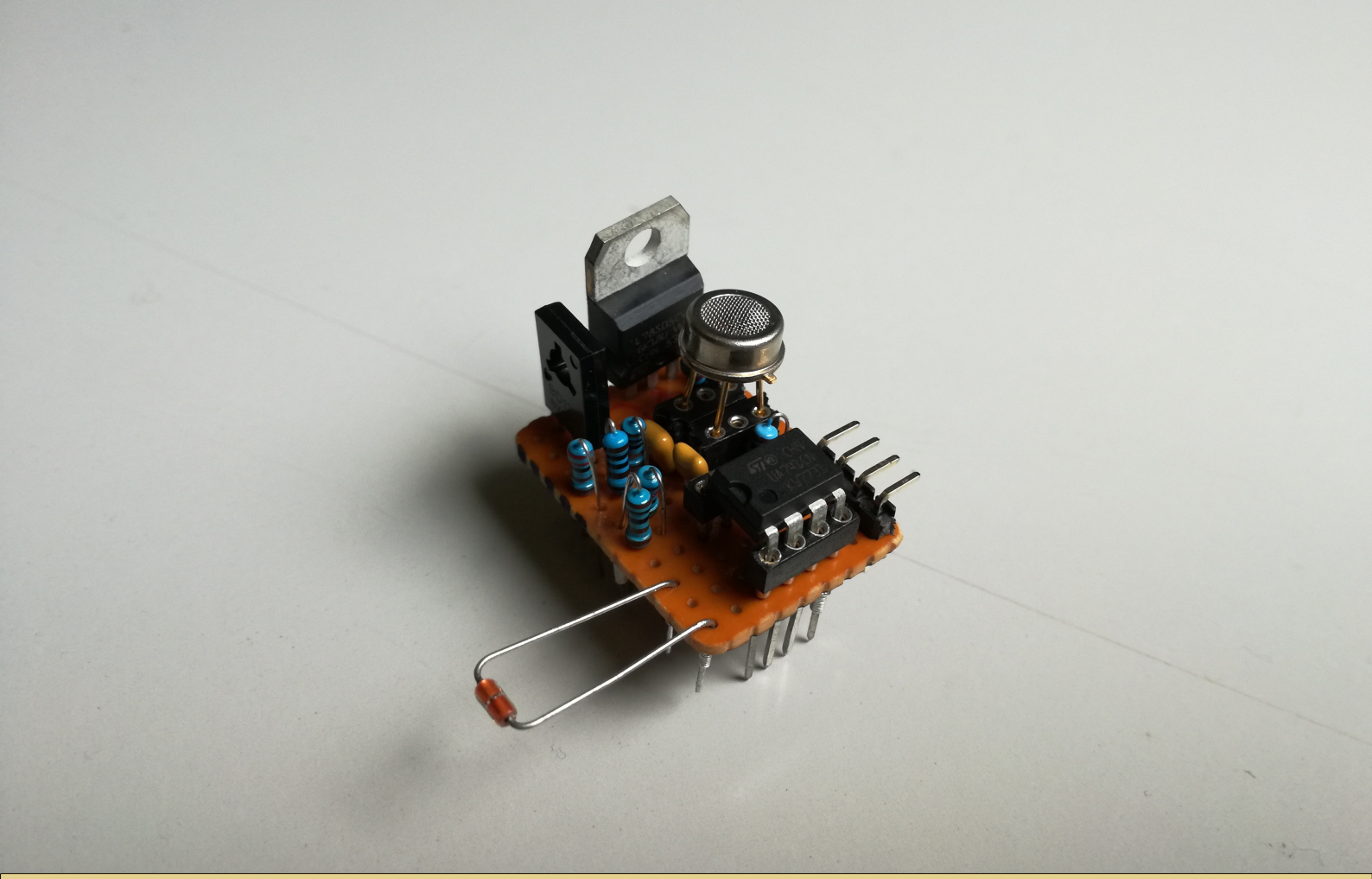


Eduardo Évora Domingues

Experiments on One-Dimensional Data Assimilation

Thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science in Electrical and Computer Engineering

February, 2017

· U C ·

Universidade de Coimbra

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Experiments on One-Dimensional Data Assimilation

Eduardo Évora Domingues

Coimbra, Fevereiro 2017

Experiments on One-Dimensional Data Assimilation

Supervisor:

Professor Lino Marques

Jury:

Professor Henrique José Almeida da Silva (President)

Professor Gabriel Falcão Paiva Fernandes (Vogal)

Professor Lino José Forte Marques (Supervisor)

Dissertation submitted in partial fulfillment for the degree of Master of Science in Electrical and Computer Engineering.

Coimbra, Fevereiro 2017

Acknowledgements

I would like to express my gratitude towards all the people that helped me finish this stage. I would like to thank my supervisor, Professor Lino José Forte Marques, for all the support and for all the ideas that I got during the time I was working in this dissertation.

I would also like to thank all the professors that thought me all the things that will make me an engineer. Studying in this department was a great and memorable experience.

To my parents, I thank all the support they gave me, and all the effort and time they spent in my education. They are the reason I got this far. Also, I thank my brother for all the time we shared through our lives.

I thank also all my friends that I made over the years for the time spent together and for sharing a good laugh from time to time. Specially my lab colleagues João Natividade e José Caetano for discussing our ideas and for the advices they gave me. I thank João Pereira for reviewing this document and giving some feedback.

Last, but not least, I thank Daniela Jordão for being by my side and following this path together. For listening to me and for helping me with her advices, and for being my best friend, thank you.

Resumo

Esta dissertação estuda o problema da assimilação de informação em ambientes fluidos, com resultados experimentais para a dispersão de odor no ar. Problemas relativos a assimilação de informação em fluidos são utilizados frequentemente em meteorologia e na monitorização de poluição ambiental. Mas nesta dissertação, o objetivo foi inferir medidas experimentais num mapa de concentração de odor, unidimensional.

Com vista a resolver este problema, foram realizadas em laboratório, algumas experiências em condições semelhantes às de um ambiente real, com sensores esparsos, e essas mesmas condições foram simuladas num computador. Com isto, os resultados experimentais foram usados para corrigir as simulações, o que levou a algoritmos capazes de estimar os valores para as variáveis de interesse ao longo de uma malha fina de pontos, com precisão. As bases para as simulações e algoritmos desenvolvidos foram métodos de diferenças finitas para equações com derivadas parciais, modelando os fenómenos físicos associados. A implementação dos métodos de diferenças finitas foram feitos usando matrizes esparsas, o que permite tornar mais eficiente o cálculo das variáveis desejadas. O trabalho desenvolvido incluiu a construção e teste dos anemómetros, tal como os módulos unidimensionais de compasso de odor utilizados para as medições finais, que são usadas para os resultados apresentados. Por último, são apresentadas comparações entre o método estudado neste trabalho e outros que podem ser utilizados para o mesmo objetivo, tais como interpolação linear e ajuste polinomial. É de realçar que uma das maiores vantagens deste método é o facto de envolver modelos matemáticos capazes de descrever os fenómenos físicos envolvidos.

Palavras Chave: Assimilação de informação, Diferenças finitas, Equações de advecção-difusão, Monitorização de poluição, Medição de odor

Abstract

This dissertation addresses the problem of data assimilation in fluid environments, with experimental results for odour dispersion in air. Problems regarding data assimilation in fluids are often used for weather forecasting and ambient pollution monitoring. But in this dissertation, the objective was to infer experimental measurements in a, one-dimensional, odour concentration map.

To tackle this problem, some experiments were assembled in a laboratory to mimic the conditions of a physical environment with sparse sensors, and similar conditions were simulated in a computer. Then the experimental results were used to rectify the simulations, which lead to algorithms capable of estimating the values for the variables of interest across a fine mesh of points, accurately. The bases for the developed simulations and algorithms were finite differences methods for partial differential equations, which model the real physical phenomenons. The implementation of the finite differences methods was done using sparse matrices, which results in a more efficient computation of the required variables. The used anemometers were built and tested as part of the work done, as well as the one-dimensional odour compass modules used to get the final measurements, that are used for the shown results. Lastly, one presents the comparison between the method studied in this work and some others that can be used for the same objective, like linear interpolation and polynomial fitting. Note that one of the biggest advantages of this method is the fact that it involves mathematical models, that describe the real physical phenomenons involved.

Keywords: Data Assimilation, Finite Differences, Advection-diffusion equations, Pollution monitoring, Odour sensing

"It is trite to regard turbulence as the last unsolved problem in classical physics and to cite many books and authorities to justify the opinion. It is likewise a cliché to list great physicists and mathematicians, such as Werner Heisenberg, Richard Feynman, and Andrei Kolmogorov, who "failed" to solve the problem despite much effort. Horace Lamb and others have been credited with wishing to seek heavenly wisdom on the subject when they arrived in heaven. With such lists and stories, youngsters are cautioned, directly and indirectly, that turbulence is beyond reasonable grasp."

— Gregory Falkovich and Katepalli R. Sreenivasan,
Lessons from Hydrodynamic Turbulence

Contents

List of Acronyms

PGM	Probabilistic Graphical Model
Kn	Knudsen number
Re	Reynolds number
RANS	Reynolds-averaged Navier–Stokes equations
NWP	Numerical Weather Predition
BLUP	Best Linear Unbiased Prediction
HMM	Hidden Markov Methods

List of Figures

1 Introduction

A single sensor can only acquire data for a single point in space at a given time instance. So the problem is how one is able to estimate the distribution of a variable, that varies in space and time, from a limited set of sparse sensors. This is important because spreading enough sensors along a field to create a good map is rather impractical and often very expensive. The goal is then to develop an algorithm capable of integrating a smaller amount of measurements into a detailed map. Problems like this are usually found in weather prediction, pollution analysis, and other fields not necessarily related to fluid dynamics. In this regard, there are already some solutions to such problems which may involve interpolation [1], probabilistic methods [2, 3, 4], or numerical methods [5] that approximate solutions to the mathematical models. The idea of using observations of the actual system to incorporate them into the mathematical model is referred to as data assimilation [6, 7, 5].

That said, this dissertation has as objective the development and implementation of a method capable of measuring and integrating a wind field like usually done in weather prediction, and an odour field to estimate the concentration of a given chemical across a space interval. But in this case, the end goal is not to study the dynamics of air, but rather the behaviour of a gaseous plume. Or maybe even to estimate the size and position of new obstacles that may, in some way, interfere with the flow. To achieve this goal, a finite difference method was used to model the Navier-Stokes equations and an advection-diffusion equation (Chapter 3). To test the concept, an easier test model was considered, consisting in the use of a diffusion equation in one dimension, and a solid material to measure the dissipation of heat by conduction. After achieving the expected results, the goal was to model the Navier-Stokes equations for one dimension and use a wind tunnel to measure the wind speed in multiple points in order to use those measurements to infer on the numerical model. The final model refers to an advection-diffusion equation and was also tested on a single dimension, using a wind tunnel with a dispenser of high concentrations of a measurable gas. The measurements were introduced in the solution as known values, which means that

at the measured points, the output of the algorithm is equal to the measurements. And for every non-measured point, its value is calculated with respect to the closest measured values and the considered mathematical models.

In order to expose this work in an understandable way, this dissertation begins with a brief explanation of some of the work that has been done by other research projects in this field of study (Chapter 2). Then a short theoretical background (Chapter 3) is provided. After listing the used methods (Section 4.1) and hardware components (Section 4.2), it was important to explain what was done in this work in a detailed and reproducible way (Chapter 5). With all the gathered data, the results were then plotted and analysed (Chapter 6). Finally, this document draws some conclusions (Chapter 7) about the work done and the obtained results.

With the work done it was possible to develop an algorithm capable of integrating one-dimensional values for odour concentrations, in the mathematical model of the dissipation of a gas through moving air. The results have potential, and adding to the usefulness of the developed algorithm in one dimension, it can be extrapolated to a two-dimensional space and serve as a mapping algorithm for multiple situations of environmental sensing through sensor networks. That said, the results also showed that the use of the Navier-Stokes equations for this one-dimensional study is not relevant, because in a narrow tube the air behaves like a very stiff elastic and the velocity is almost always constant along that dimension. However, the developed algorithm for advection-diffusion problems can be used in its current form to map the concentration of chemical substances in constrained fluid channels like ventilation ducts, water pipelines and rivers, which all behave like one dimensional problems. In these cases there are some factors that can introduce error, such as a river's irregular margin and tributaries which may induce vortices in the flow that need to be accounted for in the algorithm to maintain a low regional error.

2 Transport Mapping Solutions

In a macroscale, mathematical models such as the Navier-Stokes equations and advection-diffusion equations or Euler-Lagrange equations are often used to map and predict the flow speed patterns and the transport of particles, chemicals or heat in fluids. Models like these are often used to help predict the weather or study the concentration of chemicals like carbon dioxide in the atmosphere [8], which involves estimating their sources and sinks. Another very important field of study that uses mathematical models like these is oceanography [5], where complex methods are needed to model the maritime currents and some very important things that they can carry, like some chemical compounds or heat. These very common applications require some, of a wide range of deeply studied methods, like Bayesian estimations, Green's functions, the Kalman filter or least-squares fitting by explicit solution of extremal conditions. An increasingly more important application for these methods that can be modelled by said equations, is the study and control of air pollution [9], where adding to the referred methods, the Gaussian plume is also very commonly used. One application example would be the use of autonomous vehicles as mobile sensors [10] for optimal data collection in the ocean, meaning that a bigger area can be monitored with great accuracy by combining measurements taken at different positions and instances by the same sensor.

On smaller scales, like an urban environment, the method's precision can be greater, what allows for more intensive studies regarding pollution, like the toxic gases emitted by vehicles on the streets, which can also be done using a log-linear regression model or a Probabilistic Graphical Model (PGM) [2]. Some times, it may be advantageous to use a simple method that still follows the standard behaviour of a substance in a fluid, which is done by considering the odour cloud as being a Gaussian plume and estimating the plume that best fits the taken measurements, but that is in some cases not applicable, because the air can behave in more complex patterns as in turbulent flow, and the source of odour can be inconsistent and/or moving through space, which requires methods capable of modelling more complex situations. In terms of past work, several methods have been studied and

tested to solve these problems. Therefore, it is possible to find some scientific documents about such studies.

One of the possible ways of mapping a plume of odour is to use hidden Markov methods as done in [3] where the aim was to map the possible locations of a source of a given substance using an autonomous vehicle operating in a fluid flow and hidden Markov methods as the mathematical tools to reach that goal. These are statistical methods that rely on the use of Markov chains with unobserved states, since the process is not visible, but only its output. To put this in the perspective of the odour mapping problem, the model that guides the dispersion of the odour through the air is considered as unknown and the only usable information is that of the taken measurements, so the states of the Markov chains that lead to the measured values are unknown, but these methods focus on the estimation of the probability related to which state leading to the measurement. This dissertation focused only on the mapping problem, which is the main problem that is going to be addressed in this dissertation, but despite that, it is impractical to compare the two projects because one uses a mobile sensor in a two dimensional area and the other uses fixed sensors in a one-and-a-half-dimensional region.

