



Identification and tracking of chemical leaks and hazards with drone swarms

Einon McGrory-Perich

992697, emcgroryperi@student.unimelb.edu.au

Xing Yang Goh

1001969, xingyangg@student.unimelb.edu.au

Antony Kartomi Thomas

911656, antonyt@student.unimelb.edu.au

Executive Summary: Advances in technology have resulted in the miniaturisation of traditionally large chemical sensors that can be attached to lightweight autonomous vehicles, such as unmanned ground vehicles (UGV) and unmanned aerial vehicles (UAV). This has led to new possibilities for navigating hazardous environments, such as bushfires and chemical leaks, with greater capabilities and of no threat to human life. Potential applications could be the tracking of bushfire boundaries, or chemical plumes in disasters and using this real-time information to predict future spread for early evacuation or safe route planning. To track these boundaries using drone swarms, coverage control algorithms can be applied, which currently exist predominantly in theory. This project will first design an algorithm that will allow for identifying a boundary within a specified region and monitoring this boundary using an appropriate formation, then simulating this algorithm to verify its expected functionality. Following this, an aerial vehicle hardware implementation will be created in a hybrid real-world simulation environment for further testing of the algorithm's reliability to non-ideal dynamics and discretised delays to make way for future development in this area.

Contents

1	Introduction	1
2	Literature Review	2
2.1	Plume modelling and tracking	2
2.2	Extremum seeking control	3
2.3	Swarm formation	3
3	Methodology	4
3.1	Problem formulation	4
3.1.1	Plume model	4
3.1.2	Model assumptions	5
3.1.3	Communication network	5
3.1.4	Problem goal	6
3.2	Controller architecture	7
3.2.1	Setpoint seeking controller	7
3.2.2	Formation Controller	11
3.3	Hardware Implementation	13
4	Results & Simulations	14
4.1	Controller gain tuning	14
4.2	Performance on different plume conditions	15
4.3	Hardware results	17
5	Discussion	18
5.1	Setpoint seeking controller	18
5.1.1	Parameter Selection	19
5.1.2	Timescale Separation	19
5.1.3	Plume Model	19
5.2	Formation controller	19
5.2.1	Performance	20
5.2.2	Convergence around concentration boundary	20
5.3	Newton-like optimiser	21
5.4	Stability of controller	21
5.5	Testing on hardware	21
6	Conclusion & Recommendations	22
6.1	Future improvements	22

1 Introduction

The need for real-time monitoring of dynamic and dangerous environmental disasters has become increasingly prevalent, with growing examples of these disasters posing a significant threat to the safety of surrounding communities, where there are currently limited practices available to monitor and track these disasters as they unfold to inform an appropriate emergency response and resource allocation.

Following the devastating bushfires of 2019-2020, a Royal Commission was launched to evaluate Australia's current abilities to respond to extreme fires and monitor the air quality (Binksin et al., 2020). This identified the practices employed are of limited utility and require immediate and significant improvement to mitigate the ever-growing risks that have the potential to overwhelm presently available emergency services. Additionally, the train derailment disaster in East Palestine, Ohio which produced a toxic chemical plume was poorly managed, where true chemical levels in both air and water were consistently underrepresented (Messenger, 2023). The response to monitor and control the spill speaks to the dire necessity of better technology and preparation.



(a) 2019-2020 Bushfires



(b) 2023 Ohio train derailment

Figure 1: Examples of recent disasters involving plumes

Advancements in recent technologies such as the miniaturisation of real-time and accurate gaseous chemical sensors by RingIR pose an opportunity to address this issue. These sensors can be deployed on lightweight and scalable platforms, such as Drones, to monitor a plume's development in a safe and cost-effective manner. Utilizing an autonomous network of drones outfitted with such a sensor presents a potential low-risk and cost-effective mechanism for monitoring hazardous chemical plume boundaries.

This paper proposes a method of dynamically tracking the boundary of a chemical plume using drone swarms. Combining the benefits of adaptive, model-free controllers with a cooperative controller, the aim is to track the boundary associated with a desired chemical concentration. This approach is scalable due to its distributed nature and does not require significant knowledge about the underlying plume dynamics. We will demonstrate that the setpoint seeking controller can settle at the target concentration of the plume using the local information available to it, which differs from existing methodologies that require assumptions about the properties of the plume. Following this, we implement a distributed cooperative controller that uses a predetermined communication topology to form an equidistant formation along the perimeter associated with that concentration to properly track its boundary.

2 Literature Review

2.1 Plume modelling and tracking

Methods in monitoring and tracking air quality after bushfires utilize a variety of technology. Johnston et al. (2010) explores current strategies including fixed-site particle monitoring, which utilises gravimetric samplers to weigh residual material after filtering nearby air. These sensors require manual collection and processing, which is not ideal for monitoring real-time smoke pollution. In cases of extreme bushfires, the filter can clog and fail completely. Another strategy identified utilises a portable system using beta-attenuation monitors or nephelometers. This technology measures material concentration by observing the attenuation of beta-particles as it passes through a filter and measures the scattered light due to particle concentration respectively. It is currently common practice for US Forest Service personnel to use this type of system as it is easily transportable and well-suited for shorter-term events, however, this technology is typically less accurate than gravimetric samplers.

Examples using sun photometers or other methods of measuring aerosol optical density with ground or satellite-based systems were not found for public real-time monitoring of fire smoke, despite being commonly utilized for environmental research studies (Noh et al., 2009). The proposed sensor designed by RingIR employs a technique called Infrared Spectroscopy (IR) that characterizes chemicals by observing their interaction with infrared radiation by absorption, emission, and/or reflection (Guenzler & Gremlich, 2002). Kuze et al. (2012) explores the applicability of this technique to monitor air pollution in urban environments, showing efficacy in identifying airborne pollutants through controlled verification experiments. In combination with wind LIDAR sensors, this technology was shown to successfully identify sources of chemical pollution, namely from the nearby industrial zone of the urban area.

A necessary component in the development of the proposed algorithm is an accurately simulated environment for testing. One common approach involves utilising a Gaussian Plume model, where the transport solution of the concentration is a Gaussian function over the x and y plane at any given z height (Holzbecher, 2013). Pang and Farrell (2006) leverages this model for localising chemical plume sources using Autonomous underwater vehicles, noting that predictions based on the Gaussian model yielded high accuracy results compared to experimental data.

However, this model is not ideal for short-term simulations, where it may substantially underestimate instantaneous concentrations of a plume (Jones, 1983). Quantifying the mechanism of the processes causing these fluctuations is critical in producing an accurate representation but these processes are difficult to express. The instantaneous wind vector at a plume's source advects material away and continuously changes according to the immediate climate conditions causing intermittency where concentrations drop to either zero or close to zero (Kollmann, 2019). Even with these short-term disadvantages, Farrell et al. (2002) posed that the Gaussian model produces a reasonable representation of long-term exposure of the time-averaged concentration of a plume, where fine-scale characteristics are of little interest. According to Porter and Vasquez (2006), use of the Gaussian model has been identified as a crucial proponent in reducing computational loading as it ignores the individual diffusion of particles and produces steady state behaviour of plumes, which is supported as a reasonable assumption of chemical plume behaviour (Casbeer et al., 2005).

A more accurate 2D advection-diffusion model has been utilized to explore a distributed control algorithm for a network of robots designed to monitor dynamic plumes (Wang & Guo, 2019). This model allows for the effective modelling of dispersion, which describes the combination of diffusion and advection (Stockie, 2011). This model includes the variation of eddy diffusion coefficient K , immediate weather conditions, and time from release.

2.2 Extremum seeking control

Extremum seeking control is a real-time optimising control technique that can handle model-free, uncertain plant dynamics with time-varying optimal operating conditions. Recent advancements in the field have been driven by rigorous mathematical stability and robustness analysis of ESC with sinusoidal perturbation-based methods on non-linear plants (Krstic & Wang, 2000). Ariyur and Krstic (2003) explores dither-based signals that probe local behaviours of the unknown plant for optimisation, using averaging analysis and singular perturbations to prove convergence properties by making connections to classical two-time scale problems (Teel et al., 2003). Hazeleger et al. (2019) adapts this extremum seeking approach for setpoint control by a parameterised objective function by means of a step-like basis function in a PID-based learning controller that improves transient performance in frictional motion systems.

Extension of these principles on multi-agent systems is explored in Khong et al. (2014), where an autonomous swarm of agents are employed to locate and track a dynamically changing source, using a discrete-time gradient descent approach in the presence of noisy measurements and showing semi-global convergence to an optimum with probability one. Liao et al. (2020) explores distributed implementations of constrained extremum seeking to optimise the signal strength along the communication chain of a network on quad-rotor vehicles for homogeneous and heterogeneous signal transmission pathways.

Nesić et al. (2010) explores the estimation of derivatives of an unknown steady state reference to output map, using singular perturbation and averaging techniques to approximate optimisation methods that leverage higher order derivatives, with parameter tuning methods to achieve semi-global practical extremum seeking. Applications of using a Newton-like optimisation scheme are used with an adaptation law that reduced the controller's curvature dependence to reduce the amplitude of thermoacoustic oscillations in a gas-turbine combustor (Moase et al., 2010).

Recent methods have explored increasing the convergence speed of extremum seeking, where an observer-based extremum seeking that utilises high-frequency sinusoidal dither is explored in Moase and Manzie (2012), showing arbitrarily fast convergence of a non-linear plant to a small neighbourhood around a minimum using knowledge of the plant's input and output dynamics. Exploration of a dual-mode extremum seeking controller uses Lie bracket averaging techniques to overcome the time-scale limitations for real-time optimisation problems with time-varying optimal state trajectories (Guay & Atta, 2018).

2.3 Swarm formation

In recent years, there has been significant interest in the applications of drones for monitoring environmental hazards. Wang and Guo (2019) and Aznar et al. (2014) proposed the usage of drone swarms to track chemical plumes in marine environments. Saffre et al. (2022) and Yang et al. (2021) have suggested their use in monitoring bushfires and measuring pollution spread (Bolla et al., 2018). Research has also shown that aerial vehicles do not reduce the ability of a drone to measure chemical concentrations in the air despite their dynamics (Gerhardt et al., 2014).

