Candidate Number: 157960
Word Count: 5923

MPhys Project
University of Oxford

AO05: Instrument to sample and return airborne particles in a volcanic plume

Supervisor: Dr. Dan Peters

April 15, 2013

Abstract

This report outlines the development of an electrostatic precipitator for use onboard an unmanned aerial vehicle, inspired by the need to better quantify particulate matter in volcanic plumes and the optical properties of atmospheric black carbon. Software that allows remote control turn-on and an automated collection procedure was written. The precipitator is then used to collect a laboratory generated black carbon aerosol onto SEM stubs for microscope analysis; agglomerates are then identified and analysed. In the time permitted for this project, it was not possible to obtain a statistically significant number of particles to allow for analysis of concentration, so focus fell on determination of fractal dimensions for imaged agglomerates. Factors contributing to errors in the calculation of fractal dimension are discussed and methods to reduce them are developed and implemented.

1 Introduction

The importance of quantifying and analysing airborne ash particles was made evermore apparent by the significant disruption to air traffic caused by the 2010 eruption of Eyjafjallajökull. Prediction of the evolution of the plume following the eruption required assumptions about particle size and shape; the availability of a *in situ* sample characterising particle size and shape would improve confidence in these predictions. During the eruption only on in-situ size distribution was available. These were obtained from the Facility for Airborne Atmospheric Measurements (FAAM) research aircraft, and only in areas considered safe to fly a manned aircraft. Coupled with the fact that unmanned aerial vehicles (UAVs) have become relatively cheap recently, this project aimed to use an electrostatic precipitator (ESP) on board a UAV to collect such aerosols onto an electron microscope stub to be returned for laboratory analysis. The project built and tested the ESP on a laboratory-generated black carbon aerosol.

2 Background

The use of a point-to-plane ESP for aerosol sampling was first considered by Morrow and Mercer (1964)[1], and since then been expanded upon, notably by Miller et al. (2010)[2] into a more compact, hand-held form. This project developed an ESP with the critical design parameters given by Miller et al. Their results provide a reference for field strength and efficiency calculations.

The basic operation of an ESP (Figure 1) requires a draw through of sample air at a known rate. The tip of a needle is used to generate a strong electric field, which, at sufficently high voltage (Miller et al. used the range 5.6–6.8kV) forms a corona near the needle tip. The resulting ions flow towards the grounded plate; they collide with and add charge to passing aerosols. The charged particles are then drawn towards the grounded plate, a scanning electron microscope (SEM) grid.

The total velocity of a particle in the flow and electric field is given by[3]:

$$\mathbf{v} = \mathbf{v}_{\text{flow}} + \mathbf{v}_{\text{electric}} = \mathbf{v}_{\text{flow}} + \frac{C_c ne\mathbf{E}}{3\pi\eta d} \qquad (1)$$

where ne is the charge on the particle, d is the particle diameter, η is the dynamic viscosity of air, and C_c , the Cunningham slip correction[1], is given by:

$$C_c(\text{Kn}) = 1 + 2\text{Kn}\left[\alpha + \beta\exp\left(-\frac{\gamma}{\text{Kn}}\right)\right] \qquad (2)$$

where Kn is the Knudsen number (the ratio of mean free path to particle diameter). Buckley and

[1] Stoke's Law assumes a no slip boundary condition which no longer holds when the particle size is of the order of the mean free path. $C_c \to 1$ when $d \gg \lambda$.

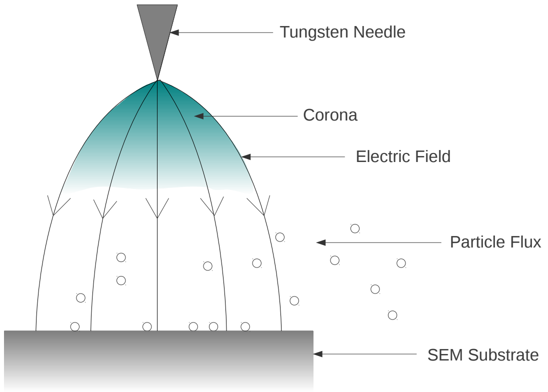


Figure 1: Basic principle of an ESP. Incoming particles are ionised by ions flowing from the corona, and are then attracted to the grounded substrate.

Loyalka (1989)[4] gave the empirical constants: $\alpha = 1.099$, $\beta = 0.518$, and $\gamma = 0.425$ for air. For example, a large particle with a micron diameter has (at S.T.P.) Kn $= 0.068$, giving $C_c = 1.15$; but a small particle with a diameter of 30nm has Kn $= 2.27$, giving $C_c = 4.52$. When we come to calculate the efficiency, we will be interested in the particles with slow velocities, i.e. those with large diameter and small C_c .

To be able to accurately calculate concentration, a precise measurement of the flow rate (i.e. the total number of particles) of air being drawn through the system at any given time must be known. Accurate measurements of flow can be obtained from the hot wire technique (measuring changes in current of a heated resistor when subjected to flow), which directly measures mass flow. To convert this to volumetric flow, the pressure and temperature of the sample gas must be known. The detected mass flow rate can be converted into the equivalent (standard) volumetric flow rate at standard conditions, using equation 3, derived from the ideal gas law.

$$\dot{m} = \frac{m_{N_2}P}{RT}\dot{V} \qquad (3)$$

where $\dot{m}$ is the mass flow rate, m_{N_2} is the molecular mass of nitrogen, R is the gas constant, and $\dot{V}$ is the volumetric flow rate.

3 Testing and Design Review

3.1 Physical Design

The final design is required to fit into the instrumentation bay of the UAV (12cm high, 12.5cm wide, and 16cm long). The UAV also has a gross weight limit, constraining the mass of the total payload to a maximum of 3kg. Smaller mass allows a longer flight time (100g would give 100 minutes flight, 2000g would give 20 minutes)[5].

3.1.1 Housing Design

The housing was machined from polycarbonate according to the specifications given in Figure 2. An 80mm long V shaped channel was milled out of a block of polycarbonate forming the upper blue segment in Figure **??**. Into this, three holes were machined to allow insertion of the tungsten needles that formed the “point” in the point-to-plane field. The upper part is joined to the lower block, with three holes machined for the SEM stubs. The needles and stubs are fixed into place by the use of o-rings and secondary plates (hatched in Figure 2). Figure 3 shows a photograph of the housing from the underneath.

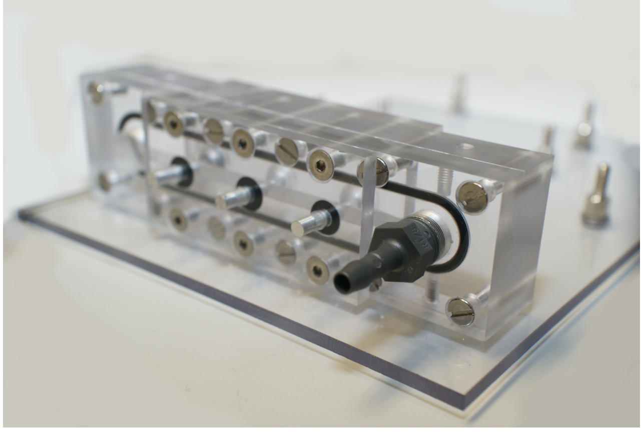


Figure 3: A photograph of the housing; the bases of the three stubs and the outlet pipe are visible.

3.1.2 Electronics

The power supply within the UAV can provide 5V with a maximum load of 1A. Four main components require power: the microcontroller, the high voltage supply for the precipitators, the flow meter, and the diaphragm pump. The UAV powered the microcontroller, which in turn powered each of the other three components.

