Seeking Scent with Robotics for Planetary Exploration

by

Christopher Nicol

A Thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of
Master of Applied Science

Ottawa-Carleton Institute for
Mechanical and Aerospace Engineering

Mechanical and Aerospace Engineering
Carleton University
Ottawa, Ontario, Canada
June 2013

Copyright ©
2013 - Christopher Nicol
The undersigned recommend to
the Faculty of Graduate Studies and Research
acceptance of the Thesis

Seeking Scent with Robotics for Planetary Exploration

Submitted by Christopher Nicol
in partial fulfilment of the requirements for the degree of
Master of Applied Science

Dr. Alex Ellery, Co-Supervisor

Dr. Ed Cloutis, Co-Supervisor

Dr. M. Yaras, Department Chair

Carleton University
2013
Abstract

This thesis presents the work towards analyzing the odour source localization problem found in nature as a suitable analogy for planetary exploration missions. The contributions are models of two environments for which the analogy may be relevant, the training of a recurrent neural network that replicates simple forms of odour source locational strategies found in nature, and the analysis and advancement of a source likelihood map algorithm for predicting the location of a chemical source given individual detection events.

Two missions that may be suitable for using the odour source localization analogy are locating methane sources on Mars and choosing a landing site for exploration of Enceladus' postulated subsurface ocean. The details in training recurrent neural networks for simple moth behaviours is shown and the results using the dynamic Mars methane plumes are discussed. The advancement of a binary chemical detection and source-localization method is shown, which improves accuracy.
Table of Contents

Abstract iii

Table of Contents iv

List of Tables viii

List of Figures x

List of Acronyms xvi

List of Symbols xviii

1 Introduction 1
  1.1 Motivation .................................................. 1
  1.2 Scope ......................................................... 5
  1.3 Overview ..................................................... 7

2 Background and Literature Review 9
  2.1 Mars Methane .................................................. 9
  2.2 Enceladus ....................................................... 12
    2.2.1 Enceladus Background ...................................... 13
    2.2.2 Previous Plume Models ..................................... 15
  2.3 Plume Tracing ................................................ 16
2.3.1 Characteristics of Moths ........................................... 17
2.3.2 Robotic Strategies .................................................. 19
2.4 Neural Networks ....................................................... 21
  2.4.1 Neuron Model ....................................................... 21
  2.4.2 Activation Functions ............................................. 22
  2.4.3 Network Structure ................................................. 23
  2.4.4 Real Time Recurrent Learning for Recurrent Networks ...... 24
  2.4.5 Forced Learning for Recurrent Networks ....................... 27
2.5 State Estimation ........................................................ 28
  2.5.1 Kalman Filter ....................................................... 28
  2.5.2 Unscented Kalman Filter ........................................ 30
2.6 Probability Theory ..................................................... 32
  2.6.1 Basic Probability Theory ....................................... 32
  2.6.2 Chain Rule ........................................................ 33
  2.6.3 Total Probability .................................................. 34
  2.6.4 Bayes Rule ......................................................... 34

3 Testing Environment ...................................................... 36
  3.1 Martian Methane Plume Dispersion Model ........................ 36
    3.1.1 Previous Investigations and Statistical Methods .......... 37
    3.1.2 Model Overview ................................................ 43
    3.1.3 Model Analysis ................................................ 49
    3.1.4 Conclusion ...................................................... 55
  3.2 Enceladus Plume Model .............................................. 55
    3.2.1 Model Description ............................................. 59
    3.2.2 Conclusion ...................................................... 64
### List of Tables

3.1 **General and Conditional Mean Data**: Data collected by Jones during his experiments on an ionized-air plume and from Farrell et al on their filament-based plume model. ........................................... 41

3.2 **Amplitude Data**: Unfiltered data collected by Jones during his experiments on an ionized-air plume and from Farrell et al for their filament-based plume model. The five parameters listed here are the standard deviation, the skewness, the kurtosis (the 2nd, 3rd and 4th moments respectively), the peak-to-mean ratio and the intermittency. 42

3.3 **General and Conditional Mean Data**: Presented is the (general) mean of the simulated measurements of the model at four positions and the conditional ‘in the plume’ mean. ........................................... 50

3.4 **Statistical Data for Modeled Plume**: The five statistical parameters; standard deviation, skewness, kurtosis, peak-to-mean ratio, and intermittency, are given for four sets of data describing four locations along the nominal wind position for a ten minute long simulation. The first three parameters are normalized with the conditional mean at that location, with thresholds 5% of the maximum levels seen at the 2 m position. ................................................................. 51

3.5 **Parameters in Stress Model**: The various parameters for the stress model and the parameters for Enceladus. ........................................... 61
3.6 **Previous Model Results**: Shown are the results from previous authors for flybys E0, E1, and E2 and for flybys E3, E5, and E7. These show the estimated activated sources. The estimated sources are compared to the stress model for the eight sources.