Drone swarms are suitable for monitoring large-scale plumes as they can form a robotic sensor network (RSN) and provide real-time coverage and information (Dunbabin & Marques, 2012). The advantages of multiple agent systems using a distributed control law are that they are more robust to vehicle failure, more scalable, can lead to faster plume monitoring and provide a greater resolution. Cooperative controllers direct a multi-agent system to work together through communication to achieve a common goal, such as searching, exploring or monitoring an environment.

There have been several proposed methods for the tracking of dynamic boundaries. According to Cao and Fierro (2006), a cooperative controller tracks the perimeter of a plume by estimating the shape and converges to the boundary using a dynamic sensor net. This method requires prior knowledge about the

boundary to track the edge, which is typically unknown. Clark and Fierro (2007) explores a different method that uses an attractive potential function for all agents once one has found the boundary before a cyclic pursuit formation controller causes all agents to spread evenly. Their algorithm only approaches the boundary once identified through random movements, so it isn't suitable for most scenarios.

More desirable methods use optimisation algorithms such as gradient-based methods to move towards the desired boundary. Wang and Guo (2019) utilises a distributed consensus Luenberger observer estimates a plume modelled with a 2D advection-diffusion partial differential equation. A cooperative controller then spreads agents out with movements tangential to the desired perimeter until all agents are equidistant. As this method uses a Luenberger observer, it requires some estimate about the plumes model and parameters, which would not be available to the algorithm. This method would also fail in circumstances where the plume properties change. This limitation can also be found in several similar approaches which either assume instantaneous knowledge about the gradient or require knowledge about the plume dynamics (Demetriou et al., 2014) (Bertozzi et al., 2005).

3 Methodology

In this section, we begin by introducing the particular models we consider. We also introduce the extremum-seeking controller that causes the system to converge to the desired concentration and the cooperative control law that results in an equidistant spread across the boundary.

3.1 Problem formulation

3.1.1 Plume model

A Gaussian plume model is chosen for our simulated environment during the development of the algorithm and testing. It allows a great deal of flexibility for testing in the variation of its initial conditions and parameters while minimizing computational requirements. Furthermore, it provides reasonable and demonstrated accuracy which is a requirement to properly validate the behaviour of our algorithm. This report considers the steady-state behaviour of a plume using a Gaussian constant source model (Holzbecher, 2013).

$$C(x, y, z) = \frac{Q}{4\pi x \sqrt{D_y D_z}} \exp\left(-\frac{v}{4x}\left(\frac{y^2}{D_y} + \frac{z^2}{D_z}\right) - \frac{\lambda}{v}x\right) \quad (1)$$

Equation 1 represents the 3D solution of this model

- $C(X)$ is the concentration of the chemical at the point $X = (x, y, z)$.
- Q is the total mass of the chemical released per unit of time.
- D_i is the diffusion coefficient in the i axis.
- λ is the decay constant of the material.
- v is the advection/velocity of the flow field.

To model an arbitrary point source plume, we introduce an origin $X_0 = (x_0, y_0, z_0)$. We modify the equation to return the concentration at any point in space

$$C(x, y, z) = \frac{Q}{4\pi(x - x_0)\sqrt{D_y D_x}} \exp\left(-\frac{v}{4(x - x_0)}\left(\frac{(y - y_0)^2}{D_y} + \frac{(z - z_0)^2}{D_z}\right)\right) \quad (2)$$

Note: The $\frac{\lambda}{v}x$ term is dropped according to model assumption 5 below.

This Gaussian plume model is used only to simulate a chemical plume environment and the proposed onboard control algorithm does not use it to inform its behaviour.

3.1.2 Model assumptions

This model makes several assumptions of environmental and chemical source characteristics, namely:

1. The chemical source is continuous and constant
2. The wind speed is constant and pointed in the x axis
3. There are no obstructing objects or changes in surrounding terrain height
4. The diffusivity of each axis is constant in both time and space
5. The chemical does not react or decay in the environment

These assumptions are made to simplify the modelling of the chemical and reflect a wider variety of environments and scenarios. They do not represent a specific case example and in fact, reflect an ideal environment useful to safely and predictably test the proposed algorithm. Additionally, there are 2 resultant characteristics of this model that are key assumptions in the design and analysis of the controllers described in Section 3.2:

1. The plume has no local maxima.
2. The plume is continuous.

3.1.3 Communication network

Assume there are n agents capable of measuring the concentration instantaneously at any point with a portable chemical sensor, each equipped with wireless transceivers to form a nearest-neighbour communication topology. An example configuration is shown in Figure 2 where $n = 4$ with the first and last agents connected to form a ring structure. The drone's position i in the wireless chain is predetermined, and the nearest-neighbour topology restricts an agent to communicate only with their directly adjacent neighbours, $i - 1$ and $i + 1$.

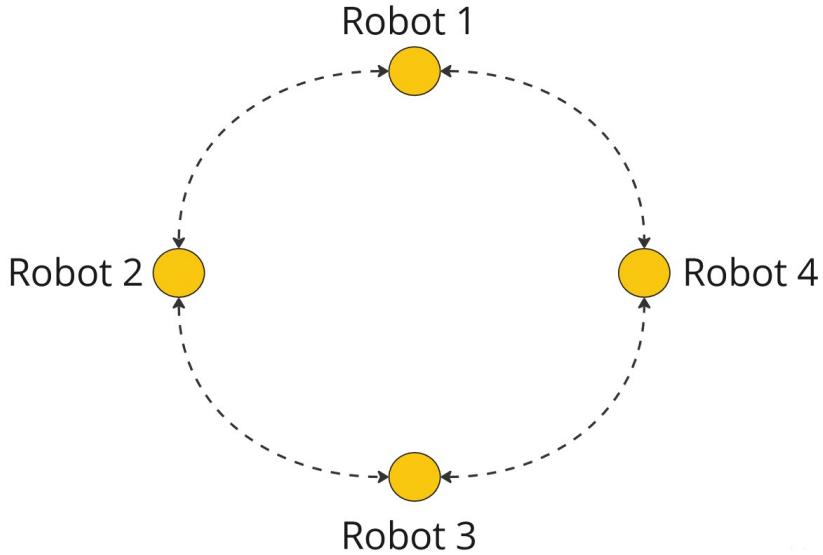


Figure 2: Communication Topology with a Ring Structure.

These agents have closed-loop stable dynamics of the form

$$\dot{x} = f(x, u), \quad (3)$$

which is determined by the stabilising inner loop controller and plant dynamics, where the input u is the desired position.

3.1.4 Problem goal

This project aims to develop an algorithm that can monitor the boundary of a chemical plume at a given reference concentration $c_{ref} \in (0, \bar{c}]$ where $\bar{c} \in \mathbb{R}^+$ is the maximum concentration within the plume. This goal of boundary tracking is to be monitored by a distributed scheme using multiple agents. This results in 2 different problems needing to be solved.

The first problem is to have all agents reach the equilibrium manifold corresponding to the desired concentration. This can be represented as

$$x_i \in C_d, \quad (4)$$

where C_d is the set of all cartesian coordinates which measure the desired concentration c_{ref}

$$C_d = \{x_d | x_d \in \mathbb{R}^2, C(x_d) = c_{ref}\}. \quad (5)$$

The second is to have all agents spread equally along the manifold C_d . This can be described by

$$\forall i, d_{i,i+1} = d_{i,i-1}, \quad x_i \neq x_{i-1} \neq x_{i+1}, \quad (6)$$

$$\text{where } d_{ij} = |x_i - x_j|, \quad j \in \{i-1, i+1\}. \quad (7)$$

This definition aims to make every agent have the same distance to their neighbours, resulting in an even distribution along the boundary.

3.2 Controller architecture

To address the problem goals defined in (4) and (6) a two-controller design is proposed, with a set point seeking component and a formation-based component.

Using a similar approach to Wang and Guo (2019), a gradient ascent method moves towards the desired set C_d . To estimate the gradient to the perimeter of interest, we use a model-free extremum-seeking controller, which only requires certain assumptions about the plume to be made, unlike previous methods. A cooperative formation controller then uses an estimate of the concentration gradient to spread equally across the boundary. Figure 3 shows a block diagram demonstrating the overall architecture.

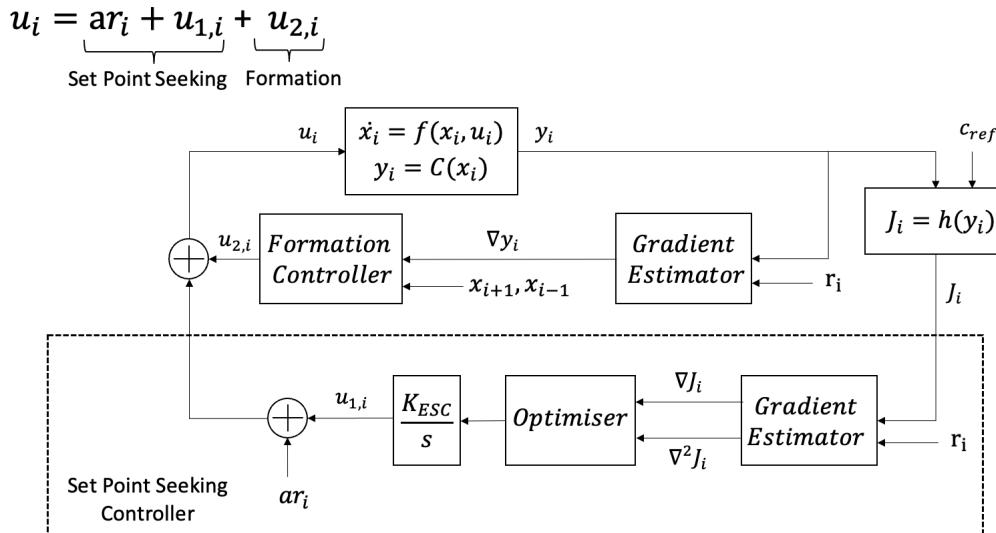


Figure 3: Block diagram of the controller architecture.

3.2.1 Setpoint seeking controller

Setpoint seeking controllers are adaptive controllers that allow a system to adapt to unknown dynamics. It is a model-free controller, which makes it suitable for tracking the concentration boundary as in real-world applications, an agent would have little underlying knowledge about the plumes model.