The ESP requires unmanned operation, software was written so that the system will run automatically, upon remote control. An Arduino Uno microcontroller is used; the Arduino is programmed using its own language, based on C and has 6 analog inputs and 14 digital outputs (6 of which are PWM[2]). An RC circuit was set up to ensure that

[2] Pulse Width Modulated - simulating an analog signal with a digital output by changing the ratio of high period

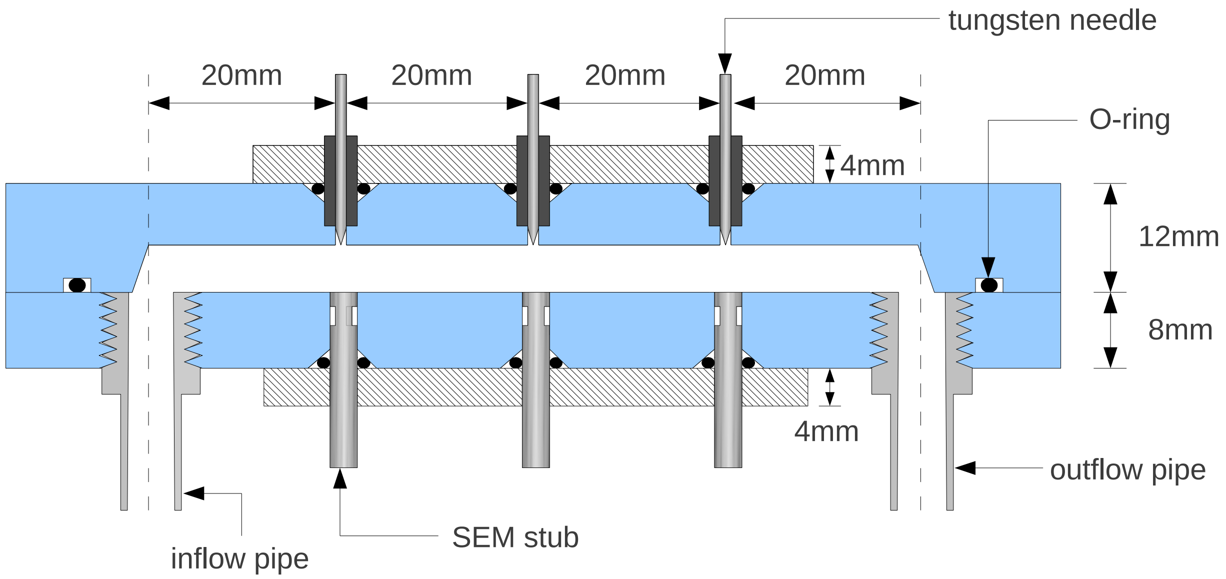


Figure 2: Scale diagram of the specifications of the housing used for the three individual precipitators. Hatched and blue areas are made from PVC.

the oscillations from the PWM did not affect the high voltage output.

To generate the necessary high voltage, miniature high voltage power supplies are required. Three *HVM Tech SMHV0580* sub-miniature regulated high voltage DC to DC converters[6] are used, one for each precipitator. They are capable of delivering a maximum of 8kV at a maximum of 125μA, and thus satisfy the requirements for the project.

It is important to ensure that arcing is not present during operation. The dielectric strength of air, ε, is 3×10^6Vm^{-1}[12]. Using

$$V_{\text{breakdown}} = \varepsilon d \tag{4}$$

with d, the distance between conducting surfaces, as 5mm, we have $V_{\text{breakdown}}$ = 15kV. Our operation at 5kV is far below this, but this is not a particularly good approximation and Miller et al. found that arcing occurred above 8kV.

3.1.3 Mechanical Parts

To draw the sample through the system, a diaphragm pump is used. Diaphragm pumps are lightweight and capable of the small flow rate required.

Mass flow meter: As with the other components, emphasis was placed on the mass flow meter being compact and lightweight; and a suitable choice was a hot wire sensor: we chose the *Honeywell AWM3100V*[8]. This also had the advantage of a very fast response time (allowing analysis such as that in Figure 6), and being relatively cheap. As the wire is prone to contamination, paper filters are fit either side of the meter to reduce the risk of particulate matter entering it (see Figure 4). The meter runs off 5V, and outputs a voltage between 1.1V and 5V, depending on the flow. This voltage is used as an analog input on the microcontroller, which compares the input to a 5V reference voltage and records a 10-bit integer proportional to the ratio (i.e. 1023 at a 5V maximum, 225 at a 1.1V minimum). This measure is used in Figure 5. The meter was calibrated at RAL using a *Gilibrator-2 Primary Air Flow Calibrator*[9].

3.2 Software Development

As mentioned in section 3.1.2, the Arduino microcontroller was programmed using its own language. The requirements of the software are that:

- it should turn on by remote control, and then run automatically
- it should turn on and regulate the three HV supplies for a given amount of time each
- it should turn on and regulate the diaphragm

to low period (mark space ratio) in a high frequency square wave pump (using a PID loop, section 3.4)

- it should read (and log, if necessary) the output of the mass flow meter.

The UAV used has a spare remote control servo output signal. This 5V PWM can be read by the microcontroller and interpreted as a message to activate the ESP sampling. This is done using an interrupt (see Appendix B.1); whenever the microcontroller detected a specific PWM mark space ratio change, it activates the sampler.

The full code written for laboratory testing of the ESP can be found in Appendix B.2.

3.3 Flow

This section addresses aerosol flow related sampling issues of the instruement running at the flow rates determined by Miller et al.[2] We chose to operate at $55\mathrm{cc\,min}^{-1}$, which gives a theoretical efficiency of 100%.

3.3.1 Laminar Flow

Ideally the flow should be laminar throughout the system, particularly in the channel. This avoids any inertial particle loss on the aerosol sampled, and reduces cross-contamination caused by potential backflow. Taking the Reynolds number:

$$\mathrm{Re} = \frac{\rho v L}{\eta} = \frac{\rho \Phi L}{\eta A} \tag{5}$$

where η is the dynamic viscosity ($\eta_{\mathrm{air}} = 1.8 \times 10^{-5}\mathrm{kg\,m}^{-1}\mathrm{s}^{-1}$), ρ is the fluid density ($\rho_{\mathrm{air}} = 1.2\mathrm{kg\,m}^{-1}\mathrm{s}^{-1}$, L is a characteristic length scale, Φ is the flow rate (volume flux), A is the cross-sectional area, and v is the flow velocity. For the V-channel, $l = 3\mathrm{mm}$ and $A = 7.5\mathrm{mm}^2$, giving, for a flow rate of of $\Phi = 55\mathrm{cc\,min}^{-1}$, a Reynolds number of 25. The inlet and outlet pipes have a similar characteristic dimension (pipe diameter of 4mm). All the Reynolds numbers in the instrument are smaller than 2000 and we would expect laminar flow.

3.3.2 Bends

Our design requires that flow entering the inlet must undergo a right-angle turn before reaching the precipitator. This is done for two reasons: firstly, the precipitator housing is already quite long, and there is a dimensional constraint on the housing that would then require difficult positioning of the remaining components; and secondly that the way in which the housing is constructed from polycarbonate would make an airtight seal at the inlet difficult from a perpendicular face (see Figure 2). To make sure that there are no losses by impaction due to the sharp 90° bend, we must satisfy the condition that the Stokes drag force is much greater than the centrifugal force, i.e.:

$$6\pi\nu v \gg \frac{mv^2}{r} \tag{6}$$

where r is the radius of curvature of the channel, ν is the kinematic viscosity of air, and m is the particle mass. This can be rearranged for a spherical particle to give:

$$v \ll \frac{9}{2}\frac{\nu r}{R^2 \rho} \tag{7}$$

where R is the particle radius and ρ is its density. Even for a large carbon particle with $R = 2\mu\mathrm{m}$, this gives the constraint $v \ll 100\mathrm{ms}^{-1}$ (corresponding to $\Phi \ll 4 \times 10^4\mathrm{cc\,min}^{-1}$), easily within the working range of the apparatus.