Another statistical approach is to consider Bayesian inference methods like was done in [4] which aimed to estimate a likelihood map for the location of a source of a chemical plume using an autonomous vehicle as a mobile sensor, and Bayesian inference methods to process the data and estimate de map. These methods consist in the use of Bayes' theorem to update the probability of a considered hypothesis as more measurements are taken and more information is considered. Bayesian inference methods were used to estimate, by continuous updates, the probabilities of the odour source being in the considered regions of the area of interest. Regardless of also using a mobile robot to take the measurements, this does not imply that the robot must reach the source and can more easily be tweaked to use multiple fixed sensor instead of a mobile one. This could also be implemented in one dimension but without much benefits, since it was developed to only estimate the most likely sources of odour and not to estimate the full odour concentration map.

All the referred methods can be used for smaller scales, but once considering enclosed environments, the physical obstacles to the flow become a dominant factor, and it becomes important to take them into account. One of the simplest ways to do that is to previously measure all the known obstacles and to introduce them into the used method as boundary conditions or some sort of assumptions, like in [11], where an interpolation and extrapolation method is used with the objective of locating a chemical source in a field with a small

number of sensors scattered around an area using eight assumptions related to the propagation of chemical particles in air and the position of eventual obstacles for the airflow. The applied method is extensive, but simpler to understand and could achieve good results non the less. The downside of this method is that for each added assumption the method becomes more and more restrictive, and that may result in an algorithm that cannot deal with unexpected conditions. However, this two-dimensional method can be adapted to consider each streamline of flowing gas as a one dimensional problem that can be solved with the method developed in this dissertation.

3 Theoretical Background

To fully understand the tools required for this work to be implemented, a theoretical introduction is needed in what regards some fluid dynamics concepts, the Navier-Stokes and advection-diffusion equations. So, in what concerns fluid dynamics, some key concepts are going to be explained as they are very important to fully understand the physics behind what was done in this work and what is that we want to achieve, as well as the fact that fluid dynamics is not a well debated theme around standard electrical engineering courses.

Despite the fact that this work is going to focus on gases in particular, this theoretical background is general enough to be considered in a liquid study around the same problems that are tackled in this one, as the difference in density is not relevant to the formulation of the fluid motion and behaviour since it only has influence over the magnitudes of forces required to produce given accelerations. The largest difference between the properties that can influence the motion of gases and liquids is their capacity to be compressed, but since the velocities in study are quite low (below a third of the velocity of sound [12, 13]), the impact of the differences in density through space is very small and can be ignored [14].

To accurately model the behaviour of a fluid, some basic assumptions and principles about fluid dynamics are needed. These assumptions and principles are turned into equations that must be satisfied if the assumptions are to be held true. The first assumption being that the fluid can be regarded as a continuum substance, and not as a large group of individual particles. For this to still result in an accurate model, the representative physical length scale needs to be much larger than the molecular mean free path, meaning, the areas through which the fluid flows must be much wider than the mean space that each particle moves between consecutive collisions [13]. To the ratio of mean free path representative of the physical length scale we call the Knudsen number (Kn) (named after the Danish physicist Martin Knudsen),

$$Kn = \frac{\lambda}{L},$$

where λ represents the mean free path and L the representative physical length scale.

For this work, the smallest bounded space considered where air can move has a diameter of 10.5 centimetres. Considering the mean free path to be $68 \cdot 10^{-9}$, we get a Kn of $6.476 \cdot 10^{-7}$. Given that this number is much lower than 0.01, the fluid can be regarded as a continuum [15, 16].

The second assumption is that the curves for pressure, velocity, density and temperature are, at least, weakly differentiable [12]. And the basic principles from which the needed equations can be derived are the conservation of mass, momentum, and energy.

One way to derive the conservation laws would be to consider a given quantity of matter or a control mass and its respective proprieties. But to study the dinamics of fluids this is not a practical method because it is difficult to track a blob of matter trough a free flowing environment. It is, then, more convenient to consider a given control volume fixed in space trough which the matter flows [12, 17].

First, let us start with the mass equation, which can neither be created nor destroyed, so for any amount of matter its mass is always constant. However, that is not true for any given volume, since mass can enter and exit said volume, so one needs to consider both the mass inside the volume and the mass entering or exiting that space. To do that, let us consider the Equation 3.1:

$$\frac{\partial}{\partial t}\int_{\Omega} \rho d\Omega + \int_{S} \rho\nu \cdot n dS = 0, \tag{3.1}$$

where ρ is the fluid density, ν kinematic viscosity, Ω is an arbitrary domain and S its boundary. Also $\frac{\partial}{\partial t}\int_{\Omega} \rho d\Omega$ is the rate of change of mass inside de considered volume and $\int_{S} \rho\nu \cdot n dS$ is the inflow of mass through the frontier of the considered volume.

This equation can finally be transformed into the following differential Equation 3.2:

$$\frac{\partial \rho}{\partial t} + \mathcal{D}(\rho\nu) = 0, \tag{3.2}$$

where $\mathcal{D}$ is the material derivative. This equation is also known as the continuity equation.

The next conservation law to consider is the momentum equation [12, 17], which can only be changed in the presence of external forces. So, according to Newton's second law of motion:

$$\frac{d(mv)}{dt} = F,$$

where t stands for time, m for mass, v for velocity, and F the resultant of the external forces.

Note that if the system if isolated, then F must be zero and the momentum is time invariant.

Using the control volume in a way analogue to the Equation 3.1, the momentum conservation equation can be written as

$$\frac{\partial}{\partial t}\int_{\Omega}\rho\nu d\Omega + \int_{S}\rho\nu\nu\cdot ndS = F_o,$$

where Ω represents the control volume, S its surface, and F_o the sum of other forces acting on the fluid. Those forces can also be considered, being those the forces acting in the fluid as a whole, like gravity, viscous forces, and pressure forces. The first can be considered as $\int_{\Omega}\rho f d\Omega$, the second as $\int_{S}\tau\cdot ndS$ and the last as $\int_{S}pndS$. The stress tensor is represented by τ and can be obtained through the following expression:

$$\tau = (-p + \lambda\nabla\cdot v)I + 2\mu\hat{D},$$

where μ is the dynamic viscosity, $\hat{D}$ is the deformation tensor and $\hat{D} = \frac{1}{2}(\nabla v + \nabla v^T)$. However, for incompressible flows like the considered in this study, $\nabla\cdot v = 0$, which results in

$$\tau = -pI + 2\mu\hat{D}.$$

Then the momentum equation 3.3 is considered.

$$\frac{\partial\rho v}{\partial t} = \rho f + \nabla\cdot(\tau - \rho vv), \tag{3.3}$$

which, using the conservation of mass, can be written as

$$\rho\frac{Dv}{Dt} = \rho f + \nabla\cdot\tau,$$

where

$$\rho\frac{Dv}{Dt} = \rho\left(\frac{\partial v}{\partial t} + v\cdot\nabla v\right).$$

So, adding the convection terms and the diffusion terms together, one will reach the Equation 3.4.

$$\rho\left(\frac{\partial v}{\partial t} + v\cdot\nabla v\right) = -\nabla p + \nabla\cdot(\mu(\nabla v + \nabla v^T)) + F. \tag{3.4}$$

The Navier-Stokes equations are only valid for non-compressible fluids, since the term $\nabla \cdot v$ was considered as negligible, but one can easily get the full Navier-Stokes equations that can model the behaviour of any Newtonian fluid in a large enough space for the equations to be relevant. So although gases are quite able to deform under pressure, high velocity gradients are needed to create considerable deformations in the gas, because gases very low viscosity and fairly low density. So, we can consider that for velocities lower than Mach 0.3, the present deformations are negligible.

A limitation of the Navier-Stokes equations is turbulent flow, because the increasingly smaller vortices would require an increasingly finer mesh. Turbulence is a chaotic flow regime characterized by the presence of very small vortices that change the overall behaviour of the fluid on that specific region [18, 19]. The higher the velocity of the fluid in respect to its viscosity, the more turbulent it is, because the viscosity is what works against shear forces in the fluid, and therefore is what mitigates turbulence [20]. To help distinguish between a laminar from a turbulent flow region one can calculate the Reynolds number (Re) [20], which characterizes the ratio of inertial forces to viscous forces and, therefore, can be used to predict if the flow will behave in a laminar or turbulent way. The equation that gives the Reynolds number is:

$$Re = \frac{\rho v L}{\mu},$$

for which ρ is the density of the fluid, v is the characteristic velocity of the fluid, L is the characteristic linear dimension and μ the dynamic viscosity of the fluid.

On top of these equations, when the study of how a substance disperses in a fluid is important, then advection-diffusion equations are needed. These are simpler than the Navier-Stokes equations and can be understood by thinking of the way that the substance concentration changes depending on a natural diffusion term, and a convective term that depends on the velocity of the fluid in each point, which can be estimated using the Navier-Stokes equations. So we have the Equation 3.5:

$$\frac{\partial c}{\partial t} = \nabla \cdot (D \nabla c) - \nabla \cdot (\vec{v} c), \tag{3.5}$$

where c is the concentration of the substance, D is the diffusion coefficient and $\vec{v}$ the velocity of the fluid. Note that c could also be a temperature if the study was about thermodynamics. So the term $\nabla \cdot (D \nabla c)$ describes diffusion and the term $-\nabla \cdot (\vec{v} c)$ is the one that describes advection.

4 Resources

4.1 Methods

4.1.1 Fluid Behaviour Modelling

In what concerns gas dispersion modelling, the most popular approach is to consider the Gaussian plume, which considers a chemical plume as having a Gaussian distribution [11] and that makes it very simple to use. However, this method fails once turbulent or irregular flow is considered and cannot be used to model the behaviour of the fluid itself, but rather of an intruder plume that spreads through the fluid. For this particular work, the Gaussian plume option was dismissed because the main objective was to model the speed of air and the dispersion of a gaseous plume from some measured values rather than a know source, which makes for a more complex problem that would lose some versatility. So, to model the behaviour of the air and the dispersion of a gas through moving air, the direct mathematical models were used for being the most accurate way despite the added problems concerning grids and convergence associated with the use of some numerical method. Being the most wildly used numerical methods, finite differences, finite volume and finite elements.

Finite differences methods [12, 21] are the simplest ones to implement, since they are based on finite intervals along the axis of reference to compute approximations for the given equations on all the considered points in space, and can archive accurate results if fine meshes are used.

Finite volume methods [12, 21] are good at preserving memory, allow for fast computational speeds, and can have unstructured meshes, what allows for higher accuracy with coarser meshes. This is because, instead of considering the neighbouring points to compute each point, they consider the neighbouring volumes to compute the value of each volume, this means that at most, the same number of points are going to be considered and this can work with any adjacent shapes and distances.

Finite elements methods [21] are more stable and accurate but are also slower to compute and use more memory than the finite volume methods. These methods are harder to implement and require special care to ensure a convergent solution. Differently to the other referred methods, these consider elements constituted from multiple points and associate matrices to those elements, which makes these methods, both more accurate and more computational demanding.

These numerical methods can easily be used to both compute the dynamic flow of air, using the Navier-Stokes equations (Chapter 3), and the dispersion of a gas through air, using advection diffusion equations. Or if one were to consider a turbulent flow at higher Reynolds numbers, other models would be needed, like Reynolds-averaged Navier-Stokes (RANS) [22, 23] for fluid flow problems or the introduction of the eddy diffusivity concept. Mathematical models like these tend to be difficult to implement, but are still very useful. And in what concerns prediction applications, they are used mainly in Numerical Weather Prediction (NWP), like said before.

For this work in question a finite difference method was used in its implicit form with the backward difference and a regular linear grid. This method was chosen because it is simple to implement and the other referred numerical methods don't offer great advantages for one-dimensional problems such as this. A regular grid was considered since an irregular one would only increase the overhaul accuracy of the method if the grid was adaptive and followed the higher gradients of the output, which would involve very complex algorithms. So to achieve good accuracy a finer grid was used at the expense of more memory and more computational effort.

4.1.2 Methods for Inverse Problems Regarding Fluid Flow

To compute the air velocity field and the odour concentration map one needs to develop an inverse model able to go from a limited set of measurements to a complete field of values. It is called an inverse problem because it starts with the results and then calculates the causes.

Among several other known methods to solve this problem there are the two-dimensional splines [1], which allow for a smooth curve that resembles a true field of values and can also compute changes in time if another dimension is added. Kriging (or Gaussian process regression), which is the best method for spacial interpolation of field values, because it gives the best linear unbiased prediction (BLUP) [24, 25]. And the best method to predict new

iterations for dynamic problems, which is the Kalman filter (or linear quadratic estimation).

However, these models have limitations that make them either hard to adapt to the problem studied here or very computational demanding. In the case of the Kriging method, it is insufficient on its own to predict values along the time dimension, so it needs to be paired with some dynamic model that can estimate a new Kriging field of values from the current one, that propagates in the right way according to the Navier-Stokes model of a flowing fluid. And the problem with the Kalman filter is that it is too slow for the amount of variables that must be considered for the prediction to be accurate through the whole area of interest.

So, the method that was tested in this work was the use of the direct model with the differential equations in question to estimate a possible field that could lead to the measured values, either by propagating the values in space and time or by iteratively alter the boundary equations until a possible field is achieved. This method also involves some problems, like it cannot make any distinction between an error in the measurement of the sensor and an error cause by a bad prediction of the inversion method.

4.1.3 Communication

To make the final setup work, a communication protocol was needed to control the microcontroller associated with the turbines and for receiving the data from the microcontrollers associated with the sensors. For that purpose, a RS485 network was put together connecting a computer and three microcontrollers that needed to exchange one 64-bit unsigned integer and eleven 32-bit integers that can be used, depending on the microcontroller that has to interpret the message, for the data read by the sensors and to set the PWM signals for the turbines. Although there are no flags to distinguish the type of data sent, however, there are two bytes designated to characterize the sender and address the receiver. The sender is designated on the first byte of the message with a letter from 'a' to 'd', and the receiver is specified on the last useful byte of the message also with a letter from 'a' to 'd'. The Figure 4.1 represents the structure of a message with the types of variables that are used by the nodes to read the messages.

Figure 4.1: RS485 message structure

4.2 Materials

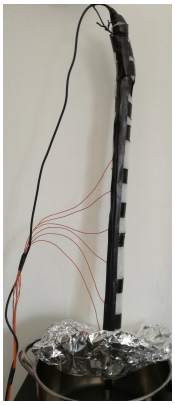
4.2.1 Electronics

The most common components used to build the test ambiances were resistors, capacitors, NTC thermistors, MOX sensors, BJT and MOSFET transistors, operational amplifiers, voltage regulators, LED's, different types of cable and connectors. The used resistors all have a 5% tolerance and enough power rating to withstand the used currents. The capacitors were used to filter noise from the voltage sources and regulators and to keep it constant at the terminals of the sensors and integrated circuits, and the capacitance was always high enough to keep the system stable. The thermistors were used to measure the wind velocity through a self-heating process (Section 5.2), every used NTC Thermistor has 2 kohm resistance with 1% tolerance, a β value of 3800 and a glass encapsulation with 2*4 millimetres. The BJT transistors were used to supply the needed voltage to the thermistors in order to reach the wanted temperature. The operational amplifiers (UA741CN) were used to control the BJT transistors. The MOSFET transistors were used to control the speed of the turbines through a PWM signal. The MOX sensors (MiCS-5521) were used to measure the concentration of oxidizable gases in the air. The voltage regulators (L78505CV) were used to reduce the voltage from 12V to 5V for the microcontrollers and for the MOX sensor circuits. The LEDs were just used as a visual feedback for the communication circuits.