B.1 **Select Flyby Data**: Shown are the speed and closest altitude of select *Cassini* encounters with Enceladus.
List of Figures

2.1 Model of Martian Methane Dispersion: Simulation of a pulse release of methane at 15°N, 310°W 5 sol after release. . . . . . . . . 11

2.2 Enceladus Jets Forming Plume: Cassini image of Enceladus showing the many individual jets of fine icy particles erupting from the south polar region and forming the plume. Courtesy NASA/JPL . . . . . . 12

2.3 South Polar Terrain Showing Tiger Stripes: The south polar terrain is shown with the tiger stripes and hot spots identified as the red squares and the eight main source locations identified as the white circles. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14

2.4 Observed Flight Behaviours in Moths: Results of observing male moths fly through pheromone plumes. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18

2.5 Braitenberg Vehicle: The figure shows two versions of the Braitenberg vehicle. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19

2.6 Model of a Neuron: The output of a previous neuron and the zeroth signal are shown with their weight and bias respectively. The activation is the sum of these weighted signals. The output is the activation energy put through the activation function. . . . . . . . . . . . . . 22

2.7 The Multi-Layer Perceptron This is an example of a MLP with 4 inputs, 1 hidden layer with 2 neurons, and 3 outputs. . . . . . . . . . 24
2.8 A Recurrent Neural Network with Internal Feedback: Recurrent neural networks are similar to the MLP except that one or more layers in the network feedback.

2.9 R1 RNN: A simple recurrent neural network with one neuron exhibiting feedback.

3.1 Sutton Model: A contour map of the Sutton model used for comparison to the model presented.

3.2 Long Term Average Comparison: Results from the simulation with the plume source at (10,0). The results presented in (3.2a) are the mean over a ten minute period compared to Sutton’s Gaussian model. In (3.2b) a snapshot of the instantaneous plume is shown. Blue color represents no concentration where red denotes the highest concentration. It can be seen that the instantaneous plume contains several ‘puffs’ of concentration at high levels with large amounts of ‘clean’ space in between.

3.3 Comparison of Means: The figures above show the mean concentration over four positions for Jones’ experiment, Farrell et al ‘s simulation and the model presented in this thesis. All data sets are normalized by that set’s concentration at 2 m.

3.4 Temporal Statistics: The charts above give the distribution of times for the length of puffs (continuous concentrations above threshold) and the time for a puff to return (continuous time below threshold) at 5 m and 15 m downwind for two different thresholds.

3.5 Particle Size in Plume: Results published by Postburg et al showing the average particle size through the plume. Jets III VI and VII are seen as an artifact of the observations.
3.6 **Model of Tiger Stripes**: The blue crosses represent the modeled boundary of the tiger stripes for comparison to Figure 2.3. The black dots indicate the center of Spitale and Porco’s eight identified main sources.  

3.7 **Stresses on Eight Source Regions** The figure shows the stress levels on the eight source regions. Positive stresses correspond to tension loading while negative stresses compare to compressive loading.  

4.1 **Centerline Concentration**: This figure shows centering concentration for the 2D Sutton model with respect to distance downwind from the source.  

4.2 **Model**: This figure shows 2D Sutton model as well as the boundaries for the desired motion of the vehicle as it travels upwind towards the source. The concentration of the methane is shown in color on a scale from 0 to 1 kg/m². The boundary of the model shown is 10 meters in both directions crosswind from the centerline of the plume, and 20 meters downwind of the source. The boundaries for the desired motion are shown as black boundaries moving downwind the source.  

4.3 **Training Set**: This figure shows an example training set. The top graph shows the input concentration, the middle graph shows the desired response velocity, and the bottom graph shows the result of the vehicle casting.  

4.4 **Testing of Network During Training**: This figure shows the output of tests that were performed during the training process.  

4.5 **Results of 2R Network**: This figure shows a trial of the vehicle navigating through the 2D Sutton plume after training for a successful trial.
4.6 **Showing Consistency of Results:** This figure shows starting locations in the plume which results in successful behaviour and non-successful behaviours. White circles denote the positions of successful trials and black x’s denote locations where the vehicle failed to behave as desired.

4.7 **Showing Robustness of Results:** This figure shows the fused paths of trials from multiple starting positions.

4.8 **Results in a Turbulent Plume:** The figure shows the path taken by the vehicle guided by the neural network through a turbulent plume. The path of the vehicle is shown up to the point where it diverges away from the source. The time-averaged plume is shown behind.

4.9 **Data from Turbulent Plume Trial:** The figure shows the input and output of the neural network while navigating through a turbulent plume.

4.10 **Screen Shot of Turbulent Plume:** A screen shot of the simulation of the neural network navigating through the turbulent plume is shown. The path is shown up to the point of the screenshot. The dynamic plume is shown behind the partial path.

5.1 **SLIM for Various Settings:** Shown are six source likelihood maps based on a wind sensor history of a steady wind in the upwards direction and a single detection event.

5.2 **Estimating the Release Time** The figure shows the estimated release time with relation to the concentration of the puff detected.

5.3 **Standard Deviation of Estimate Release Time** The figure shows the relationship with mean and standard deviation of the release time estimate.
5.4 **Initial SLIM for Original Algorithm**: This figure shows the SLIM after the first detection event experience by the original algorithm for the stationary vehicle. .................. 97

5.5 **Initial SLIM for Amended Algorithm**: This figure shows the SLIM after the first detection event experience by the amended algorithm for the stationary vehicle. .................. 97

5.6 **Data from Stationary Experiment, Original Algorithm**: This figure shows results for the stationary vehicle during the experiment for the original algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM. .................. 98

5.7 **Data from Stationary Experiment, Amended Algorithm**: This figure shows results for the stationary vehicle during the experiment for the amended algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM. .................. 98

5.8 **Data from Dynamic Experiment, Original Algorithm**: This figure shows results for the dynamic vehicle during the experiment for the original algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM. .................. 100