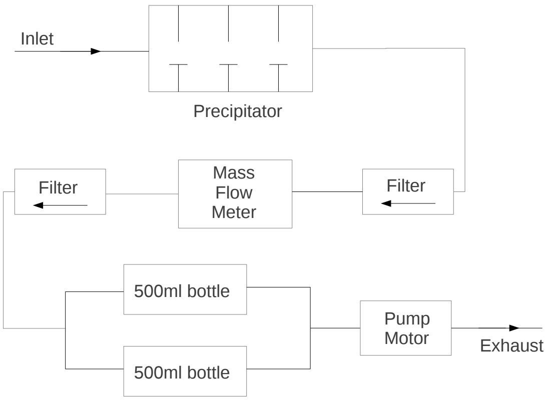


Figure 4: The path taken by sample air through the instrument.

3.4 PID loop

It is important to maintain a constant flow rate through the precipitator as this is the dominant source of number density error. For this, we use a proportional-integral-differential (PID) control loop, a feedback mechanism that depends on current error (proportional), the history of errors (integral), and the near-future prediction of errors from the current rate of change (differential). The loop takes an input and attempts to make it equal to a given set point by varying an output. For a discretised system, this can be written:

$$\sigma(t_i) = k_p\epsilon(t_i) + k_i\sum_{j=1}^{i} t_j\epsilon(t_j) + k_d\frac{\epsilon(t_i)-\epsilon(t_{i-1})}{t_i - t_{i-1}} \tag{8}$$

where k_p, k_i, and k_d are gain constants, usually determined iteratively by an algorithm, associated with the proportional, integral, and differential contributions respectively. $\epsilon(t)$ is the error at time t, given by the difference between the set point and the output; and σ is the generated change in output. It is important to note that the presence of noise in a system will render the differential part of the process unstable, and in this case k_d has been set to 0 to reduce the control to a PI loop. Application of a PID autotune library available for use with the Arduino software indicated maximum stability was reached with $k_p = 0.1$ and $k_i = 0.5$.

When the system is activated, the PID loop exhibits a typical overshoot (see Figure 5a), as the loop overcompensates for an undervalued input. To test the long term stability of this PID loop, a long run of flow output readings both with and without the control loop (Figure 5b) was conducted. We see the stability is evidently much improved when the PID loop is used.

3.4.1 Air dampers

Because a diaphragm motor was chosen as the pump, the nature of the flow directly around the motor is roughly sinusoidal (see Figure 6), which leads to uncertainty in the flow. A way around this is to use air reservoirs to smooth the flow. They are placed after the filter and immediately before the pump (to avoid affecting any aerosol. We are limited to a litre capacity by the instrument bay of the UAV. To establish the most effective setup, flow readings were taken for a variety of different reservoirs. The standard devations in the flow rate were then calculated and used as a guide for the effectiveness (see Table 1). Two 500ml reservoirs acting in parallel offered the best result (see Figure 4), and are easier to fit into the instrument bay than a single 1 litre reservoir.

3.5 Particle adhesion

We must make sure that once particles are attached to the substrate, they will not be dislodged. To do this, we consider the van der Waals sphere-to-plane interaction given by[13]:

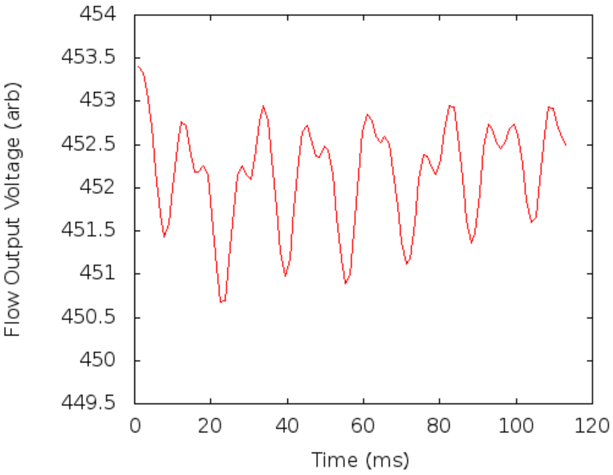


Figure 6: Small scale variations caused in the flow by the diaphragm nature of the pump.

Reservoir capacity	σ_{flow}
250	47.9 ± 3.0
500	18.3 ± 1.3
250 & 500 in series	19.7 ± 1.5
500 & 500 in series	14.2 ± 1.2
500 & 500 in parallel	10.1 ± 2.0

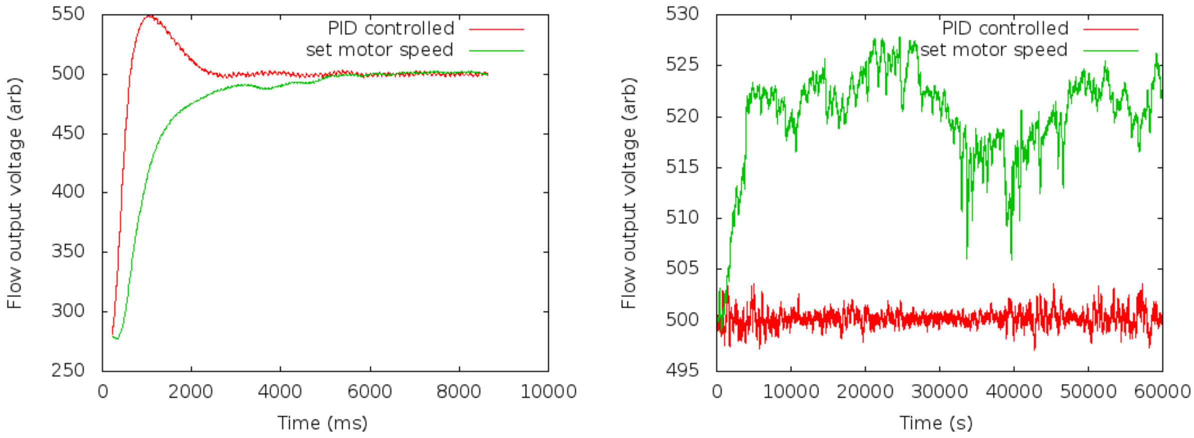
Table 1: Comparison of the standard deviation of the flow (σ_{flow}) when varying reservoirs were used to steady the flow.

$$F_{\text{vdW}} = -\frac{Hr}{6z_0^2} \tag{9}$$

where r is the spherical radius, z_0 is the distance between the plane and the centre of the sphere, and H is the Hamaker constant, roughly 1eV here. Taking a 40nm spherule directly in contact with the substrate, this gives $F_{\text{vdW}} = 7 \times 10^{-13}$N. Treating the spherule as having uniform density 2100kg m^{-3} (i.e. that of graphite), the acceleration required to overcome the van der Waals attraction would be 1.2×10^6m s^{-2}. The particles are firmly attached to the SEM stubs and are unlikely to become dislodged during handling and transport.

3.6 Electron microscopy

Hitachi, manufacturer of the SEM used, suggested a low voltage and no coating (personal communication, Michael Dixon, Hitachi, Apr 2 2013). Sputter coating (e.g. with gold) is normally used to prevent insulators from charging up in the electron beam; such charging makes the specimen appear hazy. However, there was a risk that the coating process could damage the agglomerates, so coating



(a) Start-up stability. Note typical PID overshoot.

(b) Long term stability (over a 17 hour period).

Figure 5: Comparison of flow rates for a PID control loop and a set-speed motor.

was not advised. As the soot particles were composed of carbon, a good conductor, this was not a significant problem. Lower voltage offered better contrast at the expense of poorer resolution.

4 Application

During the project, my supervisor was running experiments on soot aerosols of differing sizes. This provided the ideal opportunity to test the ESP on an aerosol of known size and number density. In addition, the soot aerosol is of particular interest as the complex morphology can be characterised using SEM techniques, demonstrating one of the advantages of an ESP sampler.

4.1 Concentration

The SEM stubs were composed of a 98%Al:2%Mg alloy. Initial inspection raised the concern that this might not have been suitable as the surface appeared speckled under an SEM (see Figure 7).