These controllers apply a dither signal to the agent's reference, changing the measured output. This change can then be used to estimate the gradient allowing the system to move towards the maximum.

Our setpoint-seeking controller is composed of three components:

1. Cost Function: The underlying function to be optimised.
2. Gradient Estimator: The method for estimating the gradient.
3. Optimiser: The technique used to update the agent's reference position incrementally.

The setpoint-seeking controller also makes several assumptions about the properties of the cost function it aims to maximise.

3.2.1.1 Assumptions for Setpoint-Seeking

1. The cost function is locally convex around the maximum equilibrium manifold.

2. The specified reference concentration within the plume results in a single path-connected equilibrium manifold of the cost function.
3. The cost function is continuous.

Assumptions 1 and 2 are required for the setpoint-seeking controller.

Assumption 3 is included for the gradient estimator but, in practice, is not required.

3.2.1.2 Cost Function

We use a similar method to Bertozzi et al. (2005), and define the cost for each agent i as

$$J_i = -(C(x_i) - c_{ref})^2, \quad (8)$$

which forms a negative parabola with a maximum cost of 0 locally around the maximum boundary. Figure 4 shows an example cost function on a Gaussian plume.

Assumption 1 is satisfied with a cost function of this form, provided the plume has no local maxima near the reference concentration. The no local maxima constraint on the plume also satisfies the second assumption. If the plume is continuous, then the cost function will also be continuous, ensuring assumption 3 is satisfied.

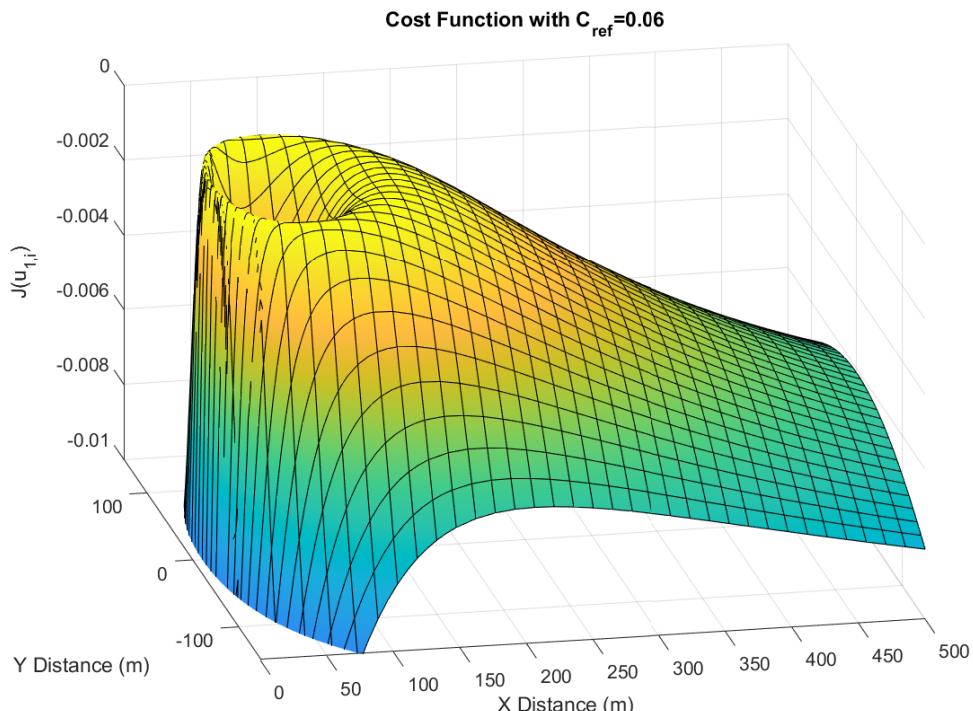


Figure 4: Example Cost Function on a Gaussian Plume model with $c_{ref} = 0.06$, and Plume Parameters: Diffusivity $D_y = 30$, $D_z = 0.5$, $v = 3\text{m/s}$, Emissivity $Q = 1000$, Elevation $H = 10\text{m}$, and Source Location $(x_0, y_0, z_0) = (0, 0, 0)$.

3.2.1.3 Gradient Estimator

To estimate the gradient, a setpoint-seeking controller uses small perturbations around the current position and measures the effect on the cost function.

We use two out-of-phase sinusoidal waves r_i , multiplied by a constant $a \in \mathbb{R}$ to perturb the reference x - y positions:

$$r_i = \begin{bmatrix} \sin(\omega t) \\ \sin(\omega t + \frac{\pi}{2}) \end{bmatrix}. \quad (9)$$

As the cost function and concentration measurements are small, we use a high-pass filter with cutoff frequency $\omega_h = \frac{\omega}{10}$ rad/s to measure the effect of the dither.

The filtered signal is demodulated by multiplying it with the original dither's sinusoidal wave r_i . A low-pass filter with cutoff frequency $\omega_l = \frac{\omega}{10}$ rad/s smooths out the oscillatory response. The low-pass filter does not significantly impact the setpoint-seeking controller but is necessary for the formation controller in Section 3.2.2.

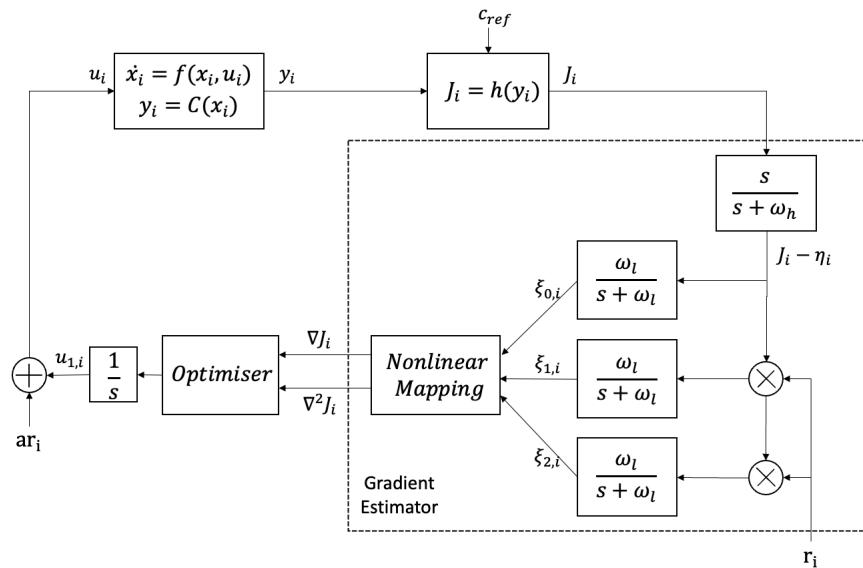


Figure 5: Block Diagram of the gradient estimator

The approximation of these derivatives can be calculated using the Taylor series expansion on the time-averaged system to achieve the following derivative estimates, using the notation $\nabla^k J := \frac{\partial J}{\partial x^k}$, where $k \in \mathbb{N}$. Tan et al. (2006) shows that the presence of the high pass filter ($J - \eta$ subsystem) in the average, is exactly the same as the subsystem J . To estimate the derivatives required to implement this modified Newton optimisation scheme, we can use the new time scale ρ where $r_i = \sin(\rho)$ which assumes a time-scale separation for this sinusoidal perturbation operating at a slower time scale (Teel et al., 2003). Given this, the model of the low-passed demodulated signal is given by ξ_0 , ξ_1 and ξ_2 .

$$\dot{\xi}_1 = -\omega_L [\xi_1 - J(u_{1,i} + a \sin(\rho)) \sin(\rho)]. \quad (10)$$

$$J(u_{1,i} + a \sin(\rho)) \sin(\rho) \approx J(u_{1,i}) \sin(\rho) + a \nabla J(u_{1,i}) \sin^2(\rho) + \frac{a^2}{2} \nabla^2 J(u_{1,i}) \sin^3(\rho). \quad (11)$$

$$\dot{\xi}_1 \approx -\omega_L \left[\xi_1 - \frac{a}{2} \nabla J(u_{1,i}) \right]. \quad (12)$$

$$\lim_{\rho \rightarrow \infty} \xi_1(\rho) \approx \frac{a}{2} \nabla J(u_{1,i}). \quad (13)$$

$$\lim_{\rho \rightarrow \infty} \xi_0(\rho) \approx J(u_{1,i}) + \frac{a^2}{4} \nabla^2 J(u_{1,i}), \quad (14)$$

$$\lim_{\rho \rightarrow \infty} \xi_2(\rho) \approx \frac{1}{2} J(u_{1,i}) + \frac{3a^2}{16} \nabla^2 J(u_{1,i}). \quad (15)$$

$$\nabla J(u_{1,i}) \approx \frac{2}{a} \xi_1, \quad (16)$$

$$\nabla^2 J(u_{1,i}) \approx \frac{8}{a^2} (2\xi_2 - \xi_0). \quad (17)$$

An example of the gradient estimation for concentration in the x direction is shown in Figure 6. In this example, the measured concentration increases as the x position increases and decreases on a decreasing x . The gradient estimator plot shows the demodulation, where the positive measures remain positive, but the negative filtered concentration is multiplied by the negative x dither to become entirely positive. The smoothing effect of the low-pass filter is also shown.

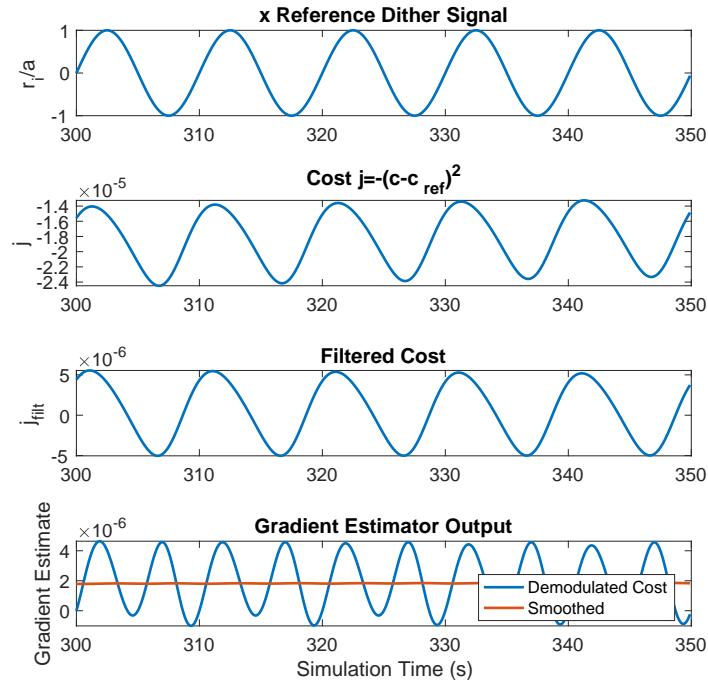


Figure 6: Gradient Estimator on Sample Scenario where the gradient in the x direction is estimated.

