becomes:

$$x\frac{\mathrm{d}^2W}{\mathrm{d}x^2} + (1-x)\frac{\mathrm{d}W}{\mathrm{d}x} + \left(\frac{\beta_n}{4} - \frac{1}{2}\right)W = 0 \tag{A.7}$$

There are two linearly independent solutions to Kummer's equation, but only one of the solutions is bounded at $x = 0$. Recall from Eq. A.3 that $R_n(0)$ is bounded and thus $W(0)$ must be bounded as well. For the bounded case, the general solution to the Kummer's equation is the Kummer's function. The general form of the Kummer's function is:

$$M(a,b,z) = \sum_{n=0}^{\infty} \frac{a^{(n)}z^n}{b^{(n)}n!} \tag{A.8}$$

Thus, the solution to Eq. A.7 is:

$$W(x) = CM\left(\frac{1}{2} - \frac{\beta_n}{4}, 1, x\right) \tag{A.9}$$

where C is an integration constant. Note that the constant will be later combined into the A_n term of Equation A.1. The eigenvalues can be calculated by substituting Eq. A.3 into Eq. A.9, and then solving for the roots of the equation. Eigenvalues for Eq. A.9 have already been calculated by Tyree and Allen (2004) and can be found in the Table A.1.

Since Eq. A.2 is a Sturm-Liouville system, the A_n terms in Eq. A.1 can be calculated using (Stolzenburg, 1988):

$$A_n = \frac{\int_0^1 \rho(1-\rho^2)R_n(\rho)f(\rho,0)\,\mathrm{d}\rho}{\int_0^1 \rho(1-\rho^2)R_n^2(\rho)\,\mathrm{d}\rho} \tag{A.10}$$

Table A.1: Graetz problem eigenvalues and coefficients using the Kummer's function solution.

n	β_n	$A_n(1)$	$A_n(\rho_a)$	$A_n(1)$	$A_n(\rho_a)$
		(10:1)		(13:1.5)	
1	2.7044	1.47644	0.23196	1.47644	0.26239
2	6.6790	-0.80612	0.46363	-0.80612	0.50724
3	10.673	0.58876	0.46056	0.58876	0.46308
4	14.671	-0.47585	0.24218	-0.47585	0.18050
5	18.670	0.40502	-0.05478	0.40502	-0.14533
6	22.669	-0.35576	-0.26100	-0.35576	-0.30982
7	26.669	0.31917	-0.27448	0.31917	-0.23316
8	30.668	-0.29074	-0.11502	-0.29074	-0.00255
9	34.668	0.26789	0.09763	0.26789	0.20142
10	38.668	-0.24906	0.22435	-0.24906	0.23686
11	42.668	0.23323	0.19345	0.23323	0.09546
12	46.668	-0.21969	0.03933	-0.21969	-0.10356
13	50.668	0.20796	-0.12785	0.20796	-0.20945
14	54.668	-0.19768	-0.19815	-0.19768	-0.15143
15	58.668	0.18859	-0.13258	0.18859	0.01557
16	62.668	-0.18047	0.01654	-0.18047	0.15987
17	66.669	0.17317	0.14542	0.17317	0.17513
18	70.669	-0.16657	0.16988	-0.16657	0.05763
19	74.670	0.16057	0.07957	0.16057	-0.09688
20	78.671	-0.15507	-0.05926	-0.15507	-0.17008
21	82.667	0.15003	-0.15149	0.15003	-0.11086
22	86.667	-0.14537	-0.13768	-0.14537	0.02962
23	90.667	0.14106	-0.03198	0.14106	0.14127
24	94.667	-0.13706	0.09068	-0.13706	0.14070
25	98.667	0.13333	0.14717	0.13333	0.03308
26	102.67	-0.12984	0.10219	-0.12984	-0.09546
27	106.67	0.12657	-0.01005	0.12657	-0.14630
28	110.67	-0.12351	-0.11136	-0.12351	-0.08363
29	114.67	0.12062	-0.13379	0.12062	0.04068
30	118.67	-0.11789	-0.06495	-0.11789	0.12964

where $f(\rho, 0)$ is the particle concentration profile of the aerosol entering the UAC:

$$f(\rho,0) = \begin{cases} 1 & 0 \leq \rho < \rho_a \\ 0 & \rho_a \leq \rho \leq 1 \end{cases} \tag{A.11}$$

and

$$R_n(\rho) = \exp\left(\frac{-\beta_n \rho^2}{2}\right) M\left(\frac{1}{2} - \frac{\beta_n}{4}, 1, \beta_n \rho^2\right) \tag{A.12}$$

To calculate the A_n terms for the region bounded by the aerosol core, it is necessary to first find $A_n(1)$ in the case where a uniform concentration profile is present across ρ (*e.g.* $f(\rho, 0) = 1$). Substituting $f(\rho, 0) = 1$ into Eq. A.10 gives (Sideman et al., 1965):

$$A_n(1) = \frac{\int_0^1 \rho(1-\rho^2) R_n(\rho) f(\rho,0)\, \mathrm{d}\rho}{\int_0^1 \rho(1-\rho^2) R_n^2(\rho)\, \mathrm{d}\rho} = \frac{-2}{\beta_n \left[\dfrac{\partial R_n(1)}{\partial \beta_n}\right]} \tag{A.13}$$

Now by substituting Eq. A.11 into Eq. A.10, the $A_n(\rho_a)$ terms corresponding with the step-profile can be expressed as:

$$A_n(\rho_\mathrm{a}) = \frac{\int_0^{\rho_a} \rho(1-\rho^2) R_n(\rho)(1)\, \mathrm{d}\rho + \int_{\rho_a}^1 \rho(1-\rho^2) R_n(\rho)(0)\, \mathrm{d}\rho}{\int_0^1 \rho(1-\rho^2) R_n^2(\rho)\, \mathrm{d}\rho} \tag{A.14}$$

Eq. A.14 can be further simplified by using Eq. A.13 and these relationships (Sideman et al., 1965)):

$$\int \rho(1-\rho^2) R_n(\rho)\, \mathrm{d}\rho = \frac{-\rho}{\beta_n^2} \frac{\partial R_n}{\partial \rho} \tag{A.15}$$

$$\int_0^1 \rho(1-\rho^2) R_n^2(\rho)\, \mathrm{d}\rho = \frac{1}{2\beta_n} \frac{\partial R_n(1)}{\partial \beta_n} \tag{A.16}$$

resulting in:

$$A_n(\rho_a) = \rho_{\mathrm{a}} A_n(1) \left[\frac{\partial R_n(\rho_{\mathrm{a}})}{\partial \rho}\right] \left[\frac{\partial R_n(1)}{\partial \rho}\right]^{-1} \tag{A.17}$$

$A_n(\rho_{\mathrm{a}})$ was then calculated numerically for 10:1 and 13:1.5 sheath-to-aerosol ratios and their values can be found in Table A.1. Note that $A_n(\rho_{\mathrm{a}})$ must be recalculated for each sheath-to-aerosol flow ratio as this ratio will change ρ_{a}. By substituting Eq. A.14 into Eq. A.1, the solution derived from SOV becomes:

$$\nu(\rho, \zeta) = \sum_{n=1}^{\infty} A_n(\rho_{\mathrm{a}}) R_n(\rho) \exp\left(\frac{-\beta_n^2}{\mathrm{Pe}}\zeta\right) \tag{A.18}$$

With Eq. A.18, the dimensionless particle concentration can be found at any point in the UAC. By integrating this solution from the centerline of the dilution tube to the aerosol bounding radius at the sample outlet, the particle penetration can be found. η can be found by using the following (Tyree and Allen, 2004):

$$\eta(\rho, \zeta) = \frac{\int_0^{\rho_{\mathrm{a}}} \nu(\rho, \zeta)\rho(1-\rho^2)\,\mathrm{d}\rho}{\int_0^{\rho_{\mathrm{a}}} \rho(1-\rho^2)\,\mathrm{d}\rho} \tag{A.19}$$

Hence, by substituting Eq. A.18 into Eq. A.19, η can be calculated with:

$$\eta(\rho_{\mathrm{a}}, 1) = \frac{4}{\rho_{\mathrm{a}}^2(2-\rho_{\mathrm{a}}^2)} \sum_{n=1}^{\infty} A_n(\rho_{\mathrm{a}}) \left[\frac{-R_n'(\rho_{\mathrm{a}})}{\beta_n^2}\right] \exp\left(\frac{-\beta_n^2}{\mathrm{Pe}}\right) \tag{A.20}$$

Appendix B

MATLAB Implementation of the Crank-Nicolson Finite Difference Method

The method described in Chapter 3 for the Crank-Nicolson finite difference method was implemented in MATLAB. Below is the code that calculates the dimensionless concentration of particles or gases in the model domain specified in Chapter 3. The inputs for the `cn_uac` are the aerosol and sheath volumetric flow rates, `Q_a` and `Q_sh` in LPM, `type` denotes whether the particle-phase or gas-phase PDE is to be solved, and `Peclet` is the Pe of the particle or gas. The function outputs four parameters: `characteristic` – the particle penetration or gas dilution factor, `nu` – the dimensionless concentration of particles or gas, and the grid coordinates (`zeta` and `rho`). Examples of the function call are included in the code's header.

```
function [characteristic, nu, zeta, rho] = ...
        cn_uac(type, Q_a, Q_sh, Peclet)

% This m-file is the Crank-Nicolson implemintation of the
% diffusion model for the Universal Aerosol Conditioner
% Originally written: October 2015
```

```
% -Kerry Chen

% Changelog
% January 2019:
% -Added "par_dia" to function call. Allows particle diffusion
%  coefficient to be calculated in code
% -Added gas_d to function call.
% February 2019:
% -Removed par_dia, gas_dif, and tube_length from function call.
%  Function now only requires phase type, aerosol, and sheath
%  flow rates, and Peclet
% -Reorganized function output parameter order

% Assumptions to be aware of when used:
% -Flows are fully devloped.
% -Flows do not mix
% -Does not account for transport losses in the sample lines
%  pre/post dilution chamber modelling region.

%--INPUT----------------------------------------------------------------
% type   = 'par' or 'gas' (single quotes are necessary) to
%          distinguish which component concentration profile
%          is to be solved
% Q_a    = the aerosol flowrate in [LPM]
% Q_sh   = the sheath flowrate in [LPM]
% Peclet = Peclet number
%              Definition of Peclet in UAC:
%              Pe = 2*(Q_a + Q_sh)/(pi * D * L) where D is
%              diffusion coeff.
%-----------------------------------------------------------------------
```

```
%--OUTPUT---------------------------------------------------------------
% zeta           = dimensionless tube length
% rho            = dimensionless tube radius
% nu             = dimensionless concentration
% characteristic = particle penetration/dilution ratio
%-------------------------------------------------------------------
% No longer used:
% par_dia     = particle diameter size (electric mobility)
% gas_d       = gas diffusion coefficient
% tube_length = length of the diluter outer tube in mm

%--Example:-------------------------------------------------------------
% [penetration, nu, zeta, rho] = cn_uac('par', 0.3, 3, 1E4);
% [dil_factor, nu, zeta, rho] = cn_uac('par', 0.3, 3, 3);
% Gas Peclet numbers are orders of magnitude smaller than particles
%-------------------------------------------------------------------
% Useful information about prototype UAC dimensions:
% Dilution Tube length: 0.5 m edge to edge of sample inlet/outlet
% Dilution Tube OD: 3/4"
% Dilution Tube ID: 0.62" (0.065" wall thickness)
% Sample Tube OD: 1/4"
% Sample Tube ID: ~3.56 mm

% Error control
tol = 1E-3;                          %tolerance to meet
err = 1;                             %initialize error
char_old = 0;                        %initialize

%the initial number of steps in zeta and rho
step = 500;                          %step scaling factor
n = 2*step + 1;                      %total number of steps in rho
zeta_max = 1;                        %maximum zeta
```

```
Q_total = (Q_a + Q_sh);              %total flow rate in system
rho_a = sqrt(1 - sqrt(1 - Q_a / Q_total)); %aerosol core radius

while err > tol
    %Calculate the concentration until the tolerance
    %is met. The number of steps doubles for each iteration.
    rho_a_index = ceil(n*rho_a);       %Index Position for rho_a
    h = 1/(n-1);                       %step size in rho (linear)

    %Set the zeta nodes
    zeta = logspace(-4,0,step);        %logrithmic spacing
    zeta(2:length(zeta)+1) = zeta;
    zeta(1) = 0;                       %set the first zeta step

    %Set the initial condition
    nu_aero = ones(rho_a_index,1);             % Aerosol Core
    nu_sheath = zeros(floor(n*(1-rho_a)),1);   % Sheath Annular
    nu_init = [nu_aero; nu_sheath];            % Combined 1st Step

    %Initialize variables
    nu = zeros(n,length(zeta));                %initialize nu
    nu(:,1) = nu_init;                         %initialize first step

    %solve the equations for each zeta step
    for j = 2:length(zeta)
        k = zeta(j) - zeta(j-1);          %calculated zeta step size
        % Set up the systems of equations to solve
        rho = zeros(1,n);
        rho(1) = 0;
        rho(n) = 1;
        A = zeros(1,n-1);
```

```
B = zeros(1,n);
C = zeros(1,n-1);
E = zeros(1,n);
for i = 2:n-1
    rho(i) = (i-1)*h;
    %Create Tridaigonal Matrix
    %A = Major-1
    %B = Major Diagonal
    %C = Major Diagonal+1,
    A(i-1) = 1/4/h/rho(i) - 1/2/h^2;
    B(i) = Peclet*(1 - rho(i)^2)/k + 1/h^2;
    C(i) = -1/2/h^2 - 1/4/h/rho(i);
    E(i) = nu_init(i-1)*(1/2/h^2 - 1/4/h/rho(i))...
        + nu_init(i)*(Peclet*(1 - rho(i)^2)/k - 1/h^2)...
        + nu_init(i+1)*(1/2/h^2 + 1/4/h/rho(i));
end

% Set the boundary conditions for concentration
B(1) = Peclet/k + 1/h^2;                %Symmetry condition
B(n) = 1/h^2;
C(1) = -1/h^2;
E(1) = (Peclet/k - 1/h^2)*nu_init(1) +...
    (1/h^2)*nu_init(2);

% Determine which end boundary condtion to use
if strcmp(type,'par') == 1
    A(n-1) = 1/4/h/rho(n) - 1/2/h^2;    %No par conc BC
    E(n) = 0;
else
    if strcmp(type,'gas') == 1
    A(n-1) = -1/h^2;           %No gas flux through wall BC
    E(n) = (1/h^2)*nu_init(n-1) +...
```

```
            (Peclet*(1 - rho(n)^2)/k - 1/h^2)*nu_init(n);
        end
    end

    % Solve the system of equations with a tridiagonal matrix
    % solver
    n_new = tridiag1(A,B,C,E);
    nu(:,j) = n_new';              %store solution into zeta@j
    nu_init = n_new';              %reset init with current sol.

end


if strcmp(type,'par') == 1
    % Calculate particle penetration
    outlet_par = trapz((1-rho(1:rho_a_index).^2)'.*...
        nu(1:rho_a_index,step+1).*rho(1:rho_a_index)');

    inlet_par = trapz((1-rho(1:rho_a_index).^2)'.*...
        nu(1:rho_a_index,1).*rho(1:rho_a_index)');

     penetration = outlet_par/inlet_par;
     characteristic = penetration;
 else

    % Calculate the Dilution Factor
     outlet_gas = trapz((1-rho(1:rho_a_index).^2)'.*...
         nu(1:rho_a_index,step+1).*rho(1:rho_a_index)');

    inlet_gas = trapz((1-rho(1:rho_a_index).^2)'.*...
        nu(1:rho_a_index,1).*rho(1:rho_a_index)');
```

```
        dilution = inlet_gas/outlet_gas;
        characteristic = dilution;
    end

    % Error check
    err = characteristic - char_old;
    char_old = characteristic;

    step = 2*step;      %double steps
    n=2*step+1;         %recalculate the number of steps
end
```

Appendix C

MATLAB Implementation of the Implicit Euler Finite Difference Method

The method described in Chapter 3 for the Implicit Euler method was implemented in MATLAB. Below is the code that calls MATLAB's built-in `bvp4c` routine. The inputs for the `ie_uac` are the aerosol and sheath volumetric flow rates, `Q_a` and `Q_sh` in LPM, `type` denotes whether the particle-phase or gas-phase PDE is to be solved, and `Peclet` is the Pe of the particle or gas. The function outputs four parameters: `char` – the particle penetration or gas dilution factor, `nu` – the dimensionless concentration of particles or gas, and the grid coordinates (`zeta` and `rho`). Examples of the function call are included in the code's header.