Carbon, however, has a significantly different atomic number to aluminium and magnesium, and therefore exhibits a significantly greater contrast than the aluminium-magnesium speckling. The stubs were polished to a mirror-finish to reduce surface texture to help identify the particles, but the polishing compound contained diamond powder of the order of a micron; the remaining scratches, although sub-micron in size, were still a hindrance in finding the small carbon particles.

The first two stubs analysed were exposed to an aerosol flow for three seconds; the first to a concentration of 50,000cm^{-3}, the second to a concentration of 30,000cm^{-3}. With the parameters used, these should correspond to 0.6 and 0.4 particles per $10 \times 10\mu$m grid respectively. This turned out to be a concentration too low for meaningful statistics: due to the scratched surface, images had to be examined much more carefully, and the largest grid that could be used for searching was only $4 \times 4\mu$m; i.e. one particle, on average, roughly every 10 frames. With the time we had available to use the SEM, this was insufficient.

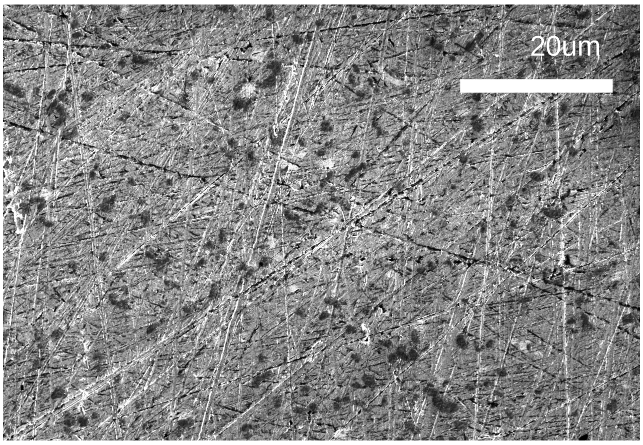


Figure 7: The speckling caused on the SEM stub surface by the presence of magnesium in the aluminium alloy. (Scratches from the polishing procedure are also visible.)

For the 21 frames taken for the first stub, three agglomerates were confirmed (which would correspond to $\sim$ 70,000$\pm$ cm^{-3}). After this, focus moved to finding particles suitable for fractal analysis (e.g. Figure 8).

4.2 Fractal Dimension

4.2.1 Definitions

In the context of a 2D shape (e.g. a projection of a particle onto a plane), the fractal dimension

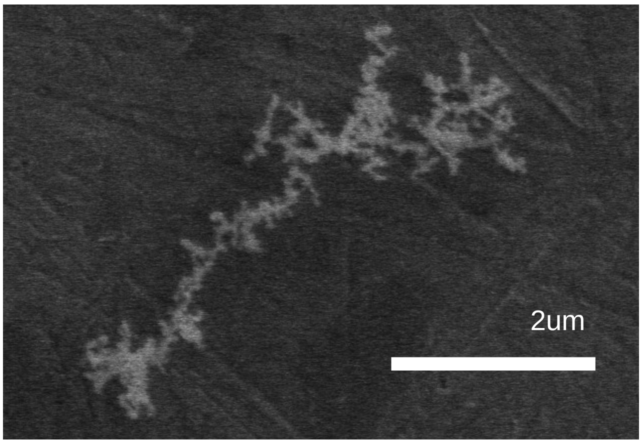


Figure 8: Example of a large agglomerate imaged with the SEM. (See Table 2D.)

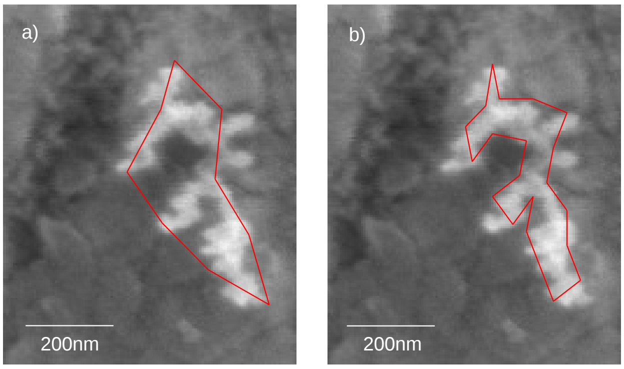


Figure 9: Calculating fractal dimension for a captured soot particle. a) using a step size of 160nm to estimate a perimeter of 1.40μm, ad b) using a step size of 80nm to estimate a perimeter f 1.84μm. Using equation 10, this gives $D_f = 1.39$. (See Table 2E.)

is a measure of complexity derived from the relationship between geometric detail and the scale at which it is observed. This can be conceived as the relationship between measured perimeter, and the length of the step size used to measure it. Formally, given by Hinds (1999)[10]:

$$D_f = 1 - \frac{\log(L_1/L_2)}{\log(\lambda_1/\lambda_2)} = 1 - \frac{\mathrm{d}(\log L)}{\mathrm{d}(\log \lambda)}, \tag{10}$$

where λ is the step size, L is the perimeter, D_f is the fractal dimension, and k is a constant. and subscripts 1 and 2 indicate successive measurements (e.g. Figure 9), which with further measurements can be used to calculate the gradient. Friedlander (2000)[14] showed that fractal dimension can be used to determine the number, N_p of primary particles with diameter, d_p in an agglomerate particle of given fractal dimension, D_f and with a given characteristic radius, R, using:

$$N_p = k_f \left(\frac{R}{d_p}\right)^{D_f} \tag{11}$$

where the constant $k_f \approx 1.7$ is given for typical soot particles by Hinds[10]. k_f is a structure factor often referred to as the *fractal precursor*.[15][16] R is usually taken to be the radius of gyration, R_g[16][17][14][15]. R_g is defined thus:

$$R_g^2 = \frac{1}{N_p}\sum_{i=1}^{N_p}(\mathbf{r}_i - \mathbf{r}_{\mathrm{cm}})^2 \tag{12}$$

where $\mathbf{r}_i$ and $\mathbf{r}_{\mathrm{cm}}$ are the distances of the ith particle and the centre of the mass from a given reference point respectively.

Taking the radius of gyration of the particle in Figure 9 as 350nm, and the primary particle diameter as 30nm, we can estimate $D_f \approx 1.39$ and $N_p \approx 51$.

To automate calculation of fractal dimension, several methods can be used[18]. In the example of Figure 9, we used the *Richardson procedure* (popularised by Mandelbrot after L.F. Richardson's measurements of the British coastline). This is perhaps the most easy to conceptualise, but in reality it is a computationally inefficient method that is difficult to implement.[20] More commonly used is the *mosaic amalgamation method*, also known as the *box counting method* which relies on the definition of fractality as a quantification of a fractal's ability to fill space. That is:

$$D_f = -\lim_{\lambda \to 0} \frac{\log N(\lambda)}{\log \lambda} \tag{13}$$

where λ is the mosaic tile width, and $N(\lambda)$ is the number of tiles containing part of the outline of the fractal. A plot of $\log N\lambda$ against $\log \lambda$ is known as a Richardson plot.

4.2.2 Problems

There are four issues to consider with measuring fractal dimension in this way. One is a property of the way the particles are collected and viewed: the issue of projection,[17][19]; the other three result from the way the image is recorded: resolution[16], noise, and focus.

Projection: Observing soot particles on a stub under an electron microscope offers only a single 2D projection of any given agglomerate. This presents an immediate problem: a

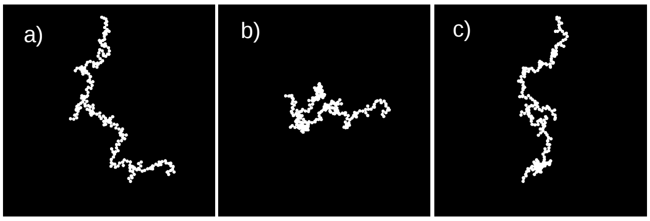


Figure 10: (a) xy plane (b) xz plane (c)yz plane
Projections of a simulated agglomerate ($k_f = 1.7$, $D_f = 1.4$, $d_p = 5$px, $N_p = 220$) onto three orthogonal 640×640 grids.