5.9 **Data from Dynamic Experiment, Amended Algorithm**: This figure shows results for the dynamic vehicle during the experiment for the amended algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM. .................. 101
B.1 **Penetration Depths for Different Penetrator Designs**: This chart shows the depth of penetration for different Weight/Diameter Combinations for a 300 m/s impact into ice. 113

B.2 **Initial Altitude vs Velocity**: The plot shows the combination of initial altitude and speed which result in an impact velocity of 300 m/s. 115

B.3 **Initial Altitude vs Time**: The plot shows the combination of initial altitude the time in free fall given that the impact velocity will be 300 m/s. 115
# List of Acronyms

<table>
<thead>
<tr>
<th>Acronyms</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN</td>
<td>Artificial Neural Network</td>
</tr>
<tr>
<td>BPTT</td>
<td>Back Propagation Through Time</td>
</tr>
<tr>
<td>CDA</td>
<td>Cosmic Dust Analyzer</td>
</tr>
<tr>
<td>CIRS</td>
<td>Composition Infrared Spectrometer</td>
</tr>
<tr>
<td>DAR</td>
<td>Diffusion-Advection-Reaction</td>
</tr>
<tr>
<td>EKF</td>
<td>Extended Kalman Filter</td>
</tr>
<tr>
<td>FFNN</td>
<td>Feed Forward Neural Network</td>
</tr>
<tr>
<td>HMM</td>
<td>Hidden Markov Methods</td>
</tr>
<tr>
<td>INMS</td>
<td>Ion and Neutral Mass Spectrometer</td>
</tr>
<tr>
<td>ISS</td>
<td>Imaging Science Subsystem</td>
</tr>
<tr>
<td>MLP</td>
<td>Multi-Layered Perceptron</td>
</tr>
<tr>
<td>MLS</td>
<td>Mars Science Laboratory</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Density Function</td>
</tr>
<tr>
<td>RNN</td>
<td>Recurrent Neural Network</td>
</tr>
<tr>
<td>RTRL</td>
<td>Real Time Recurrent Learning</td>
</tr>
<tr>
<td>SLIM</td>
<td>Source Likelihood Map</td>
</tr>
<tr>
<td>SPT</td>
<td>South Polar Terrain</td>
</tr>
<tr>
<td>UKF</td>
<td>Unscented Kalman Filter</td>
</tr>
<tr>
<td>VIFF</td>
<td>Vector In Forward Flight</td>
</tr>
<tr>
<td>VIMS</td>
<td>Visual and Infrared Mapping Spectrometer</td>
</tr>
</tbody>
</table>
## List of Symbols

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C(p, t)$</td>
<td>Concentration at a defined position and time</td>
</tr>
<tr>
<td>$\bar{C}(p, t_i, t_f)$</td>
<td>Average concentration at position from initial to final time</td>
</tr>
<tr>
<td>$D$</td>
<td>Diffusion coefficient</td>
</tr>
<tr>
<td>$\vec{u}(p, t)$</td>
<td>Wind vector</td>
</tr>
<tr>
<td>$\vec{a}(p, t)$</td>
<td>Advection vector</td>
</tr>
<tr>
<td>$\vec{v}(p, t)$</td>
<td>Vortex vector</td>
</tr>
<tr>
<td>$\vec{t}(p, t)$</td>
<td>Turbulence vector</td>
</tr>
<tr>
<td>$Q(p, t)$</td>
<td>Reaction value</td>
</tr>
<tr>
<td>$\Gamma(p)$</td>
<td>A record of concentrations at a given position</td>
</tr>
<tr>
<td>$\bar{\Gamma}(p)$</td>
<td>Average concentration of concentration record</td>
</tr>
<tr>
<td>$\tilde{\Gamma}(p)$</td>
<td>Conditional average concentration of concentration record</td>
</tr>
<tr>
<td>$\hat{\Gamma}(p)$</td>
<td>Maximum concentration of concentration record</td>
</tr>
</tbody>
</table>
\( \gamma \)  
Threshold concentration for detection event

\( \sigma_{\Gamma} \)  
Standard deviation of concentration record

\( S_{\Gamma} \)  
Skewness of concentration record

\( K_{\Gamma} \)  
Kurtosis of concentration record

\( I \)  
Intermittency

\( t_p \)  
Time in a puff

\( t_g \)  
Time in a gap

\( Z(p, t) \)  
Wind surface

---

**Enceladus Model**

\( C(R, h) \)  
Concentration at position defined by radius and altitude

\( R \)  
Radius from normal vector to position

\( h \)  
Altitude from moon surface

\( Q \)  
Source strength

\( \Psi_0 \)  
Jet size

\( u_c \)  
Characteristic velocity

\( k_c \)  
Diffusion constant

\( \theta \)  
Longitude

\( \phi \)  
Colatitude

\( m \)  
Mean anomaly

xix
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$</td>
<td>Angle between fault and lines of longitude</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>Normal stress</td>
</tr>
<tr>
<td>$\tau$</td>
<td>Shear stress</td>
</tr>
</tbody>
</table>

**Neural Networks**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v$</td>
<td>Activation energy</td>
</tr>
<tr>
<td>$y$</td>
<td>Output energy</td>
</tr>
<tr>
<td>$g$</td>
<td>Activation function</td>
</tr>
<tr>
<td>$w_{mn}$</td>
<td>Weight from neuron $n$ to neuron $m$</td>
</tr>
<tr>
<td>$b$</td>
<td>Bias value</td>
</tr>
<tr>
<td>$e$</td>
<td>Innovation (error function)</td>
</tr>
<tr>
<td>$\eta$</td>
<td>Learning rate parameter</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Local gradient</td>
</tr>
</tbody>
</table>