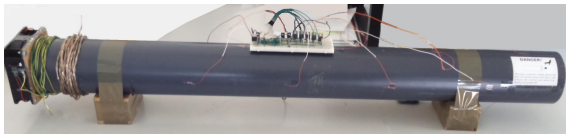
Regarding the microcontrollers, the most important was the PIC24FV16KM202, as it was used most of the time for tasks such as read analogic voltage values and output PWM signals. To program it, a PicKit3 was used and the code was compiled using the MPLAB X IDE version 3.35. The other microcontroller used was an ATmega328P with an Arduino Uno for minor tasks. Other integrated circuits were needed for the microcontrollers to communicate with each other in grids larger than two nodes, so some RS485 transducers (SN75176A) were used to set up a RS485 network. Also, to communicate with a computer, a FTDI232 was used.

4.2.2 Prepared Physical Environments

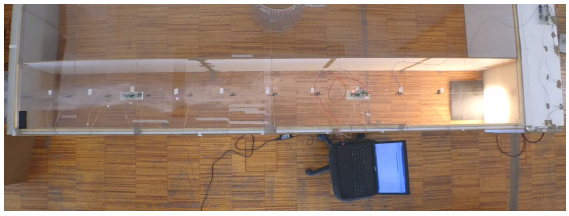
In order to execute the intended practical tests, some hardware setups were put together, being the first for testing an idea using a 40 cm stainless steel rod as a heat conductor. Six NTC thermistors were coupled to the rod using thermal conductive paste and the resulting



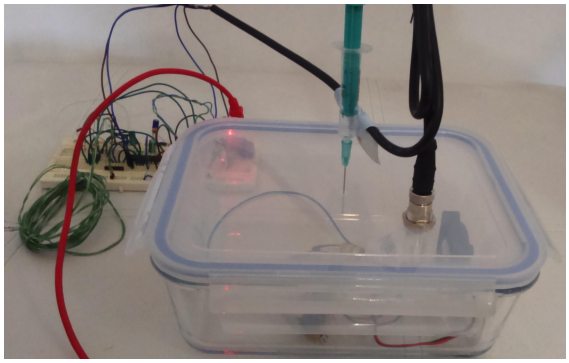
(a) Heat dissipation setup



(b) Wind behaviour setup



(c) Odour dispersion setup



(d) Odour sensor calibration setup

Figure 4.2: Experimental setups assembled throughout this work

bundle was wrapped with packing foam wrap to help isolating it from the air temperature. Then on unwrapped tip of the rod was taped a foil cone opening towards the rest of the rod to deflect hot air convention currents and that tip was dipped in boiling water to maintain a constant temperature of 100 °C. This setup is represented in Figure 4.2a.

The next one was intended to measure the airspeed at different points along a tube. For that, a one meter long tube with a diameter of 10.5 cm was used together with a 12 centimetre DC brush-less fan to draw air trough the tube. A small cardboard adapter was cut to help coupling the fan to the tube. Self heating NTC thermistors were mounted inside the tube through tiny holes in the plastic, and honeycomb filters were placed inside the tube to straighten the flow. The circuit board was glued on top of the tube to be close to the thermistors. This setup is represented in Figure 4.2b.

The most important setup has some similarities to the prior but in a bigger scale, this time a square section tunnel was built out of a rectangular section wind tunnel with the use of particle board walls and an acrylic top, being the bottom the laboratory's floor. The section has a size of 0.5 meters and the whole tunnel is 4 meters long. The turbines are positioned at the end of the tunnel not to introduce turbulence in the wind, and are controlled simultaneous by the PWM channels of the used microcontroller through a MOSFET transistor. Every

microcontroller in the tunnel and sensor network was connected with a RS485 network to a main computer. And the sensors were individual circuits capable of measuring wind speed and odour concentration. This setup is represented in Figure 4.2c.

The last assemble was not a test but was also important since it was meant to help calibrating the odour sensors. An hermetic sealed glass container with a plastic top was perforated to fit a connector to power the needed circuits inside and a syringe was used to inject a known quantity of 96% ethanol alcohol. A small fan was also placed inside the container to mix the air-ethanol mixture. This setup is represented in Figure 4.2d.

5 Developed Work

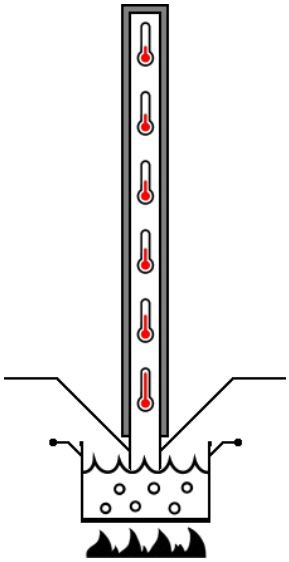
To develop and test the algorithms that were used to map the wind and odour fields, a simulator in Matlab was coded and examples of possible wind and odour fields in one dimension started to be run. Then, to test the various algorithms, some setups (Section 4.2.2) were assembled to extract real data from the anemometers and gas sensors. Then, the acquired data was used to estimate a possible realistic field of values across the considered domain.

5.1 Simulators

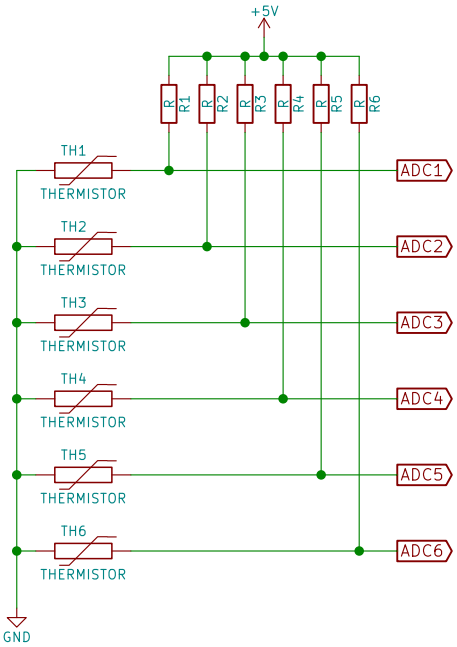
The simulators are similar Matlab programs that apply the differential equations to a field of initial values and given boundary conditions. The result is a field of values that is coherent with the physical model and the given boundary conditions. Simulators were used to model the diffusion of heat in a steel rod, to model the flow of air trough a pipe, and to model the spread of a gas through moving air. The objective of these simulations was to have a good point of reference for what should be the real measurements and areas in between the sensors. The reason for rods and tubes is that those are very close to what can be considered as a one-dimensional structure. With these simulations run, visual marks were set to be the best performances possible of the inversion algorithm. And with those results one could start to program a way to invert the experimented results in a way that would result in the full map of the variable of interest across the medium.

As a proof of concept, the first test model to be simulated was the diffusion of temperature in a steel rod, which may seem too distant from the purpose of this work, but it is a model that is very easy to test in a real environment with very high reliability and the modelled behaviour is fairly similar to that of a gas dissipating trough still air. To test this, the simulator was programmed based on a simple heat equation with thermodynamic losses.

Also, an apparatus was set up to gather real data for the inversion. A thin steel rod was dipped in boiling water in one end and isolated in the rest to minimise losses, six thermistors were thermally coupled to the rod as the Figure 5.1a shows represented by thermometers, and a piece of foil was placed around the beginning of the rod to deflect the hot convecting air (Section 4.2.2).



(a) Assembled temperature measuring system diagram



(b) Temperature sensing circuit

Figure 5.1: Temperature diffusion experimental setup

The electrical resistance of the thermistors was measured using simple tension dividers and an ATmega328P microcontroller to read and send to a computer the analogic signals (10 bit resolution) from the tension dividers. The raw data was logged for later processing.

Before considering the measurements taken with the setup represented in Figure 5.1, a simulation was run, considering the diffusion of heat in the rod with boundary conditions related to the actual real conditions of the test. It is important to compare the simulation to the measurements, to analyse the differences and look for factors that are not being considered and compromise the accuracy of the inversion.

Since the physical rod has 0.4 meters, the considered domain was $[0, 0.4]$ and the time domain $[0, 3600]$ seconds, because the experiment lasted one hour. To keep the conditions similar, one end of the rod was considered to be at a constant 100 °C and the other end as being well isolated. The initial temperature across the medium is of 20 °C, and for any instance beyond that, the temperature is defined by a simple diffusion equation (Equation 5.1) with a loss term, that was necessary because of the present thermal losses trough the isolation.

With this, the considered mathematical model is defined by

$$\frac{\partial T}{\partial t}(x,t) = D\frac{\partial^2 T}{\partial x^2}(x,t) - K\left(T(x,t) - T_a\right), \tag{5.1}$$

for $x \in (0, 0.4)$ and $t \in (0, 3600)$, where T_a is the ambient temperature and K is the thermal loss coefficient. And it has the boundary conditions

$$T(0,t) = 100, \quad \forall\ t \in [0, 3600],$$

$$\frac{\partial T}{\partial x}(1,t) = 0, \quad \forall\ t \in [0, 3600].$$

Since the exact analytical solution of partial differential equations is in general very complex to calculate accurately, the solution needs to be approximated in a set of considered points. For those points, a uniform mesh was considered with N + 1 points with a stepsize of $h = 0.005$, which means that $0 = x_0 < x_1 < ... < x_N = 0.4$ where $x_i = ih$. Regarding the time domain, the time-step is $\Delta t = 1$ s and $t^m = m \cdot \Delta t$ with $m = 0, ..., M$. It is then possible to calculate $T_i^m \approx T(x_i, t^m)$, for $i = 0, ..., N$, $m = 0, ..., M$. Now, in order to discretize the equation 5.2, the backward difference method was used for the time derivative and a centred second order approximation was used for the spatial derivative [21].

$$\frac{T_i^{m+1} - T_i^m}{\Delta t} = D\frac{T_{i-1}^{m+1} - 2T_i^{m+1} + T_{i+1}^{m+1}}{h^2} - K(T_i^m - T_a). \tag{5.2}$$

From this form, it is possible to solve the equation as a matrix system 5.3

$$AT^{m+1} = T^m - K(T^m - T_a)$$

$$\Downarrow \tag{5.3}$$

$$\begin{bmatrix} 1+\frac{2D\Delta t}{h^2} & -\frac{D\Delta t}{h^2} & & & 0 \\ \ddots & \ddots & \ddots & & \\ & -\frac{D\Delta t}{h^2} & 1+\frac{2D\Delta t}{h^2} & -\frac{D\Delta t}{h^2} & \\ & & \ddots & \ddots & \ddots \\ 0 & & & -2\frac{D\Delta t}{h^2} & 1+\frac{2D\Delta t}{h^2} \end{bmatrix} \begin{bmatrix} T_1^{m+1} \\ \vdots \\ T_{i-1}^{m+1} \\ T_i^{m+1} \\ T_{i+1}^{m+1} \\ \vdots \\ T_N^{m+1} \end{bmatrix} = \begin{bmatrix} T_1^m \\ \vdots \\ T_{i-1}^m \\ T_i^m \\ T_{i+1}^m \\ \vdots \\ T_N^m \end{bmatrix}^{1} - K \begin{bmatrix} T_1^m - T_a \\ \vdots \\ T_{i-1}^m - T_a \\ T_i^m - T_a \\ T_{i+1}^m - T_a \\ \vdots \\ T_N^m - T_a \end{bmatrix}.$$

[1] For T_1^m, the boundary condition needs to be summed like so, $T_1^m + \frac{D\Delta t}{h^2} * T(0, t^m)$

Now, after the experimental test and the simulation, one can use the measured values obtained with the setup from Figure 5.1 to rectify the simulation. This allows the simulation to be approximated to the reality, since the mathematical models always have some differences from the real phenomena. To do this, the spatial domain was divided in equal portions separated by the points where the sensors are (where the values are known). In those smaller domains, the same numerical method (see Equation 5.3) is applied with two boundary conditions equal to the nearest measured values. So the considered boundary conditions were

$$T(10s, t) = T_s(t), \quad \forall \ t \in [0, 3600],$$
$$\frac{\partial T}{\partial x}(10s, t) = 0, \quad \forall \ t \in [0, 3600],$$

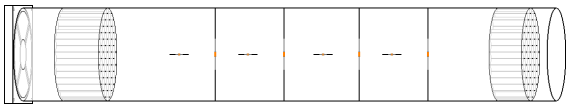
where $T_s(t)$ with $s \in \{1, ..., 6\}$ is the measured temperature by the s sensor at the instant t, and the number 10 means that there is a sensor at each 10 length steps.

To introduce these conditions to the system, matrix A needs to be tweaked to allow for the boundary conditions to be implemented. So the T_s^m points coincident with the positions of the sensors were overwritten by the measured $T_s(t)$ values. And in order for this substitution to take effect, the T_s^m should not be computed. So the $A(s, s)$ points were set to 1 and the values $A(s, s-1)$ and $A(s, s+1)$ were set to zero. Also the loss term was ignored for the measured values.

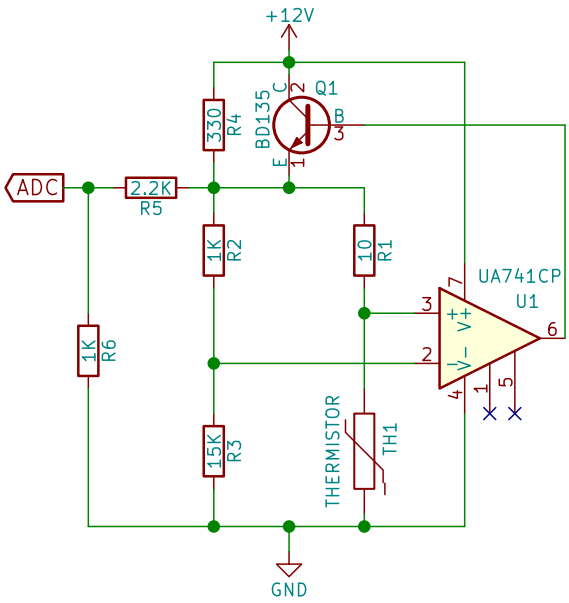
5.2 Simulation and Testing

After testing the method with the temperature dissipation model, a more complex set up was tried, and this time the aim was to model the Navier-Stokes equations 3.4 for a non-compressible (negligible compressibility) fluid in a laminar flow regime. To test this, a tube was used as a wind tunnel, with honeycomb filters and a DC brush-less turbine. The wind speed sensors were seven self-heating thermistors and were placed at 10 cm intervals. Also another thermistor was used to measure the temperature of the air as it entered the tube. This setup is shown in Figure 5.2.