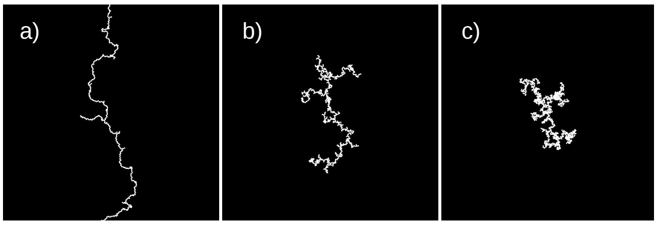


Figure 11: (a) $D_f = 1.2$, $N_p = 400$ (b) $D_f = 1.5$, $N_p = 700$ (c)$D_f = 1.7$, $N_p = 1000$
Large simulated agglomerates with varying fractal dimension for $k_f = 1.5$ and $d_p = 2$px.

3-dimensional sphere of agglomerated particles would have $D_f = 3$, but its projection, a disk, would only appear to have $D_f = 2$. Using fractal agglomeration software[21], 3D fractals with $1 < k_f < 2.15$, $1.1 < D_f < 2.5$, 2px$< d_p <$ 30px, and $10 < N_p < 500$ were generated, these were then projected onto the xy, yz, and zx planes, creating three images, each 640 × 640px. Figures 10 and 11 show resulting examples.

For a true fractal, it can be proven analytically[22] that:

$$D_f^{\mathfrak{L}} = \min\{\mathfrak{L}, D_f^{\mathfrak{D}}\} \tag{14}$$

where $D_f^{\mathfrak{L}}$ is the projected fractal dimension, $\mathfrak{L}$ is the dimensionality of the surface onto which the fractal has been projected (e.g. $\mathfrak{L} = 2$ for projection onto a plane), and $D_f^{\mathfrak{D}}$ is the original fractal dimension. That is to say, were the soot particles true fractals with $D_f < 2$, then the projection would share that characteristic; else we would measure $D_f = 2$. However, the soot particles are not true fractals, and this result does not hold if the primary particles overlap each other [23]. Using the mosaic amalgamation method in ImageJ[24] to calculate fractal dimensions of the projections of simulated particles, an assumption on how those particles attach to the surface is required. The most favourable state maximises contact between the agglomerate and the surface; or in terms of projection, the largest projected area. This results in a lower fractal dimension; Figure 12 shows the relationship between the actual fractal dimension of an agglomerate and the dimension of the projection that has maximal surface area, a least-squares linear fit gives:

$$D_f^2 = (0.450 \pm 0.020) D_f^3 + (0.731 \pm 0.029). \tag{15}$$

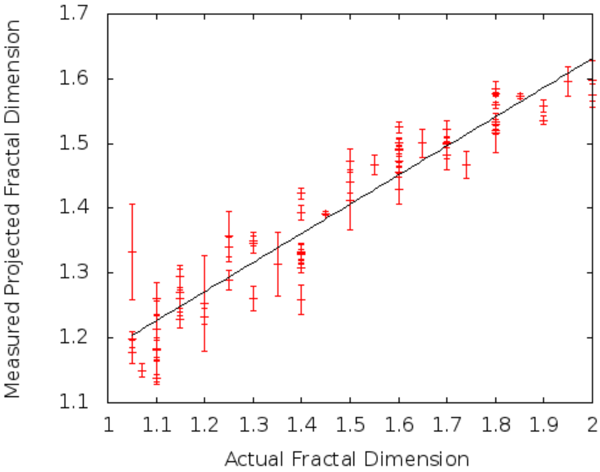


Figure 12: Measurements of projected fractal dimension (D_f^2) of simulated fractals with known D_f^3.

Resolution: Again, as these soot particles are not true fractals ($d_p \neq 0$, $N_p \neq \infty$), there are effects that must be considered in the limits of analysis. At very small scales (e.g. $\lambda < d_p/4$) , one recovers the individual particles and their shape. At vanishingly small scales ($\lambda \to 0$), we would see only a straight line, and would recover the Euclidean result $D_f = 1$. Beyond that, the circular shape and packing would be measured (once $\lambda \lesssim d_p$), in this regime, analysis of the generated fractals (for example, Figure 13) gave $D_f = 1.11 \pm 0.02$; this is referred to as textural fractal dimension. Structural fractal dimension (the quantity in which we are interested) will be measured when $\lambda \gtrsim d_p$. This boundary can be useful in approximating the primary particle size from a Richardson plot (e.g. Figure 13).

At the other end, λ will either become comparable to the particle width, $2r_g$, or the image width, l_{img}, and we would measure D_f incorrectly.

Noise: Even near-zero spatially uniform noise is damaging to fractal dimension estimates obtained using a mosaic method; for example, a single stray pixel away from the particle will trigger box miscounts at all levels (see Figure 14, calculated using a simulated SEM image). However, background

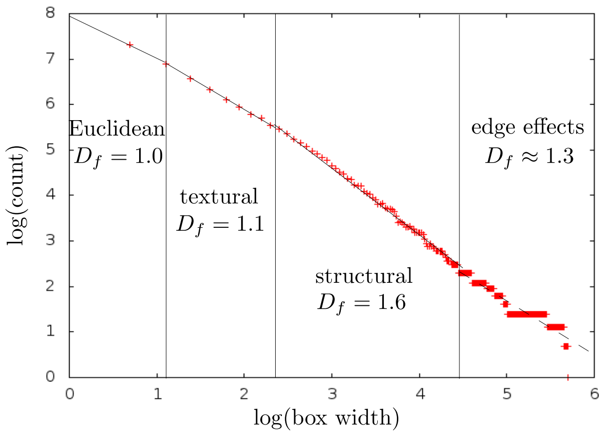


Figure 13: The effect of changing mosaic box size on the determined fractal dimension for an agglomerate with $d_p = 10$. Note the transition between textural and structural at $\sim e^{2.3} \approx 10$.

noise can be very easily removed, and only noise localised to the particle perimeter need be considered. We define the clearance radius as the distance from the particle beyond which all noise is removed, and the noise power, P_N:

$$P_N = \log \frac{A_n}{A_i} \tag{16}$$

where A_n is the number of noisy pixels, and A_i is the image area (in pixels), e.g. for a 640×640px image with a count of $A_n = 1000$ gives $P_N = -6.0$. The relationship between clearance radius and measured structural fractal dimension for two agglomerates and two noise powers is demonstrated in Figure 15. We can see that if we remove noise further away than a few pixels from the particle then D_f gains some uncertainty, but changes very little.

Focus: The effect of defocussing an image can be simulated by applying a Gaussian blur filter[25]. Figure 16 displays the result of applying Gaussian blurs of increasing standard deviations to a given test particle. This process removes small-scale complexity, reducing the fractal dimension.

For an image, the blur width can be approximated by measuring the extent of grey (not white or black) pixels across what should have been a sharp black-white edge (we now refer to this as the extent, ϵ). This can be taken with the measured fractal dimension and calibrated using simulated data (e.g. Figure 17). The relationship between ϵ and the Gaussian standard deviation, σ, can be derived directly from the mathematical definition of the Gaussian, for an 8-bit grayscale:

$$\epsilon = \sigma\sqrt{2 \ln 255} \approx 3.33\sigma. \tag{17}$$

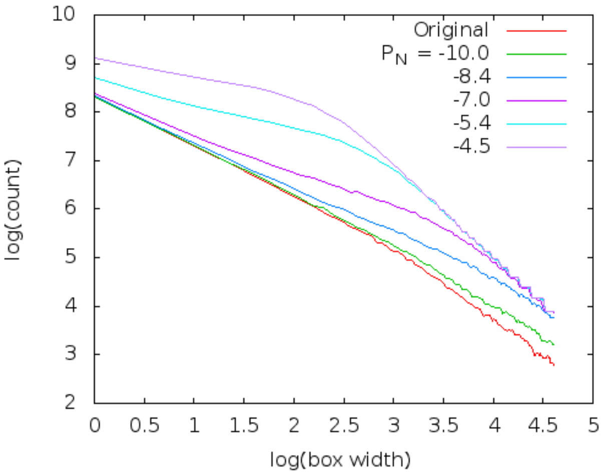


Figure 14: Effect of white noise at varying powers on a 640×640 grid containing a single agglomerate ($d_p = 20$, $N_p = 42$, $D_f = 1.39$).