3.2.1.4 Optimiser

Using the derivative estimations outlined in section 3.2.1.3, optimisation of the cost function defined in equation 4 is achieved using a modified Newton optimisation gradient ascent scheme to converge to

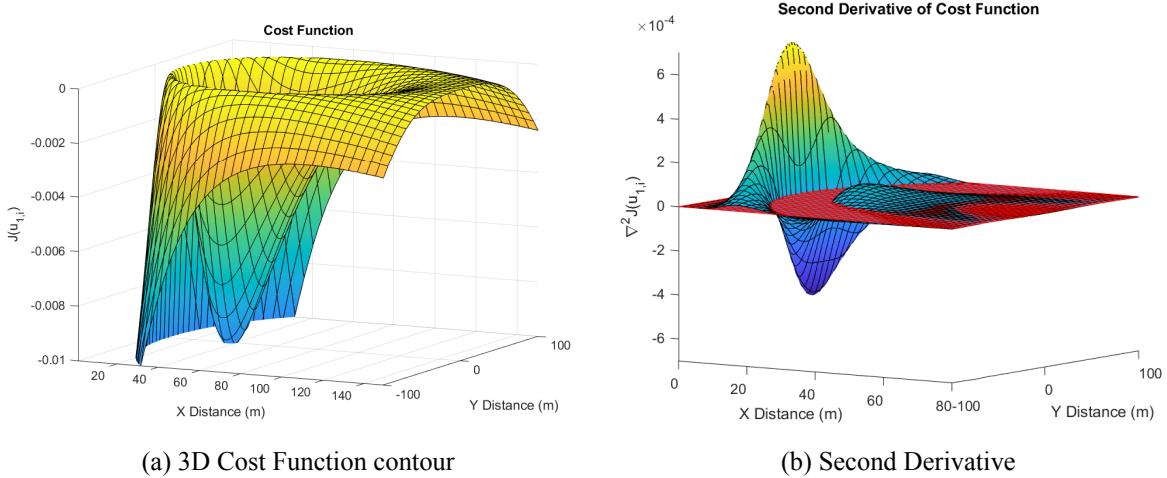


Figure 7: Plume cost function and the second derivative

the global maxima equilibrium manifold as shown in Figure 4. This method utilises the gradient and second derivative of the cost function for convergence. Moase et al. (2010) describes the convergence of perturbation-based ES schemes as proportional to $\nabla^2 J(u_{1,i})$, which could result in slow convergence when $\nabla^2 J(u_{1,i})$ is low and unstable operations when $\nabla^2 J(u_{1,i})$ is too high. Thus, using a Newton-like optimisation method could reduce the effect of uncertainty around different plume conditions with varying curvatures, which is especially critical for the deployment of this algorithm on unknown models that could have fluctuating contours.

Using these derivative approximations, the modified Newton optimisation scheme can be implemented using a similar scheme to the implementation in (Nesić et al., 2010). Note that we avoid division by zero errors by introducing an empirically identified $\varepsilon = 0.00001$ as the second derivative of the plume can be very small due to the large scale of the plume which regions of shallow gradients. Additionally, the second derivative of the plume has to be negative definite during the Newton gradient descent scheme, otherwise, the first-order gradient descent method is used instead to avoid convergence issues such as oscillations and instability due to the changing descent direction from the non-negative definite Hessian around the equilibrium manifold as observed in figure 7b.

$$\text{Optimiser} = \begin{cases} -\frac{\nabla J(u_{1,i})}{\nabla^2 J(u_{1,i})} & \text{if } |\nabla^2 J(u_{1,i})| \geq \varepsilon \text{ and } \nabla^2 J(u_{1,i}) < 0 \\ \frac{\nabla J(u_{1,i})}{\varepsilon} & \text{otherwise} \end{cases} \quad (18)$$

$$\dot{u}_{1,i} = K_1 \text{Optimiser}. \quad (19)$$

3.2.2 Formation Controller

A distributed cooperative control law is used to accomplish the project's second goal, where all agents are spread equally around the boundary of interest, defined in (6). Due to the ring communication topology, every agent can measure the positions of their neighbours at any time. The control law for the formation controller was chosen to be

$$\dot{u}_{2,i} = K_2(d_{i,i+1} - d_{i,i-1})\hat{t}_{i,r}, \quad (20)$$

where $\hat{t}_{i,r}$ is the unit vector tangential to the gradient of the plume ∇y_i and $K_2 \in \mathbb{R}^+$ is the gain given to the formation controller.

The direction of the formation control updates is always perpendicular to the concentration gradient to ensure its updates don't affect the setpoint seeking controller. The gradient of the cost function can not be used in the formation controller, as once it reaches the maximum (when $x_i \in C_d$) the derivative will be 0. Instead, we perform a second gradient estimation on the concentration using the same method in Section 3.2.1 to produce ∇y_i .

To calculate the perpendicular direction to the concentration gradient, we use the equation

$$\hat{t}_{i,r} = H \frac{\nabla y_i}{\|\nabla y_i\|}, \quad (21)$$

where $H = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ is the rotation matrix corresponding to a 90° rotation, and ∇y_i is a vector pointing in the direction of an increasing concentration. The low pass filter in the gradient estimator is used to make the unit vector towards increasing concentration constant rather than oscillating.

The vector of the concentration gradient ∇y_i is calculated using the same gradient estimation method used in the setpoint-seeking controller using the already applied dither signal.

We enforce another time scale separation by ensuring K_2 is sufficiently small. There are two main reasons for doing this:

1. The gradient estimate ∇y_i depends on the slow dithering signal, so it can not move too much; otherwise, it would affect the measurement.
2. Large updates in the position $u_{2,i}$ will push the agent away from the equilibrium manifold C_d where the measured concentration is the target concentration.

The overall behaviour of the controller then pushes the agent along the right direction of the contour when $d_{i,i+1} > d_{i,i-1}$, as seen in Figure 8. Alternatively, the agent will move to the left if the opposite is true. When they are at the same distance, the agent remains stationary. By consensus, all agents will eventually spread equidistant to each other and enter an equilibrium state.

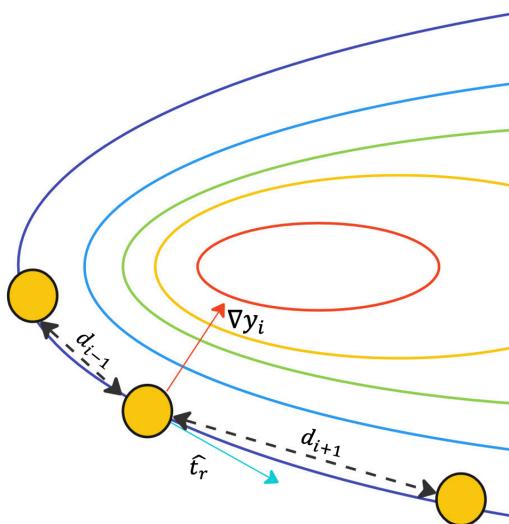


Figure 8: Diagram of the formation controller behaviour

3.3 Hardware Implementation



Figure 9: Drones flying in Vasey laboratory

The proposed algorithm and control law was tested on a DJI Tello Drone, a mini drone quadrotor within the University of Melbourne’s Flight (Vasey) Lab (pictured in Figure 9). By linearising the drone dynamics around the hover equilibrium, the following equations of motion describe its motion:

$$\ddot{\phi} = \frac{\tau_\phi}{I_x} \quad \ddot{r}_x = -gu_{1,i} \quad (22)$$

$$u_{1,i} = \frac{\tau_{u_{1,i}}}{I_y} \quad \ddot{r}_y = g\phi \quad (23)$$

$$\ddot{\psi} = \frac{\tau_\psi}{I_z} \quad \ddot{r}_z = \frac{F_t}{m}. \quad (24)$$

$(\phi, u_{1,i}, \psi)$ are the Euler angles (roll, pitch, yaw). $r \in \mathbb{R}^3$ is the position vector of the drone, F_t is the total motor thrust and m is the drone’s mass. τ_ϕ , $\tau_{u_{1,i}}$ and τ_ψ are the control torques produced by varying different combinations of motor speeds and I_x , I_y and I_z are the drone’s moments of inertia.

The plume boundary tracking control law was tested in the Vasey Lab using 3 Tello drones. For external positioning, a Vicon infrared camera system was used to accurately monitor the drone positions in a 5x3 metre space. As the drones are not capable of autonomous navigation or onboard processing, they receive commands via UDP from an external computer running the algorithm on MATLAB. A simulated Gaussian plume provided ‘concentration’ values by feeding in the measured position values, for which they used the gradient ascent optimiser in the setpoint seeking controller. Whilst processing was handled on an external computer, the algorithm could be deployed and distributed to multiple drones with onboard computing capabilities.

4 Results & Simulations

4.1 Controller gain tuning

Performing simulations on agents with a point mass model we use a constant swarm formation controller gain $K_2 = 5 \times 10^{-5}$, a dither signal period of $\omega = \frac{\pi}{5}$ and dither amplitude $a = 3$, we simulate the results on a Gaussian plume model with 6 agents. Note that we are using the Newton optimiser in Section 3.2.1.4, which scales the gain by dividing by the second derivative or $\varepsilon = 0.00001$, which means that the gain K_1 is not directly acting of the gradient magnitude.