**Kalman Filter**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x$</td>
<td>State vector</td>
</tr>
<tr>
<td>$\hat{x}_{ij}$</td>
<td>Estimate of state vector at time $i$ with measurements through time $j$</td>
</tr>
<tr>
<td>$F, f$</td>
<td>State transition matrix, function</td>
</tr>
<tr>
<td>$G$</td>
<td>Input matrix</td>
</tr>
<tr>
<td>$H, h$</td>
<td>Measurement matrix, function</td>
</tr>
</tbody>
</table>

xx
\( Q \)  System noise covariance matrix

\( R \)  Measurement noise covariance matrix

\( K \)  Kalman gain

\( P_x \)  State covariance matrix

\( P_y \)  Measurement covariance matrix

\( P_{xy} \)  State-measurement variance matrix

---

**Source Likelihood Maps**

\( \mathbf{p} \)  Position

\( u \)  Wind vector

\( v(t_1, t_2) \)  Wind transport vector

\( f \)  Probability density function

\( N\{\mu, \sigma\} \)  Denotes a normal distribution with mean \( \mu \) and standard deviate \( \sigma \)

\( S_{ij}(t_1, t_2) \)  Probability of puff created at \( t_1 \) in region \( R_j \) based on detection in region \( R_j \) at \( t_2 \)

\( B_{ij}(t_0, t_2) \)  Probability of puff created in region \( R_j \) based on detection in region \( R_j \) at \( t_2 \)

\( \gamma_{ij}(t_0, t_2) \)  Probability of puff created in region \( R_j \) based on non detection in region \( R_j \) at \( t_2 \)

\( W(t_1, t_2) \)  Integrated white noise from \( t_1 \) to \( t_2 \)
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_i$</td>
<td>Region in SLIM</td>
</tr>
<tr>
<td>$\eta$</td>
<td>White noise process</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Dirac delta function</td>
</tr>
<tr>
<td>$\sigma^2$</td>
<td>Variance</td>
</tr>
</tbody>
</table>

$\Pi$  SLIM
Chapter 1

Introduction

1.1 Motivation

In many planetary exploration missions, the target being sought usually leaves markers or traces that provide clues to its location. The most sought after targets in planetary exploration are signs of extraterrestrial life, both past or present. NASA’s ‘follow the water’ philosophy is an example of a target location as all known life requires water to thrive. Many missions focus their resources on locating regions where water exists or may have existed. Sedimentary rocks are markers for past water and used to identify landing sites for rover missions. Before picking a landing site for the most recent Mars lander, the Mars Science Laboratory (MSL), scientists attempted to use data from orbiters to determine which locations would have the best chance of finding these desired rock formations [1]. Water and sedimentary rock formations are but one example of the target and marker analogy.

In many cases, markers have a defined spatial relationships with the target in question. These known spatial relationship are called the ‘forcing function’ as this function determines the distribution of the markers in relationship to the origin, or target. The forcing functions are key in modeling the spread of markers. Once the distribution of markers is known, then the inverse problem is analogous to the
odour source localization problem seen in nature. Many animals and insects exhibit behaviours that allow them to find the source of pheromones to a mate or odour to food. These seeking behaviours result from individual events of detection above threshold levels and knowledge of the wind, which acts as the forcing function. The behaviours these animals and insects exhibit fall into two main groups: chemotaxis, behaviour based mainly on concentration gradients; and anemotaxis, behaviour primarily based on the forcing function. Turbulent odour plumes are characterized as sparse areas of high concentration where the gradient of the concentration does not always point to the source. These areas, called puffs, are an analogy to the markers and the wind is the forcing function. If the turbulent plume adequately represents the distribution of markers about a science target, then the behaviours exhibited by insects and animals in solving the odour source localization problem may be suitable for a planetary vehicle seeking such a science target.

This thesis presents the analogy of the odour source localization problem as the model for autonomous science. This project was funded through a partnership with the Advanced Concepts Team at the European Space Agency. The odour source localization problem is taking cues from the local vicinity and updates the predicted location of the most valuable target and determine where to travel next. Two planetary exploration missions analogous to the source localization problem are locating the sources of Martian methane and navigating a high speed lander to target the icy jets at Enceladus’ southern pole.

Martian methane has gained attention in the scientific community. Detection of atmospheric methane on Mars reveals a planet that is more intriguing than previously thought. If the source of the methane is found to be biotic in nature, it would be the first evidence that life exists or existed on Mars. The spatial resolution of these methane detections has been on the order of kilometers and does not sufficiently
constrain the source of the plume, the scientific point of interest. Although locating methane sources is not a primary scientific objective, the MSL Curiosity rover has instruments on board to allow for the measurement of methane in the Martian atmosphere and soil [2]. Should Curiosity find significant quantities of methane in the Martian atmosphere a mission to locate the source of that methane would be the next logical step. This thesis discusses a surface exploration system that localizes the source of the plume during the traverse phase of a rover mission. The sensor suite includes a methane detector that provides the concentration measurement. A wind detector provides the mean wind vector and is utilized to predict the forcing function acting on the methane gas in the target area. The odour source localization problem would be directly analogous to following the clues in methane concentration and wind measurements to the source of the methane.