`ie_uac` calls the functions `MODEL_ODEfun`, which evaulates the ordinary differential equation, and `MODEL_bcfun` or `MODEL_bcfun_par` which are functions that set the boundary conditions. The MATLAB code for these files are shown after `ie_uac`.

```
function [char, nu, zeta, rho] = ...
    ie_uac(type, Q_a, Q_sh, Peclet)

% This m-file is the Euler Implicit-BVP4C implemintation of the
% diffusion model for the Universal Aerosol Conditioner.
% April 2017

%--INPUT----------------------------------------------------------------
% type   = 'par' or 'gas' (quotes are necessary) to distinguish
%          which component concentration is to be solved
% Q_a    = the aerosol flowrate in LPM
% Q_sh   = the sheath flowrate in LPM
% Peclet = Peclet number
%          Definition of Peclet in UAC:
%          Pe = 2*(Q_a + Q_sh)/(pi * D * L) where D is diffusion
%                                                          coeff.

%--OUTPUT---------------------------------------------------------------
% zeta = non-dimensional length
% rho  = non-dimensional radius
% nu   = non-dimensional concentration
% characterisitic is the return handle for penetration or dilution ratio

%--Example--------------------------------------------------------------
% [penetration, nu, zeta, rho] = ie_uac('par', 0.3, 3, 1E4);
% [dil_factor, nu, zeta, rho] = ie_uac('par', 0.3, 3, 3)
% Gas Peclet numbers are orders of magnitude smaller than particles

% the initial number of steps in zeta and rho
step = 100;                     %step scaling factor
n = 2*step;                     %total number of steps in rho
```

```
Q_total = Q_a + Q_sh;

% Generate mesh in r-direction between [a,b] where rho(a)=0 and rho(b)=1
rho = zeros(n,1);                          %(column)
h = 1/(n-1);                               %step size in rho (linear)
for i = 2:n-1
    rho(i) = (i-1)*h;
end
rho(n) = 1;
rho(1) = 1E-6;               % Can't have rho(1) = 0 otherwise I get a
                             % divide by zero error in the ODEfun
% Generate mesh in z-direction between 0 and 1
zeta = logspace(-4,0,step);                %logrithmic spacing (row)
zeta(2:length(zeta)+1) = zeta;             %push vector over 1 space
zeta(1) = 0;                               %set the first zeta step

% Discretize Initial Condition (initial concentraction) into a vector
% Might be possible to do this using a step function for BVP4C
rho_a = sqrt(1-sqrt(1-Q_a/Q_total));  %aerosol core radius
Rad1 = ceil(n*rho_a);          %Index Position for aerosol core radius

%calculations done on half domain. aerosol concentration starts at r=0
%create vector that simulates the step function for the initial
%concentration profile
nu_zero1 = ones(Rad1,1);                   %(column)
nu_zero2 = zeros(floor(n*(1-rho_a)),1);    %(column)
nu_zero = [nu_zero1; nu_zero2];            %(column)

% Initialize concentration variables
nu = zeros(n,length(zeta));                %initialize nu (matrix)
nu(:,1) = nu_zero;                         %set initial conc. in matrix
```

```
%get interpolated polynomial of the IC for BVP4C calculation
nu_old = interp1(rho,nu_zero,'pchip','pp');

% Setup initial guess for BVP4C
nu_int = [-10 -10]; %[-10 -1000]
ini_sol = bvpinit(rho,nu_int);
options = bvpset('RelTol',1e-3,'Nmax',5e5);
initial_run = 1;  %tell the bvp4c solver to use the interpolated poly
                  %for the first run of the loop


% Solve the equations for each zeta step (zeta @ j=1 is known already)
for j = 2:length(zeta)
    k = zeta(j) - zeta(j-1);      %calculated the zeta step size

    % Build system of equations to solve
    % Functions MODEL_ODEfun, MODEL_bcfun, and MODEL_bcfun_par
    % are stored in separate m-files
    if strcmp(type,'par') == 1
        sol = bvp4c(...
        @(r,n_new)MODEL_ODEfun(r,n_new,nu_old,k,Peclet,initial_run),...
        @MODEL_bcfun_par, ini_sol,options);
    else
        sol = bvp4c(...
        @(r,n_new)MODEL_ODEfun(r,n_new,nu_old,k,Peclet,initial_run),...
        @MODEL_bcfun, ini_sol,options);
    end

    Temp = deval(sol,rho);   %calculate actual conc. values
    nu(:,j) = Temp(1,:);     %add nu @ current zeta to matrix
    nu_old = sol;            %pass current sol as prev_sol
    initial_run = 0;
```

```
end


if strcmp(type,'par') == 1

    outlet_par = trapz((1-rho(1:Rad1).^2).*...
        nu(1:Rad1,step+1).*rho(1:Rad1));

    inlet_par = trapz((1-rho(1:Rad1).^2).*...
        nu(1:Rad1,1).*rho(1:Rad1));

    penetration = outlet_par./inlet_par;
    char = penetration;
else
    outlet_gas = trapz((1-rho(1:Rad1).^2).*...
        nu(1:Rad1,step+1).*rho(1:Rad1));

    inlet_gas = trapz((1-rho(1:Rad1).^2).*...
        nu(1:Rad1,1).*rho(1:Rad1));

    dilution = (inlet_gas./outlet_gas);
    char = dilution;

end
```

```
function dndr = MODEL_ODEfun(r,n_new,prev_sol,h,Pe,run)
%n_new = nu @ j+1
%n_old = nu @ j
%Let n_new(1) denote nu (non-dim concentration)
%    n_new(2) denote d(nu)/dr @j+1
```

```
%    r is the non-dim radial coordinate
%    Pe is the Peclet number
%    Set run to 1 to use nu_zero. Set to 0 for subsequent
%    runs in the the loop
%System of Equations
if run == 1
    n_old = ppval(prev_sol,r);
elseif run == 0
    n_old = deval(prev_sol,r);
    n_old = n_old(1,:);
end
dndr = [ n_new(2);
    Pe*(1-r^2)*(n_new(1)-n_old)/h - 1/r*n_new(2)];
```

```
function res = MODEL_bcfun(Ra,Rb)
%n_new = nu @ j+1
%n_old = nu @ j
%Let n_new(1) denote nu (non-dim concentration)
%    n_new(2) denote d(nu)/dr @j+1
%    Boundary conditions are dependent on r/rho, hence denoted by R
%    Ra(1) is the left hand (symmetry) BC for nu
%    Rb(1) is the right hand BC for nu
%    Ra(2) is the left hand BC for flux, d(nu)/dr
%    Rb(2) is the right hand BC for flux
res = [
    Ra(2);
    Rb(2)
    ];
```

```
function res = MODEL_bcfun_par(Ra,Rb)
%n_new = nu @ j+1
%n_old = nu @ j
%Let n_new(1) denote nu (non-dim concentration)
%    n_new(2) denote d(nu)/dr @j+1
%    Boundary conditions are dependent on r/rho, hence denoted by R
%    Ra(1) is the left hand (symmetry) BC for nu
%    Rb(1) is the right hand BC for nu
%    Ra(2) is the left hand BC for flux, d(nu)/dr
%    Rb(2) is the right hand BC for flux
res = [
    Ra(2);
    Rb(1)
    ];
```

Appendix D

Controlling the UAC with an Arduino Microcontroller

To control the UAC, an Arduino Uno was programmed to act as the device's control system. The Arduino is responsible for controlling the UAC's blowers and interfacing with the pressure sensors and thermocouple. To control the blowers, the Arduino interfaces with two 10-bit digital potentiometers, over the Serial Peripheral Interface (SPI) protocol, which controls the voltages sent to each blower. The absolute pressure sensor (Model SSCSRNN1.6BA7A3, Honeywell Sensing) and Honeywell Zephyr mass airflow sensors (Model HAFUHT0020L3AXT) communicate with the Arduino over the Inter-Integrated Circuit (I^2C) bus. The pressure sensor (Model HSCDRRN002NG2A3, Honeywell Sensing) used in the laminar flow meter and the thermocouple's analog-to-digital (ADC) integrated circuit (MAX31855) communicate over SPI. The PID control loops are initiated using the open source Arduino library provided by Brett Beauregard.

The sheath and exhaust flow rates are measured by the Zephyrs. To obtain valid results in the gas-only dilution tests, it was necessary to ensure that the Q_{a} and Q_{s} were the same. This was accomplished by setting the aerosol and exhaust flow rates as the control variables for the sheath and exhaust flows, respectively. Using