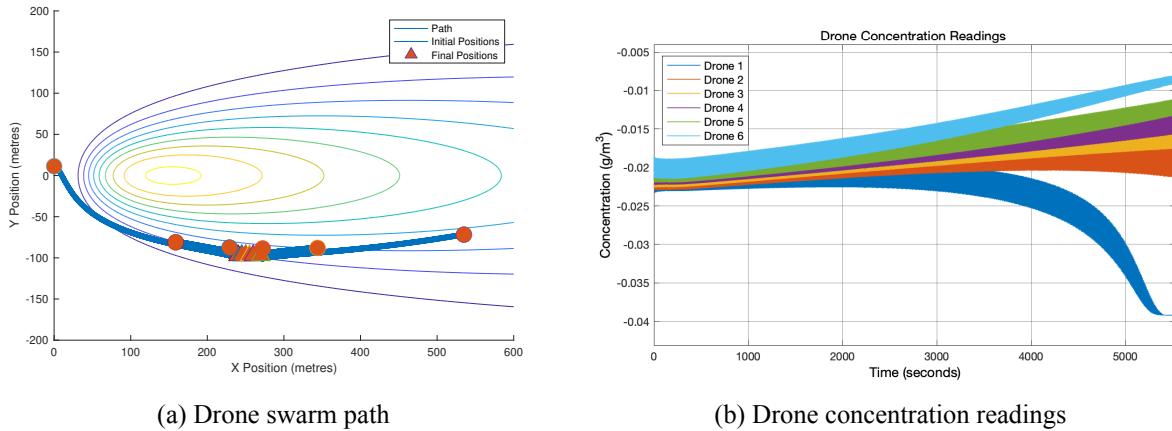


Figure 10: $K_1 = 0.0001$

Using $K_1 = 0.0001$ leads to divergence as it violates the $K_1 \gg K_2$ assumption. Therefore, using a K_1 value that is too small results in a slow gradient climb, with the swarm formation component dominating the control signal, leading to an unstable performance with the divergence of the swarm.

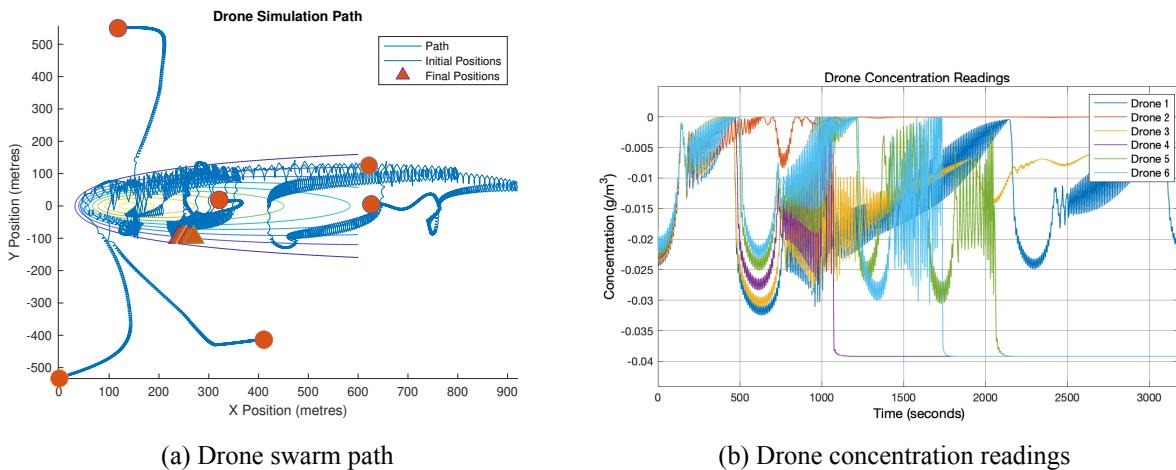
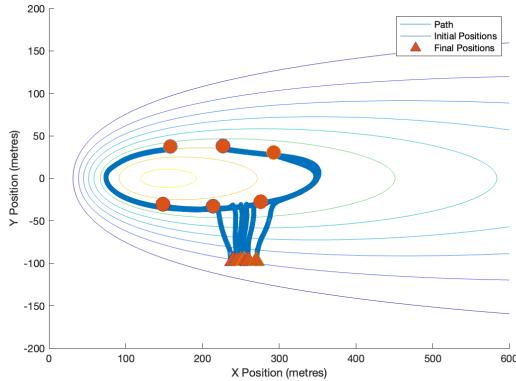
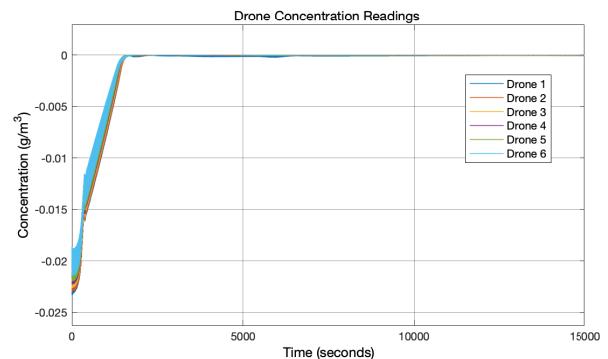


Figure 11: $K_1 = 1$

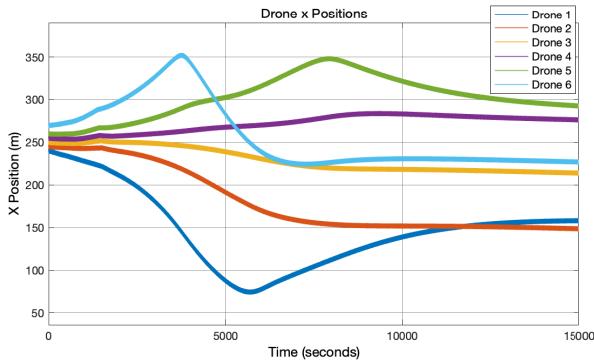
Using $K_1 = 1$ leads to divergence as it has substantial overshoot properties that the agent cannot recover from, leading to large oscillatory behaviour and agents that diverge far away from the specified set point.



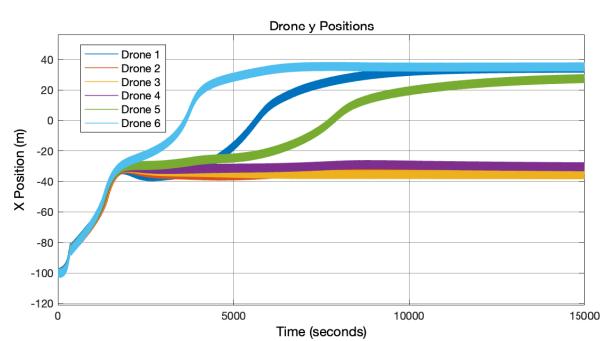
(a) Drone swarm path



(b) Drone concentration readings



(c) Drone x positions

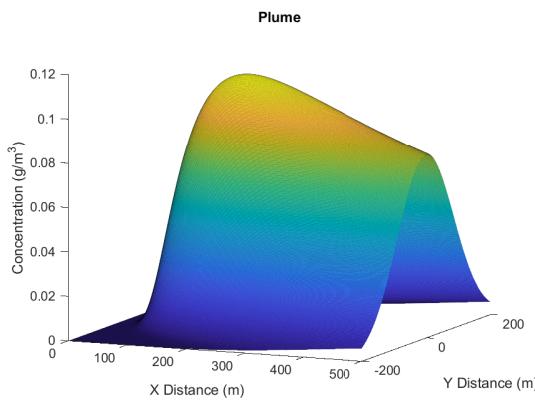


(d) Drone y positions

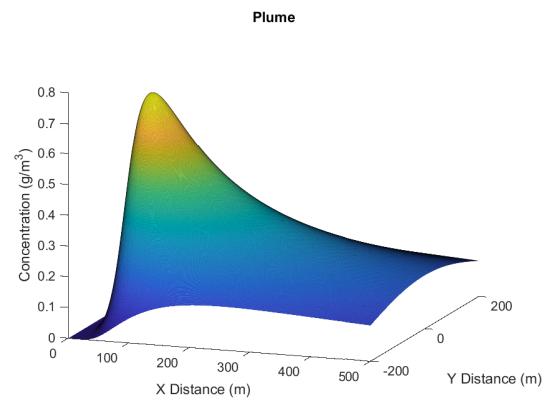
Figure 12: $K_1 = 0.0015$

Using $K_1 = 0.0015$ leads to a good set point seeking convergence with minimal overshoot and a balanced swarm formation controller $K_2 = 5 \times 10^{-5}$, resulting in convergence to the equilibrium manifold concentration. Note that the set-point seeking controller converges in 1600s, with the slower formation controller taking 15000s to converge.