To measure the air speed around the thermistors, each thermistor was heated such that an increase in air speed would result in a high enough unbalance to the heating circuit for it to be easily measured. To do this, two approaches were tried. The first one was to use a current mirror to supply the each thermistor with a constant current, which means that with an increase in resistance on a thermistor results in an increase in supplied voltage, so it



(a) Assembled wind speed measuring system diagram



(b) Anemometer circuit

Figure 5.2: Experimental setup for air movement behaviour

is self-balanced and easy to measure. But the response times were poor, given the fact that the thermistor had to cool down and heat up to keep up with the different balance points. To work around that problem a second circuit was designed and tested (see Figure 5.2b), but this time the thermistor was part of a Wheatstone bridge, where the output of the bridge was compared by a operational amplifier and fed to the transistor that feeds the bridge. This way the bridge would self-balance by providing more power to the bridge, and therefore heating it up and lowering its resistance.

This one-dimensional wind test did not achieve the expected results because a pressure or velocity gradient in one dimension propagates at the speed of sound and the sampling time of 0.15 s was not small enough to characterise any wave moving at such a speed. In an unconstrained environment air can move more freely and velocity gradients can move at speeds closer to the velocity in question. But inside a narrow tube, air only moves in one dimension and pressure differences cannot escape trough open space. That said, the response of a fluid to a force in a constrained tunnel differs from the response to the same force in an open environment.

To better understand the mechanics in question, one can see the fluid as a long and narrow mass of particles that is going to be pulled or pushed in one end. Those particles are held together with pressure (atmospheric pressure) and any force acting on the system will act on the particles as to reach the new point of equilibrium. This system is then analogue to a chain

of springs and masses and the speed at which a pressure disturbance propagates through the medium is the acoustic speed in that medium [26] [27]. This means that the Navier-Stokes equations for this problem are only relevant for two or three-dimensional analysis or for long enough one-dimensional cases. Nonetheless, the tests were done and results were plotted so that this problem may be worked around better in future research.

In what concerns the simulation of the behaviour of air across the one-meter tube, the air was considered to be stationary until 0.5 seconds after the simulation had started. At that point the speed at one end of the tube was considered to be 3 m/s, as if a turbine was turned on, and then the speed in the same point was considered as being zero again 0.5 seconds after, as if the turbine had stopped. For the rest of the domain of $[0,1]m$ the air velocity is defined by the equation 3.4. The time domain was $[0,20]s$, and the considered boundary conditions are

$$V(0,t) = 0, \quad \forall \ t \in [0, 0.5[, \ ]1, 20],$$

$$V(0,t) = 3, \quad \forall \ t \in [0.5, 1],$$
$$\frac{\partial V}{\partial x}(1,t) = 0, \quad \forall \ t \in [0, 20].$$

The solution for this equation was then approximated in the set of $N+1$ points $0 = x_0 < x_1 < ... < x_N = 1$ where $x_i = ih$ with h = 0.001. The time domain was considered to have a finite number of times instances spaced by $\Delta t = 0.01$ and $t^m = m * \Delta t$ with $m = 0, ..., M$. Now it is possible to calculate $V_i^m \approx V(x_i, t^m)$, for $i = 0, ..., N$, $m = 0, ..., M$. So to discretize the equation 3.4, the backward difference method was used for the first-order derivatives and a centred second-order approximation was used for the second-order spatial derivative.

$$\frac{V_i^{m+1} - V_i^m}{\Delta t} + V_i^m \frac{V_{i+1}^{m+1} - V_{i-1}^{m+1}}{h} = -\frac{p_{i+1}^{m+1} - p_{i-1}^{m+1}}{h\rho} + \nu \left(\frac{\frac{V_{i+1}^{m+1} - V_i^{m+1}}{h} - \frac{V_i^{m+1} - V_{i-1}^{m+1}}{h}}{h} \right), \quad (5.4)$$

where $x \in (0,1)$, $t \in (0,20)$ and $\nu = \mu/\rho$.

From this form, it is possible to solve the equation as a matrix system 5.5

$$AV^{m+1} = BV^m + F$$

$$\Downarrow \tag{5.5}$$

$$\begin{bmatrix} -\frac{1}{\Delta t} - \frac{2\nu}{h^2} & -\frac{V_1^m}{h} + \frac{\nu}{h^2} & & & 0 \\ \ddots & \ddots & \ddots & & \\ & \frac{V_i^m}{h} + \frac{\nu}{h^2} & -\frac{1}{\Delta t} - \frac{2\nu}{h^2} & -\frac{V_i^m}{h} + \frac{\nu}{h^2} & \\ & & \ddots & \ddots & \ddots \\ 0 & & & \frac{V_N^m}{h} + \frac{\nu}{h^2} & -\frac{1}{\Delta t} - \frac{2\nu}{h^2} \end{bmatrix} \begin{bmatrix} V_1^{m+1} \\ \vdots \\ V_{i-1}^{m+1} \\ V_i^{m+1} \\ V_{i+1}^{m+1} \\ \vdots \\ V_N^{m+1} \end{bmatrix} =$$

$$= \begin{bmatrix} -\frac{V_1^m}{\Delta t} \\ \vdots \\ -\frac{V_{i-1}^m}{\Delta t} \\ -\frac{V_i^m}{\Delta t} \\ -\frac{V_{i+1}^m}{\Delta t} \\ \vdots \\ -\frac{V_N^m}{\Delta t} \end{bmatrix}^2 + \begin{bmatrix} \frac{P_1^m - P_0^m}{\rho h} \\ \vdots \\ \frac{P_{i-1}^m - P_{i-2}^m}{\rho h} \\ \frac{P_i^m - P_{i-1}^m}{\rho h} \\ \frac{P_{i+1}^m - P_i^m}{\rho h} \\ \vdots \\ \frac{P_N^m - P_{N-1}^m}{\rho h} \end{bmatrix},$$

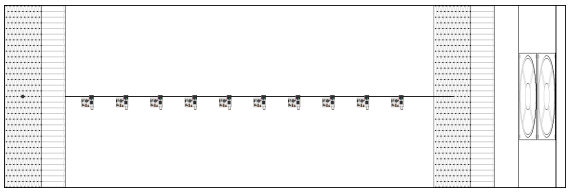
where instead of changing the values of the matrix A to isolate the last point, the value of the point V_N^m was overwritten by the value of the point V_{N-2}^m, which has the same mathematical result and is faster to compute.

One of the biggest problems this time is that there is some error associated to the use of the elements . The solution was then to use the last computed version of those elements, which would result in a substantial error, so the idea was to use the newly computed values and feed them back in the same equation in order to recalculate them more accurately. This was done until the difference between two computations of the same value was smaller than 1%. Other problem is the calculation of the F pressure component of the matrix, which was basically done by using $F = AV^{m+1} - BV^m$.

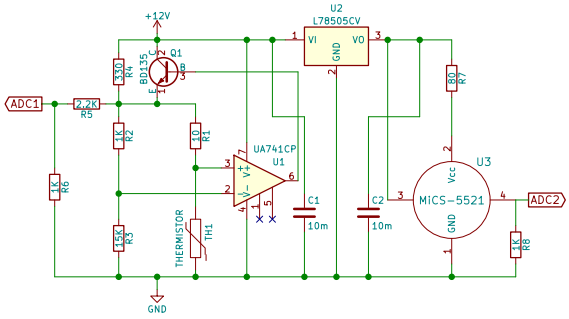
The next model to be tested was the dispersion of a gas through air, and the mathematical model needed was an advection-diffusion equation. For this, a longer and wider tube was used. This tube had honeycomb filters on each side and multiple turbines in the sink end.

[2]For V_1^m, the boundary condition needs to be summed like so, $V_1^m + (\frac{V_1^{m+1}}{h} + \frac{v}{h^2}) * V(0, t^m)$

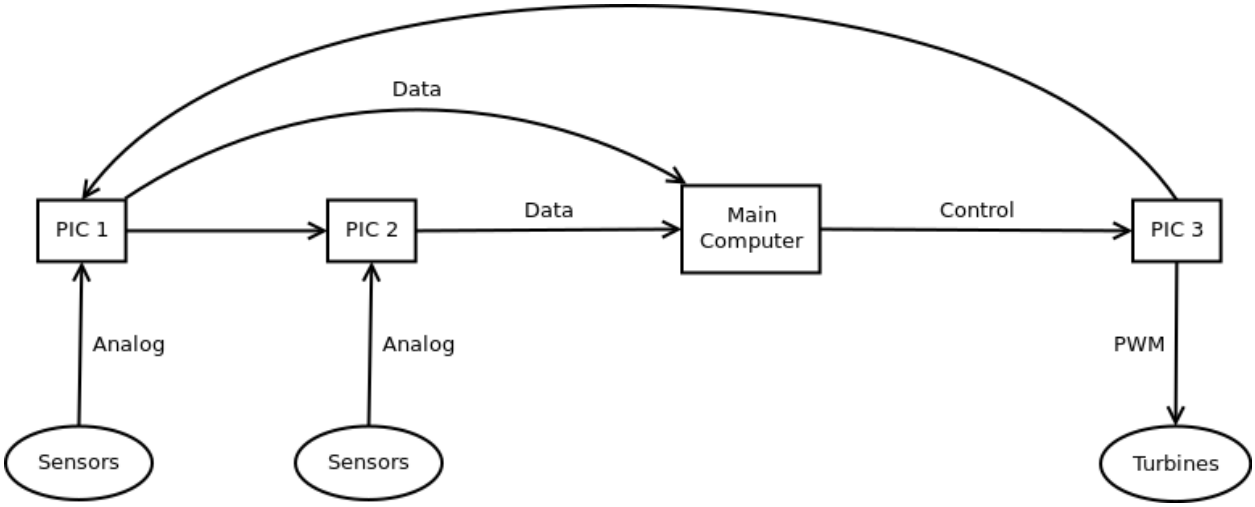
The turbines were controlled by a PWM pulse generated by a PIC microcontroller and the measurements were taken with the use of ten small identical circuits containing an anemometer similar to the ones used before and a MOX odour sensor. To control the turbines and log the measured data with the same computer, a RS485 network was put together and it involved 3 PIC microcontrollers and one computer. One PIC received the PWM values to set the turbines, the other two read the analogic values from the sensors and sent them to the computer, and the computer logged the data and sent the PWM values through the network.



(a) Odour sensing in a wind tunnel diagram



(b) Anemometer and odour sensor circuit



(c) Experimental setup communication and functioning diagram

Figure 5.3: Experimental setup for odour dissipation in a working wind tunnel

The circuit for the anemometer is the same that was used previously, it consists of a Wheatstone bridge with an amplifier and a transistor as a feed back loop such that the circuit reaches an equilibrium when the thermistor has a specific resistance, which happens only if the thermistor reaches the right temperature. If more air flows around the thermistor, more power is going to be needed to reach the needed temperature, and so, the velocity of the fluid can be extrapolated from the power that is being applied to the circuit. The MOX sensor is powered with 2.4 Volts and it's resistive component is measured with a simple voltage divider.

Concerning the functioning of the experimental setup 5.3c, the nodes send their messages in turns, following the shown sequence of Main Computer, PIC 3, PIC 1, PIC 2, Main Computer. The "Data" arrows contain the measured values from the sensors, and the "Control" arrow contains the PWM values for the turbines, any uncharacterised arrow represents the message that gives the turn to the next node, as well as "Data" and "Control" arrows.

Regarding the Simulation, this test was done considering that the odour concentration at the first point of the tube is equal to the odour concentration measured by the first sensor of the framework in Figure 5.3. This is because the simulation needs to meet close conditions to those of the measuring setup, in order for the extracted curves to be comparable. Also, the last point of the simulation is an isolated point. The considered spatial domain is then $[0, 4]m$ and the time domain $[0, 30]s$. The initial conditions are of $0\mu g/L$ for the entire domain at $t = 0$, and for $t > 0$ the concentration of alcohol vapour is estimated with respect to the equation 3.5, and the considered boundary conditions were

$$C(0, t) = C_1(t), \quad \forall \ t \in]0, 30],$$

$$\frac{\partial C}{\partial x}(1, t) = 0, \quad \forall \ t \in [0, 30],$$

where $C_1(t)$ is the value measured by the first odour sensor in the network.

To approximate the solution for the equation 3.5, the set of $N + 1$ points $0 = x_0 < x_1 < ... < x_N = 1$ was considered, where $x_i = ih$ with h = 0.01. The time domain was considered to have a finite number of times instances spaced by $\Delta t = 0.02$ and $t^m = m * \Delta t$ with $m = 0, ..., M$. Now it is possible to calculate $C_i^m \approx C(x_i, t^m)$, for $i = 0, ..., N$, $m = 0, ..., M$. For the discretization, the backward difference method was used for the first-order derivatives and a centred second-order approximation was used for the second-order spatial derivative.

$$\frac{C_i^{m+1} - C_i^m}{\Delta t} + \frac{(VC)_{i+1}^{m+1} - (VC)_{i-1}^{m+1}}{2h} = D \left(\frac{\frac{C_{i+1}^{m+1} - C_i^{m+1}}{h} - \frac{C_i^{m+1} - C_{i-1}^{m+1}}{h}}{2h} \right). \tag{5.6}$$

The V^{m+1} terms refer to the velocity of the fluid and needs to be input to the system by other means, which means that the ideal method would be to use the Navier-Stokes equations to estimate V^{m+1} first, and that would be the course of this work if a two dimensional system were to be considered. But as seen before, the Navier-Stokes equations are irrelevant in a small one dimensional problems like this one. So to get a reliable value for V^{m+1} the direct measures from the anemometers were considered, and since the flow velocity is constant in

the whole length of the tunnel, the measures were averaged.

From the form 5.6, it is possible to solve the equation as a matrix system 5.7:

$$AC^{m+1} = BC^m - KC^m$$

$$\Downarrow \tag{5.7}$$

$$\begin{bmatrix} \frac{1}{\Delta t} + \frac{D}{2h^2} & -\frac{D}{2h^2} + \frac{V_2^{m+1}}{2h} & & & 0 \\ \ddots & \ddots & \ddots & & \\ & -\frac{D}{2h^2} - \frac{V_{i-1}^{m+1}}{2h} & \frac{1}{\Delta t} + \frac{D}{2h^2} & -\frac{D}{2h^2} + \frac{V_{i+1}^{m+1}}{2h} & \\ & & \ddots & \ddots & \ddots \\ 0 & & & -\frac{D}{2h^2} - \frac{V_{N-1}^{m+1}}{2h} & -\frac{1}{\Delta t} - \frac{2\nu}{h^2} \end{bmatrix} \begin{bmatrix} C_1^{m+1} \\ \vdots \\ C_{i-1}^{m+1} \\ C_i^{m+1} \\ C_{i+1}^{m+1} \\ \vdots \\ C_N^{m+1} \end{bmatrix} =$$

$$= \begin{bmatrix} -\frac{C_1^m}{\Delta t} \\ \vdots \\ -\frac{C_{i-1}^m}{\Delta t} \\ -\frac{C_i^m}{\Delta t} \\ -\frac{C_{i+1}^m}{\Delta t} \\ \vdots \\ -\frac{C_N^m}{\Delta t} \end{bmatrix}^{3} + K \begin{bmatrix} C_1^m \\ \vdots \\ C_{i-1}^m \\ C_i^m \\ C_{i+1}^m \\ \vdots \\ C_N^m \end{bmatrix},$$

where similarly to what was done in the previous simulation, the value of the point C_N^m was overwritten by the value of the point C_{N-2}^m. Also, a loss term was added, where K is the loss coefficient.