the aerosol flow rate as the sheath control variable ensured that the sheath flow was equal to the exhaust flow since when Q_{a} is equal to Q_{s}, Q_{sh} must be equal to Q_{exh}. When Q_{a} is larger than the setpoint, the sheath blower will increase the sheath flow rate which will decrease Q_{a}. When Q_{a} is lower than desired, the sheath flow rate will be decreased. This control scheme avoids any potential problems caused by the inaccuracies between the flow sensors. The code used to program the Arduino is found below.

```
/*********************************************************************************
Universal Aerosol Conditioner Sheath Flow Control

   Kerry Chen
   April 29, 2016

   Updated for new voltage range on digital pot. on May 10, 2016 (PID parameters)
   Updated on May 17, 2016 for communication w/ UAC GUI
   Updated on June 2, 2016 for incorporating temperature reading (analog)
   Updated on June 10, 2016 for SPI Thermocouple and rewrote pressure and zephyr functions
   Updated on September 2, 2016
   -removed heater
   -multiplexor chip added
   Credits to:
   -Brett Beauregard for Arduino PID Library
   -Adafruit team / Adafruit forum members for the MAX31855K Library


   Uses MAX548* Digital Potentiometer to vary the Ametek blowers' speeds
   Communicate with the digital pot. over SPI
   The MAX548* chip has 10-bits "steps" worth of control
   Honeywell Zephyr mass flow sensors are used to control the Ametek blowers setpoint
   Zephyrs communicate over I2C
   PID control used to maintain desired setpoint
   NV = nonvolatile memory
```

```
   To use the MAX31855K, install the Adafruit_MAX31855K Library fork:
   https://github.com/wmacevoy/Adafruit-MAX31855-library


   The above library uses the base library built by Adafruit but incorporates the
   non-linearity compensation using the NIST tables. Original fork code written by
   jh421797 and incorporated into the forked library by wmacevoy on the Adafruit forums.
   You can still call the original functions written by Adafruit with this library.

   To do list:
   - Temperature PID control for the process heater
   - Temperature PID control for the heating tape


  May 3, 2017
  Changed GUI interface to work with Megunolink instead of Processing

******************************************************************************/
#include <ArduinoTimer.h>
#include <CircularBuffer.h>
#include <CommandHandler.h>
#include <CommandProcessor.h>
#include <DataStore.h>
#include <DeviceAddress.h>
#include <EEPROMStore.h>
#include <Filter.h>
#include <MegunoLink.h>
#include <MessageHeaders.h>

#include <SPI.h>
#include <Wire.h>
#include <PID_v1.h>
#include <math.h>
#include <Adafruit_MAX31855.h>
#include <Adafruit_KHighResMAX31855.h>
#include <RunningAverage.h>

#define TCAADDR 0x70  // I2C address for TCA multiplexor (address is adjustable)
#define Zephyr_3XT 0x39 // I2C address for 3XT Zephyr
#define Zephyr_4XT 0x49 // I2C address for 4XT Zephyr
```

```
#define p_sensor 0x78  // I2C address for Absolute Pressure Transducer
#define p_sensor2 0x28  // I2C address for Vented Pressure Transducer

CommandHandler<15,50> SerialCommandHandler;
InterfacePanel Panel("Test");

TimePlot SheathPIDPlot("SheathPlot"), ExhaustPIDPlot("ExhaustPlot");

const int CS_thermo = 7; // Chip select pin (SS) for the thermocouple using the MAX31855K
    breakout board
const int SPI_Pin = 8; // connect Arduino Pin 8 to Pin 6 on the MAX548* to set SPI Mode on
     (pin 4/9 on dsub)
const int CS1 = 9;     // connect Arduino Pin 9 (SS) to Pin 3 on the MAX548* (pin 1 on
    dsub)
const int CS2 = 10;    // connect Arduino Pin 10 (SS) to Pin 3 on the MAX548* (pin 6 on
    dsub)
// connect Arduino Pin 11 or pin 4 on ICSP (MOSI) to Pin 5 on the MAX548*  (pin 3/8 on
    dsub)
// connect Arduino Pin 13 or pin 3 on ICSP (SCK) to Pin 4 on the MAX548* (pin 2/7 on dsub)
// connect Arduino Pin 12 or pin 1 on ICSP (MISO) to DO pin on MAX31855K breakout
// connect ground to dsub pin 5

boolean set_nvmem = false;   // triggers write to NV if true, copy from NV if false. See
    void setup

double Setpoint_intake, Input_intake, Output_intake;  // Define variables for PID
double Setpoint_exhaust, Input_exhaust, Output_exhaust;
double Sheath_pressure, Sheath_temperature;       // variable to store pressure/temp
    reading
double Sample_flow, Sheath_flow, Exhaust_flow;

unsigned int Output_intake_int = 0;
unsigned int Output_exhaust_int = 0;
unsigned long serialTime; //this will help us know when to talk with megunolink

// Specify the links and initial tuning parameters
// PID syntax: PID(&Input, &Output, &Setpoint, Kp, Ki, Kd, Direction)
float Kp_i = 150;  //old 10
float Ki_i = 150; //old 200
float Kd_i = 0;
```

```
float Kp_e = 10;
float Ki_e = 30;
float Kd_e = 0;

int min_sheath = 0;
int min_exhaust = 0;
int max_sheath = 1023;
int max_exhaust = 1023;

PID PID_intake(&Input_intake, &Output_intake, &Setpoint_intake, Kp_i, Ki_i, Kd_i, REVERSE)
    ;
PID PID_exhaust(&Input_exhaust, &Output_exhaust, &Setpoint_exhaust, Kp_e, Ki_e, Kd_e,
    DIRECT);

// Create thermocouple class with the Adafruit library
Adafruit_KHighResMAX31855 process_thermocouple(CS_thermo);

RunningAverage RA_sample(10);
RunningAverage RA_sheath(10);
RunningAverage RA_exhaust(10);

void setup() {
  SPI.begin();                 // Initialize SPI Interface
  SPI.setBitOrder(MSBFIRST);   // Set MSBFIRST according to MAX548* Data Sheet

  pinMode(CS1, OUTPUT);
  digitalWrite(CS1, HIGH);     // Set pin high to set slave to idle

  pinMode(CS2, OUTPUT);
  digitalWrite(CS2, HIGH);     // Set pin high to set slave to idle

  pinMode(SPI_Pin, OUTPUT);
  digitalWrite(SPI_Pin, HIGH); // Set pin high to enable SPI interface

  Wire.begin();                // Join I2C bus
  Serial.begin(19200);          // Communication rate (9.6 kHz)

  /*
    This first if block is to write to the NV memory what starting speed you want the
```

```
        blowers to run at
    i.e. Here I call digitalPotWrite twice, one for each blower
    I call the function which tells the pot I want the blowers running at 0 steps
    The pot writes this 0 value to its "wiper". The wiper stores the current position in
        volatile memory
    After that's established, call digitalPotSetMemory which tells the pot. to write the
        current
    wiper position into NV memory. This block of code should only be run once to preserve
        the write cycle
    of the NV.
  */
    digitalPotWrite(CS1, 0);   // Set wiper position to 0
    digitalPotWrite(CS2, 0);
//  Changed the startup setting to just have the blowers run at min. speed.
//  if (set_nvmem == true) {
//    digitalPotWrite(CS1, 0);   // Set wiper position to 0
//    digitalPotWrite(CS2, 0);
//    digitalPotSetMemory(CS1);  // Save wiper position to NV
//    delay(12);                 // Digital Pot needs 12 ms to complete write operation
//    digitalPotSetMemory(CS2);
//    delay(12);
//  }
// Now that the initial starting voltage/motor speed is established, this if block just
    copies the
// set motor speed/pot step from the NV memory to the wiper

//  if (set_nvmem == false) {
//    digitalPotNVtoWiper(CS1);  // Copy NV to wiper
//    digitalPotNVtoWiper(CS2);
//  }
  delay(2000);  // Delay 2 seconds to allow everything to power up properly

  ////////////////////////////////////PID SETUP
      ////////////////////////////////////////

  Setpoint_intake = 0.32;         // Setpoint in LPM
  Setpoint_exhaust = 3;

  PID_intake.SetOutputLimits(min_sheath, max_sheath); // Expand the range to 10-bits
  PID_exhaust.SetOutputLimits(min_exhaust, max_exhaust); // Expand the range to 10-bits
```

```
  PID_intake.SetMode(AUTOMATIC);    // Turns on PID control
  PID_exhaust.SetMode(AUTOMATIC);
  ////////////////////////////////////END OF PID SETUP
      ////////////////////////////////////

//  SerialCommandHandler.AddVariable(F("Setpoint_Intake"), Setpoint_intake);
//  SerialCommandHandler.AddVariable(F("Setpoint_Exhaust"), Setpoint_exhaust);
//  SerialCommandHandler.AddVariable(F("Sheath_ki"), Ki_i);
//  SerialCommandHandler.AddVariable(F("Sheath_kp"), Kp_i);
//  SerialCommandHandler.AddVariable(F("Sheath_kd"), Kd_i);
//  SerialCommandHandler.AddVariable(F("Exhaust_ki"), Ki_e);
//  SerialCommandHandler.AddVariable(F("Exhaust_kp"), Kp_e);
//  SerialCommandHandler.AddVariable(F("Exhaust_kd"), Kd_e);
//  SerialCommandHandler.AddVariable(F("Min_step_sh"), min_sheath);
//  SerialCommandHandler.AddVariable(F("Min_step_ex"), min_exhaust);
//  SerialCommandHandler.AddVariable(F("Max_step_sh"), max_sheath);
//  SerialCommandHandler.AddVariable(F("Max_step_ex"), max_exhaust);

  SerialCommandHandler.AddCommand(F("setSPs"), cmd_setSPs);
  SerialCommandHandler.AddCommand(F("setPIDLimits"), cmd_setPIDLimits);
  SerialCommandHandler.AddCommand(F("setPIDGains_i"), cmd_setPIDGains_i);
  SerialCommandHandler.AddCommand(F("setPIDGains_e"), cmd_setPIDGains_e);

//  SheathPIDPlot.SetTitle(F("Sheath Flow Rate"));
  SheathPIDPlot.SetTitle(F("Sample Flow Rate"));
  SheathPIDPlot.SetXlabel(F("Time"));
  SheathPIDPlot.SetYlabel(F("Flow Rate (LPM)"));

  ExhaustPIDPlot.SetTitle(F("Exhaust Flow Rate"));
  ExhaustPIDPlot.SetXlabel(F("Time"));
  ExhaustPIDPlot.SetYlabel(F("Flow Rate (LPM)"));

  RA_sample.clear(); // explicitly start clean
  RA_sheath.clear();
  RA_exhaust.clear(); // explicitly start clean
}

void loop() {
```

```
SerialCommandHandler.Process();

Sheath_pressure = pressure_read();
Sheath_temperature = process_thermocouple.readCelsius();
Sample_flow = aerosol_inlet_read();
RA_sample.addValue(Sample_flow);

//-PID Start----------------------------------------------------------------------------

tcaselect  //select port 0 on multiplexer (zephyrs share the same hardware i2c address)
RA_sheath.addValue(zephyr_read(Zephyr_3XT, 's'));
Input_intake = RA_sample.getAverage();            //use aerosol flow rate as control
    variable
PID_intake.Compute();
Output_intake_int = (int)round(Output_intake);
digitalPotWrite(CS2, Output_intake_int);          // Set the value only once between 0 -
     1023

tcaselect(1);
RA_exhaust.addValue(zephyr_read(Zephyr_3XT, 'e'));
Input_exhaust = RA_exhaust.getAverage();
PID_exhaust.Compute();
Output_exhaust_int = (int)round(Output_exhaust);
digitalPotWrite(CS1, Output_exhaust_int);         // Full scale: 0 - low speed, 1023 -
    high speed
//  digitalPotWrite(CS2, 0);

//-PID End------------------------------------------------------------------------------

// Check if it's time to communicate with Megunolink
if (millis() > serialTime)
{
  SerialSend_Meguno();
  serialTime += 250;  //500 originally
  PID_intake.SetTunings(Kp_i, Ki_i, Kd_i);
  PID_exhaust.SetTunings(Kp_e, Ki_e, Kd_e);
  PID_intake.SetOutputLimits(min_sheath, max_sheath); // Expand the range to 10-bits
  PID_exhaust.SetOutputLimits(min_exhaust, max_exhaust); // Expand the range to 10-bits
}
```

```
}
//End of loop


// Save previous wiper position to non-volatile memory
void digitalPotSetMemory(int csPin) {
  digitalWrite(csPin, LOW);      // select slave
  byte command = B00100000;     // Command + Address bits: 00100000 to write = 20 in HEX.
  SPI.transfer(command);
  digitalWrite(csPin, HIGH);    // de-select slave
}

// Copy wiper position saved in NV memory to current wiper position
void digitalPotNVtoWiper(int csPin) {
  digitalWrite(csPin, LOW);     // select slave
  byte command = B00110000;     // Command + Address bits: 00110000 to copy = 30 in HEX.
  SPI.transfer(command);
  digitalWrite(csPin, HIGH);    // de-select slave
}


/*
  This function is responsible for setting the voltage for the blowers. It takes two
      inputs:
  the pin the motor is connected to and the desired step.
  The # of steps ranges from 0 - 1023 (10 bits)
  where 0 is the low end of your voltage range and 1023 is the max voltage.
  The MAX548* accepts a 3 byte input. The first byte is the command byte which tells the
  chip what command you want to envoke (read from and write to NV or set new wiper
      position)
  The next two bytes are used to store your desired wiper position, i.e. an integer value.
*/
void digitalPotWrite(int csPin, int value) {
  digitalWrite(csPin, LOW);           // select slave
  byte command = 0x0;                 // Command + Address bits: 00000000
  value = (value << 6);               // Only 2 bits are used in the Data byte 1 since pot
      . is 10-bit
  // MSBFIRST so shift trailing 10 bits to the front
  // value is the step number/wiper position you want
  byte byte0 = highByte(value); // Data Byte 0, makes a byte out of first 8 bits of value
```

```
  byte byte1 = lowByte(value);  // Data Byte 1, takes trailing 8 bits (only first 2 bits
      of lowbyte read)

  // Send complete instruction (24-bits to digital pot.)
  // Arduino SPI library only allows 1 byte to be transfered at a time
  // Send 3 separate bytes
  SPI.transfer(command);
  SPI.transfer(byte0);
  SPI.transfer(byte1);
  digitalWrite(csPin, HIGH);          // de-select slave
}

// This is the function to control which output to use on the Multiplexer.
void tcaselect(uint8_t i) {
  if (i > 7) return;

  Wire.beginTransmission(TCAADDR);
  Wire.write(1 << i);
  Wire.endTransmission();
}

float zephyr_read(int address, char location)  //need to delay it by 1 ms to get fresh
    data
{
  int receive;
  byte b0, b1, status_bits;
  Wire.requestFrom(address, 2);     // request 2 bytes from slave device #112 <-73
  if (2 <= Wire.available())          // if two bytes were received
  {
    b0 = Wire.read();           // Read High Byte
    b1 = Wire.read();           // Read Low Byte
    status_bits = b0 >> 6;      // First two bits of the High Byte are to
                                // indicate the status
    // 00 = normal, 11 = Diagnostic error (see I2C Comm. pdf)
    switch (status_bits) {
      case 0:
        // Combine b0 and b1 to form int
        receive = (((unsigned int)(b0 & 0x3f) << 8) | b1);
        break;
```

```
      case 3:
        Wire.endTransmission();
        Serial.print("Status Error");
        break;
    }

    float flow = 20.0 * (receive / 16384.0 - 0.1) / 0.8; // transfer function (in SLPM)

    // Pressure = Density * R_specific * Temperature. R_specific can be omitted
    // since it is cancelled out when the flow rate is adjusted from standard to
    // operating conditions.
    // Omitting R_specific also allows us to use different sheath gases without
    // having to modify the code but a correction factor must be applied according
    // to Honeywell's guidelines for the zephyrs

    /*  Since this is a multipurpose function for the zephyrs, I need to select the
        correct calibration curve. As both zephyrs used here have the same hardware
            address,
        each zephyr will be assigned a char. Compare char values where "s" denotes the
        sheath location and "e" denotes the exhaust to get the correct calibrated value
        (where the exhaust is considered the true value).
    */
    switch (location) {
      case 's':  // sheath flow meter
        {
  flow = -0.0000609858*pow(flow,3) + 0.0013232697*flow*flow + 0.9854126865*flow -
      0.201397499;

          break;
        }
      case 'e': // exhaust flow meter
        {
        // this zephyr reads flow rates very close to bubble flow meter
          break;
        }
    }
    float flow_lpm = flow * 101.325 / 273.15 * (Sheath_temperature + 273.15) / (
        Sheath_pressure * 100);
    return flow_lpm;
//    return flow;  //uncomment for SLPM reading
```

```
  }
  delay(1);
}

float pressure_read()
{
  float pressure = 0;
  int pressure_bytes;
  byte b0, b1, status_bits;
  Wire.requestFrom(p_sensor, 2);            // Sends content of first two registers
  if (2 <= Wire.available()) {              // if two bytes were received
    b0 = Wire.read();                       // Read High Byte
    b1 = Wire.read();                       // Read Low Byte
    status_bits = b0 >> 6;                  // First two bits of the High Byte are
                                            // to indicate the status
    // 00 = normal, 01 = Command Mode (shouldn't see this one)
    // 10 = Stale data,  11 = Diagnostic error (see I2C Comm. pdf)
    switch (status_bits) {
      case 0:
      // Combine b0 and b1 to form int
        pressure_bytes = (((unsigned int)(b0 & 0x3f) << 8) | b1);
      // Pressure transfer function (in bar)
        pressure = ((pressure_bytes - 1638) * 1.6) / (13107);
        break;

      case 2:
        delay(2);
        // If stale data was received (read requests faster than 1 ms)
        // Delay by 2 ms and try reading again
        Wire.requestFrom(p_sensor, 2);
        if (2 <= Wire.available()) {
          b0 = Wire.read();
          b1 = Wire.read();
          pressure_bytes = (((unsigned int)(b0 & 0x3f) << 8) | b1);
          pressure = ((pressure_bytes - 1638) * 1.6) / (13107); // pressure output in bar
        }
        break;

      case 1:
        Wire.endTransmission();
```

```
        Serial.print("Command Mode Error");
        break;

      case 3:
        Wire.endTransmission();
        Serial.print("Diagnostic Error");
        break;
    }
  }
  return pressure;
}

float aerosol_inlet_read()
{
  float pressure;
  int pressure_bytes;
  byte b0, b1, status_bits;
  Wire.requestFrom(p_sensor2, 2);  //Sends content of first two registers
  if (2 <= Wire.available()) {         // if two bytes were received
    b0 = Wire.read();
    b1 = Wire.read();
    status_bits = b0 >> 6;

    switch (status_bits) {
      case 0:
        pressure_bytes = (((unsigned int)(b0 & 0x3f) << 8) | b1);
        pressure = ((pressure_bytes - 1638.4) * 2) / (13107.2); // pressure output in
            inH2O
        pressure = pressure * 248.84; // pressure output in Pa
        //        Serial.print("Pressure: ");
        //        Serial.println(pressure_bytes,DEC);
        break;

      case 2:
        // Case 2 is when stale data is read. This occurs when we read from the sensor
        // too quickly (i.e before the old reading is flushed from system)
        delay(2);
        Wire.requestFrom(p_sensor, 2);  //Sends content of first two registers
        if (2 <= Wire.available()) {         // if two bytes were received
          b0 = Wire.read();
```

```
          b1 = Wire.read();
          pressure_bytes = (((unsigned int)(b0 & 0x3f) << 8) | b1);
          pressure = ((pressure_bytes - 1638.4) * 2) / (13107.2); // pressure output in
              inH2O
          pressure = pressure * 248.84; // pressure output in Pa
        }
        break;

      case 1:
        Wire.endTransmission();
        Serial.print("CommandModeError");
        break;

      case 3:
        Wire.endTransmission();
        Serial.print("DiagnosticError");
        break;
    }
  }
  float diameter = (0.055 * 25.4 / 1000);
  float viscosity = 1.81E-5;
  float tube_length = 0.10;
  float flow = PI * pow(diameter, 4) * pressure / 128 / viscosity / tube_length * 60000;
  flow = -0.32113 * pow(flow, 4) + 1.28826 * pow(flow, 3) - 1.94425 * flow * flow + 2.4234
       * flow - 0.01687;
  return flow;
}

void SerialSend_Meguno()
{
//  Panel.SetNumber("Sheath_flow", (float)Input_intake);
//  Panel.SetNumber("Sheath_flow", (float)Sheath_flow);
  Panel.SetNumber("Sheath_flow", (float)RA_sheath.getAverage());
  Panel.SetNumber("Exhaust_flow", (float)Input_exhaust);
//  Panel.SetNumber("Exhaust_flow", (float)RA_exhaust.getAverage());
//  Panel.SetNumber("Sample_flow", (float)Sample_flow);
  Panel.SetNumber("Sample_flow", (float)RA_sample.getAverage());
  Panel.SetNumber("Sheath_pressure", (float)Sheath_pressure);
  Panel.SetNumber("Temperature", (float)Sheath_temperature);
  Panel.SetNumber("Intake_PID", Output_intake_int);
```

```
  Panel.SetNumber("Exhaust_PID", Output_exhaust_int);
  Panel.SetNumber("Current_sheath", (float)Setpoint_intake);
  Panel.SetNumber("Current_exhaust", (float)Setpoint_exhaust);

  Panel.SetNumber("Current_kp_i", (float)Kp_i);
  Panel.SetNumber("Current_ki_i", (float)Ki_i);
  Panel.SetNumber("Current_kd_i", (float)Kd_i);
  Panel.SetNumber("Current_kp_e", (float)Kp_e);
  Panel.SetNumber("Current_ki_e", (float)Ki_e);
  Panel.SetNumber("Current_kd_e", (float)Kd_e);
  Panel.SetNumber("Min_step_sh_cur", min_sheath);
  Panel.SetNumber("Min_step_ex_cur", min_exhaust);
  Panel.SetNumber("Max_step_sh_cur", max_sheath);
  Panel.SetNumber("Max_step_ex_cur", max_exhaust);

  SheathPIDPlot.SendData("SheathSet", (float)Setpoint_intake);
  SheathPIDPlot.SendData("SheathCurrent", (float)Input_intake);

  ExhaustPIDPlot.SendData("ExhaustSet", (float)Setpoint_exhaust);
  ExhaustPIDPlot.SendData("ExhaustCurrent", (float)Input_exhaust);
}

void cmd_setPIDLimits(CommandParameter &Parameters)
{
  min_sheath = Parameters.NextParameterAsInteger();
  max_sheath = Parameters.NextParameterAsInteger();
  min_exhaust = Parameters.NextParameterAsInteger();
  max_exhaust = Parameters.NextParameterAsInteger();
}

void cmd_setSPs(CommandParameter &Parameters)
{
  Setpoint_intake = Parameters.NextParameterAsDouble();
  Setpoint_exhaust = Parameters.NextParameterAsDouble();
}

void cmd_setPIDGains_i(CommandParameter &Parameters)
{
  Ki_i = Parameters.NextParameterAsDouble();
  Kp_i = Parameters.NextParameterAsDouble();
```

```
  Kd_i = Parameters.NextParameterAsDouble();
}

void cmd_setPIDGains_e(CommandParameter &Parameters)
{
  Ki_e = Parameters.NextParameterAsDouble();
  Kp_e = Parameters.NextParameterAsDouble();
  Kd_e = Parameters.NextParameterAsDouble();
}
```

To control the UAC in real-time, a graphical user interface (GUI) was designed using the MegunoLink Pro software. This software allows the user to change variables used in the Arduino in real-time. The GUI is shown in Figure D.1. The GUI reports several parameters: the current sample (aerosol), sheath, and exhaust volumetric flow rates; the absolute pressure; sheath air temperature; and the setpoints for the sample and exhaust flow rates. For debugging purposes, the GUI also shows the current control effort for each blower. Real-time plots of the sample and sheath flow rates are plotted below the parameter control boxes. These plots also show the current setpoint for each flow. On the right side of the GUI one will find the control boxes for the PID gains. The GUI allows the user to change the PID gains in real-time. By using these control boxes and the flow rate plots, the user can tune the PID in real-time without having to reprogram the flash memory on the Arduino. These control boxes also allow the user to manually control the blowers if desired. Note that any changed settings are not saved on the Arduino when the UAC is powered off.