4.2 Performance on different plume conditions



(a) Gentle plume contour with $C_{ref} = 0.09$



(b) Steep plume contour with $C_{ref} = 0.5$

Figure 13: Varying plume conditions

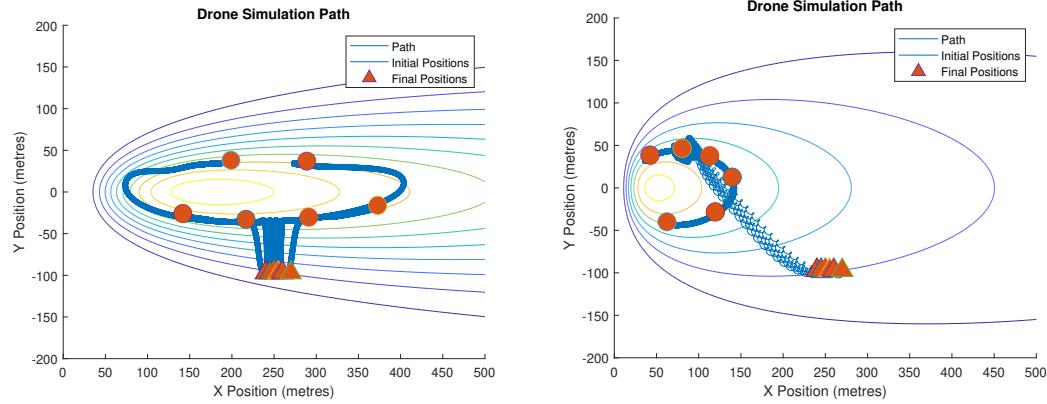


Figure 14: Drone swarm path using gradient ascent method

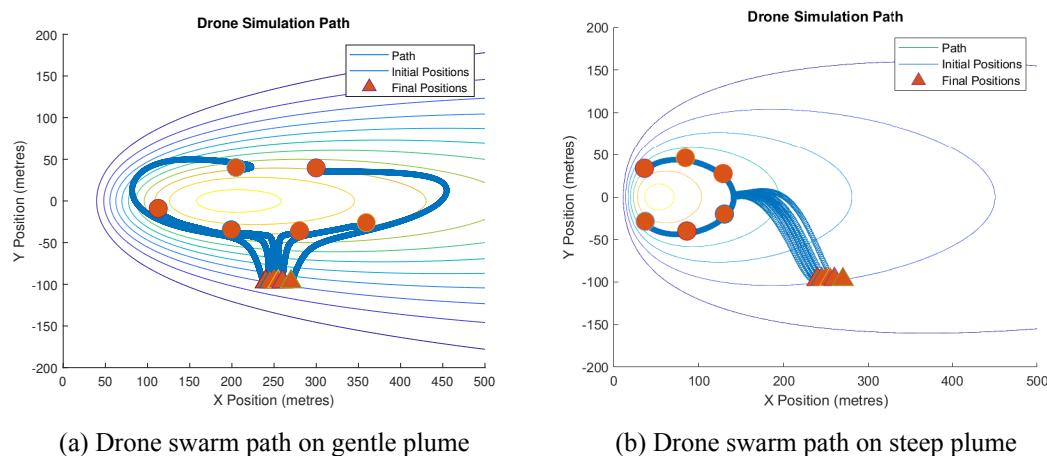


Figure 15: Drone Swarm Path using modified Newton optimiser

The controller is able to handle variations of the plume contour without changing any controller gain parameters in a 10,000 second simulation, solely changing C_{ref} for the set-point seeker to optimise towards. This behaviour demonstrates the robustness of the Newton optimiser, which utilises information about the plume's second derivative to scale the setpoint seeking controller based on the plume contour. Comparing this to using a fist order gradient ascent method, the behaviour of the controller is greatly influenced by this change in plume contour, with a dramatic overshoot response in Figure 14b. The formation controller failed to converge in the required time, as all agents had not yet become equidistantly spaced.

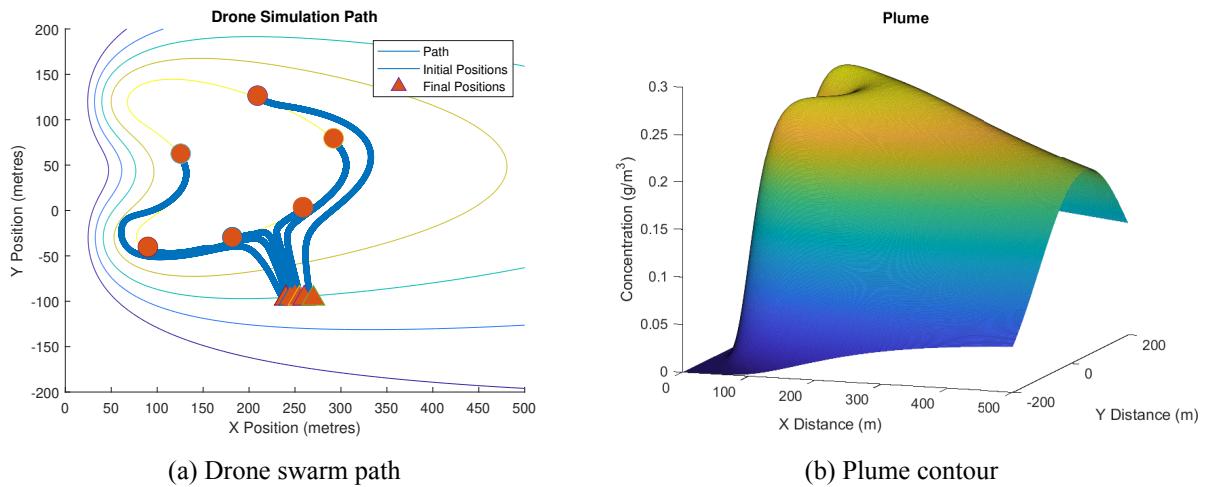


Figure 16: Two plume stacks with $C_{ref} = 0.3$

Figure 16 demonstrates the limitations of the swarm formation controller on different plume shapes, with this caused by having two plume stacks 100m next to each other. Since the controller has no knowledge of the plume dynamics, it converges to a formation that does not cover the top protrusion of the plume contour as it has reached an optimised distance-based formation between the adjacent agents.

4.3 Hardware results

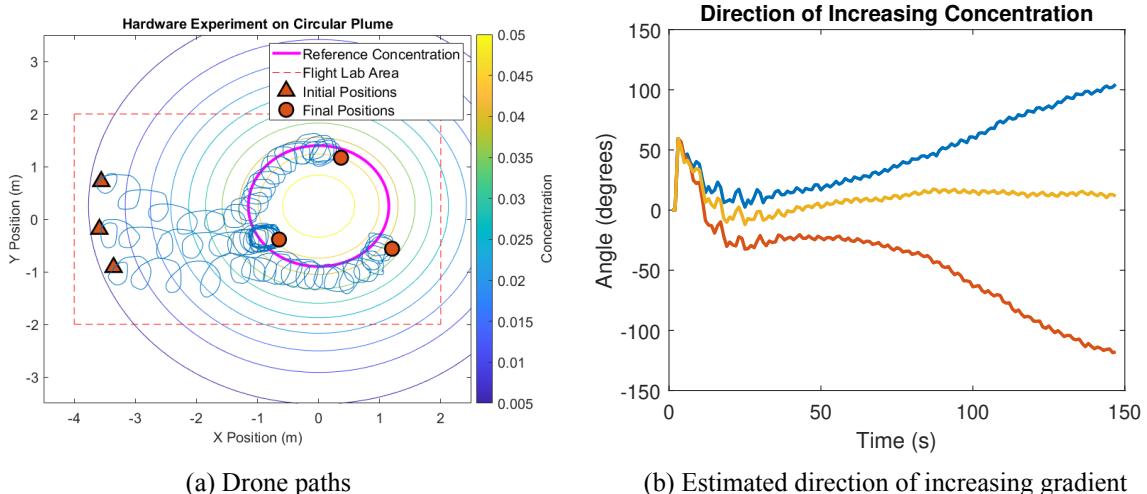


Figure 17: Drone swarm during hardware test on circular plume

Utilising a simpler circular Gaussian plume model, a test was completed on DJI Tello drones to validate the controller's design. Figure 17a demonstrates the convergence of both the Setpoint and Swarm Formation controllers in such an environment.

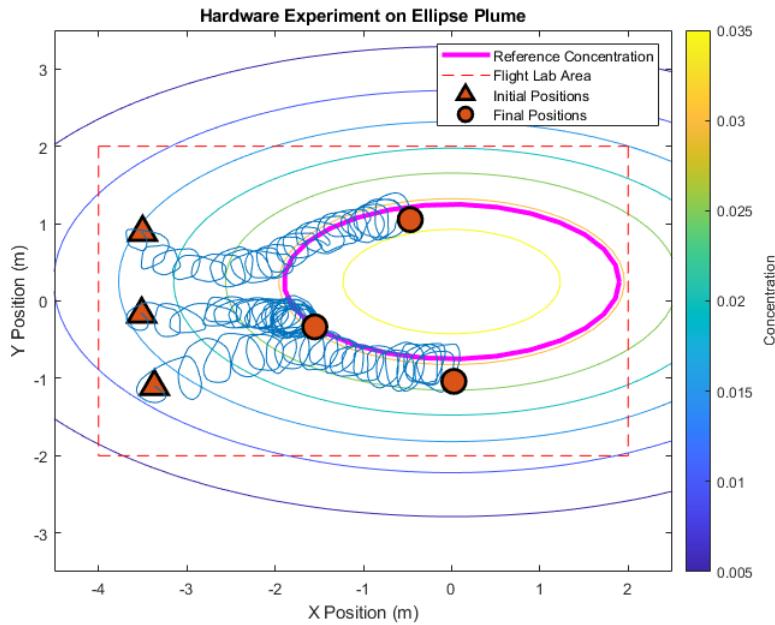


Figure 18: Drone swarm path during hardware test on ellipse plume

Further experimentation on plumes more closely resembling those in simulated testing demonstrated a limitation of the formation controller, where the drones became equidistant but not across the entire contour. Figure 18 shows that while all drones converge to the reference concentration, the right half of the boundary has been neglected after convergence.

All hardware experiments completed relied on the Vicon motion capture system to track the positional data of the drones. This system has extremely high accuracy and does not reflect a more realistic experiment, where drones would likely rely on less precise systems such as GPS. Despite this, the hardware results support the validity of the controller design in more ideal testing scenarios.

5 Discussion

This project has demonstrated how a drone swarm can identify and track the boundary of a hazardous chemical plume using limited communication and local measurements. The chosen approach combines a set point-seeking controller to position drones at a given reference concentration, and when combined with a distributed cooperative controller, can spread out to form an equidistant boundary between the agents. This control algorithm requires no prior knowledge about the plume’s model, making it a potential method for investigating environments where there is insufficient information about the model.

5.1 Setpoint seeking controller

The setpoint seeking controller is a viable option for positioning a drone at a desired reference concentration. It is an adaptive method suitable for environments where parameters are not sufficiently known, making it viable for emergency responses where information is not always available. Existing methods, such as in (Wang & Guo, 2019), currently rely on Luenberger observers to estimate the gradient at a particular location using local sensor measurements communicated across a robotic sensor network. Typically the plume model will be unknown to the system, so observer-based methods are not often practical. This makes our algorithm’s proposed setpoint seeking controller a suitable alternative for estimating the

gradient and converging to the desired reference concentration, as it is a model-free technique.

5.1.1 Parameter Selection

In Section 4, the importance of parameter tuning is demonstrated. Figures 10, 12 and 11 demonstrate how improper selection of the setpoint seeking gain K_1 and the formation controller gain K_2 can result in unstable performance. These issues arise when the required timescale separations are not respected. When K_2 is not sufficiently smaller than K_1 , the formation controller dominates the setpoint seeking controller, forcing it away from the reference boundary rather than towards it. As the formation controller also disturbs the gradient estimation method, it measures an incorrect direction of the increasing gradient, causing it to move away from the concentration set it is on. Similarly, when K_1 is too large, the drone moves before accurate gradient values are calculated, resulting in unstable performance. When tuned appropriately, the algorithm was shown to work on a variety of different plume contours.