After the experimental test and simulation, one can use the measured values obtained with the setup from Figure 5.3 to rectify the simulation. This allows the simulation to be approximated to the reality, since the mathematical models don't consider everything from the real phenomenons. To do this, the spatial domain was divided in equal portions separated by the points where the values are known from the sensors. In those smaller domains, the same numerical method (see Equation 5.7) is applied with two boundary conditions equal to the nearest measured values. The considered boundary conditions were then,

$$C(33s, t) = C_s(t), \quad \forall\ t \in [0, 30],$$

[3]For C_1^m, the boundary condition needs to be summed like so, $C_1^m - (\frac{D}{2h^2} + \frac{V^{m+1}}{2h}) * C(0, t^m)$

$$\frac{\partial C}{\partial x}(33s, t) = 0, \quad \forall\ t \in [0, 30],$$

where $C_s(t)$ with $s \in \{1, ..., 10\}$ is the measured temperature by the s sensor at the instant t, and the number 33 means that there is a sensor at each 33 length steps.

To introduce this conditions to the system, the matrix A needs to be tweaked to allow for the boundary conditions to be implemented. So the C_s^m points coincident with the positions of the sensors were overwritten by the measured $C_s(t)$ values. And in order for this substitution to take effect, the C_s^m should not be computed. So the $A(s, s)$ points were set to 1 and the values $A(s, s-1)$ and $A(s, s+1)$ were set to zero. Also the loss term was ignored for the measured values.

5.2.1 Sensor Calibration

To calibrate the anemometer component of the sensor, the turbines were set to a fixed speed and the real flow speed was measured with an accurate hot-wire anemometer. This step was repeated for 10 different speeds between 0 and 1 m/s. The measured values were then normalized and the exponential curves were plotted according to the real wind speed values, as shown in the example of Figure 5.4, that shows the calibration curve for one of the anemometers.

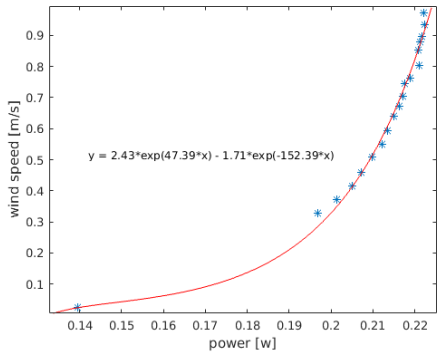


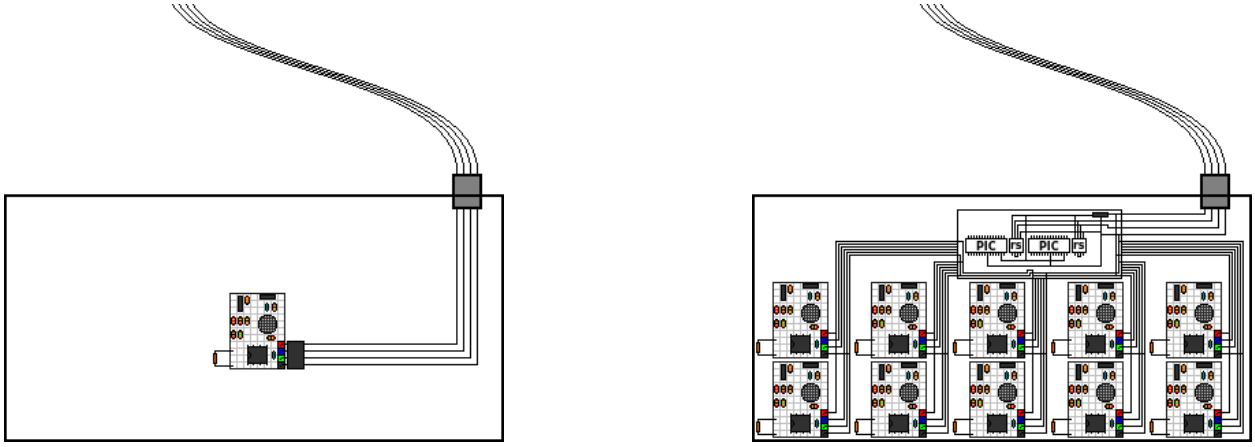
Figure 5.4: Calibration curve for an anemometer

To calibrate the odour sensors, the first sensor was sealed in a 1.65 litre glass container with one small fan to mix the air (see Figure 5.5a), where a small amount of evaporated ethanol (96%) was introduced, and the values acquired by the ADC were logged and averaged to set the value for actual concentration of ethanol. The method to dilute the ethanol was to inject 78.9 mg of liquid ethanol in a 1 L bag full of air and wait for it to evaporate, next a millilitre of that air was removed from the bag and injected in the calibration container resulting in a concentration of 47.81 ng/L. this procedure was then repeated for increasingly

concentrations of alcohol, such as 0.43 μg, 2.39 μg and 4.78 μg.

The next step was then to enclose all of the sensors in the container and inject small amounts of alcohol, the calibration value was given by the previously calibrated sensor (see Figure 5.5b). The reason for the sensors not to be calibrated all at the same time is because it would take a lot of space inside the container and the total volume of gas would be very different from the total capacity of the container. So the first sensor had to be calibrated alone with all the other circuitry outside the container, and after the first calibration it becomes irrelevant the amount of space used inside the box, because there is one sensor that can measure the actual concentration of ethanol in the air.

The resistive component of the odour sensor changed in a logarithmic way, so the measured value for the voltage divider describes a curve similar to $Y = \frac{a}{-log(b*X)+c}$, and the value ranges that the sensor reaches in normal conditions are present in the linear portion of the curve and therefore the voltage curve measured can be approximated as being linear.



(a) First sensor calibration diagram

(b) All sensors calibration diagram

Figure 5.5: Calibration method for odour sensors inside a sealed container

The data acquisition circuits are the same used before minus the microcontroller that controlled the fan. Since there is a small fan inside the container but its power is constant, given its simple task of keeping the air mixed for the sensors to be in the presence of a uniform concentration of ethanol.

After the calibration points were measured, the calibration curves were plotted, using linear regression for the odour concentration, as shown in the example of the Figure 5.6, that is the calibration curve for one of the odour sensors.

As can be extracted from this curve, a linear regression method fits the measured values with enough accuracy. The reason why there are a lot more points for the lowest values is because the majority of the measured values by the sensors are going to be very small, so it

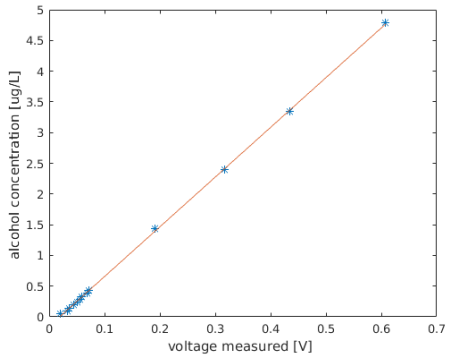


Figure 5.6: Odour sensor calibration linear regression fit

is important to have good accuracy for the lowest concentrations measured.

5.3 Inversion Algorithm

First of all, as said before (Section 5.2), the Navier-Sokes equations for one-directional flows are not relevant for the dimensions of the experimental setup, because any perturbation in the turbines travels near the speed of sound, and therefore, any gradients concerning the flow of air in space are negligible and very hard to detect. But regarding the advection-diffusion equation for the gas sensing, the results were quite promising. The speed of the flow in the tube apparatus can be set to any value between 0.2 and 1 m/s at any given time, which is quite slow and allows for simulations in which the diffusion term of the equation is meaningful. That said, the diffusion coefficient of the sensed gas is really far from the one that can be calculated as the molecular diffusion coefficient. That is because in real conditions, the gases diffusion is in its majority carried out by small eddy currents and vortices, even when the flow is considered as laminar. So, to counter that, the diffusivity coefficient was chosen based on the acquired data and not on the molecular diffusion coefficient, which would be $1.6 \cdot 10^{-5} m^2/s$. In future work, this coefficient should be calculated or estimated in order to minimize the error of a representative experimental test.

So, to extrapolate the odour diffusion map from the acquired data, the idea is that in each measured point the error is considered as negligible and those points are directly passed on to the final map. As an example, in the case of the odour propagation experiment, the theoretical model is considered for the spaces in between measurements and the concentration of alcohol is given by the advection-diffusion equation parametrized as said before. This means that for any point of the plot, its value is determined by the two nearest known

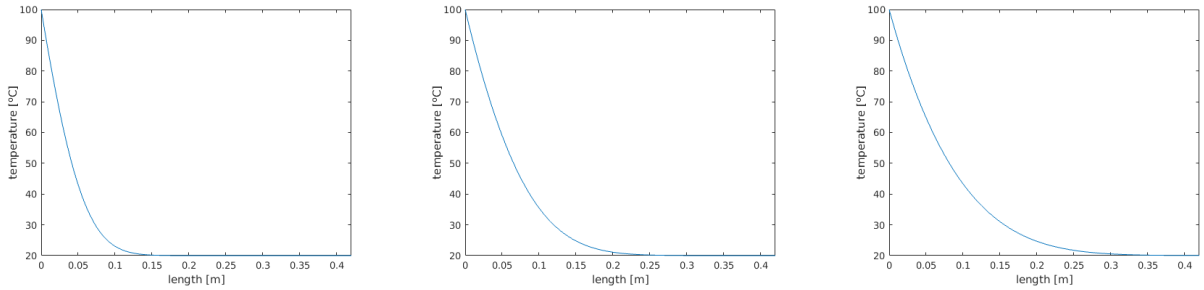
values and the propagation from those values in space and time regarding the considered advection-diffusion equation for the known velocity and rate of diffusion. To simulate the explained method, the boundary conditions were set in a way that takes the measured points as sources of odour concentration to the direction to which the wind is moving and sinks to the direction from which the wind came, and their values were accounted as accurate. In the numerical method, this means that for the C_i^m values at the same positions as of the mounted sensors, the C_i^{m+1} values were set to the measures and not computed by the system. To avoid computing the values for those points, the A matrix was tweaked in such a way that i values corresponding to those C_i^m is equal to 1 and all the values in the same row are zero, as well as all i values of any matrix that is summed in the system.

6 Experimental Results

From what was made to achieve these frameworks and algorithms, some results were taken, plotted and analysed. From those results, these algorithms can follow the dissipation of a fluid in a much more coherent way, regarding the physics of particle flow, and it is clear that the subsequent error is smaller in average than that of some interpolation methods. Also all the graphs plotted in this chapter were obtained using *Matlab*.

6.1 Diffusion

Starting with the simple case for heat diffusion, the simulation was run for a 40 cm long material with a thermal diffusivity of $4.2 \cdot 10^{-6}\ m^2/s$ which is the thermal diffusivity of stainless steel. The considered ambient temperature was of $20°C$, the left boundary condition was set to $100°C$ and the other boundary was set as isolated, the first guess for the rest of the space was $20°C$ as it was the ambient temperature.



(a) Simulation for $t = 300s$ (b) Simulation for $t = 900s$ (c) Simulation for $t = 1800s$

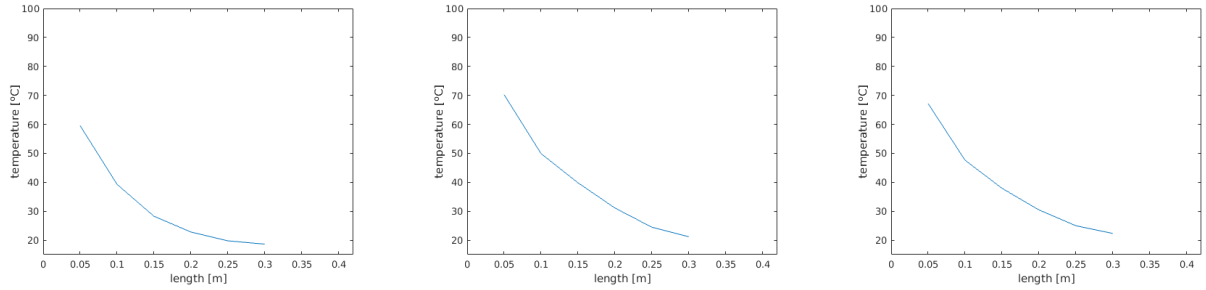
Figure 6.1: Simulation of temperature diffusion in stainless steel

For this simulation, the step in space was of 5 mm and the step in time of 1 s since the dynamics in question are quite slow.

As can be easily seen in the plots from Figure 6.1, the thermal energy quickly gets dispersed form the hottest points to the coldest, behaving like a negative exponential curve. If thermal losses were not considered, the steady state would be a constant temperature of

$100°C$.

The next step was then to replicate this curves using the experimental setup as explained before. And as shown in Figure 6.2, there is some deviation that occurs in consequence of non simulated phenomenons, like fluctuations on air temperature around de system.



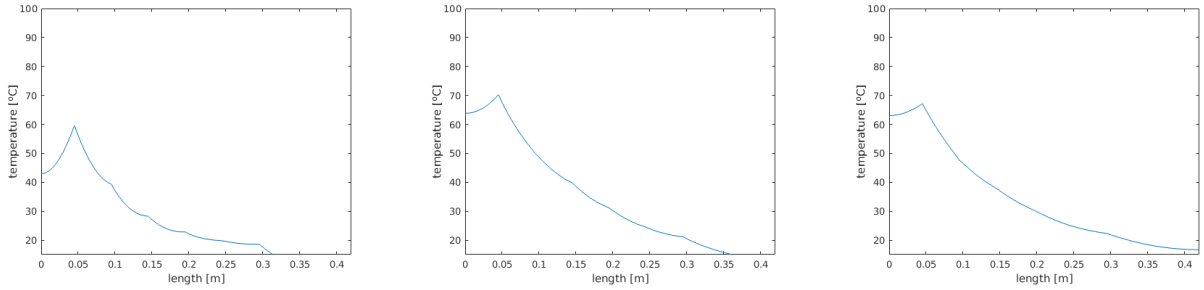
(a) Temperatures at $t = 300s$ (b) Temperatures at $t = 900s$ (c) Temperatures at $t = 1800s$

Figure 6.2: Measured values of temperature in six points along the material

This deviation consists in an early increase in temperature in the furthest measuring points and resulting in a much quicker steady-state condition, as can be seen in the negligible difference between the plots of the Figures 6.2b and 6.2c. This happens because the air around the steel rod heats up, despite of the foil deflecting the convecting currents, and that increase in air temperature is transferred to the steel, despite of the isolating foam layer. This heat management failure always happens to some extent but can be diminished with better setups and accounted for in the simulator by setting up a dynamic ambient temperature value. The reason this was not made here was because it has little relevance for the inversion method and the accuracy for this first case test only needed to be sufficient for the work to move forward with a tested concept.