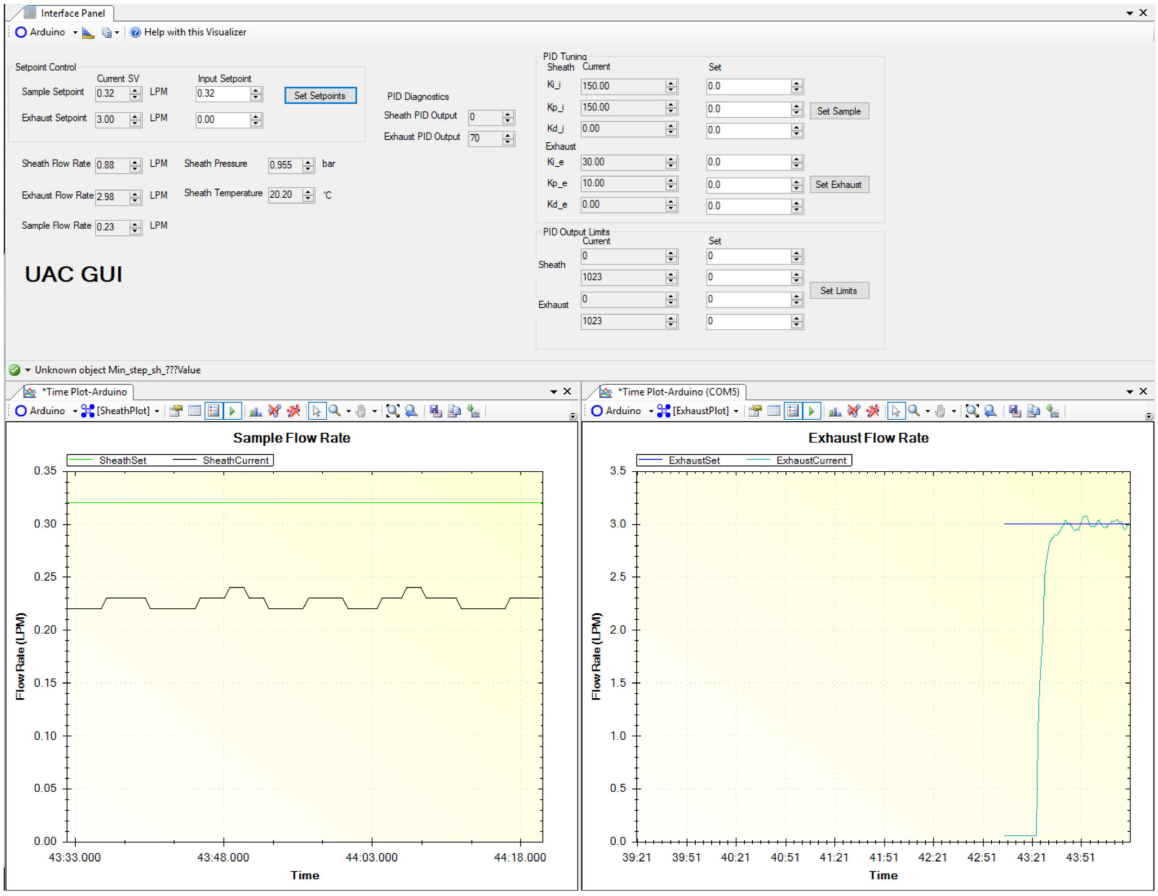


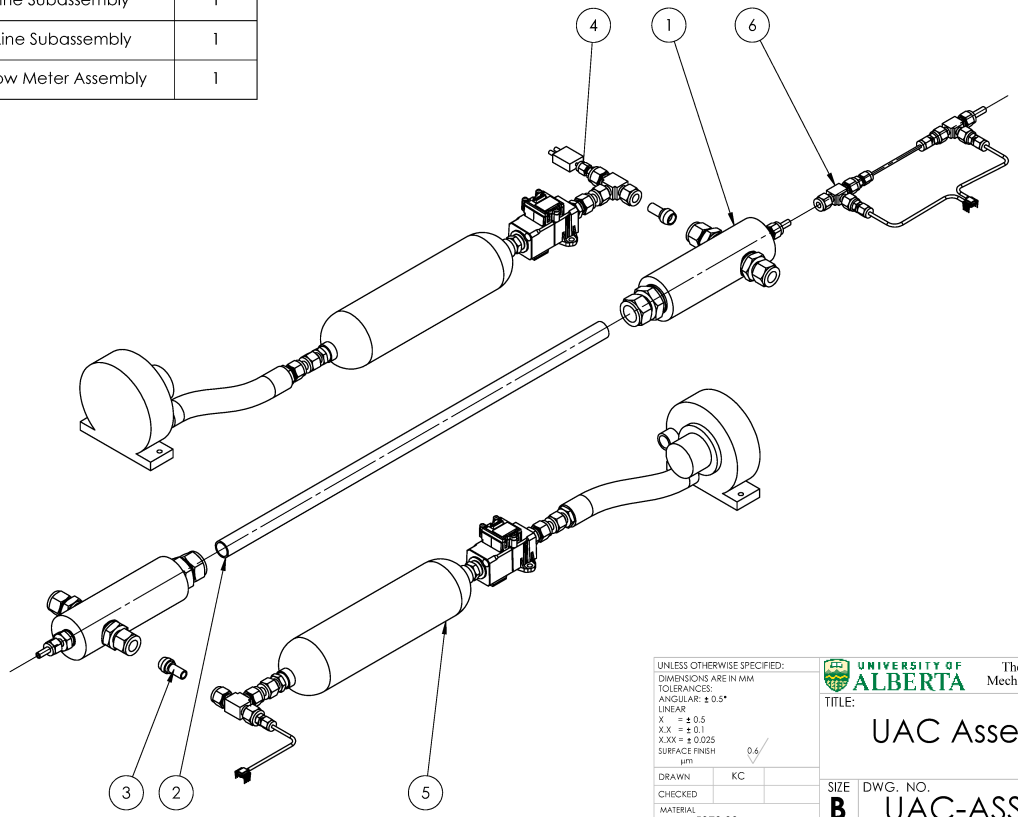
Figure D.1: The UAC's Graphical User Interface

Appendix E

Drawing Package for the UAC

Engineering drawings for the UAC are presented in this Appendix. The drawings are sorted by top level down. Individual part drawings are shown in their respective subassembly.

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Manifold Subassembly for Schematic	Manifold Assembly	2
2	0.75 in Stub	Dilution Tube (0.75" OD 0.065" WT)	1
3	SS-811-PC-6	1/2" to 1/4" OD Swagelok Reducing Port Connector	2
4	Sheath Line Subassembly	Sheath Line Subassembly	1
5	Exhaust Line Subassembly	Exhaust Line Subassembly	1
6	Laminar Flow Meter Assembly	Laminar Flow Meter Assembly	1



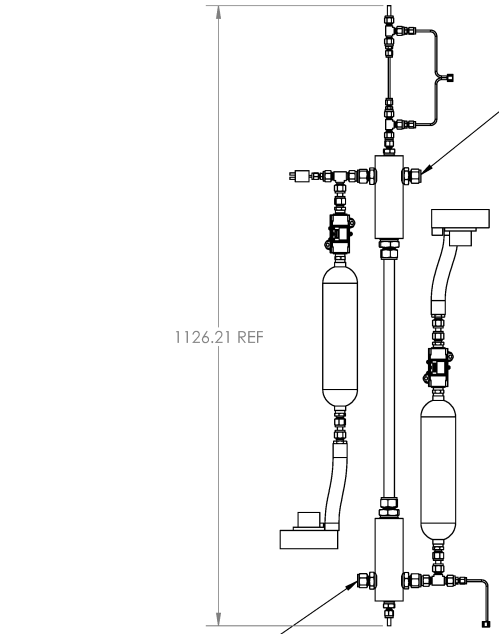
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH 0.6 µm
DRAWN KC
CHECKED
MATERIAL 5878.08
DO NOT SCALE DRAWING

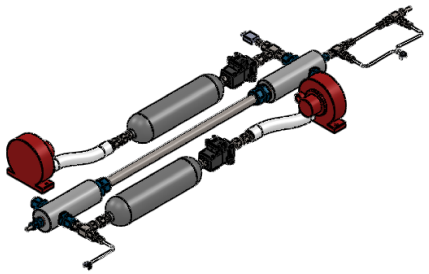
UNIVERSITY OF ALBERTA — The Department of Mechanical Engineering

TITLE: UAC Assembly

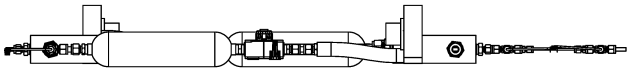
SIZE B | DWG. NO. UAC-ASSEM | REV 1

SCALE: 1:4 | MASS: 5878.08 | SHEET 1 OF 2





Note: For experimental testing, the sheath and exhaust flows were split into two streams and connected to each 1/2" Swagelok fitting on the manifolds. The laminar flow meter was also connected to the UAC with a 180 degree bend to keep the test setup as compact as possible.



UNLESS OTHERWISE SPECIFIED:	
DIMENSIONS ARE IN MM	
TOLERANCES:	
ANGULAR: ± 0.5°	
LINEAR	
X = ± 0.5	
X.X = ± 0.1	
X.XX = ± 0.025	
SURFACE FINISH 0.6 µm	
DRAWN	KC
CHECKED	
MATERIAL 5878.08	
DO NOT SCALE DRAWING	

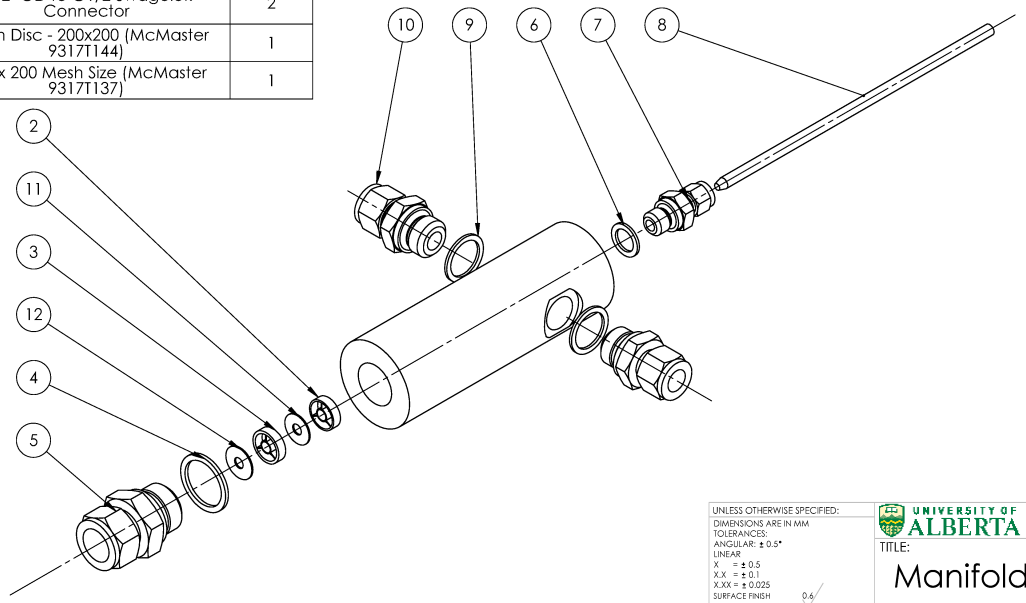
UNIVERSITY OF ALBERTA — The Department of Mechanical Engineering

TITLE: UAC Assembly

SIZE	DWG. NO.	REV
B	UAC-ASSEM	1

SCALE: 1:7 | WEIGHT: | SHEET 2 OF 2

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Manifold	Manifold	1
2	Alignment Plate	Alignment Plate (Small)	1
3	Alignment Plate (large)	Alignment Plate (Large)	1
4	CU-12-RP-2	3/4" Copper Gasket	1
5	SS-1210-1-12RP	Bored Through	1
6	CU-4-RP-2	1/4" Copper Gasket	1
7	SS-400-1-4RP	Bored Through	1
8	Sample Tube	Sample Tube (1/4" OD 0.049" W.T.)	1
9	CU-8-RP-2	1/2" Copper Gasket	2
10	SS-810-1-8RP	1/2" OD to G1/2 Swagelok Connector	2
11	Mesh Disc (Small)	Mesh Disc - 200x200 (McMaster 9317T144)	1
12	Mesh Disc	200 x 200 Mesh Size (McMaster 9317T137)	1