The icy jets at the southern pole of Enceladus are believed to originate from a subsurface ocean. Data from in-situ measurements of Enceladus’ icy jets have been gathered from the Cassini-Huygens mission. The habitability of the subsurface oceans and the presence of life remain unanswered questions. These questions can be answered by probing into the subsurface with a planetary penetrator. Planetary penetrators are self-contained spacecraft vehicles that act as high-speed landers with the goal of traversing a distance through the solid material during impact. Planetary penetrators are typically characterized as low-mass (3-10 kg), high impact velocities (300 m/s), with a long cylindrical shape and an oblunt nose cone [3]. One operation method for a penetrator mission is to release the penetrator from a main spacecraft during a flyby of a science target and, using a propulsion system, decrease velocity to allow the gravity of the target body to deorbit the spacecraft. Three-axis attitude control would be necessary to have a full range of controllable thrust directions.

The targeting requirements for the proposed Enceladus mission are much tighter than for other lander missions. While rover missions have a landing ellipse with a
major axis of tens of kilometers [4], the target ellipse of the penetrator will have a radius dependent on the size of the icy jets. In addition, the actual target location is unknown and must be estimated in real-time throughout the descent. In a traditional penetrator system, the penetrator is deorbited through the use of reverse thrusters and is then released to free-fall to the surface. Lateral motion control is required to maneuver the penetrator towards the target while the penetrator is in free-fall. The ability to vector in forward flight (VIFF) allows the fine control of the spacecraft, which increases the likelihood of hitting a target but adds complexity to the spacecraft control.

The targeting environments in the two proposed planetary exploration missions are different. For Enceladus, the targeting of a single icy jet out of hundreds is more analogous to the source localization problem in laminar conditions, with smooth transitions between lower and higher gradient areas, and multiple sources. The Enceladus plume problem is opposite to the one-source turbulent conditions that are expected in the Martian methane mission. The vehicles for both proposed missions are also different: a rover for the Martian targeting environment will be a slow moving vehicle in approximately planar motion; a penetrator for the Enceladus mission will be a high-speed spacecraft moving in three-dimensions. The vehicles used in the exploration of these environments have implications for mission planning. The rover will be traveling at a slower rate as compared to the wind forcing function experienced on Mars [5]. In the case of the planetary penetrator, maneuvering the penetrator outside its ballistic trajectory is difficult. It can be seen that the Martian methane and Enceladus missions provide for the two extremes in conditions that would make the source localization problem difficult. Modeling these two environments provides a robust testing platform for the autonomous science algorithms developed.

Animals and insects employ reactive control to locate the odour source being sought. The source is realized when the animal or insect successfully reaches the
source. For successful localization to occur, the animal or insect traces the plume, whether following gradient or the forcing function, until it leads to the source. Plume tracing is effective because it ends with direct confirmation of the source location. From a robotic view, this method is attractive because it results in an egocentric control; that is, the rover responds to its current environment without regard for its past location. Reactive control results in low computational algorithms for controlling the vehicle. Egocentric control allows for unbounded search of the odour source, which is especially important in planetary missions where prior information on the target may be lacking.

Robotic vehicles can take advantage of mapping algorithms which, to date, no animal or insect has been seen to employ. Source Likelihood Maps (SLIM) divide the map into regions with assigned probability based on where the source is likely to be. These maps can be updated as new information about the plume becomes available. Benefits of the mapping approach include the decouple of the localization and control algorithms for the vehicle. The algorithm is able to estimate the source location without the vehicle tracing the plume in the direction of the source. Although the location of the source may not be confirmed without the need to traverse to the estimated source location, the path planner can balance mission priorities and perform other tasks while continuously updating the estimated source location. The cost of mapping algorithms is the potentially high computation needed to perform the algorithm as compared to egocentric controls.

1.2 Scope

The main objective of this thesis is to validate the odour source localization strategy as a valid analogy for particular planetary exploration missions. This is done by analyzing the environments that planetary exploration may occur, and the various
strategies in current existence for solving the source localization problem. The work is divided up into three parts: the development of suitable plume models, simulating simple insect behaviour using neural networks, and improving probability based methods for predicting source location.

1. The models for the two proposed missions are developed. The first model is the plume caused by Martian methane dispersion from a single source. The second model is the plume caused by the sum of Enceladus’ numerous icy jets. These two models are meant to simulate the characteristics of their individual environments that make the plume tracing problem difficult. They are not developed as scientific models, although the results of the model should replicate the measurements from their respective environments. Note that for this thesis, testing of the algorithms is based on the Martian methane model only.

2. Insects and other animals solve the source localization problem biologically. They do this with limited measurements and actuation and, in some cases, limited computation. Neural networks are mathematical models of how individual neurons in the brain interact with each other. A neural network is trained on a simplified version of the behaviours seen in moths while tracing odour plumes to form a reactive controller for a 2D vehicle. The results show the strengths and weaknesses of neural networks for control, for both this problem and in general use for robotics.

3. Source likelihood maps (SLIM) help to decouple the navigation and control of the vehicle. The SLIM method can be useful when balancing the source seeking behaviour with other operations such as obstacle avoidance. The SLIM algorithm presented by Pang and Farrell [6] has been shown to converge to a solution for the source location more quickly than other methods. There are issues with the accuracy of that estimate. The accuracy is improved by making
use of the concentration value as opposed to the binary sensor used by Pang and Farrell [6]. The need for such an addition is shown as well as the results of the addition.

1.3 Overview

The remainder of this thesis will proceed as follows. Chapter 2 will detail the current state of modeling the Martian and Enceladus environments as well as background information needed to appreciate the controller development process.