With all the needed data gathered it was time to test the inversion algorithm, with the results shown in Figure 6.3.

Since the measured points are considered in the simulation as being heat sources, the highest temperature on the model is always the highest measured temperature, because there is no assumed direction for the propagation of temperature, and so there is no way of knowing if there is a hotter heat source than the highest value measured. It is possible to estimate the real heat source if a location is considered, randomly selected or guessed with some criteria, but those ideas result in more constraint solution and would not work for more complex problems and heat distributions.



(a) Inversion at $t = 300s$ (b) Inversion at $t = 900s$ (c) Inversion at $t = 1800s$

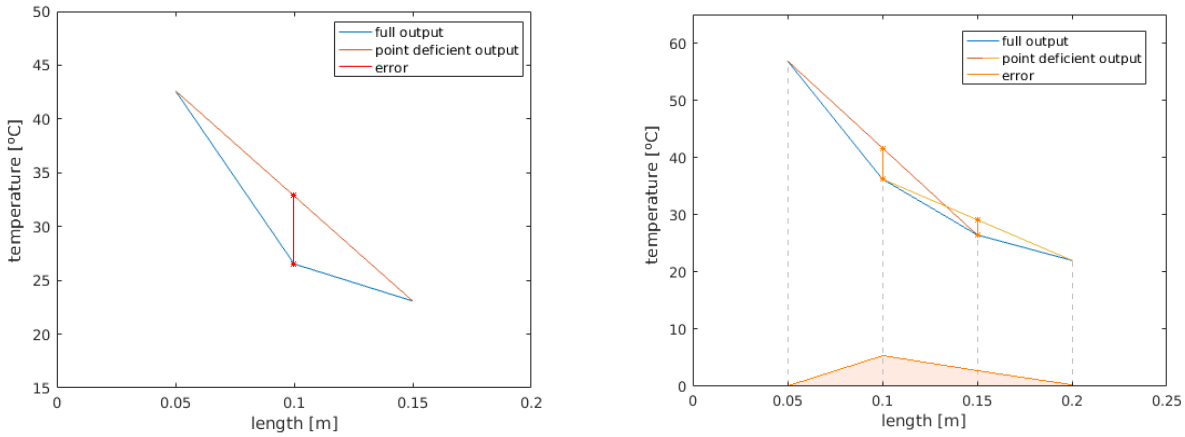
Figure 6.3: Inversion algorithm to estimate the temperature across the medium

One can further note that this method is very dependent of how accurately is the mathematical model being implemented, which means that the error associated with this inversion of information will grow very fast with some details and assumptions regarding the simulated model, so this means that extra care is needed for the simulation to take into account the most important phenomenons that rule the dynamics in question.

To understand how closely a method can fit a curve representative of real world conditions, the analysis has to pass through some form of error accounting or estimation. Without knowing the exact shape of the real curve it is not possible to know the absolute error in every point of the curve, but it can be approximated trough estimation. One way to do it is to compare the computed curve to the optimal fit to the values in question. But since the exact solution is not studied in this work and the best fit possible is the actual aim for this dissertation, that method is not applicable. Alternatively, what is going to be studied is the error related to an increase in distance from the nearest known value. To do that the idea is to stop considering some of the known values along the curve, and study how the estimation for those otherwise known points compares to the real data.

Let us consider tree points fitted with a trapezoidal interpolation for simplicity as can be seen in Figure 6.4. The idea is to stop considering the midpoint and directly interpolate the other two points together. And as can be seen in Figure 6.4a, there is a gap between the ignored point and the line that links the other two points together, that gap is going to be considered as the error that would exist at that point.

At this point, the explained idea is not yet very useful to analyse the accuracy of the methods, so to implement it in a useful way the choice was to interpolate all the even points and all the odd points separately like the Figure 6.4b suggests, and to consider all the gaps



(a) Relative error for a measurement point

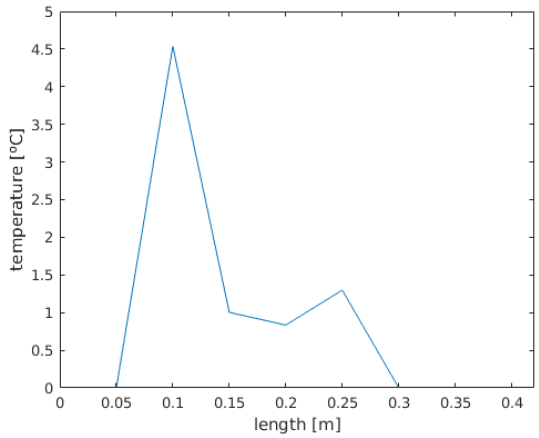
(b) Error estimation technique

Figure 6.4: Error estimation demonstration schemes

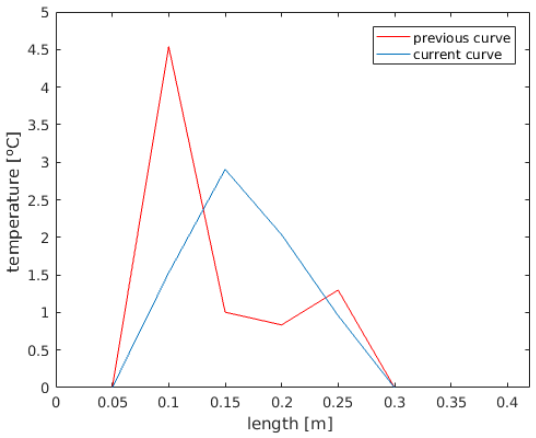
between the known points and the lines that connect its neighbours. The values for the estimated error are then not representative of the error of the full interpolation, but rather, an error representative of a different condition with less known points in a given area, which may not be closely related to the real error, but it is related with the error of the interpolation method and can be used to evaluate it. The error values are then going to be connected using trapezoidal interpolation regardless of the interpolation curve that is being analysed, to guarantee fairness in the comparison of two different methods for estimating the curve of values. The number that is then going to be compared between methods is the mean value of the error plotted in Figure 6.4b.

The plots shown in Figure 6.5 display the mean absolute error values for the sensor positions computed as explained before for both trapezoidal interpolation 6.5a and the developed inversion method 6.5b.

As shown in Figure 6.5, the estimated error for the numerical inversion is not a lot better than the simplest interpolation possible. This means that for this very simple case it may be better to use some interpolation method, but more complex heat distributions could have different results. However, the rest of the work was to be done with more complex systems, like odour dissipation, so this setup was not used to run further tests.



(a) Error for the linear interpolation (average = 1.92 °C)

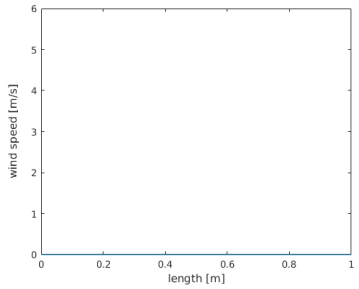


(b) Error for the inversion method (average = 1.857 °C)

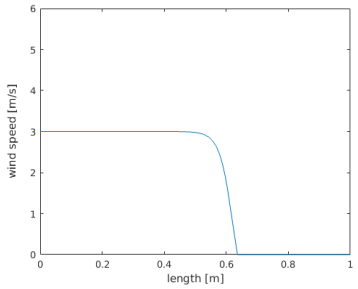
Figure 6.5: Estimated error curves for temperature diffusion

6.2 Navier-Stokes

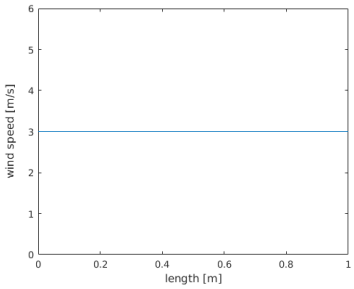
The first test towards using environmental data was set around the motion of air as a fluid and the Navier-Stokes equations. The simulation was run for a one meter long path with a fluid viscosity of $\mu = 1.81 \cdot 10^{-5}\ kg/m \cdot s$ and a initial null speed. After the simulation had begun, a boundary condition was set to 3 m/s, as if a turbine was turned on in the system, and the opposite boundary condition was set as a boundary with null derivative.



(a) Simulation for t = 0 s



(b) Simulation for t = 0.2 s

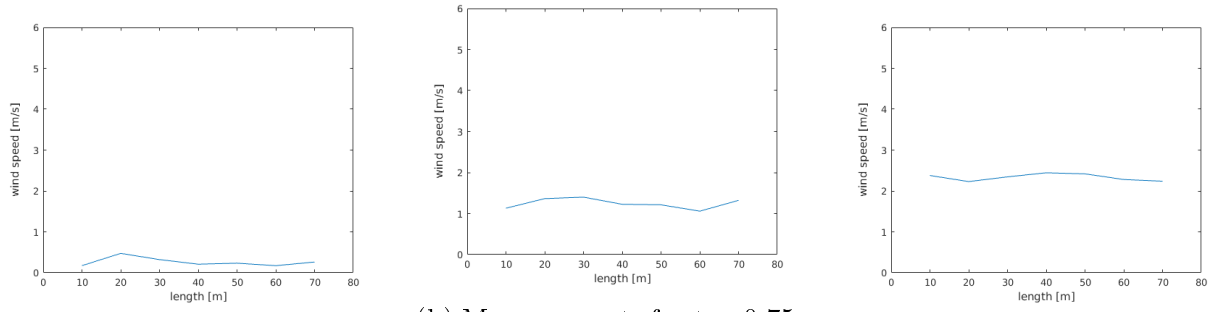


(c) Simulation for t = 0.4 s

Figure 6.6: Air wave propagation simulation

As represented in Figure 6.6, the speed at which the velocity gradient moves is greater than that of the air for the reasons detailed in (Section 5.2). However, in this simulation the velocity gradient does not travel at the speed of sound because of the limitations of the implementation of the numerical method and the simplifications to the Navier-Stokes equations. But despite that, the measured data suggests that there is some phenomenon that is not accounted for in the simulation, and what it is the delay that is imposed by the

turbines mechanical angular inertia. What this means is that the slower speed up of the turbine and the thermal inertia of the sensors eliminates any characteristic from the curve that could help to detect the front wave.



(a) Measurements for t = 0 s (b) Measurements for t = 0.75 s (c) Measurements for t = 1.4 s

Figure 6.7: Measured wind speed across one meter

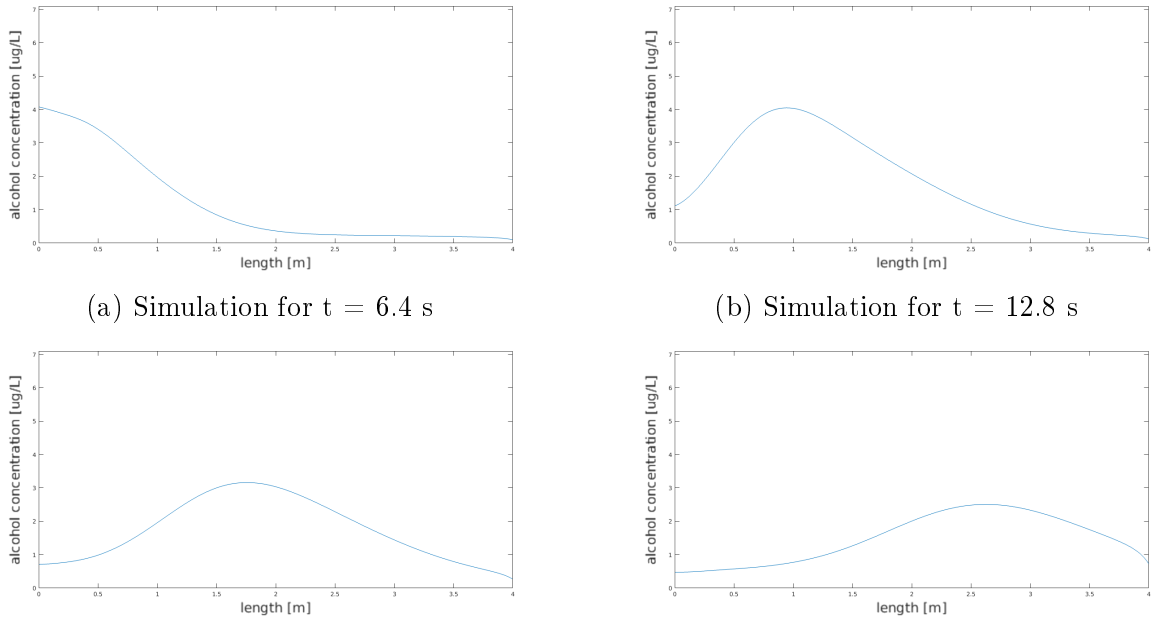
This means that one must chose between creating a pressure wave similar to that of an explosion in order to be able get measurements similar of the ones predicted in the simulation, or tweaking the simulation as of having a slow raise in velocity like the one pushed by a turbine. Both choices are impractical, the first one is difficult to get right and needs ultra sensitive anemometers, and the second would dis-characterise the wave just like it can be seen from the measurements in Figure 6.7.

To work around this problem, the only valid solution would be to re-design both the simulator and the physical setup to be used as two-dimensional tunnels, but the plan for this dissertation did not reach the point where all the tests would be redone for one more dimension, and it did not make much sense to try adding a dimension just for the case of fluid behaviour, since the main focus would be to study the dissipation of odour through air.

Regardless of the air velocity through the tube being approximately constant in space, it is still useful to measure it. And although it is not analogue to a two-dimensional situation, averaging all the values measured for wind velocity at a given instant is a very good estimation for the speed of the flow of air along every point of the tube. This is the way that the air velocity is acquired to be input on the advection-diffusion model studied ahead.

6.3 Advection-Diffusion

The final setup was meant to test the diffusion of gases through moving air. For the simulation, a four meter long path was considered, and the diffusion coefficient had an experimental value of 0.08 m^2/s. Furthermore, the air velocity to consider was measured using the method described above. For this simulation, the step in space was of 1 cm and the step in time of 0.02 s.