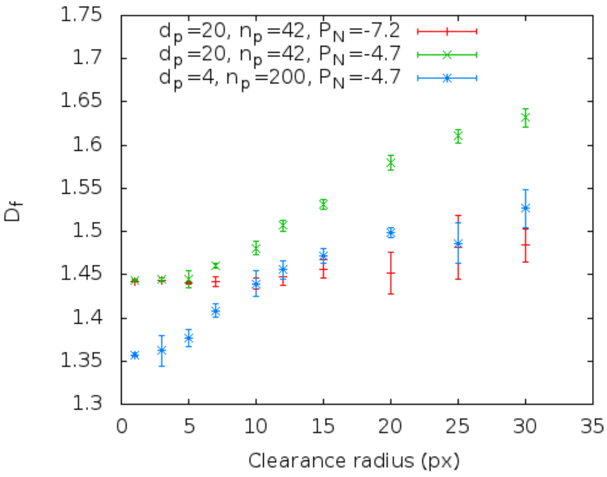


Figure 15: Effect on measured fractal dimension of noise clearance up to a certain radius from the particle. Both particles have $R_g = 200$px, and are on a 640×640 image.

The information displayed in Figure 17 suggests that there is a point ($\sigma \sim 0.7 d_p$) beyond which information about the fractal dimension becomes irretrievable. Using, as an example, Figure 8,with $d_p \approx 30$nm, and image size 640×480px; we have (using equation 17) $\epsilon \sim 10$px. The behaviour of the paths in the example depend slightly on the shape of the fractal (e.g. for varying k_f, or if the projection is not simply connected), but these effects are dominated by variance in D_f. Further, although smaller agglomerates (i.e. lesser N_p) have a steeper gradient than larger ones, this effect is only significant in the limit of very small agglomerates (e.g. $N_p \lesssim 20$).

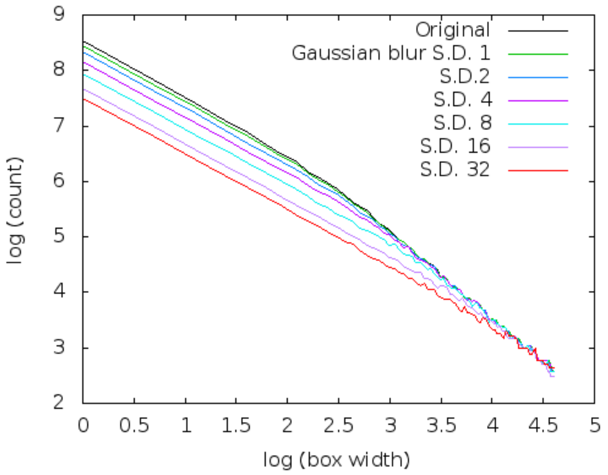


Figure 16: Effect on Richardson plot of a particle ($d_p = 10$) by defocussing with Gaussian blur. $\log 10 \approx 2.5$, so D_f would be measured as the gradient of the line beyond that point.

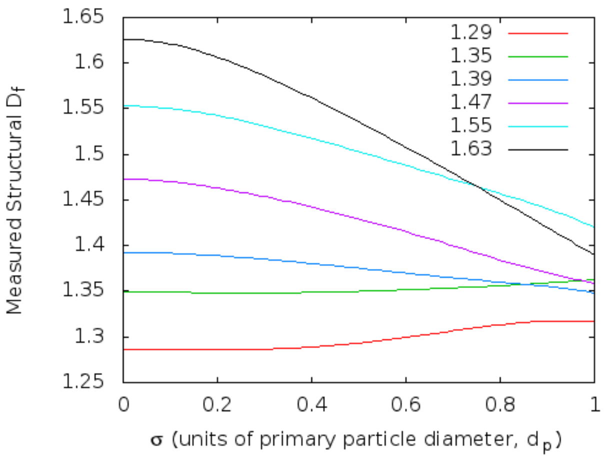


Figure 17: Effect of Gaussian blurs of increasing standard deviation on measured structural fractal dimension for simulated particles (with $200 < N_p < 400$.

4.3 Analysis

4.3.1 Processing

In total, 6 images were suitable for calculating structural fractal dimension. Using imaging software, the images were first smoothened to assist noise reduction; this filter replaces each pixel with the average of its 3×3 neighbourhood[24], roughly equivalent to a Gaussian blur of unit pixel standard deviation. Within the images analysed, 1px $\approx 0.15\sigma/d_p$, and reference to Figure 17 shows that this process will easily keep fractal number within the error given by equation 15. Thereafter, the image was thresholded to a binary (2-bit) image such that the shape of the agglomerate was conserved, then further noise cleared and edges detected. Finally a box counting (mosaic amalgamation) algorithm was used with box widths between $2d_p$ and $W/6$, where W is the length of the lesser of the two edges of the image, or the longest dimension of the particle (whichever was smaller). Results derived from mosaic amalgamation method can be sensitive to the position of the first box, so this was offset three times to ensure that the result was independent (within error) of the starting point. Further, as the noise removal process was performed manually, that too was repeated to ensure consistency.

4.3.2 Results

The images used were sufficiently sharp and (after processing) noiseless that neither defocussing nor noise had to be factored in. For the former, the extent, ϵ, was measured and compared with d_p.

Image	R_g(nm)	D_f^2	D_f^3
A	1106	1.465 ± 0.016	1.631 ± 0.076
B	794	1.495 ± 0.009	1.698 ± 0.070
C	1051	1.481 ± 0.017	1.667 ± 0.077
D	2475	1.505 ± 0.010	1.720 ± 0.071
E	326	1.456 ± 0.032	1.611 ± 0.098
F	1055	1.428 ± 0.013	1.549 ± 0.073

Table 2: Results of analysis on SEM images, including conversion to D_f^3 using equation 15.

These results suggest a mean fractal dimension of 1.65 with variance $3.8{\times}10^{-3}$. This is in accordance with studies by Lapuerta et al. (2006)[23] and Köylü et al. (1995)[17].

5 Conclusion

An ESP was designed for use on board a UAV. Using the specifications given by Miller et al., it was built and subsequently tested in the laboratory using a known aerosol. Resulting stubs were analysed with an SEM, showing that greater exposure time should have been used to allow for statistically significant evaluations of particle concentration. Subsequent investigation involved analysis of the soot agglomerate fractal dimension. Many factors contribute to the uncertainty of this procedure, and each was carefully considered. Of 6 agglomerates analysed in detail, values of $1.43 < D_f^2 < 1.51$ and $1.55 < D_f^3 < 1.72$ were found. More time would have allowed

Further work is needed to better quantify the relationship between the 3D fractal dimension of an agglomerate and its projected 2D fractal dimension; this would significantly reduce the error generated when transforming. Further time would have allowed for testing of the precipitator in a real-world scenario, e.g. pollution in a train station.

Acknowledgements

Extensive thanks must be given to my supervisor, Dan Peters, without whom none of this work would have been possible, and further thanks go to Paul Pattinson for his cheerful lessons on the SEM.

References

[1] Morrow, P.E., Mercer, T.T. (1964). A Point-to-Plane Electrostatic Precipitator for Particle Size Sampling. *American Industrial Hygiene Association.* **25**, 8-24

[2] Miller, A., et al. (2010). A Handheld Electrostatic Precipitator for Sampling Airborne Particles and Nanoparticles. *Aerosol Science and Technology.* **44**(6), 417-427.

[3] Fierz, M., Kaegi, R., and Burtscher, H. (2007) Theoretical ad Experiental Evaluation of a Portable Electrostatic TEM Sampler. *Aerosol Science and Technology.* **41**(5), 520–528.