Chapter 3 will detail the Martian methane model testing environment for the control algorithms. The plume dynamics and past modeling of atmospheric odour dispersion as well as details of the new model will be presented. The analysis of the plume model statistics are compared to experimental data and another dispersion model. The case will be made that this model is suitable for testing algorithms meant to mimic the odour source localization strategies found in nature. The properties of natural dispersion that make this task difficult are seen in the model. A model of Enceladus' icy jets is developed and detailed in this chapter. First the requirements of the model that make the model useful for testing the developed algorithm are described. These requirements force the development of a model that differs from those developed by others. The model is presented and shown to match the general trends seen in past models.

Chapter 4 discusses the development of the neural network controller, the results of testing and the conclusions that can be drawn from the experiments. Chapter 5 details the source likelihood mapping algorithm of Pang and Farrell [6]. An analysis of the algorithm is performed which shows the need for an additional component of the algorithm for improved accuracy. The addition to the algorithm and the results of testing are presented. Chapter 6 summarizes the contributions of this thesis and
presents future work that can be performed to follow up on the works presented.
Chapter 2

Background and Literature Review

This chapter will outline some of the previous knowledge required to realize the results of this thesis. The environments of Mars and Enceladus are discussed and previous models of their respective plumes are shown. Next, odour source localization is discussed for animals and insects. A history of robotic odour source localization for terrestrial purposes and a discussion on the requirement differences between planetary science missions and terrestrial odour source localization tasks will follow. The basics in neural networks and recurrent neural networks will be detailed for understanding the development of the reactive neural controller. The Kalman filter and the Unscented Kalman filter are detailed, as the UKF will be adapted into training rules for the recurrent neural network. Finally, Bayes theorem and other properties of probability densities related to source likelihood mapping (SLIM) are discussed.

2.1 Mars Methane

Three individual measurements of atmospheric Martian methane occurred in 2003 and 2004. Formisano et al. [7] reported detection from the Planetary Fourier Spectrometer on board Mars Express orbiter. At the same time, Krasnopolsky et al. [8] used the Fourier Transform Spectrometer at the Canada-France-Hawaii Telescope to observe
methane. Finally, Mumma et al. [9] published results of detection during the northern summer of 2003 using a ground based high-dispersion infrared spectrometer from Earth. Mischna et al. [10] used these measurements and a global circulation model to predict from which part of the planet the methane originated (See Figure 2.1).

More extraordinary than the discovery of methane in the Martian atmosphere is the spatial and temporal variability of the methane concentration from consecutive measurements. Lefèvre and Forget [11] note that the seasonal variability noted by two of the measurements does not agree with the 300 year lifetime of methane expected in the Martian atmosphere based on known chemistry. Zahnle et al. [12] further question the results of the observations pointing out that the filtering of Earth’s atmosphere out of the ground-based measurements is difficult and can easily produce false results. In order to gather more evidence for or against the presence of methane on Mars, Curiosity, that landed on Mars in early August 2012, features an advanced sensor suite to measure methane concentrations both in the atmosphere and in soil samples [13].

Confirmation of atmospheric methane by Curiosity would imply either geological or biological activity occurring on Mars. The small time scales of the methane measurements imply that an active source exists and that some sort of destruction process is taking place. Chastain and Chevrier [14] noted that methane clathrates are a viable source of the methane on Mars. Methane clathrate deposits are expected on Mars, and the dissociation of the gas can occur in many natural processes. Chassefière [15] further discusses the possibility of metastable clathrates and states the correlation with methane and water vapor measurements to experiments performed. Metastable clathrates help explain the time scale of methane production/losses; lifetime on the order of 6 months is expected as opposed to atmospheric models that predict 300 years. Furthermore, it has been proposed that Martian dust storms could act as a possible methane sink, further reducing its lifetime [16].
Figure 2.1: Model of Martian Methane Dispersion: Simulation of a pulse release of methane at 15°N, 310°W 5 sol after release. This model was formed through a global circulation model by Mischna et al. [10]

Mischna et al. [10] used the measurements of Mumma et al. [9] to constrain the plume source location and release time. This was the first attempt to constrain the source location of the methane plume. Mischna et al. [10] used the Mars Weather Research and Forecasting General Circulation Model to model the advection patterns across the entire planet. Two resolutions were used, 1.25° × 1.25° and 5° × 5°, that result in resolutions of 75 km and 300 km at the equator, respectively. The four parameters that the authors could alter for their experiments were source location and season, emission duration, source strength, and species lifetime. The source strength was assumed to be the entire methane supply detected in the plume by Mumma et al. [9]. The species lifetime was assumed to be 1000 sols (the Martian day) as the median of the range estimated by Mumma et al. [9]. Through fitting the model to the measurements and data, Mischna et al. [10] determined that the plume source must cover a wide range of the planet covering a parallelogram 80° latitude by 15° longitude pointing north-west/south-east. The emission duration was found to be short and occurring no more than 1-2 sols prior to the measurements being taken.
2.2 Enceladus

Initial interest in the Saturian moon Enceladus was due to its orbital position in the densest part of Saturn’s E-ring. Models of the E-ring showed instability, suggesting the E-ring should collapse over time without a source of new particles [17]. Enceladus was thought to be the source needed for stability. The Voyager mission included a flyby of the moon’s northern hemisphere and viewed a diverse geography with large number of craters in the north with less craters and smooth plains in the south. The diversity in crater density suggested some form of geological activity inside the moon [18], fueling more interest in its study. The Cassini-Huygens spacecraft was launched in 1997 and entered orbit around Saturn in 2004. The spacecraft has made several close flybys of Enceladus, whose events are numbered chronologically (E#), and was able to prove that Enceladus is the source of the E-ring through observation of erupting water gas and dust particles forming a plume out of the south polar region of the moon (Figure 2.2).