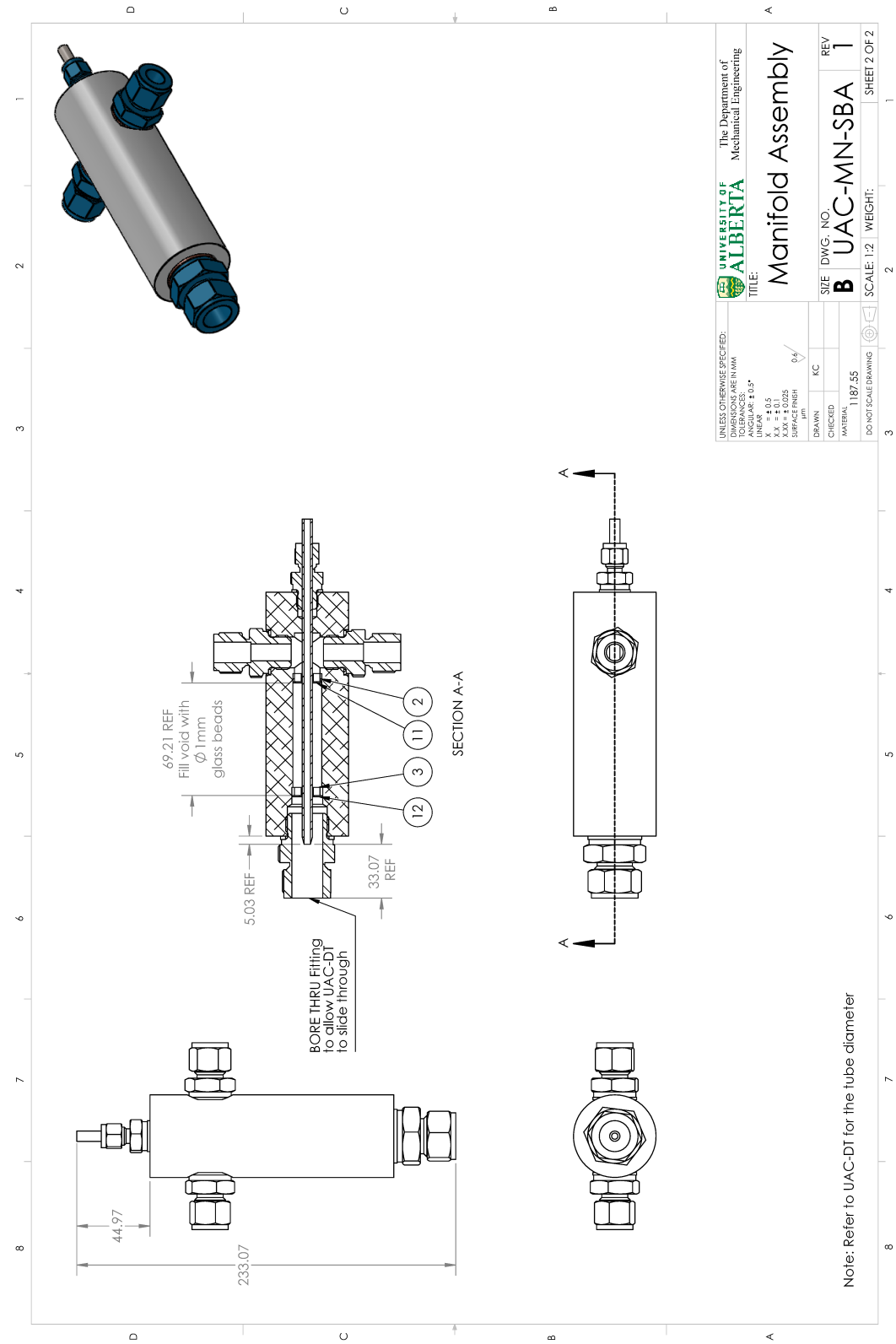
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH 0.6 µm
DRAWN KC
CHECKED
MATERIAL 1187.55
DO NOT SCALE DRAWING

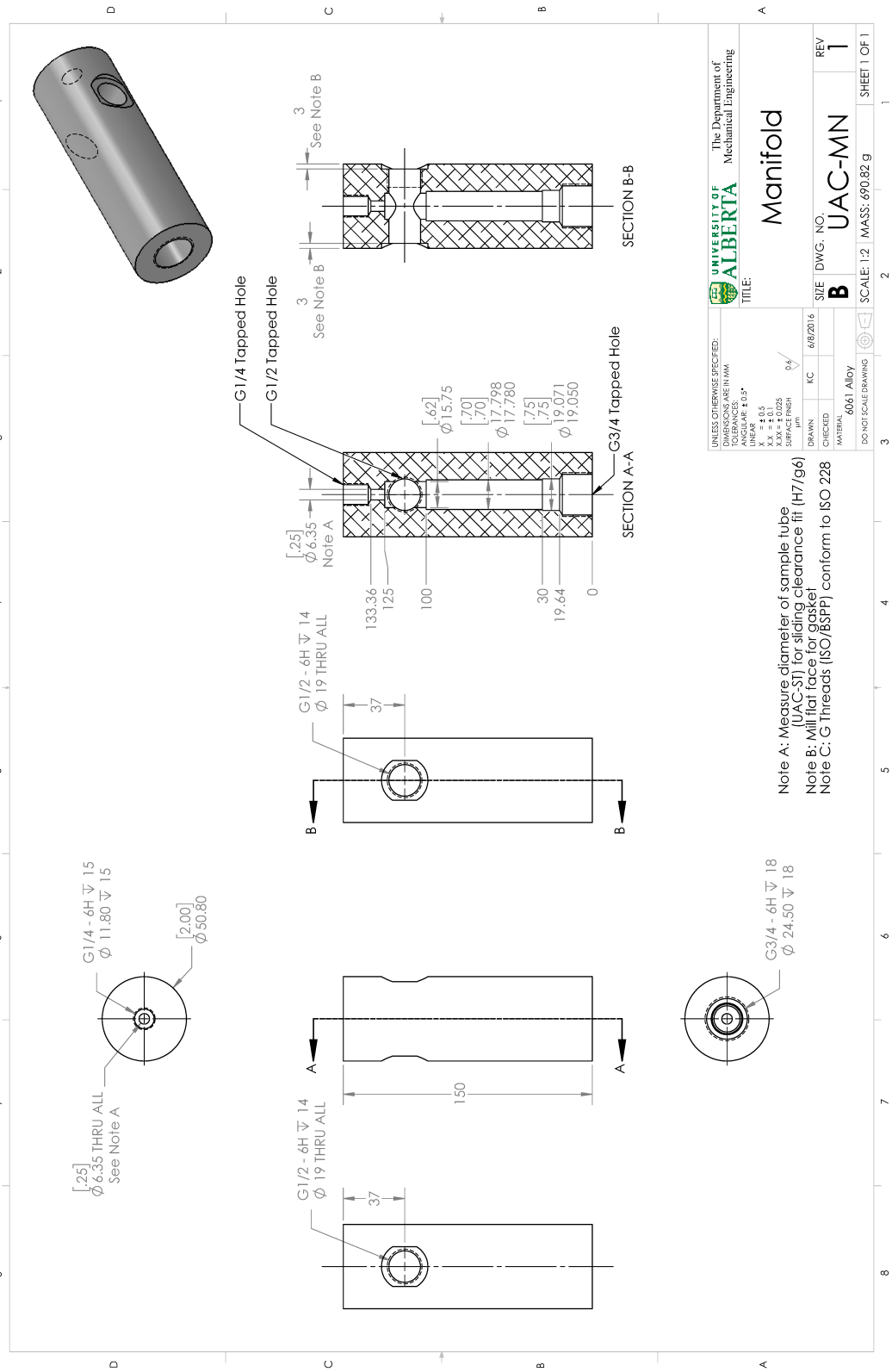
UNIVERSITY OF ALBERTA — The Department of Mechanical Engineering