This shows that the performance and stability of the algorithm heavily depend on the selection of parameters, for which tuning can be complex. Other parameters, such as the dither size and frequency, also play an important role in performance. These were not explicitly demonstrated in the results but were found to affect the convergence rate, as they measured greater gradients and affected the separation between the fast and slow timescales.

5.1.2 Timescale Separation

The required timescale separation becomes one of the limiting factors of the control algorithm. As the controller estimates the gradient, a perturbation is applied to the drone, for which the fast vehicle dynamics must be able to converge. This behaviour introduces the need for a sufficient timescale separation and highlights a key limitation of the method—the rate of convergence. In the case where this timescale separation is not enforced, the algorithm fails due to incorrect gradient estimation. Different methods of extremum seeking can remove these timescale separations, such as by instantaneously measuring the gradient using smaller drone formations (Khong et al., 2014).

5.1.3 Plume Model

Our approach has only considered the steady state behaviour of the Gaussian plume, which only considers the plume over a long period. Jones (1983) mentions that Gaussian plume models are unsuitable for modelling short-term behaviour, a scenario in which the proposed algorithm could be deployed. This is because the Gaussian model typically represents the average concentration value over 15 minutes and modifications to the algorithm may be required to account for this using different plume modelling techniques. We also limit our approach to considering a single locally convex maxima of interest rather than multiple sources, which would be more realistic for a variety of scenarios. In bushfire events, smoke plumes are emitted from many sources and move quickly as the fire spreads. Therefore, it is not reasonable to assume a static point source in all use cases.

5.2 Formation controller

The formation controller was shown to be capable of spreading out equidistantly along the boundary of the plume by communicating with the previous and next drones in the communication chain. This results in a distributed and leaderless configuration that reduces dependence on a single leader and the degree of communication, as they only have to communicate with their two closest neighbours rather than all drones in the swarms.

5.2.1 Performance

The formation controller was shown to work on simulations conducted in Section 4 and on the hardware tests that used a circular Gaussian plume model. As the formation controller operates at the slowest time scale to not force the drones away from the desired reference concentration, it is the limiting component of the algorithm. In Figure 10, the setpoint seeking controller converges to the target concentration at approximately 2000 seconds, but the formation controller doesn't converge until roughly 10,000 seconds. This suggests that the performance of the formation controller is insufficient and would need to be improved. In our tests, we have only considered starting the drones in one location, but appropriately spacing the drones out at the start would increase performance.

5.2.2 Convergence around concentration boundary

In most cases, the controller spreads the agents equally around the desired concentration despite the slow performance. However, there were cases where the algorithm failed, typically when fewer agents were in the formation. Figure 19 demonstrates a significant limitation of the formation controller, where limited numbers of drones cannot spread around the boundary as they reach an undesirable equilibrium. The hardware test in Figure 18 further confirms this and shows the method is not robust.

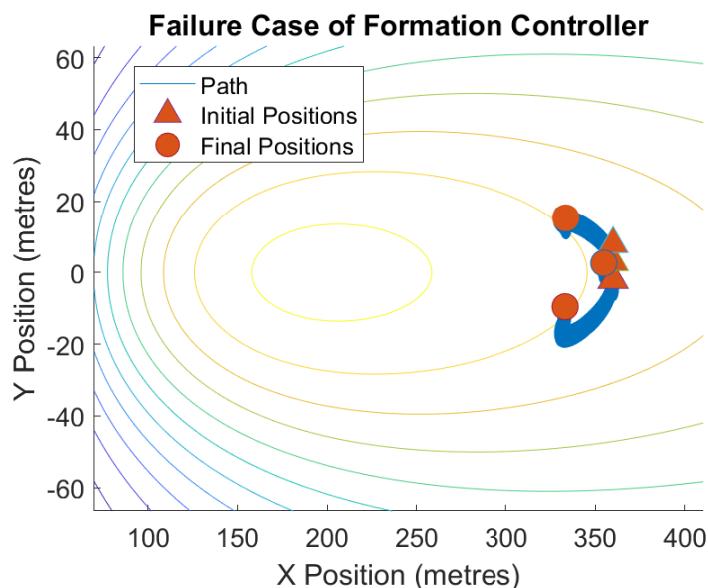


Figure 19: Failure of the formation controller on simple cases

A potential method to rectify this problem is to make all agents traverse the plume with a constant velocity once the desired reference concentration has been reached. This will provide temporal information about the boundary of the plume, which could be used to fit an appropriate reference boundary using the drones' path, given the plume's evolution rate is low enough. Additionally, this could allow the agents to be pushed out of the undesirable equilibrium state, as seen in Figure 19, causing the distances between all agents to change. However, this approach does not exclude this equilibrium state of the formation controller, so further research and validation would be required to overcome this limitation.

Figure 16a shows that the algorithm could still perform reasonably well on some non-convex boundaries but depends on the contour. The boundary tracker reached an equilibrium state but did not converge to the desired equilibrium around the contour. This is due to the algorithm not having any knowledge about the actual structure of the plume and only using the direct distances between agents to inform its movement.

5.3 Newton-like optimiser

As seen in Section 4.2, by implementing the Newton Optimiser, we were able to increase the robustness and convergence rate of the setpoint seeking controller. When considering the shape of Gaussian plumes in Figure 14, and the cost function in Figure 4, significant variation in map curvature can be seen. In these circumstances, a gradient ascent method would not perform reliably as the rate of convergence changes depends on the variable gradient and requires individual tuning for desired performance. This contour-dependent response is undesirable given the problem statement of real-world deployment in unknown plume conditions, with plumes that could exhibit vastly different structures based on the environmental conditions and chemical source. Additionally, the overshoot response seen in figure 14b could lead to instability of the controller as a significant enough overshoot could cause the agents to leave the region of interest with sufficient concentration readings.

The more complicated Newton-like optimiser avoids this by scaling by the second derivative, ensuring a similar convergence rate for all drones regardless of their starting locations. The Newton method demonstrates more desirable behaviour than the gradient ascent method on the different plumes, with no additional tuning. Introducing a Newton-like optimiser made the algorithm significantly more robust and practical in real-life applications. However, we assume this only operates within a locally convex region of the cost function, and so the second derivative is always negative. We use an epsilon threshold to scale the first derivative to avoid dividing by zero. Under the assumption that we operate in a locally convex region of the cost function, we don't consider the case where the second derivative can change sign. In cases where non-locally convex regions are considered, the second derivative becomes 0 when the sign changes and can multiply a potentially non-small gradient with the large scaling factor, which could lead to divergence. In scenarios where non-local regions of the plumes are considered, different Newton-like methods can be used that involve a minimum fixed step rather than a large scaling (Moase et al., 2010).

5.4 Stability of controller

The controller's stability was only tested via extensive testing of the drones via simulations. The formation controller has also been shown to not converge to the desired equilibrium and would require further improvements to be practical. Note that a sufficient time scale separation between the setpoint seeking controller and formation controller is required with $K_1 \gg K_2$ to prevent divergence as seen in Figure 10.

5.5 Testing on hardware

The hardware tests demonstrated that a discrete-time version of the algorithm could be implemented on hardware with nonlinear dynamics. The setpoint seeking and formation controller functioned as expected on the limited tests conducted.

These tests also highlighted how important it was for the vehicle to be given sufficient time to reach the reference position. Initial implementations of the outer loop controller had a sampling time that did not allow the drone to converge to the desired position. This resulted in the gradient estimator producing inaccurate values causing the drones to move in unexpected directions. By lowering the outer loop sampling time, the drones reached the expected position before the following reference was generated, producing the expected result in Figure 17a.

The hardware testing also made apparent the impact of the low pass filter on the controller, something the simulations had not. The low pass filter introduced noticeable delays to each drone's direction of the increasing gradient. Lowering the filter's time constant reduced the delay but caused the gradient direction to oscillate. The low pass filter cut-off frequency selection becomes a trade-off between how

quickly the gradient needs to be measured and how oscillatory the tangent direction can be. Figure 17b shows a hardware test using a low pass filter with a time constant of $\frac{15}{\omega} = 20s$, which effectively attenuated the high-frequency oscillations without introducing a significant delay.

Testing conditions in the Vasey Lab were ideal, where no external environmental elements or obstructions were present, and the positions of the drones could be accurately measured. While these tests support a preliminary validation of the controller's design, further testing in more realistic conditions is necessary to explore further contributing factors to the drone's dynamics in more practical scenarios.

6 Conclusion & Recommendations

This project has demonstrated how a drone swarm can identify and track the boundary of a hazardous chemical plume using a distributed algorithm that only required adjacent communication and local measurements. The chosen approach combines a setpoint seeking controller to position drones at a given reference concentration, and when combined with a distributed cooperative controller, can spread out to form an equidistant boundary between the agents. This control algorithm requires no prior knowledge about the plume's model, making it a potential method for investigating environments where there is insufficient information about the model.

While the setpoint seeker has shown viability in a variety of tests, the formation controller demonstrated limitations in practical cases. Further consideration into its development should be taken to address these shortcomings or the controller may not be practical beyond our controlled simulation environment.

6.1 Future improvements

In more practical scenarios, it is also reasonable to assume that environmental factors will impact the drones' ability to perform and it is hence recommended to consider potential methods to reduce harsh environmental exposure. In the case of bushfire monitoring, the extreme heat may severely damage onboard components and so it may be necessary to avoid regions of high temperatures. This can be achieved through the utilization of no-fly zones around identified high-risk zones, though requires a methodology to identify said zones. These no-fly zones can also be used in areas where no chemical material is present.

The current algorithm also does not consider the safety of the drones with respect to others in the network and the environment. Future works should investigate methods of preventing drones from colliding with each other or surrounding obstacles. Such a mechanism could be modelled by a barrier function where drones are forced away from each other to mitigate this risk. Such a technique can also be applied to observed obstructions in the drones' pathing, to further reduce risks associated with practical collision risks.