Figure 2.2: Enceladus Jets Forming Plume: Cassini image of Enceladus showing the many individual jets of fine icy particles erupting from the south polar region and forming the plume. Courtesy NASA/JPL
2.2.1 Enceladus Background

*Cassini* has made numerous flybys of Enceladus, many of which were scheduled as part of *Cassini’s* extended missions. *Cassini* has multiple instruments on board that are able to take in-situ measurements of Enceladus’ interior through sampling the plume during flybys. These instruments include the Imagining Science Subsystem (ISS), the Visual and Infrared Mapping Spectrometer (VIMS), the Composite Infrared Spectrometer (CIRS), the Cosmic Dust Analyzer (CDA), the Ion and Neutral Mass Spectrometer (INMS), and the Ultraviolet Imaging Spectrograph. *Cassini* also employs its magnetometer to measure the distortion in Saturn’s gravitational field.

The southern plains viewed by *Voyager* turned out to be fine fractures when viewed in higher resolution by *Cassini*. Four long, distinct fissures running 130 km long, 2 km wide, with ridges 100 m high on each side named ‘tiger stripes’ run through the south polar region [18]. These tiger stripes are the most prominent fractures in the south polar region, and are individually named Damascus, Baghdad, Cairo, and Alexandria. The plume whose origin is the south polar region is made up of individual jets as seen in Figure 2.2. Spitale and Porco [19] used images from the ISS instrument to triangulate the position of the most prominent jets that were able to be identified in the images and these locations corresponded to locations on the tiger stripes. The locations of the sources on the tiger stripes are denoted in Figure 2.3 as the white circles with the individual sightings marked as diamonds in different colors.

Postberg et al. [21] have stated that the particles sampled from the E-ring and plume imply that there is a subsurface ocean on Enceladus that is in contact with its rock core. Nimmo et al. [22] proposed that shear stresses could cause local pockets of liquid water near the surface, but there would also need to be an additional large subsurface ocean. Tobie et al. [23] did one of the most extensive studies of the effect of subsurface heating due to tidal forces on Enceladus. They concluded that there
needed to be a decoupling layer between the south polar terrain ice and the core of the moon for the dissipation of the tidal heat to occur as observed by Cassini. A global subsurface ocean is not required, however, as long as the ocean covers the majority of the south polar region province. This liquid layer could be one of the most accessible locations in the solar system where extraterrestrial life could possibly exist.

The majority of habitability studies conducted to date focus on Europa as the most likely location for finding extraterrestrial life in the solar system [24, 25, 26]. Like Enceladus, Europa is thought to experience tidal heating that forms a large subsurface ocean between its ice crust and rock core [27]. The disadvantage to studies of Europa is the large ice crust that is thought to be up to hundreds of kilometers thick that separates the possible habitat from study. Even if signs of life were to permeate through the ice to the surface, Europa is blasted with radiation from the Sun contaminating the top meters of ice. Therefore the minimum effort to detect life would consist of drilling meters into the ice [28]. In contrast, the geysers on Enceladus’ southern polar region allow for an in-situ measurement of the interior composition.
Many models of Enceladus’ interior have already been derived from the composition of the plume as measured by Cassini. In addition, there is evidence to support that the material that falls to the moon’s surface as it travels through the E-ring makes its way into the interior of the body. Thus it is proposed that water particles erupted from the jets make their way into the E-ring of Saturn. There, they react with radiation and some particles turn to hydrogen-peroxide (H$_2$O$_2$). As Enceladus travels through the E-ring, it collects E-ring particles on its surface including those that may have formed in the E-ring; these particles permeate through the ice crust into the interior. Having a geochemical cycle as described would greatly increase the likelihood of life developing [28].

2.2.2 Previous Plume Models

Many studies have attempted to determine Enceladus’ plume structure by comparing models to measurements made by Cassini during its many flybys. Tian et al. [29] modeled the plume based on UVIS measurements, Smith et al. [30] modeled the plume based off INMS measurements, and Tenishev et al. [31] used a combination of sensors from different encounters. Saur et al. [32] used magnetic field observations to show the variability in the plume between the E0, E1 and E2 encounters using a Gaussian plume model. Dong et al. [33] used INMS data to fit a Maxwellian velocity model of the plumes. The Gaussian and Maxwellian models are two common models used to represent Enceladus’ plume structure. Saur et al. [32] and Dong et al. [33] modeled the plume as a superposition of eight plumes originating from the eight locations defined by Spitale and Porco [19]. Using eight superimposed plumes instead of one was found to achieve more representative results of the instrument data than a single plume source.