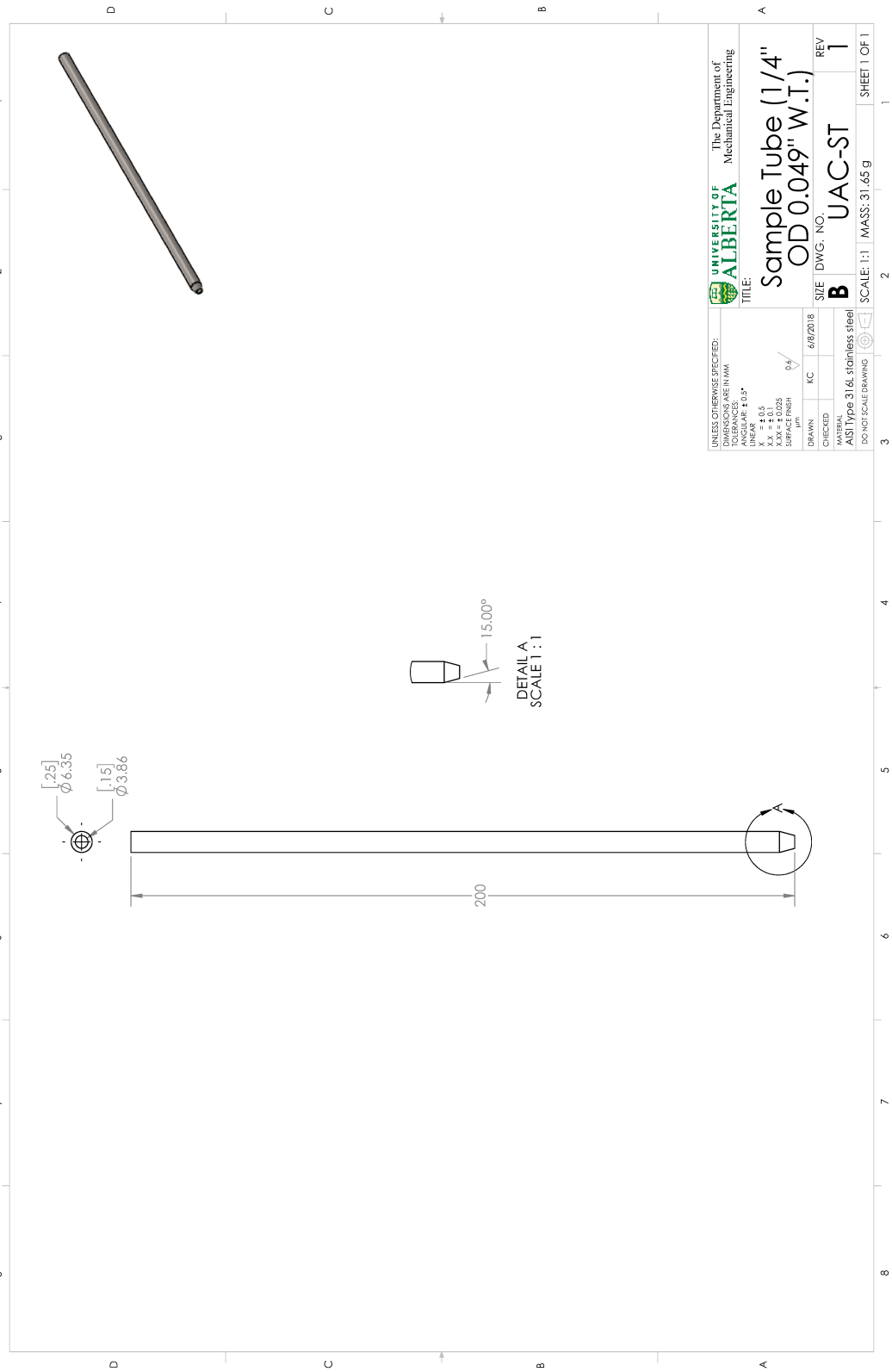
TITLE: Manifold Assembly

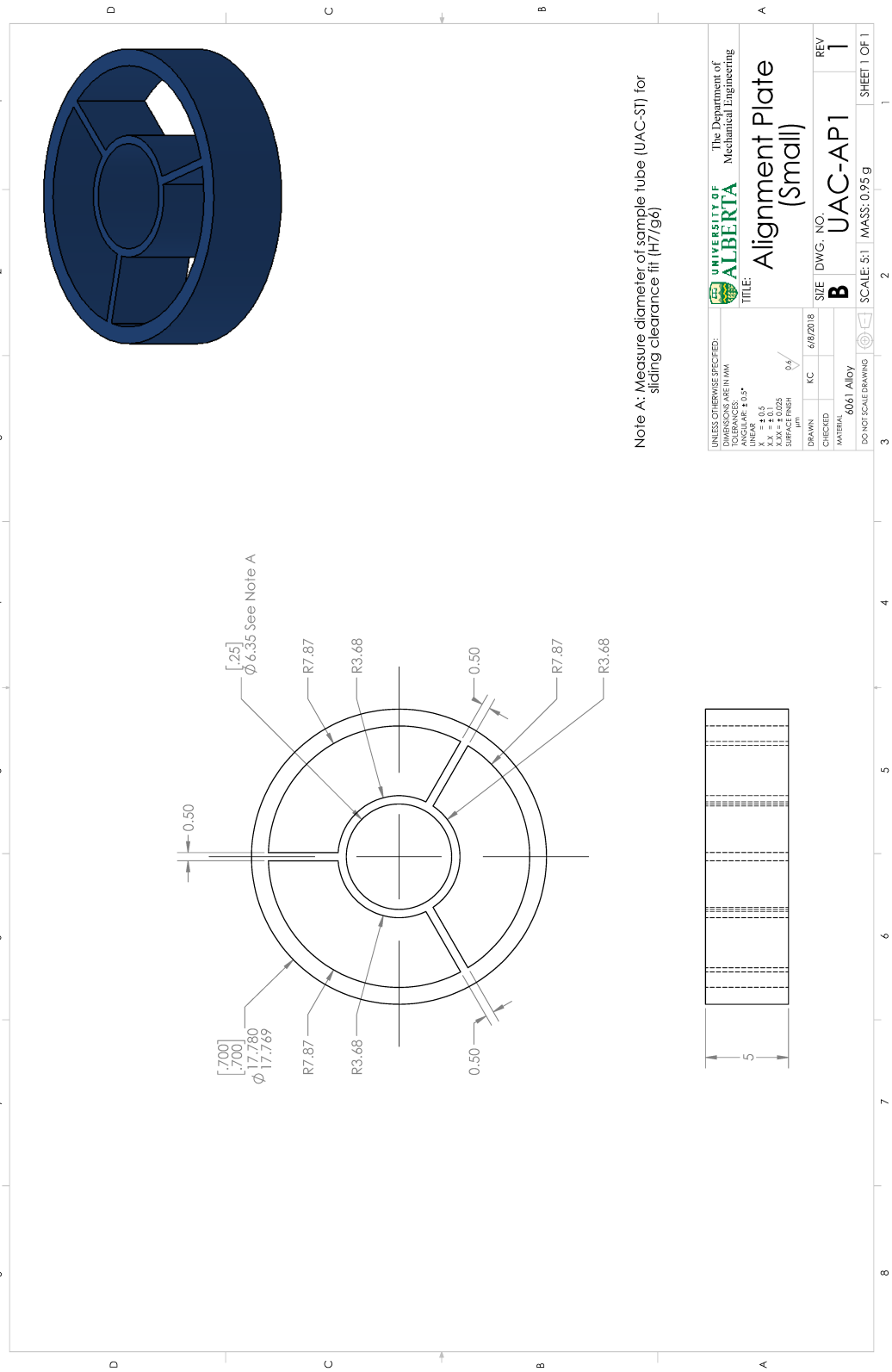
SIZE B | DWG. NO. UAC-MN-SBA | REV 1

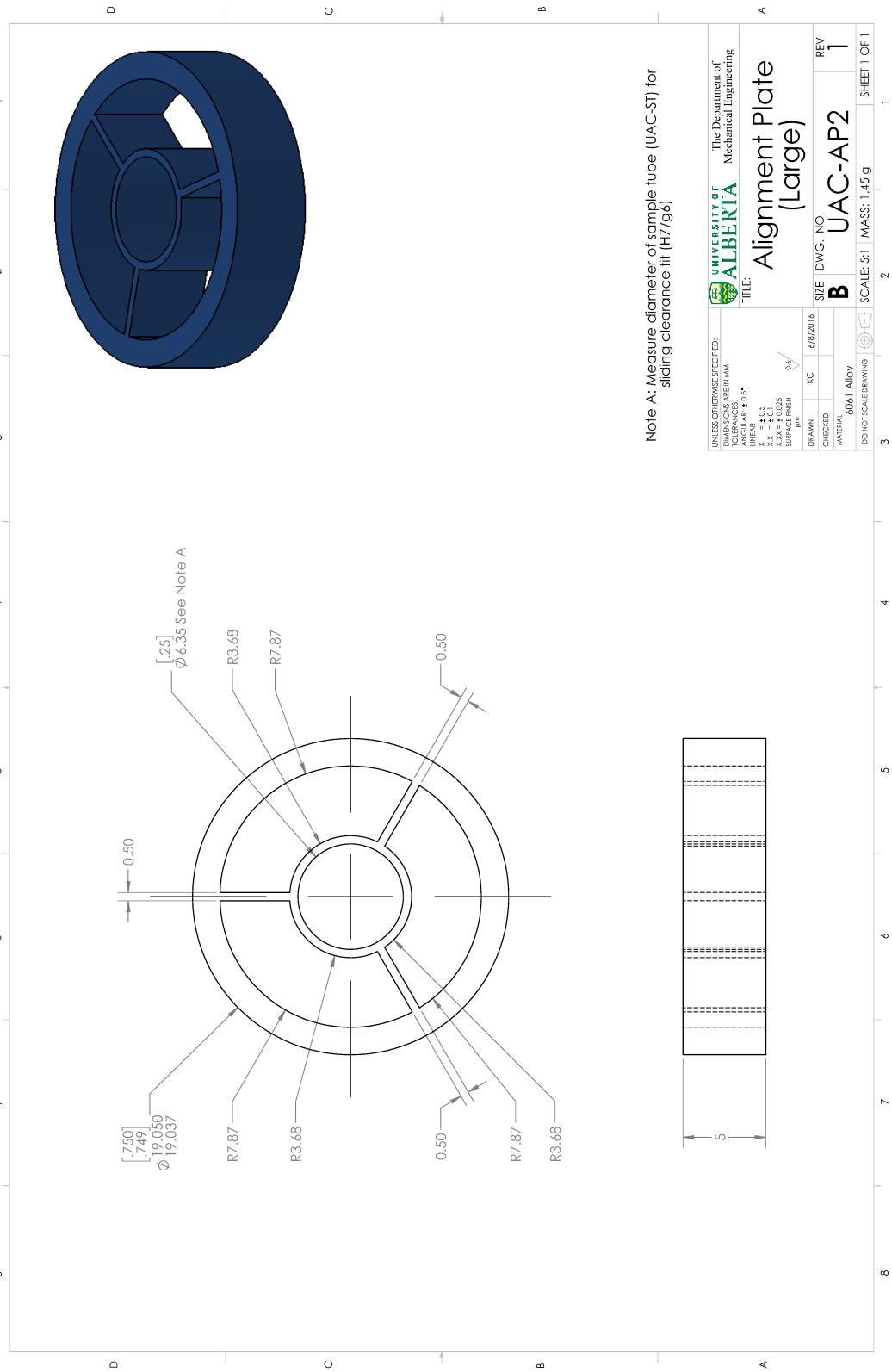
SCALE: 1:2 | MASS: 1187.55 | SHEET 1 OF 2

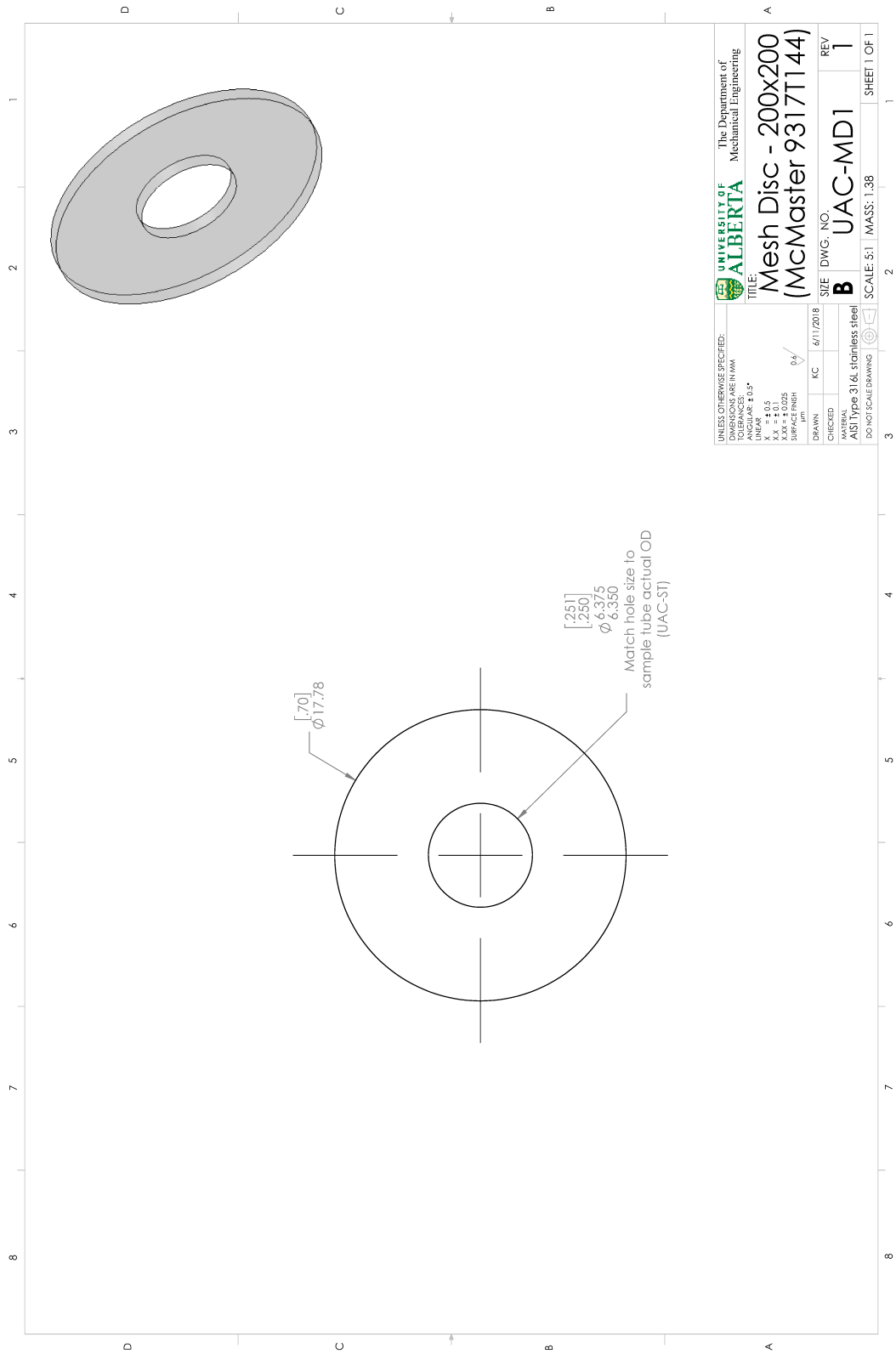

69.21 REF
Fill void with
Ø 1mm
glass beads
5.03 REF
33.07
REF
BORE THRU Fitting
to allow UAC-DT
to slide through
SECTION A-A
44.97
233.07
Note: Refer to UAC-DT for the tube diameter
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Manifold Assembly
SIZE B
DWG. NO. UAC-MN-SBA
REV 1
SCALE: 1:2
WEIGHT:
SHEET 2 OF 2
DRAWN KC
CHECKED
MATERIAL 1187.55
DO NOT SCALE DRAWING

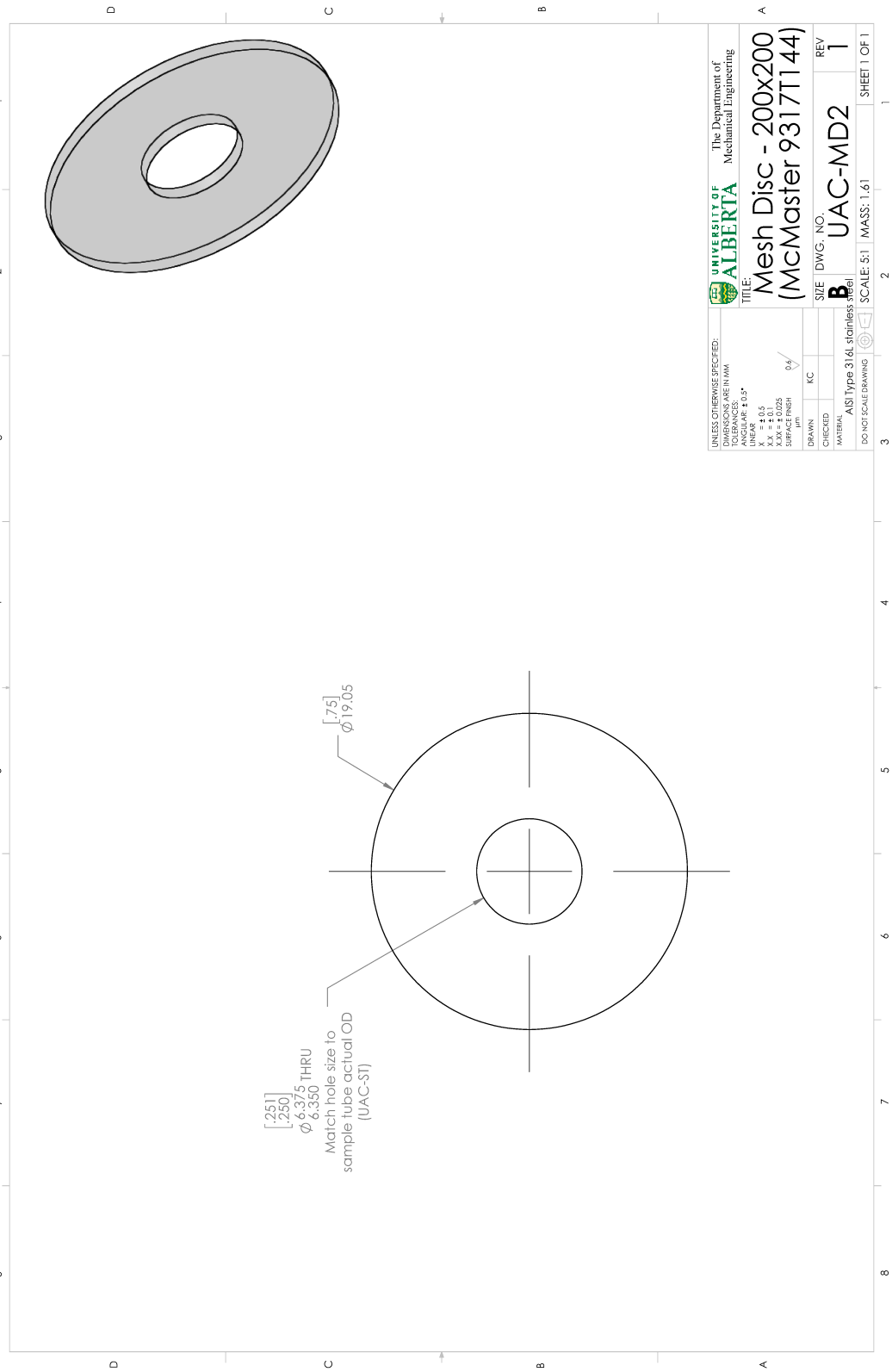

G1/4 Tapped Hole
G1/2 Tapped Hole
G3/4 Tapped Hole
SECTION A-A
SECTION B-B
See Note B
See Note A
G1/2 - 6H ↧ 14
⌀ 19 THRU ALL
G1/4 - 6H ↧ 15
⌀ 11.80 ↧ 15
G3/4 - 6H ↧ 18
⌀ 24.50 ↧ 18
[.25]
⌀6.35 THRU ALL
See Note A
[2.00]
⌀50.80
[.25]
⌀6.35
Note A
[.62]
⌀15.75
[.70]
[.70]
⌀17.798
17.780
[.75]
[.75]
⌀19.071
⌀19.050
133.36
125
100
30
19.64
0
37
150
Note A: Measure diameter of sample tube (UAC-ST) for sliding clearance fit (H7/g6)
Note B: Mill flat face for gasket
Note C: G Threads (ISO/BSPP) conform to ISO 228
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
μm
0.6
DRAWN KC 6/8/2016
CHECKED
MATERIAL
6061 Alloy
DO NOT SCALE DRAWING
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Manifold
SIZE B
DWG. NO. UAC-MN
REV 1
SCALE: 1:2
MASS: 690.82 g
SHEET 1 OF 1


[.25]
Ø6.35
[.15]
Ø3.86
200
15.00°
DETAIL A
SCALE 1 : 1
A
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
0.6
DRAWN
KC
6/8/2018
CHECKED
MATERIAL
AISI Type 316L stainless steel
DO NOT SCALE DRAWING
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Sample Tube (1/4" OD 0.049" W.T.)
SIZE
B
DWG. NO.
UAC-ST
REV
1
SCALE: 1:1
MASS: 31.65 g
SHEET 1 OF 1

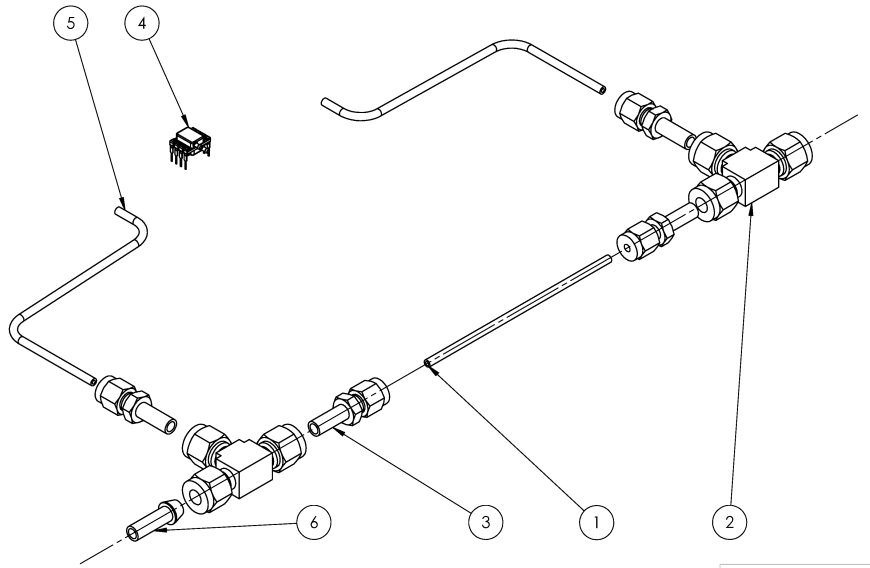

Note A: Measure diameter of sample tube (UAC-ST) for sliding clearance fit (H7/g6)
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Alignment Plate (Small)
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
DRAWN KC 6/8/2018
CHECKED
MATERIAL
6061 Alloy
DO NOT SCALE DRAWING
SIZE B
DWG. NO. UAC-AP1
REV 1
SCALE: 5:1
MASS: 0.95 g
SHEET 1 OF 1
[.25] Ø 6.35 See Note A
R7.87
R3.68
0.50
[.700] [.700] Ø 17.780 17.769
5


Note A: Measure diameter of sample tube (UAC-ST) for sliding clearance fit (H7/g6)
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
0.6
DRAWN
KC
6/8/2016
CHECKED
MATERIAL
6061 Alloy
DO NOT SCALE DRAWING
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Alignment Plate (Large)
SIZE
B
DWG. NO.
UAC-AP2
REV
1
SCALE: 5:1
MASS: 1.45 g
SHEET 1 OF 1
[.25] Ø6.35 See Note A
R3.68
R7.87
0.50
[.750] [.749] Ø19.050 Ø19.037
5


[.70]
ϕ17.78
[.251]
[.250]
ϕ 6.375
6.350
Match hole size to
sample tube actual OD
(UAC-S1)
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
0.6
DRAWN
KC
6/11/2018
CHECKED
MATERIAL
AISI Type 316L stainless steel
DO NOT SCALE DRAWING
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Mesh Disc - 200x200
(McMaster 9317T144)
SIZE
B
DWG. NO.
UAC-MD1
REV
1
SCALE: 5:1
MASS: 1.38
SHEET 1 OF 1