(a) Simulation for t = 6.4 s

(b) Simulation for t = 12.8 s

(c) Simulation for t = 16.5 s

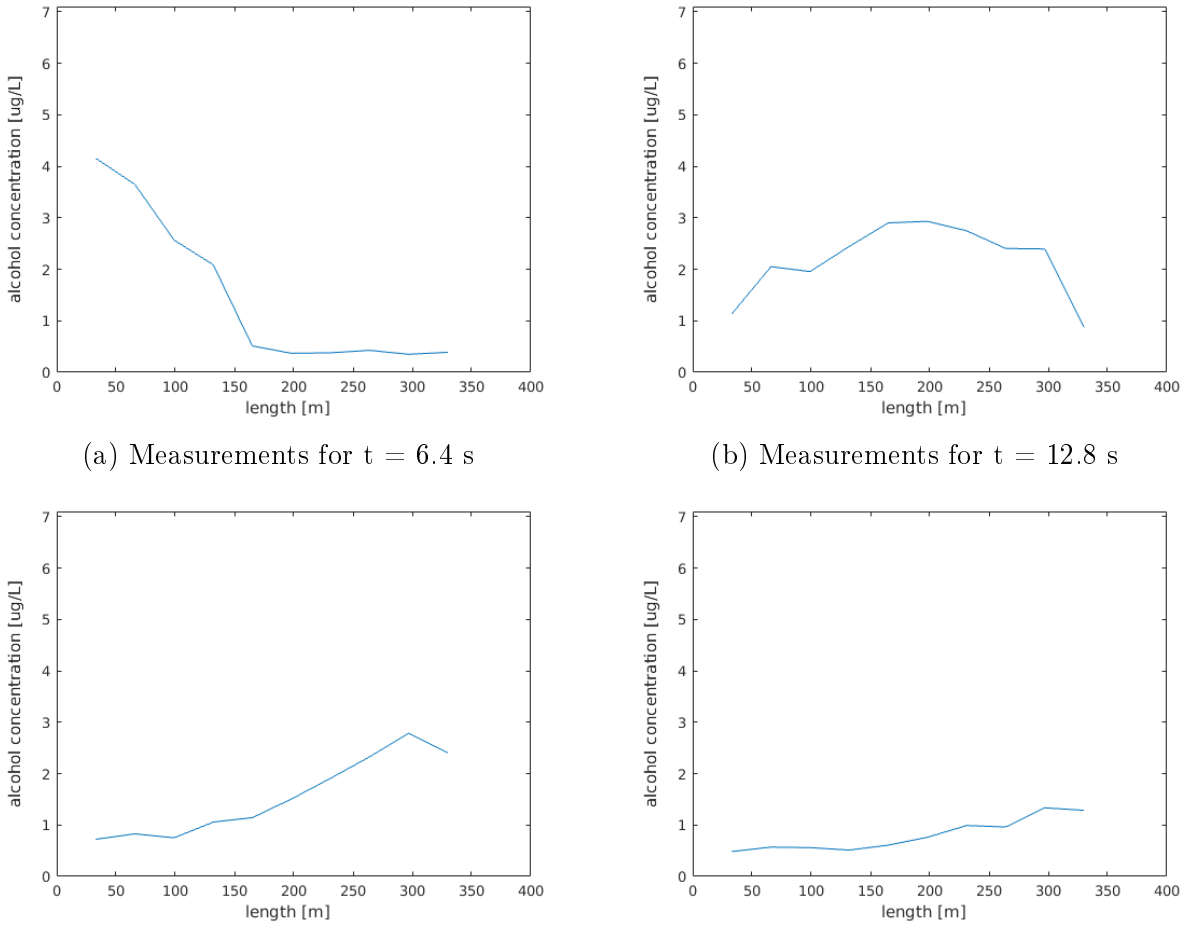
(d) Simulation for t = 21 s

Figure 6.8: Simulation for odour dispersion in a 4 m domain

As plotted in Figure 6.8, a wave was reproduced by changing the first boundary condition to a fixed value and returning it to zero shortly afterwards. As the wave propagates at the same velocity as the air, its shape changes because of the diffusion component of the equation. This change in shape would occur at the same rate if there was no air movement, but in a real world environment the rate of dissipation goes up along the velocity of the air, because an increase in velocity also increases the probability of small vortices forming, but for this test that effect was considered as negligible.

For the environmental gas propagation test, the four meter long tube was used with ten sensors (Section 4.2.2) evenly spread across 3 meters of that tube. The odour was introduced as a cloud of alcohol vapour which was created by bubbling air through a 96% alcohol solution and storing the contaminated air inside a large closed cylinder near the entrance of the wind tunnel. To release the mixture of air and alcohol vapour, the cylinder was opened on both

sides and the air was slowly aspirated into the tunnel.



(a) Measurements for t = 6.4 s

(b) Measurements for t = 12.8 s

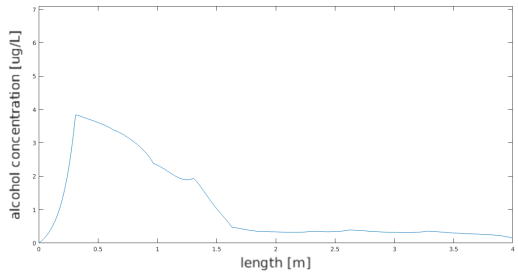
(c) Measurements for t = 16.5 s

(d) Measurements for t = 21 s

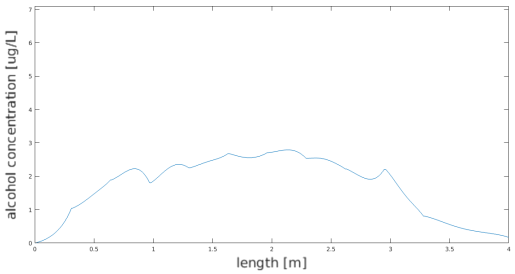
Figure 6.9: Odour sensing for ethanol concentration in a wind tunnel

Similarly to what was simulated, a wave measured in Figure 6.9 propagates at the same speed as the air that carries it. Although there is some apparently random deviation in the measured values that is caused by the uniformities of the gaseous cloud. Also, the reason behind the consideration of a loss term in the mathematical model is because the gaseous cloud is not as wide as the tunnel, which means that some of the gas will disperse in the directions perpendicular to the direction of the tunnel, which effectively results in lost gas to the measured dimension. If the tunnel was long enough, the gas would reach all the walls and this effect would slow down, so, if this test gets redone in a longer tunnel it becomes important to release the gas in such a way that it is evenly spread across the cross-section of the tunnel at the beginning.

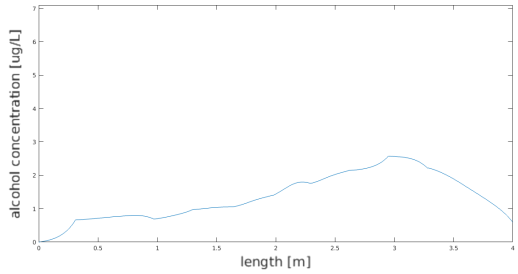
With all the needed data gathered it was then possible to test the developed inversion algorithm as explained before (Section 5.3).



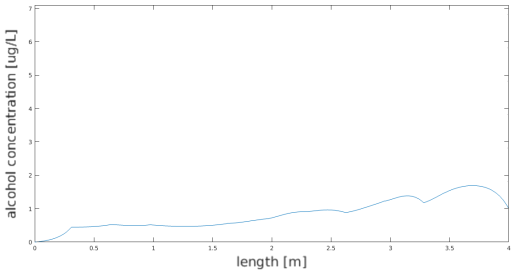
(a) Inversion at t = 6.4 s



(b) Inversion at t = 12.8 s



(c) Inversion at t = 16.5 s

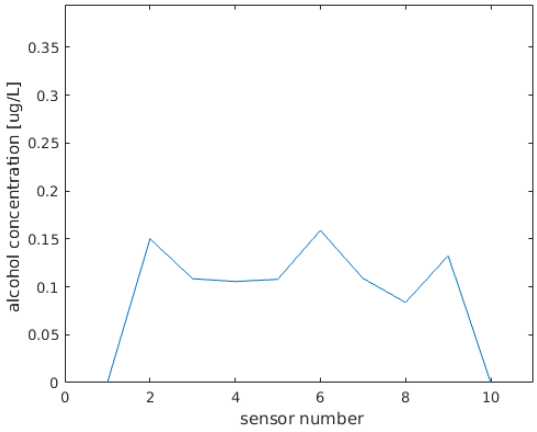


(d) Inversion at t = 21 s

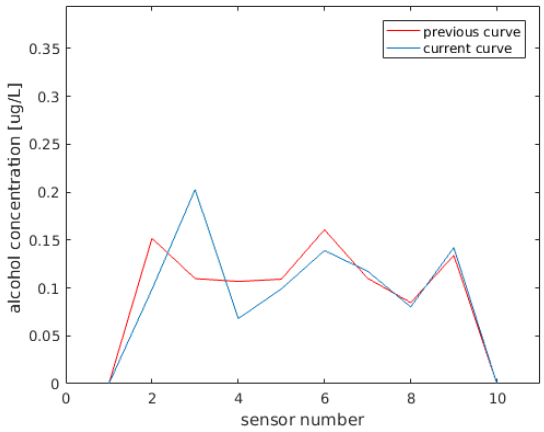
Figure 6.10: Ethanol concentration data assimilation algorithm in a wind tunnel

The extracted curves 6.10 roughly describe what is the expected behaviour from the gas cloud dispersing trough moving air, but some measured values differ from the predicted with the mathematical model which is the cause for the characteristic texture of this curve. Those errors in prediction are caused by small and slow vortices that spread the gas in a inconsistent way. So this problem can only be solved or mitigated by changing the physical methods used in these tests. Although some of the values may fluctuate away from the expected, this integration is a very good approximation of the real world model. In fact the estimated error is quite low given all the unaccounted disturbances in the behaviour of the gas. The averaged estimated error for this method was plotted in Figure 6.11a compared to a simple trapezoidal interpolation 6.11b. The respective standard deviations can be found in 6.11c and 6.11d.

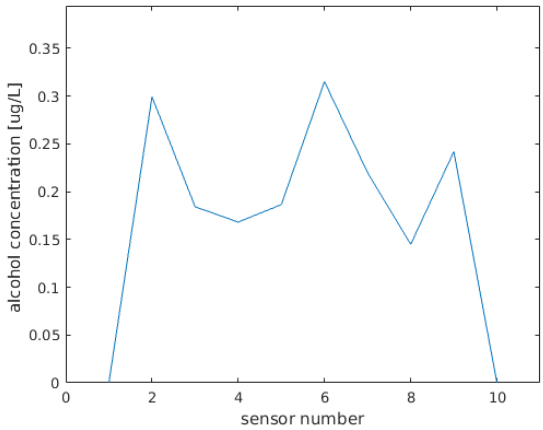
From this simple comparison, it can be noted that the overall estimated error for the developed algorithm is better than a trapezoidal interpolation, but it still has a considerable error which can compromise its usefulness for some applications in this form. To confirm these results, the data was also fitted with third degree polynomial curves 6.12 and the error was estimated in the same way as before.



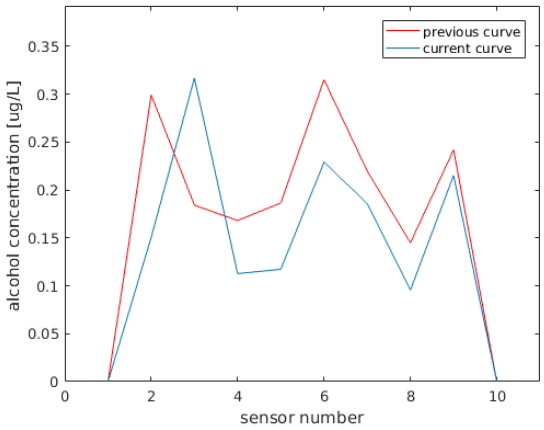
(a) Trapezoidal interpolation mean values (average = 0.154 $\mu g/L$)



(b) Inversion algorithm mean values (average = 0.151 $\mu g/L$)



(c) Trapezoidal interpolation standard deviation (average = 0.28 $\mu g/L$)



(d) Inversion algorithm standard deviation (average = 0.227 $\mu g/L$)

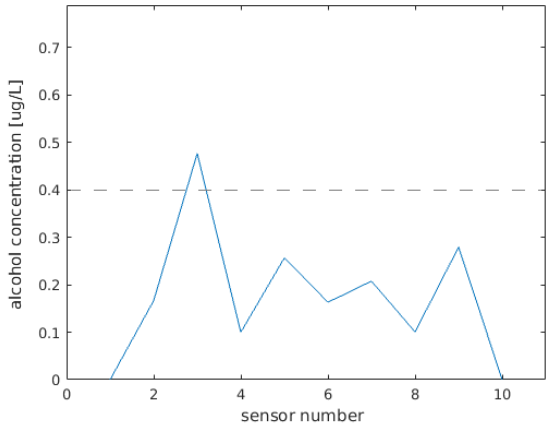
Figure 6.11: Trapezoidal interpolation and inversion algorithm error statistics comparison

As this result shows, the polynomial fitting generates an even higher error when estimated in the described way, the fitting could be done with a higher order polynomial equation, but that was tried and the results were worse, so it was kept out of this document for lack of relevance.

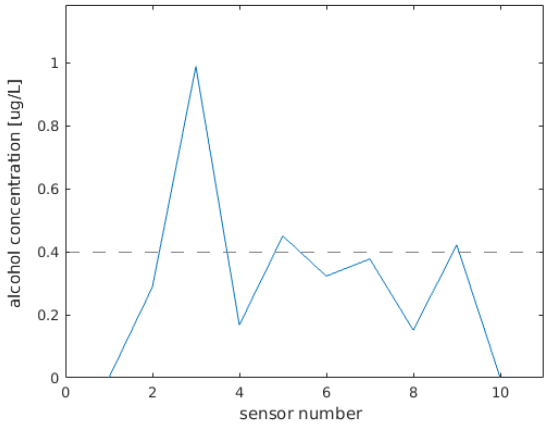
Although the estimated error of the developed method is fairly good, it can still be improved. For that propose, came the idea of correcting the velocity term based on the lag or advance of the odour propagation between sensors, which is an indicator of an incorrect wind speed term. To adjust the velocity term, the first idea was to change the velocity only in the time dimension, since we know that the wind speed is constant along the length of the tube.

The resulting error from this additional correction to the velocity term is much lower than before, so it may be worth the extra computational effort required.