[4] Buckley, R.L., and Loyalka, S.K. (1989) Cunningham Correction Factor and Accommodation Coefficient: Interpretation of Millikan's Data. *Journal of Aerosol Science.* **20**(3) 347-349.

[5] QuestUAV. Available: `http://questuav.co.uk/quest300.php`. Last accessed 18 Mar 2013.

[6] HVM Technologies. Available: `http://hvmtech.com/pdf/SMHVSpecSheet.pdf`. Last accessed 8 Apr 2013.

[7] Jackson, J.D. (2000). Charge Density on a Thin Straight Wire, Revisited. *American Journal of Physics.* **68**(9)), 789.

[8] Honeywell Sensing. Available: `http://sensing.honeywell.com/product%20page?pr_id=25734`. Last accessed 4 Mar 2013.

[9] Sensidyne. Available: `http://www.sensidyne.com/air-sampling-equipment/calibration-equipment/gilibrator-2.php`. Last accessed 8 Apr 2013.

[10] Hinds, William C. (1999). Aerosol Technology, Properties, Behavior, and Measurement of Airborne Particles. *John Wiley & Sons*

[11] Tyndall, A.M. (1938). The Mobility of Positive Ions in Gases. *Cambridge University Press.*

[12] Rigden, John S. (1996). Macmillan Encyclopedia of Physics. *Simon & Schuster*, p353.

[13] Gady, B., et al. (1996). Identification of Electrostatic and van der Waals Interaction Forces Between a Micrometer-Size Sphere and a Flat Substrate. *Physical Review B.* **53**(12) 8065-8070.

[14] Friedlander, Sheldon K. (2000) Smoke, Dust, and Haze Fundamentals of Aerosol Dynamics Topics in Chemical Engineering. *Oxford University Press.* 2nd Edition.

[15] Gwaze, P., et al. (2006) Comparison of Three Methods of Fractal Analysis Applied to Soot Aggregates From Wood Combustion. *Journal of Aerosol Science.* **37** 820-838.

[16] Sorensen, C.M., and Roberts, G.C. (1996) The Prefactor of Fractal Aggregates. *Journal of Colloid and Interface Science.* **186** 447-452.

[17] Köylü, Ü.Ö., et al. (1995) Fractal and Projected Structure Properties of Soot Aggregates. *Combustion and Flame.* **100**(4) 621-633.

[18] Kaye, B.H. (1994) A Random Walk through Fractal Dimensions. *Wiley-VCH.* 2nd Edition.

[19] Kaye, B.H., Leblanc, J.E., ad Clark, G.C. (1984) A Study of the Physical Significance of Three-Dimensional Signature Waveforms. *Particle Systems Characterization* **1** 59-65.

[20] Russ, J.C. (1994) Fractal Surfaces *Plenum Press.*

[21] Smith, A.J.A. (2011) Microphysical Modelling of Aerosols in the ORAC Retrieval. *D.Phil. Thesis.* University of Oxford, U.K.

[22] Falconer, K. (2003) Fractal Geometry, Mathematical Foundations and Applications. *Wiley.* 2nd Edition.

[23] Lapuerta, M., Ballesteros, R., and Martos, F. (2006) A Method to Determine the Fractal Dimension of Diesel Soot Agglomerates. *Journal of Colloid and Interface Science* **303**. 149-158.

[24] ImageJ. Available: `http://rsb.info.nih.gov/ij/`. Last accessed 1 Apr 2013.

[25] Hecht, E. (2003) Optics. *Pearson.* 4th Edition.

[26] Cheng, Y. S., Yeh, H. C., et al. (1981). Collection Efficiencies of a Point to Plane Electrostatic Precipitator. *American Industrial Hygiene Association.* **42**(8), 605-610.

A Efficiency

To estimate the greatest flow speed that could theoretically guarantee 100% efficiency, we use equation 1. There are two effects that will dictate the eventual charge attached to a particle: *diffusion charging* and *field charging*.

Diffusion charging is caused by Brownian motion inducing random collisions between ions and the aerosol: charge number as a function of time is given by Hinds (1999)[10]:

$$n(t) = \frac{d_p kT}{2Ke^2} \ln \left[1 + \frac{\pi d_p K \bar{c_i} e^2 n_i t}{2kT} \right] \tag{18}$$

where d_p is the particle diameter, $\bar{c_i}$ is the average thermal ion speed, $K_E = 1/4\pi\epsilon_0$ and n_i is the ion concentration.

Field charging is caused by interaction between the aerosol and the ion current following the field lines, charge number as a function of time is again given by Hinds:

$$n(t) = \left(\frac{3\epsilon}{\epsilon + 2} \right) \left(\frac{Ed_p^2}{4Ke} \right) \left(\frac{\pi K e \mu_i n_i t}{1 + \pi K e \mu_i n_i t} \right) \tag{19}$$

where $\epsilon \approx 10$ is the relative permittivity of black carbon, our test aerosol, and μ_i is the electrical mobility of dissociated molecular nitrogen, $\mu_i(N_2^+) = 1.8 \times 10^{-4} \text{m}^2\text{V}^{-1}\text{s}^{-1}$[11].

We now need to estimate n_i. Taking the point to plane distance as l, and the effective cross-sectional area of a cone as πw^2 (w is the radius of a cylinder with the same length and volume as the cone), we can calculate the drift velocity, v_{drift} for E $= 1.2 \times 10^6 \text{Vm}^{-1}$[2] with:

$$v_{\text{drift}} = \mu_i E. \tag{20}$$

Then, using:

$$Q = \frac{Il}{v_{\text{drift}}} \tag{21}$$

and:

$$en_i = \frac{Q}{l\pi w^2} \tag{22}$$

we get:

$$n_i = \frac{I}{eE} \frac{1}{\pi w^2 \mu_i}. \tag{23}$$

Putting values in ($T = 300$K, $c_i = 240\text{ms}^{-1}$[10], $n_i = 1.77 \times 10^{16}\text{m}^{-3}$, I $= 2\mu$A, $w = 1.7$mm), we get, for equation 18:

$$n_{\text{diff}}(t) = (9 \times 10^6) d_p \cdot \ln \left[1 + 3.7 \times 10^{11} (d_p t) \right] \tag{24}$$

and, for equation 19:

$$n_{\text{field}}(t) = (4.25 \times 10^{14}) \underbrace{\left(\frac{12010t}{1 + 12010t} \right)}_{\to 1 \text{ very rapidly}} d_p^2. \tag{25}$$

We can then evaluate these for a few particle sizes to get a feel for the competing effects (see Table 3).

d_p	n_{diff}	n_{field}	C_c	$v_{\text{elec}}(\text{ms}^{-1})$
1μm	115	425	1.15	0.70
300nm	31	38	1.53	0.40
100nm	9	4	2.87	0.42
30nm	3	0	7.93	0.88
10nm	1	0	22.6	2.56

Table 3: Effects of field and diffusion charging for one second on particles of varying diameters, and the resulting velocity.

By combining equations 24 and 25 with equation 1, a theoretical efficiency curve for these parameters can be obtained. Figure 18 shows an overlay of this curve on experimental data from Miller et al.

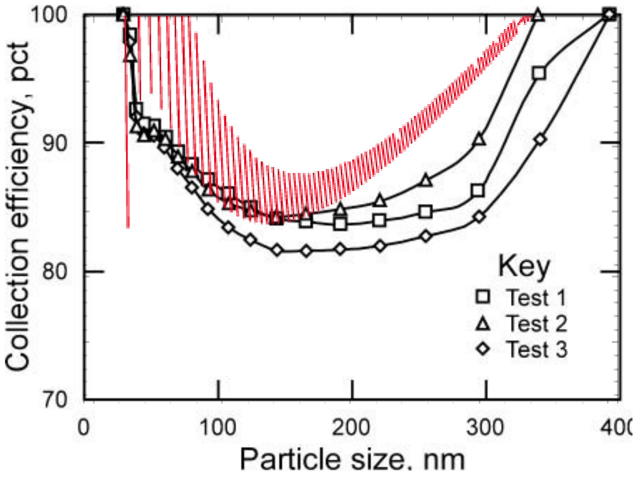


Figure 18: Theoretical efficiency curve (red) with same parameters as experimental data from Miller et al.[2] (black).