[.75]
ϕ19.05
[.251]
[.250]
ϕ 6.375
ϕ 6.350 THRU
Match hole size to
sample tube actual OD
(UAC-ST)
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
μm
0.6
DRAWN
KC
CHECKED
MATERIAL
AISI Type 316L stainless steel
DO NOT SCALE DRAWING
The Department of
Mechanical Engineering
TITLE:
Mesh Disc - 200x200
(McMaster 9317T144)
SIZE
B
DWG. NO.
UAC-MD2
REV
1
SCALE: 5:1
MASS: 1.61
SHEET 1 OF 1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Laminar Tube	1/8" OD 0.028" W.T. 316 Stainless Tube	1
2	SS-400-3	1/4" OD Swagelok Union Tee	2
3	SS-200-R-4	1/4" OD to 1/8" OD Swagelok Reducer	4
4	HSCDRRN002NG2A3	HSCDRRN002NGA3 Pressure Sensor	1
5	0.125 in OD Tygon	1/8" OD 1/16" ID Tygon Tubing	2
6	SS-401-PC	1/4" OD Swagelok Port Connector	1



UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH 0.6 µm
DRAWN KC
CHECKED
MATERIAL 220.88
DO NOT SCALE DRAWING

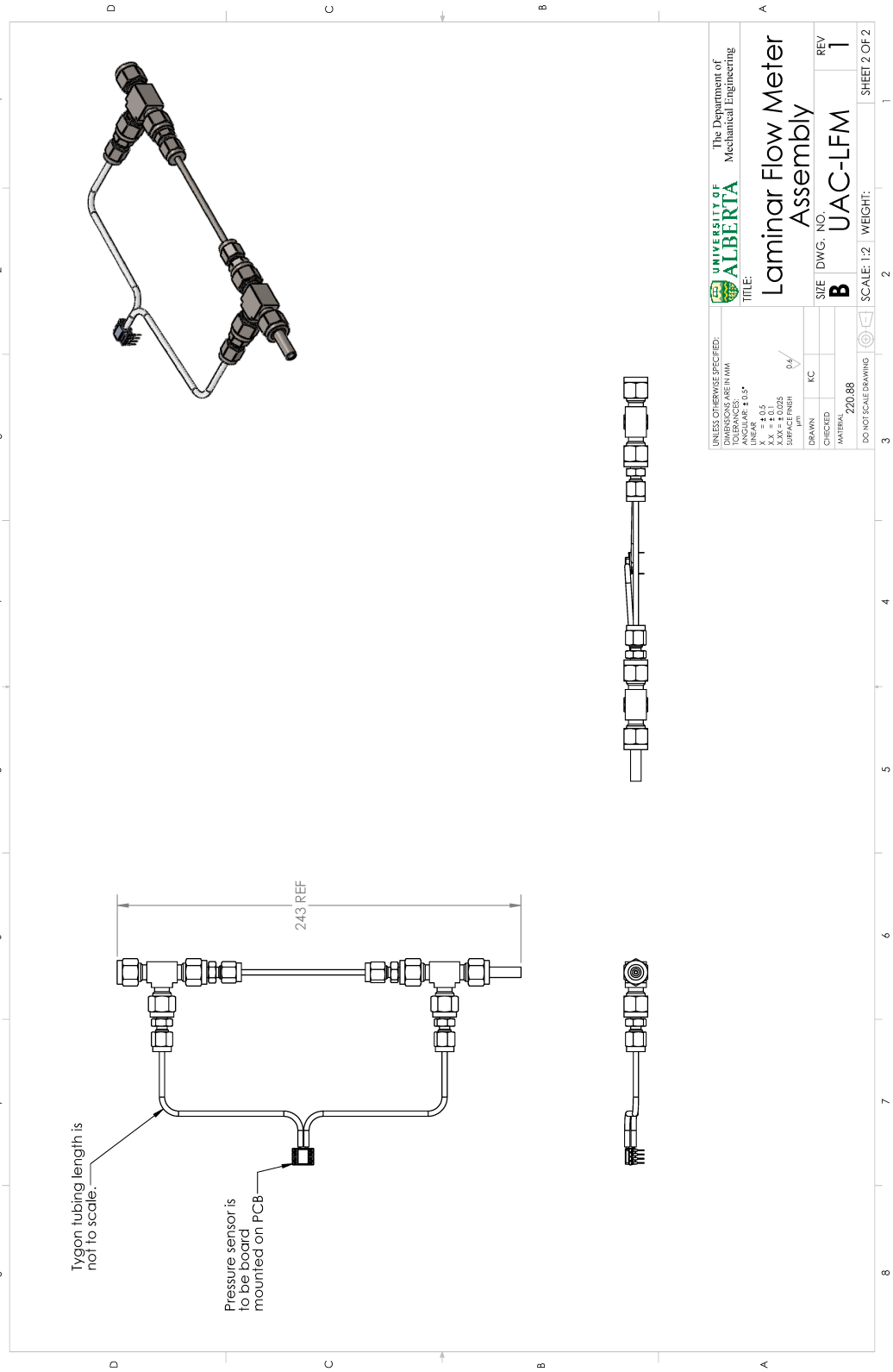
UNIVERSITY OF ALBERTA — The Department of Mechanical Engineering

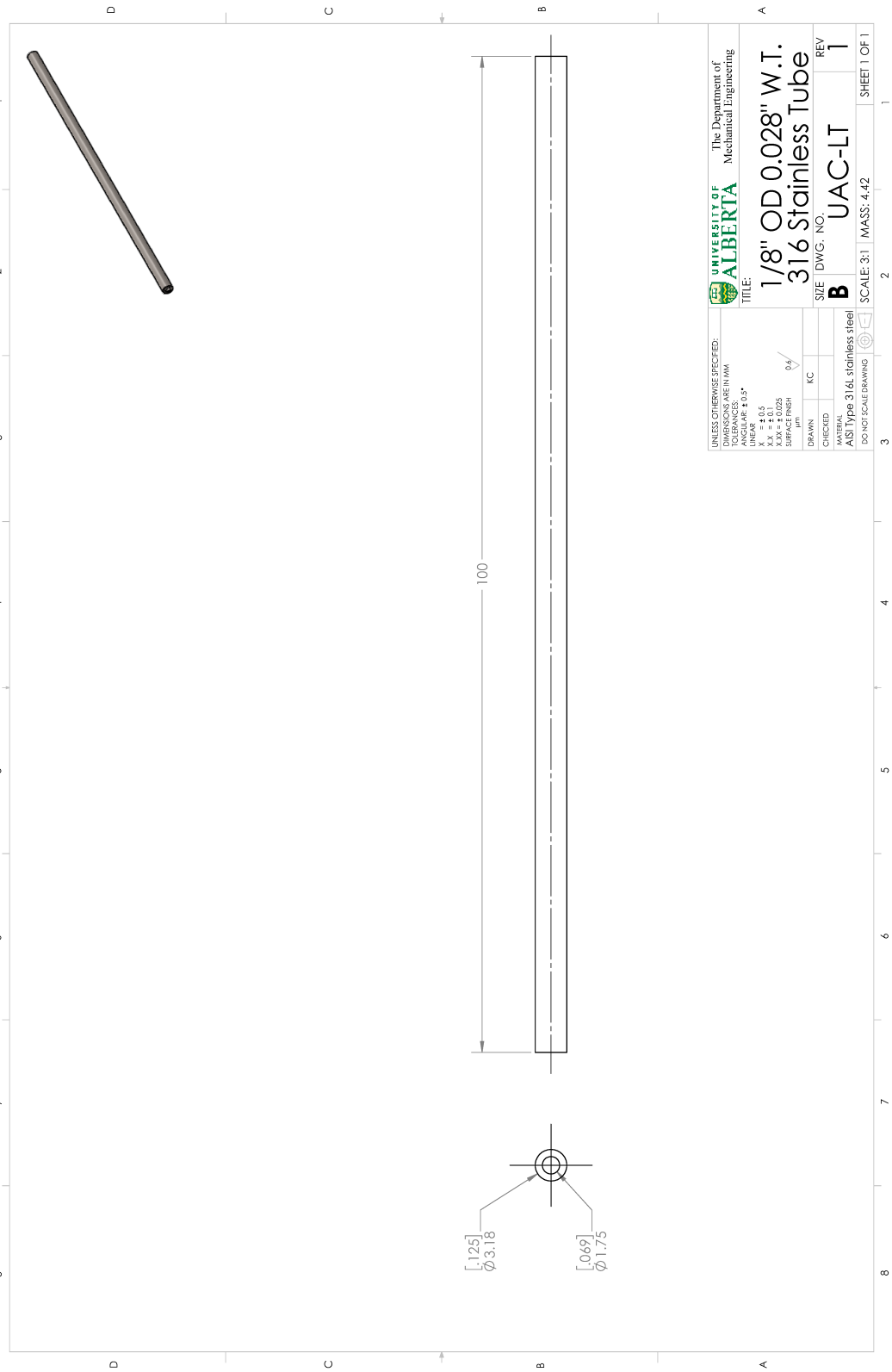
TITLE:

Laminar Flow Meter Assembly

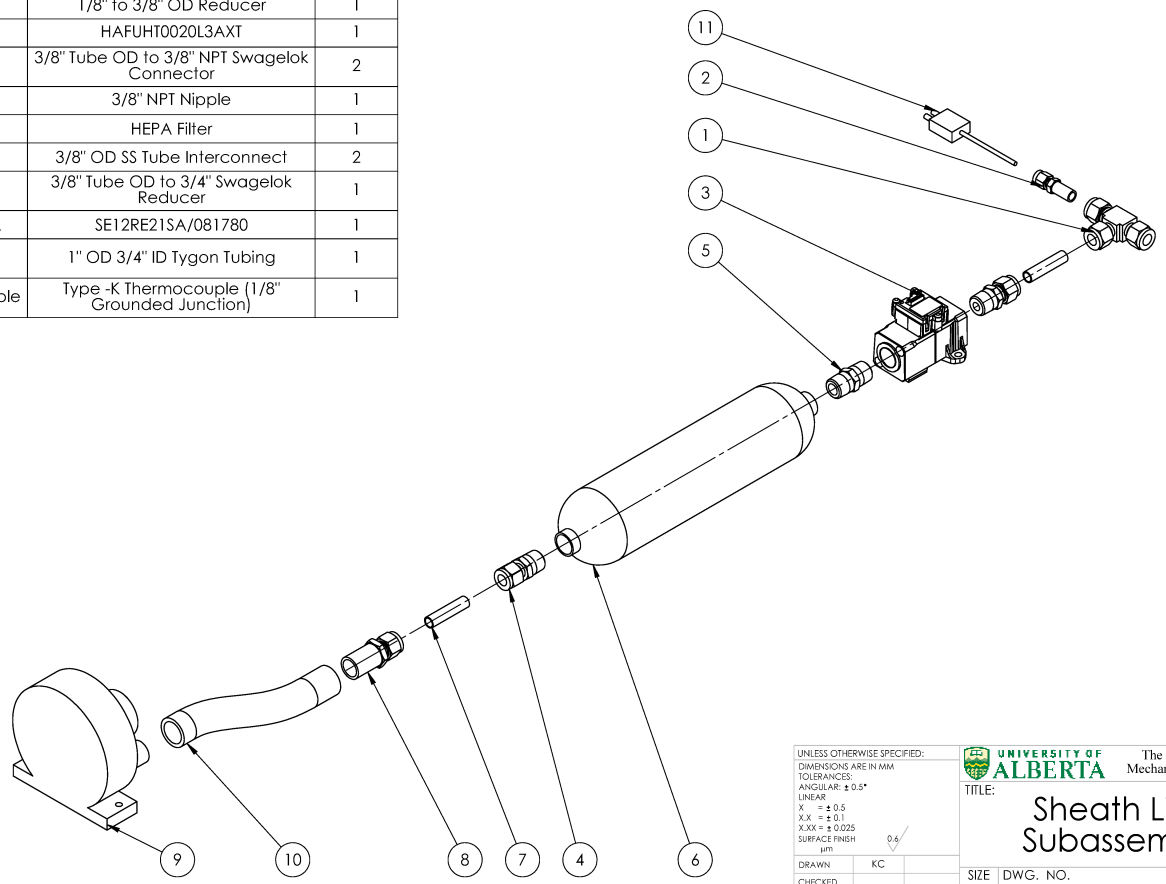
SIZE	DWG. NO.	REV
B	UAC-LFM	1

SCALE: 2:3 | MASS: 220.88 | SHEET 1 OF 2


243 REF
Tygon tubing length is not to scale.
Pressure sensor is to be board mounted on PCB
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
0.6
DRAWN KC
CHECKED
MATERIAL
220.88
DO NOT SCALE DRAWING
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Laminar Flow Meter Assembly
SIZE B
DWG. NO. UAC-LFM
REV 1
SCALE: 1:2
WEIGHT:
SHEET 2 OF 2


100
[.125]
Ø3.18
[.069]
Ø1.75
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
1/8" OD 0.028" W.T.
316 Stainless Tube
SIZE
B
DWG. NO.
UAC-LT
REV
1
SCALE: 3:1
MASS: 4.42
SHEET 1 OF 1
DRAWN
KC
CHECKED
MATERIAL
AISI Type 316L stainless steel
DO NOT SCALE DRAWING
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
0.6

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	SS-600-3	3/8" OD Swagelok Union Tee	1
2	SS-200-R-6	1/8" to 3/8" OD Reducer	1
3	Honeywell Zephyr	HAFUHT0020L3AXT	1
4	SS-600-1-6	3/8" Tube OD to 3/8" NPT Swagelok Connector	2
5	SS-6-HN	3/8" NPT Nipple	1
6	Pall HEPA Filter	HEPA Filter	1
7	Tube Interconnect	3/8" OD SS Tube Interconnect	2
8	SS-600-R-12	3/8" Tube OD to 3/4" Swagelok Reducer	1
9	Ametek MINISPIRAL	SE12RE21SA/081780	1
10	Outlet Tubing to Blower	1" OD 3/4" ID Tygon Tubing	1
11	Type-K Thermocouple	Type -K Thermocouple (1/8" Grounded Junction)	1