Another idea in the same frame of though was to also adjust the velocity term in the



(a) Polynomial fitting mean values (average = 0.276 $\mu g/L$)



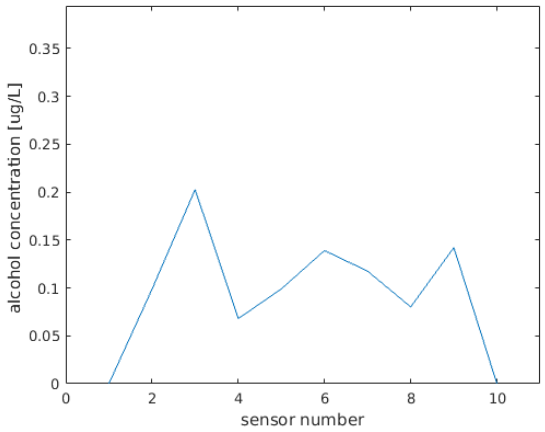
(b) Polynomial fitting standard deviation (average = 0.499 $\mu g/L$)

Figure 6.12: Polynomial fit error statistics

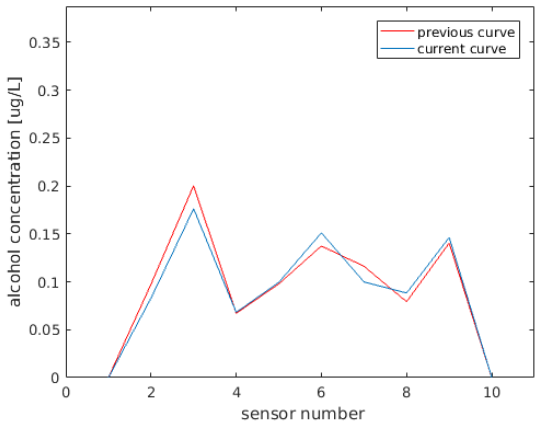
spatial dimension, which would automatically lose its equivalence to the actual air speed, but the gas particles can, sometimes, change their position relatively to the airflow because of the vortices created by small imperfections in the tunnel. This makes for an even more accurate algorithm, although it may not be the best choice in some applications because the velocity term can diverge from reality and this method would be not practical to implement on a real-time system.

These are the best results archived in this work, and can still be further improved both by improving the algorithm and the quality of the environmental framework and sensors.

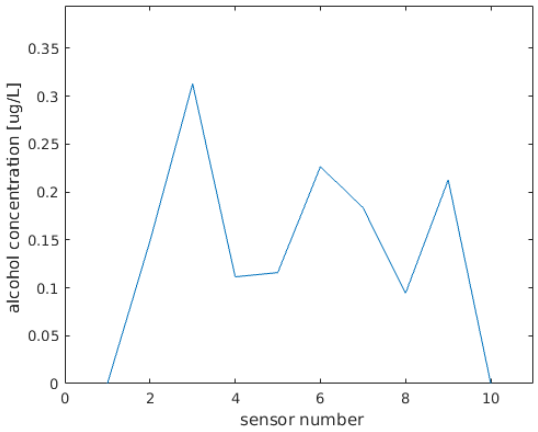
Also, the resulting velocity term is more inconsistent across the spatial dimension in the presence of turbulence, which means that this can be further analysed in order to detect and quantify the presence of chaotic vortices or turbulent regions in the flow.



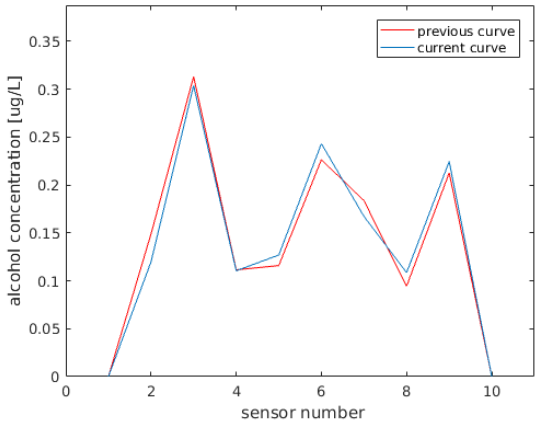
(a) Inversion algorithm mean values (average = 0.151 $\mu g/L$)



(b) Adjusted velocity method mean values (average = 0.147 $\mu g/L$)

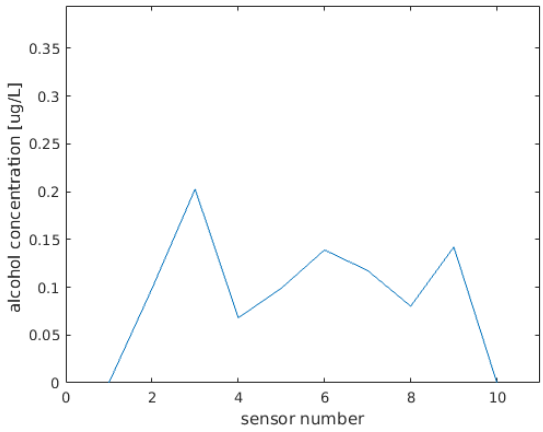


(c) Inversion algorithm standard deviation (average = 0.227 $\mu g/L$)

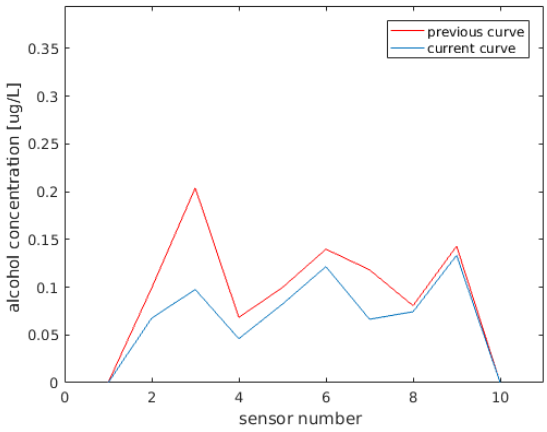


(d) Adjusted velocity method standard deviation (average = 0.226 $\mu g/L$)

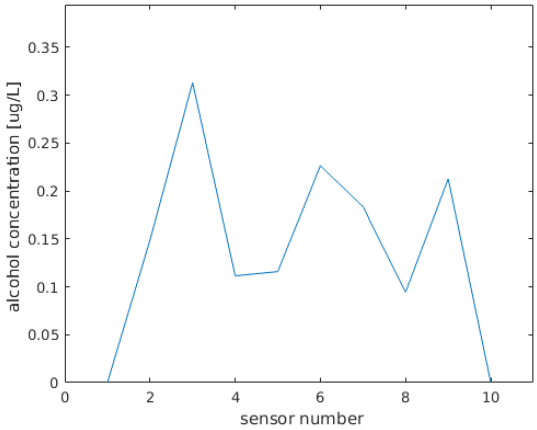
Figure 6.13: Improved velocity term method error statistics



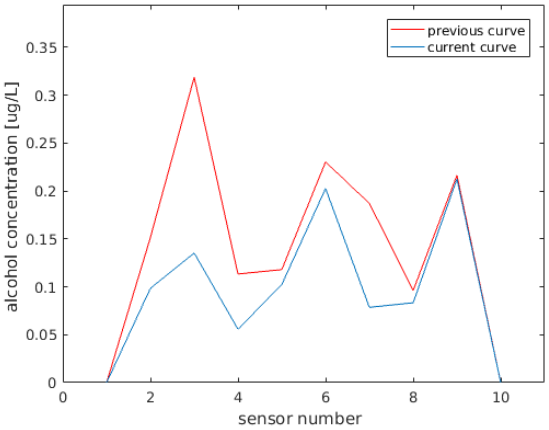
(a) Inversion algorithm mean values (average = 0.151 $\mu g/L$)



(b) Adjusted velocity method mean values (average = 0.109 $\mu g/L$)



(c) Inversion algorithm standard deviation (average = 0.227 $\mu g/L$)



(d) Adjusted velocity method standard deviation (average = 0.1535 $\mu g/L$)

Figure 6.14: Improved velocity term method error statistics assuming non constant flow

7 Conclusion

Given the achieved results, it is reasonable to conclude that the developed method as interesting potential in mapping odour concentrations, and with the adaptive velocity term, it can even be useful to determine the velocity of the fluid without the use of anemometers.

This is a method that focus on being coherent with the physical phenomena involved, which means that it can be more reliable without losing any generality which is the most common issue around this kind of problems. Usually, the accuracy of a method and its generality diverge, which forces researches to choose a good balance between the two for one specific problem, but by directly using an adequate mathematical model, it is possible to work around that trade-off and get an accurate method capable of solving more general problems. This method can also be very useful to predict the behaviour of a fluid in a larger scale with very few correction points, because although the error increases with distance, it increases at a slower rate than other methods. In regard to the Gaussian plume method, the developed algorithm can replace it in almost every application that does not require very high computational speeds, because the Gaussian plume is equivalent to a simple or averaged case for an advection diffusion equation. The developed method has the particularity of following the physical model for the behaviour of the considered fluid mechanics, which has valuable prediction features while keeping a low enough estimated error to still be useful in a wide range of applications.

In a choice between this method and some of the others referred, one can face some facts that may, in some cases, lead to circumstances where this method is not the best choice. One of those facts is the computational power demand, which is higher than that of some of the other methods, and although a matrix was used instead of a loop, the algorithm is still quite heavy to compute. Also, to use this algorithm correctly and in a way that minimizes its error, some care is needed. The initial and the boundary conditions need to be set, a useful diffusion coefficient needs to be estimated, since the molecular diffusion coefficient does not represent the true behaviour of the fluid. Also all losses due to the chemical exiting the

system trough processes like oxidation or deposition need to be considered. Every physical factor that can influence the behaviour of the gases can be considered and will decrease the error of the method.

7.1 Future Work

Although this study produced promising results, there are several aspects that can be further developed to improve upon this work, such as the implementation of a more robust algorithm for velocity feedback, diffusion coefficient and loss coefficient estimation. Overall, almost every constant or variable can be tweaked in a dynamic way in order to improve the prediction capability of the algorithm. At a later stage it may even be possible to estimate the error of each sensor and adjust the respective calibration curve.

The best improvement that can be made to the current algorithm is to add one more spatial dimension, this easily enables the study of most two-dimensional areas with the exception of very uneven surfaces like mountains and valleys. Although this work was developed to study and solve one-dimensional problems, it can be adapted or combined with another method like the one used by [11] where a number of assumptions are considered regarding the behaviour of air and the chemicals that can be spread by it. The idea to merge the two distinct methods would be to draw the path of the fluid with close flow lines in which line can be considered as a one dimensional problem and there is a transfer process between lines trough diffusion. Although this idea might work, it is quite complex to implement and the true mathematical models are likely to be the best path to chose.

Among the known problems, there is one that can become more evident after the addiction of a second spacial dimension, which is turbulence. The greater the characteristic length of the objects, the higher the Reynolds number, and that comes with a more chaotic flow. One idea to solve this is to replace the Navier-Stokes equations for the RANS equations, and maybe to use a dynamic map for gas diffusivity in which the diffusion coefficient can change according to the turbulence of the flow.

A possible application of a method that is coherent with the physical model of the problem is to differentiate between the errors caused by noise-full measurements and by uncounted factors, such as compromised sensors or changes in the environment. This may enable the development of an algorithm capable of adapting to changes in the environment and estimate the positions and some characteristics of new obstacles to the flow or substances capable of neutralising some airborne chemicals.

Lastly, a possible improvement to one of the two-dimensional methods just described would be to add the third spatial dimension, making it a method that can fully describe any problem with Newtonian fluids and suspended or diluted chemicals that disperse according to their diffusivity and the movement of the fluid.

References

[1] Gary D. Knott. *Interpolating Cubic Splines.* Birkhäuser Basel, 2000.

[2] Ali Marjovi and Adrian Arfire and Alcherio Martinoli. High Resolution Air Pollution Maps in Urban Environments Using Mobile Sensor Networks. *International Conference on Distributed Computing in Sensor Systems*, pages 11–20, 2015.

[3] Jay A. Farrell, Shuo Pang, and Wei Li. Plume Mapping via Hidden Markov Methods. *IEEE Transactions on Systems Man and Cybernetics - Part B: Cybernetics*, Vol. 33(No. 2), 2003.

[4] Shuo Pang and Jay A. Farrell. Chemical Plume Source Localization. *IEEE Transactions on Systems Man and Cybernetics - Part B: Cybernetics*, Vol. 36(No. 5), 2006.

[5] Andrew F. Bennett. *Inverse Modeling of the Ocean and Atmosphere.* Cambridge University Press, 2002.

[6] W. Lahoz, B. Khattatov, and R.Ménard. *Data Assimilation, Making Sense of Observations.* 2010.

[7] Jan Mandel, Jonathan D. Beezley, Janice L. Coen, and Minjeong Kim. *Data Assimilation for Wildland Fires.* 2009.

[8] Ian G. Enting. *Inverse Problems in Atmospheric Constituent Transport.* Cambridge University Press, 2002.

[9] Noel de Nevers. *Air Pollution Control Engineering.* McGraw-Hill Higher Education; 2nd edition, 2000.

[10] Naomi Ehrich Leonard, Derek A. Paley, Francois Lekien, Rodolphe Sepulchre, David M. Fratantoni and Russ E. Davis. Collective Motion, Sensor Networks, and Ocean Sampling. *Proceedings of the IEEE*, Vol. 95(No. 1):48–74, 2007.

[11] Xiang Gao and Levent Acar. Using a mobile robot with interpolation and extrapolation method for chemical source localization in dynamic advection-diffusion environment. *International Journal of Advanced Robotic Systems*, Vol 5(No 2), 2016.

[12] Joel H. Ferziger and Milovan Peric. *Computational Methods for Fluid Dynamics*. 1997.

[13] COMSOL Inc. What Are the Navier-Stokes Equations?, December 2016. See `https://www.comsol.com/multiphysics/navier-stokes-equations`.

[14] G. K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge University Press, 2000.

[15] James Sturnfield. Understanding the transition flow region through comsol multiphysics modeling. *COMSOL Conference, Boston, MA*, 2015.

[16] James Ransley. Molecular flow module: Simulate rarefied gas flows in vacuum systems, December 2016. See `https://www.comsol.com/blogs/molecular-flow-module-simulate-rarefied-gas-flows-in-vacuum-systems/`.

[17] Grétar Tryggvason. The equations governing fluid motion. Spring, 2011. See `http://www3.nd.edu/~gtryggva/CFD-Course/2011-Lecture-4.pdf`.

[18] R. Rubinstein, C. L. Rumsey, M. D. Salas and J. L. Thomas. Turbulence Modeling Workshop. In *ICASE Interim Report, No. 37*, 2001.

[19] Philip J. W. Roberts and Donald R. Webster. Turbulent Diffusion. See `http://docplayer.net/14756664-Turbulent-diffusion-philip-j-w-roberts-and-donald-r-webster-1.html`, 2015.

[20] Tom Benson. Reynolds Number, December 2016. See `https://www.grc.nasa.gov/www/BGH/reynolds.html`.

[21] Endre Suli. *An Introduction to the Numerical Analysis of Partial Differential Equations*. University of Oxford, 2005.

[22] Christopher Rumsey. Turbulence Modeling Resource, December 2016. See `https://turbmodels.larc.nasa.gov/`.

[23] Walter Frei. Which Turbulence Model Should I Choose for My CFD Application?, December 2016. See https://www.comsol.com/blogs/which-turbulence-model-should-choose-cfd-application/.

[24] Michael L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. 1999.

[25] D. A. Harville. *BLUP (Best Linear Unbiased Prediction) and Beyond*. 1987.

[26] MathPages. Propagation of Pressure and Waves, December 2016. See http://www.mathpages.com/home/kmath569/kmath569.htm.

[27] Julian L. Davis. *Mathematics of Wave Propagation*. Princeton University Press, 2000.