One limitation of the developed swarm formation controller creates an impractical reliance on the initial positioning of the drones to achieve full coverage of plume boundaries. Future works should include further consideration of proper strategies to mitigate this issue, as this behaviour is presently inherent in the controller's design. One potential method would be to introduce constant movement along the contour to provide temporal data about its shape to fit a boundary contour along the reference concentration measurements.

Though the setpoint seeking controller's main strength is its adaptability to unknown and changing environments, further testing should be conducted to verify this. This could involve testing it on simulations that use dynamic plumes rather than a static Gaussian plume. Sensor noise should also be investigated in further detail.

This report considered a network of agents at constant fixed height only, and the further inclusion of the z dimension may reveal higher-quality information regarding the plume's characteristics and real-time development. It is important to note that the accuracy of the proposed IR spectroscopic sensor is dependent on its distance from the material, and the drones' respective heights should reflect this to maximize its potential.

Acknowledgements

We wish to extend our gratitude toward our project supervisor Professor Chris Manzie, for his continual guidance and support throughout the project. His provision of patient instruction and feedback formed the basis of our work, and we greatly appreciate all the time he dedicated towards our project.

We appreciate the support from the Faculty of Engineering and Information Technology and Vasey Lab in lending their resources and facilities. Our testing would not have been possible without their equipment and flight laboratory.

We will also like to thank our industry partner RingIR for their support and correspondence. Motivation for this project's goals was inspired by their work, providing a professional partnership over the course of the project.

References

- Ariyur, K. B., & Krstic, M. (2003). *Real-time optimization by extremum-seeking control*. John Wiley & Sons.
- Aznar, F., Sempere, M., Pujol, M., Rizo, R., & Pujol, M. J. (2014). Modelling oil-spill detection with swarm drones. *Abstract and Applied Analysis*, 2014. <https://doi.org/10.1155/2014/949407>
- Bertozzi, A. L., Kemp, M., & Marthaler, D. (2005). Determining environmental boundaries: Asynchronous communication and physical scales. *Lecture Notes in Control and Information Sciences*, 309. https://doi.org/10.1007/978-3-540-31595-7_2
- Binksin, M., Bennett, A., & Macintosh, A. (2020). *Royal commission into national natural disaster arrangements report*. Commonwealth of Australia.
- Bolla, G. M., Casagrande, M., Comazzetto, A., Moro, R. D., Destro, M., Fantin, E., Colombatti, G., Aboudan, A., & Lorenzini, E. C. (2018). Aria: Air pollutants monitoring using uavs. *5th IEEE International Workshop on Metrology for AeroSpace, MetroAeroSpace 2018 - Proceedings*. <https://doi.org/10.1109/MetroAeroSpace.2018.8453584>
- Cao, Y., & Fierro, R. (2006). Dynamic boundary tracking using dynamic sensor nets. *Proceedings of the IEEE Conference on Decision and Control*. <https://doi.org/10.1109/cdc.2006.376942>
- Casbeer, D. W., Beard, R. W., McLain, T. W., Li, S. M., & Mehra, R. K. (2005). Forest fire monitoring with multiple small uavs. *Proceedings of the American Control Conference*, 5. <https://doi.org/10.1109/acc.2005.1470520>
- Clark, J., & Fierro, R. (2007). Mobile robotic sensors for perimeter detection and tracking. *ISA Transactions*, 46. <https://doi.org/10.1016/j.isatra.2006.08.001>
- Demetriou, M. A., Gatsonis, N. A., & Court, J. R. (2014). Coupled controls-computational fluids approach for the estimation of the concentration from a moving gaseous source in a 2-d domain with a lyapunov-guided sensing aerial vehicle. *IEEE Transactions on Control Systems Technology*, 22. <https://doi.org/10.1109/TCST.2013.2267623>
- Dunbabin, M., & Marques, L. (2012). Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics and Automation Magazine*, 19. <https://doi.org/10.1109/MRA.2011.2181683>

- Farrell, J. A., Murlis, J., Long, X., Li, W., & Cardé, R. T. (2002). Filament-based atmospheric dispersion model to achieve short time-scale structure of odor plumes. *Environmental Fluid Mechanics*, 2(1), 143–169. <https://doi.org/10.1023/A:1016283702837>
- Gerhardt, N., Clothier, R., & Wild, G. (2014). Investigating the practicality of hazardous material detection using unmanned aerial systems. *2014 IEEE International Workshop on Metrology for Aerospace, MetroAeroSpace 2014 - Proceedings*. <https://doi.org/10.1109/MetroAeroSpace.2014.6865908>
- Guay, M., & Atta, K. T. (2018). Dual mode extremum-seeking control via lie-bracket averaging approximations. *2018 Annual American Control Conference (ACC)*, 2972–2977.
- Guenzler, H., & Gremlich, H. U. (2002). *Ir spectroscopy an introduction* [INSTRUMENTATION RELATED TO NUCLEAR SCIENCE AND TECHNOLOGY]. Wiley-VCH. http://inis.iaea.org/search/search.aspx?orig_q=RN:34024616
- Hazeleger, L., Beerens, R., & van de Wouw, N. (2019). A sampled-data extremum-seeking approach for accurate setpoint control of motion systems with friction. *IFAC-PapersOnLine*, 52(16), 801–806.
- Holzbecher, E. (2013). *Environmental modeling: Using matlab®*. <https://doi.org/10.1007/978-3-642-22042-5>
- Johnston, F., Williamson, G., & Bowman, D. (2010). *Air Quality and Climate Change*, 44(2), 17–21. <https://search.informit.org/doi/10.3316/informit.281358847612840>
- Jones, C. (1983). On the structure of instantaneous plumes in the atmosphere. *Journal of Hazardous Materials*, 7(2), 87–112. [https://doi.org/https://doi.org/10.1016/0304-3894\(83\)80001-6](https://doi.org/https://doi.org/10.1016/0304-3894(83)80001-6)
- Khong, S. Z., Tan, Y., Manzie, C., & Nešić, D. (2014). Multi-agent source seeking via discrete-time extremum seeking control. *Automatica*, 50(9), 2312–2320. <https://doi.org/https://doi.org/10.1016/j.automatica.2014.06.009>
- Kollmann, W. (2019). The structure of turbulent flows. In *Navier-stokes turbulence: Theory and analysis* (pp. 307–332). Springer International Publishing. https://doi.org/10.1007/978-3-030-31869-7_20
- Krstic, M., & Wang, H.-H. (2000). Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36(4), 595–601.
- Kuze, H., Goto, Y., Mabuchi, Y., Saitoh, H., Alimuddin, I., Bagtasa, G., Harada, I., Ishibashi, T., Tsujimoto, T., & Kameyama, S. (2012). Urban air pollution monitoring using differential optical absorption spectroscopy (doas) and wind lidar. *2012 IEEE International Geoscience and Remote Sensing Symposium*, 3638–3641. <https://doi.org/10.1109/IGARSS.2012.6350628>
- Liao, C.-K., Manzie, C., & Chapman, A. (2020). A distributed algorithm for uav-based communication networks using constrained extremum seeking. *IFAC-PapersOnLine*, 53(2), 5429–5434.
- Messenger, N. (2023, February 23). Ohio River Valley Institute. Retrieved May 5, 2023, from <https://ohiorivervalleyinstitute.org/an-overview-of-the-norfolk-southern-train-derailment-and-hazardous-chemical-spill-in-east-palestine-ohio/>
- Moase, W. H., & Manzie, C. (2012). Fast extremum-seeking for wiener–hammerstein plants. *Automatica*, 48(10), 2433–2443.
- Moase, W. H., Manzie, C., & Brear, M. J. (2010). Newton-like extremum-seeking for the control of thermoacoustic instability. *IEEE Transactions on Automatic Control*, 55(9), 2094–2105.
- Nesić, D., Tan, Y., Moase, W. H., & Manzie, C. (2010). A unifying approach to extremum seeking: Adaptive schemes based on estimation of derivatives. *49th IEEE conference on decision and control (CDC)*, 4625–4630.
- Noh, Y. M., Müller, D., Shin, D. H., Lee, H., Jung, J. S., Lee, K. H., Cribb, M., Li, Z., & Kim, Y. J. (2009). Optical and microphysical properties of severe haze and smoke aerosol measured by integrated remote sensing techniques in gwangju, korea. *Atmospheric Environment*, 43(4), 879–888. <https://doi.org/https://doi.org/10.1016/j.atmosenv.2008.10.058>
- Pang, S., & Farrell, J. (2006). Chemical plume source localization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 36(5), 1068–1080. <https://doi.org/10.1109/TSMCB.2006.874689>

- Porter, M. J., & Vasquez, J. R. (2006). Bio-inspired navigation of chemical plumes. *2006 9th International Conference on Information Fusion*, 1–8. <https://doi.org/10.1109/ICIF.2006.301656>
- Saffre, F., Hildmann, H., Karvonen, H., & Lind, T. (2022). Monitoring and cordoning wildfires with an autonomous swarm of unmanned aerial vehicles. *Drones*, 6. <https://doi.org/10.3390/drones6100301>
- Stockie, J. M. (2011). The mathematics of atmospheric dispersion modeling. *SIAM Review*, 53(2), 349–372. <https://doi.org/10.1137/10080991X>
- Tan, Y., Nešić, D., & Mareels, I. (2006). On non-local stability properties of extremum seeking control. *Automatica*, 42(6), 889–903. [https://doi.org/https://doi.org/10.1016/j.automatica.2006.01.014](https://doi.org/10.1016/j.automatica.2006.01.014)
- Teel, A. R., Moreau, L., & Nesic, D. (2003). A unified framework for input-to-state stability in systems with two time scales. *IEEE Transactions on Automatic Control*, 48(9), 1526–1544.
- Wang, J. W., & Guo, Y. (2019). Leaderless cooperative control of robotic sensor networks for monitoring dynamic pollutant plumes. *IET Control Theory and Applications*, 13. <https://doi.org/10.1049/iet-cta.2018.5112>
- Yang, J., Qian, J., & Gao, H. (2021). Forest wildfire monitoring and communication uav system based on particle swarm optimization. *Journal of Physics: Conference Series*, 1982. <https://doi.org/10.1088/1742-6596/1982/1/012068>