B Arduino Software

B.1 Remote Control Activation

```
int servoIn = 0; //digital pin 2 corresponds to interrupt 0//
float t =0; //length of peak in ms//
volatile int i =0; //counter//
volatile int counterarray[50];
volatile long timearray[50];
int waiting = true;
int t_req = 0; //insert required t for turn on (micros)//
int t_lenience = 0; //insert allowed error in t_req allowed for turn on//
long eventally = 0;
long oddtally = 0;
void setup(){
  attachInterrupt(servoIn , PWMread, CHANGE);
};

void PWMread(){
  counterarray[i] = i;
  timearray[i] = micros();
  i++;
}

void loop(){
  while(waiting){
    if (i=50){
      i = 0;
    };
    for(int j=0; j<50; j++){
      if(j%2){
        eventally = eventally + timearray[j];
      }
      else{
        oddtally = oddtally +timearray[j];
      }
    }

    t = (oddtally - eventally)/25;
    if (abs(t-t_req) <= t_lenience){
      waiting = false;
      Serial.println(”t=”);
      Serial.print(t);
    }
  }


  if(!waiting){
    //start main program
  }
}
```

B.2 Main Control

```
//invite PID and average libraries//
#include <PID_v1.h>
//serial read//
int serialData = 0;
//Define board in use, choose UNO or DUE//
char BOARD_IN_USE[] = ”UNO”;
//time constant of RC circuit for PWM smoothing//
```

```
//Assign pins on board//
const int pumpSpeed = 3;
const int pumpDirection = 12;
const int precipitator1 = 6;
const int precipitator2 = 7;
const int precipitator3 = 8;
int HVlevel;
const int flowMeter = A4;
const int AVmon1 = A1;
const int AImon1 = A2;
const int AVmon2 = A3;
//const int AImon2 = A4;//
const int AVmon3 = A5;
const int AImon3 = A6;
//define turn on times, delay, inital pumpspeed etc//
int t_prec1 = 3000; //time between prec1 off and prec2 on//
int t_prec2 = 3000;//time between prec2 off and prec3 on//

int pause = 0; //sampling time//


//set up PID//
double Setpoint, Input, Output;


PID motorPID(&Input, &Output, &Setpoint, 0.1, 0.5, 0, DIRECT);

/*define stats collection function to return average Imon, Vmon
 pumpSpeed, flowMeter*/
int Imon1, Imon2, Imon3, Vmon1, Vmon2, Vmon3;


void collect(int Imon, int Vmon, double Output, double Input){
  unsigned long ImonTot = 0, VmonTot = 0, pumpSpeedTot =0, flowMeterTot=0;
  float ImonStdev, VmonStdev, pumpSpeedStdev, flowMeterStdev;
  unsigned long ImonSq=0, VmonSq=0, pumpSpeedSq=0, flowMeterSq=0;
  float ImonAvg, VmonAvg, pumpSpeedAvg, flowMeterAvg;
  unsigned long now = millis();
  int i=0;
  while(millis()-now <pause){ //summing for averages//

    runPID();
    ImonTot += Imon;
    VmonTot += Vmon;
    pumpSpeedTot += Output;
    flowMeterTot += Input;
    //summing squares for stdev calc//

    ImonSq += sq(Imon);
    VmonSq += sq(Vmon);
    pumpSpeedSq += sq(Output);
    flowMeterSq += sq(Input);
    i++;
    delay(10);
    Serial.print(Output);
    Serial.print(”\t”);
    Serial.print(pumpSpeedTot);
    Serial.print(”\t”);
    Serial.print(sq(Output));
    Serial.print(”\t”);
    Serial.println(pumpSpeedSq);

  }
  int finish = millis()-now;
  //calculate averages and stdevs//
  ImonAvg = ImonTot/i;
  VmonAvg = VmonTot/i;
  pumpSpeedAvg = pumpSpeedTot/i;
  flowMeterAvg = flowMeterTot/i;

  ImonStdev = sqrt(ImonSq/i - sq(ImonAvg));
  VmonStdev = sqrt(VmonSq/i - sq(VmonAvg));
  pumpSpeedStdev = sqrt(pumpSpeedSq/i - sq(pumpSpeedAvg));
  flowMeterStdev = sqrt(flowMeterSq/i - sq(flowMeterAvg));

  //print results to serial//
  Serial.println(i);
  Serial.println(pumpSpeedSq);
  Serial.println(sq(pumpSpeedAvg));


  Serial.print("Imon:");
  Serial.print("\t");
  Serial.print("avg:");
  Serial.print(ImonAvg);
  Serial.print("\t");
  Serial.print("stdev:");
  Serial.println(ImonStdev);

  Serial.print("Vmon:");
  Serial.print("\t");
  Serial.print("avg:");
  Serial.print(VmonAvg);
  Serial.print("\t");
  Serial.print("stdev:");
  Serial.println(VmonStdev);

  Serial.print("pumpSpeed:");
  Serial.print("\t");
  Serial.print("avg:");
  Serial.print(pumpSpeedAvg);
  Serial.print("\t");
  Serial.print("stdev:");
  Serial.println(pumpSpeedStdev);

  Serial.print("flowMeter:");
  Serial.print("\t");
  Serial.print("avg:");
  Serial.print(flowMeterAvg);
  Serial.print("\t");
  Serial.print("stdev:");
  Serial.println(flowMeterStdev);



}




void runPID(){
  Input = analogRead(flowMeter);
  motorPID.Compute();
  analogWrite(pumpSpeed, Output);
}

void setup(){
    Serial.begin(9600);
  if(BOARD_IN_USE == ”UNO”){
    HVlevel = 9;
  }
  else{
    HVlevel = 0;//needs to reference DAC0, check syntax
  };
  pinMode(pumpSpeed, OUTPUT);
  pinMode(pumpDirection, OUTPUT);
  pinMode(precipitator1, OUTPUT);
  pinMode(precipitator2, OUTPUT);
  pinMode(precipitator3, OUTPUT);
  pinMode(HVlevel, OUTPUT); //not sure of DAC syntax, will check//
  motorPID.SetMode(AUTOMATIC); //turn PID on//
  //Set internal reference to 1.1V//
  analogReference(DEFAULT);

};


void loop(){
  runPID();


  if(Serial.available()){
    runPID();

    serialData = Serial.read();
    switch(serialData){

    case ’P’: //P for pump//
      Setpoint = Serial.parseInt();
      Serial.print(”Pump on: ”);
      Serial.println(Setpoint);
      break;

    case ’H’: //H for high voltage supply//
      HVlevel = Serial.parseInt();
      Serial.println(”HV on”);
      break;

    case ’Q’: //Q for pump off//
      Setpoint = 0;
      Serial.println(”Pump off”);
      break ;

    case ’X’: //X for HV off//
      HVlevel = 0;
      Serial.println(”HV off”);
      break;

    case ’A’: //A turn on esp 1, etc.//
      pause = Serial.parseInt();
      digitalWrite(precipitator1, HIGH);
      collect(Imon1, Vmon1, Output, Input);
      digitalWrite(precipitator1 , LOW);
      break;

    case 'B':
        pause = Serial.parseInt();
      digitalWrite(precipitator1 , HIGH);
      collect(Imon1, Vmon1, Output, Input);
      digitalWrite(precipitator1 , LOW);
      break;

    case 'C':
        pause = Serial.parseInt();
      digitalWrite(precipitator1 , HIGH);
      collect(Imon1, Vmon1, Output, Input);
      digitalWrite(precipitator1 , LOW);
      break;
    }
  }
}
```