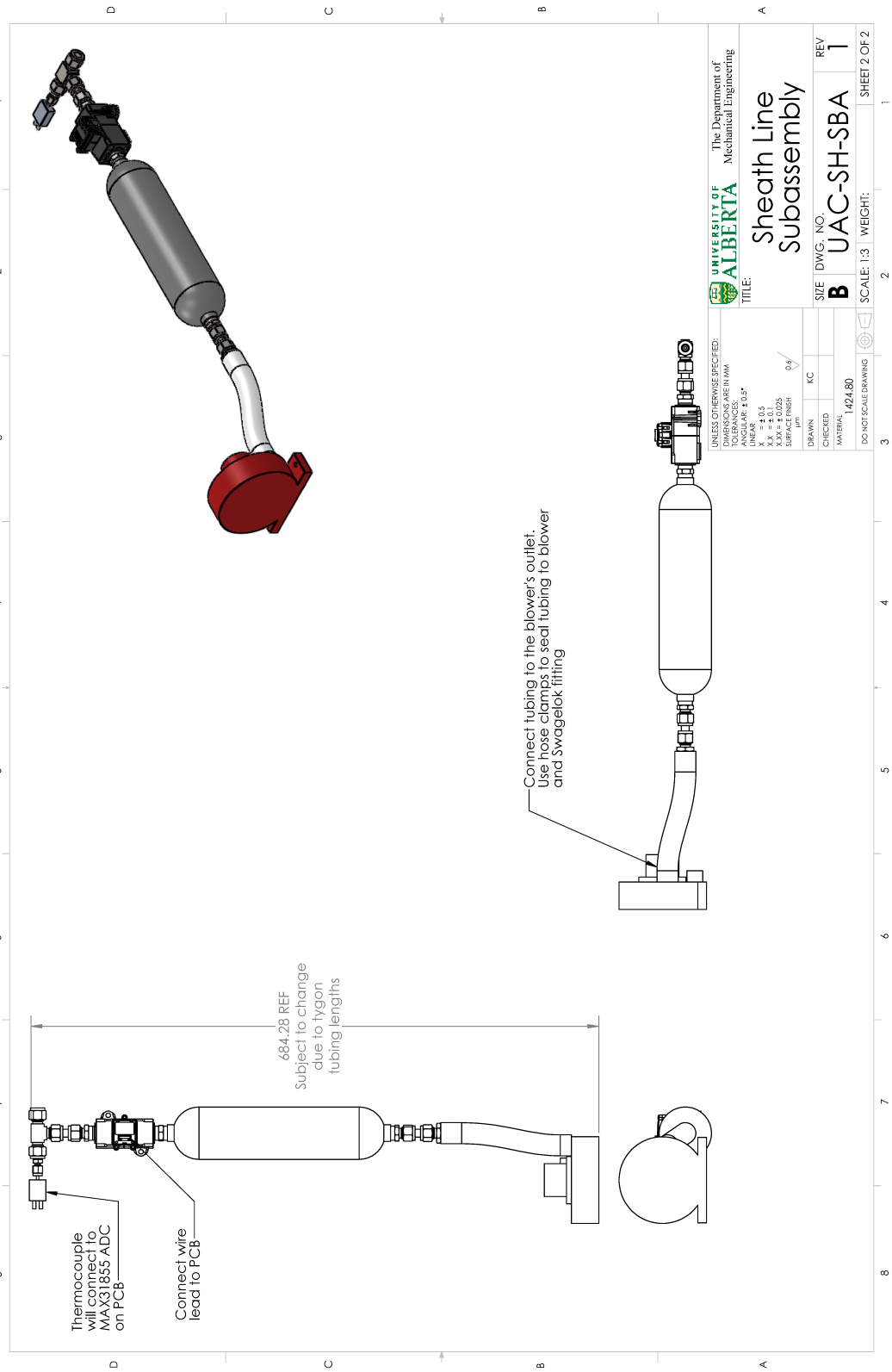
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH 0.6 µm
DRAWN KC
CHECKED
MATERIAL 1424.80
DO NOT SCALE DRAWING

UNIVERSITY OF ALBERTA — The Department of Mechanical Engineering

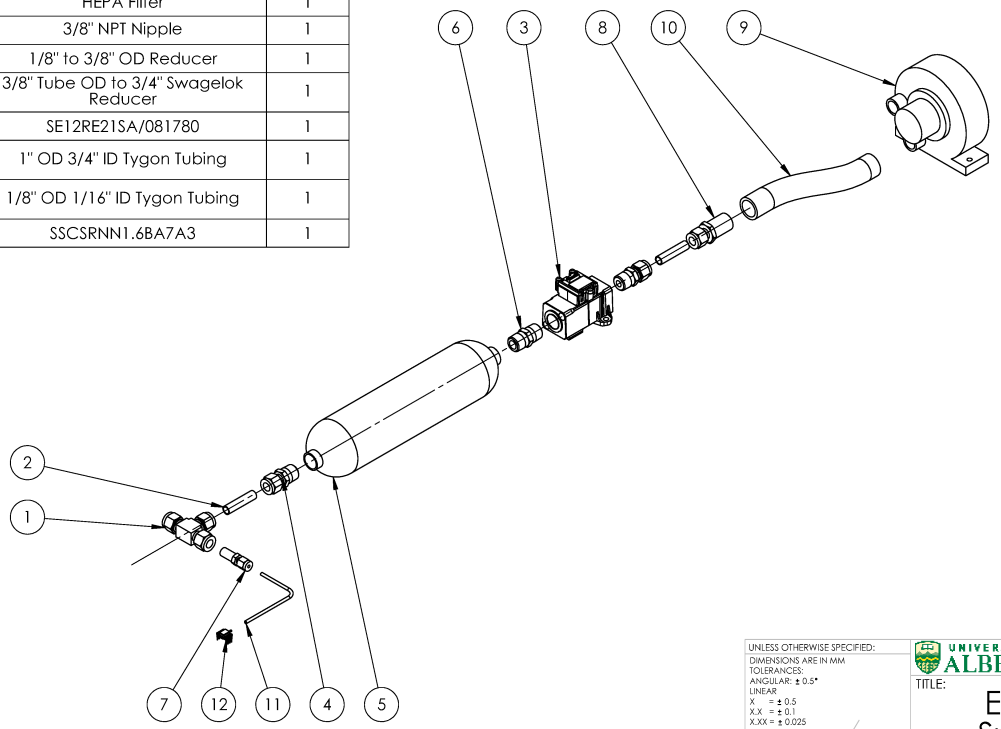
TITLE: Sheath Line Subassembly

SIZE B | DWG. NO. UAC-SH-SBA | REV 1

SCALE: 1:3 | MASS: 1424.80 | SHEET 1 OF 2


Thermocouple will connect to MAX31855 ADC on PCB
Connect wire lead to PCB
684.28 REF
Subject to change due to tygon tubing lengths
Connect tubing to the blower's outlet. Use hose clamps to seal tubing to blower and Swagelok fitting
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
0.6
DRAWN KC
CHECKED
MATERIAL
1424.80
DO NOT SCALE DRAWING
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Sheath Line Subassembly
SIZE B
DWG. NO. UAC-SH-SBA
REV 1
SCALE: 1:3
WEIGHT:
SHEET 2 OF 2

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	SS-600-3	3/8" OD Swagelok Union Tee	1
2	Tube Interconnect	3/8" OD SS Tube Interconnect	2
3	Honeywell Zephyr	HAFUHT0020L3AXT	1
4	SS-600-1-6	3/8" Tube OD to 3/8" NPT Swagelok Connector	2
5	Pall HEPA Filter	HEPA Filter	1
6	SS-6-HN	3/8" NPT Nipple	1
7	SS-200-R-6	1/8" to 3/8" OD Reducer	1
8	SS-600-R-12	3/8" Tube OD to 3/4" Swagelok Reducer	1
9	Ametek MINISPIRAL	SE12RE21SA/081780	1
10	Outlet Tubing to Blower	1" OD 3/4" ID Tygon Tubing	1
11	0.125 in OD Tygon for Exhaust line	1/8" OD 1/16" ID Tygon Tubing	1
12	SSCSRNN1.6BA7A3	SSCSRNN1.6BA7A3	1



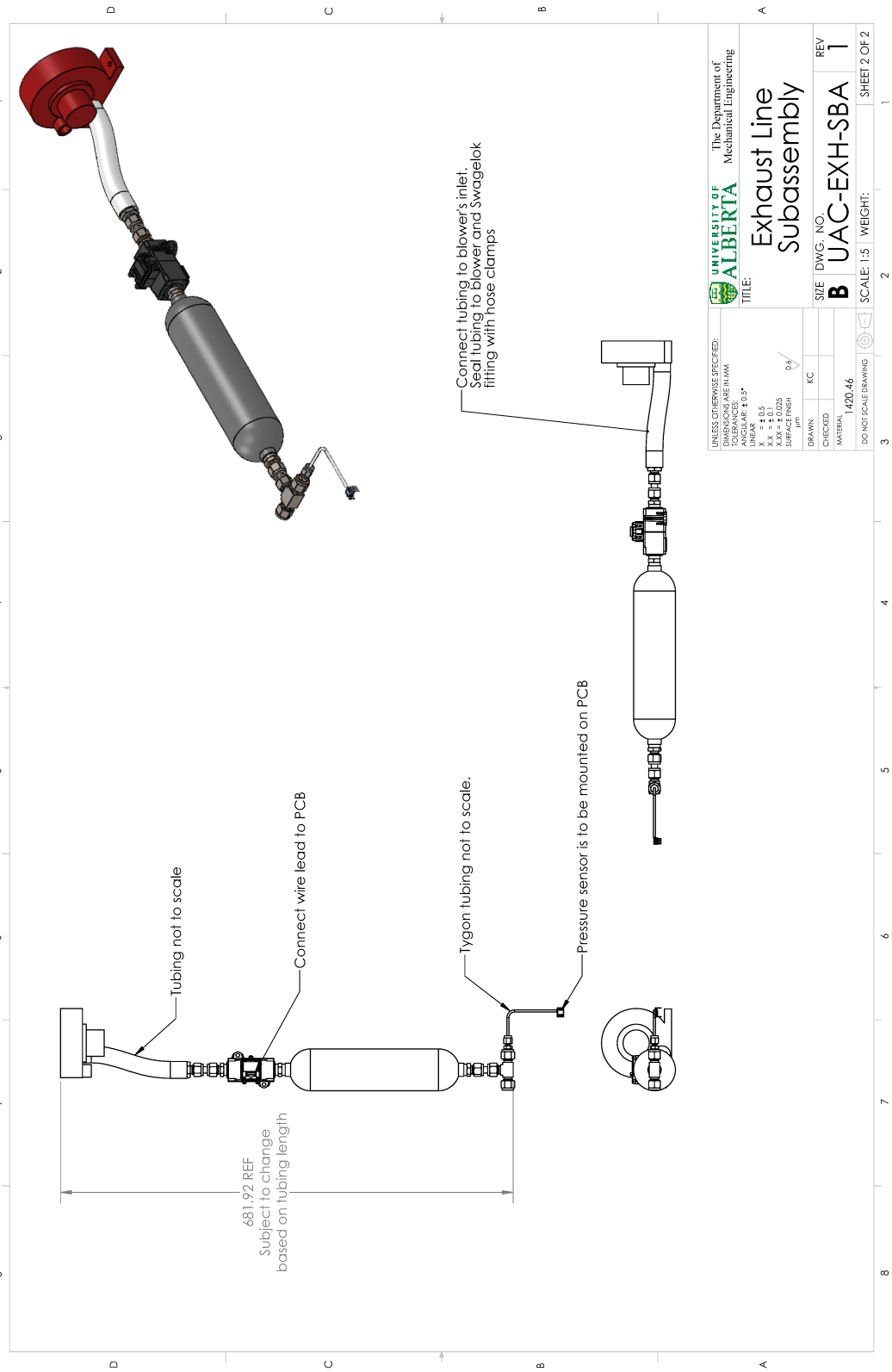
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH 0.6 µm
DRAWN KC
CHECKED
MATERIAL 1420.46
DO NOT SCALE DRAWING

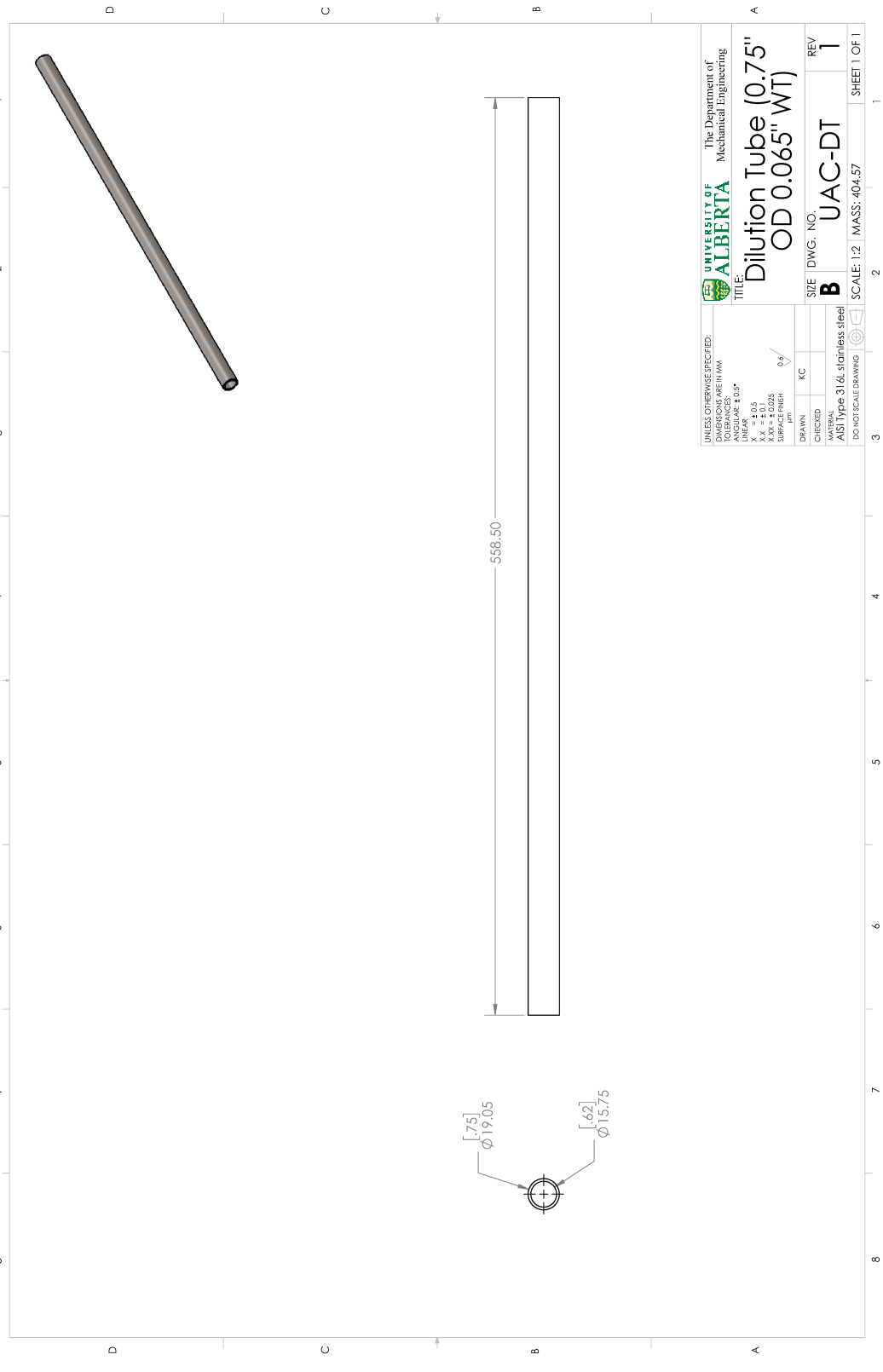
UNIVERSITY OF ALBERTA — The Department of Mechanical Engineering

TITLE: Exhaust Line Subassembly

SIZE B | DWG. NO. UAC-EXH-SBA | REV 1

SCALE: 1:4 | MASS: 1420.46 | SHEET 1 OF 2


Connect tubing to blower's inlet.
Seal tubing to blower and Swagelok
fitting with hose clamps
Tubing not to scale
Connect wire lead to PCB
Tygon tubing not to scale.
Pressure sensor is to be mounted on PCB
681.92 REF
Subject to change
based on tubing length
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Exhaust Line Subassembly
SIZE
B
DWG. NO.
UAC-EXH-SBA
REV
1
SCALE: 1:5
WEIGHT:
SHEET 2 OF 2
DRAWN
KC
CHECKED
MATERIAL
1420.46
DO NOT SCALE DRAWING


558.50
[.75]
Ø19.05
[.62]
Ø15.75
UNIVERSITY OF ALBERTA
The Department of Mechanical Engineering
TITLE:
Dilution Tube (0.75" OD 0.065" WT)
SIZE
B
DWG. NO.
UAC-DT
REV
1
SCALE: 1:2
MASS: 404.57
SHEET 1 OF 1
DRAWN
KC
CHECKED
MATERIAL:
AISI Type 316L stainless steel
DO NOT SCALE DRAWING
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MM
TOLERANCES:
ANGULAR: ± 0.5°
LINEAR
X = ± 0.5
X.X = ± 0.1
X.XX = ± 0.025
SURFACE FINISH
µm
0.6