The Gaussian plume model presented in [32] is shown in Equation 2.1. The variable $n_0$ is the density at the center of the plume at the surface, $r$ is the radial
distance from the center of the moon, $R_E$ is the radius of Enceladus, $\Theta$ is the angular distance from the plume center, $H_\Theta$ is the angular width of the plume, and $H_d$ is the depletion length scale that limits the total gas content of the plume.

$$n_n(r, \Theta) = n_0 \left( \frac{R_E}{r} \right)^2 \exp \left[ -\frac{\Theta^2}{H_\Theta^2} \right] \exp \left[ -\frac{r - R_E}{H_d} \right]$$

(2.1)

The Maxwellian velocity plume presented in [33] is shown in Equation 2.2. The variable $n_s$ is the number density of the source, $r_s$ is the radius of the source location, $M$ is the thermal Mach number, and $\theta$ is the angle between the vector from the source location to the position in question and the vector normal to the surface of the moon at the source location.

$$n(r > r_s, \theta, M) = \frac{n_s r_s^2}{\pi r^2} \left\{ \frac{2M \cos \theta}{\sqrt{\pi}} e^{-M^2} + e^{-M^2 \sin^2 \theta} \left[ 1 + 2M^2 \cos^2 \theta \right] \left[ 1 + \text{erf}(M \cos \theta) \right] \right\}$$

(2.2)

The source emission rates for the eight sources was solved by fitting the data to the measurements of Cassini, which suggests that the plume models may only be relevant to the well mixed plume at high altitudes above the surface.

2.3 Plume Tracing

The plume source localization problem has been well observed in nature. Recently, attempts to solve the problem robotically have been made to for assisting engineers in various fields. Examples include determining the location of pipeline leaks [34], locating the source of gas leaks or fire in a building [35], and locating underwater mines. These robotic solutions are substitutes for trained animals, such as dogs and dolphins, that have been used for completing similar tasks. Whether the solver is robotic or biological, the problem remains the same; locate the origin of the target
using only odour detections and forcing function measurements. The source localization problem is difficult due to the nature of the plumes formed. In most engineering applications, the plume is turbulent. Turbulence results in a plume that is sparse; that is, a plume that has a very low background level and discrete ‘puffs’ of high concentration separated by large areas of low concentration or ‘clean air’. In many circumstances, the gradient or direction of the forcing function does not point to the source. This section will detail the solutions seen to date, both in nature and in previous robotic solutions.

\subsection{2.3.1 Characteristics of Moths}

Plume tracing is a well observed behaviour in nature. Lobsters have been observed to forage based on scent and robotic replication of their behaviours have been implemented \cite{36}. Long range olfactory based navigation has been viewed in sea birds \cite{37}. Many insects have been seen to forage and search for mates using olfactory methods in response to scents and pheromones including moths \cite{38}, wasps \cite{39}, and mosquitoes \cite{40}. Moths tracing odour plumes in search for potential mates is currently the best documented biological scent-tracing behaviour, with extensive observations performed by Farkas and Shorey \cite{41}, Mafra-Neto and Carde \cite{38}, and Murlis et al. \cite{42} among many.

The survey of techniques for plume tracing in insects conducted by Murlis et al. \cite{42} mostly focuses on work done by studying the attraction of male moths to females. They note that there are two distinct phases of plume-tracing: acquiring the plume and following the plume. In the acquisition phase, the moth exhibits a casting pattern, where they move perpendicular to the wind direction back-and-forth, progressively widening their passes until they reconnect with the plume. Once the plume is located the moth heads upwind in bursts until the plume is inevitably lost again, or until the female is located visually. Mafra-Neto and Carde \cite{38} also did experiments with
Figure 2.4: Observed Flight Behaviours in Moths: Mafra-Neto and Carde [38] show their results of observing male moths fly through pheromone plumes. The moth’s position is tracked and plotted as it searches for the odour source. Two different plumes are used: ribbon plumes where the moths typically travel slow and zigzagging (left), and turbulent plumes where moths cast until they contact an odour puff (blue arrow) and then quickly move upwind (right) When the odour is not detected for a defined duration, the moth restarts its casting pattern.
Figure 2.5: Braitenberg Vehicle: The figure shows two versions of the Braitenberg vehicle: one where each chemical sensor powers the wheel on that side proportional to the concentration measured (left), and the other where the sensors power the opposite side wheels (right).

moths and noticed that the moths did not fly upstream in the plume in all cases. Two types of plumes were used: ribbon plumes characterized as a continuous but meandering plume with low Reynolds number, and a turbulent plume that consisted of odour pulses separated by gaps of lower concentration or clean air. In situations where the ribbon plume was encountered, the moths flew at a slow rate with a zigzag pattern. They fly upwind at high velocities when the turbulent plume is encountered. These patterns were plotted and are shown in Figure 2.4.

2.3.2 Robotic Strategies

The earliest robotic techniques for locating an odour source consisted of only instruments for measuring the concentration of the substance of interest. The most fundamental of these designs is the Braitenberg vehicle [43] that uses two chemical sensors directly connected to each wheel, shown in Figure 2.5. The level of concentration proportionally drives the motor of the wheel it is connected to and thus the instantaneous chemical gradient controls the vehicle. In turbulent plumes the instantaneous gradient and the direction to the plume source are often not aligned, and
therefore more information is needed to efficiently find the source. Most current techniques for robotic scent tracking are aided by a wind sensor, or the equivalent sensor for detecting the forcing function in the relevant fluid. Measurements of the plume can be compared with models of the plume’s structure and used to estimate position and model parameters with a Kalman filter [44], although this works best with laminar plumes or long-averaged measurements. With knowledge of the forcing function, information about the plume structure can be inferred, allowing more advanced guidance systems to anticipate where more information can be found as opposed to heading directly for the estimated source location [45]. Source likelihood maps (SLIM) can be achieved with this combination of measurements. Farrell and Pang proposed a method for creating a SLIM based on hidden Markov methods (HMM) [34] and later improved upon their results by using Bayesian methods [6]. Kowadlo and Russell [46, 47] developed a reactive source localization system based on naive physics that, instead of using a SLIM, would control the instantaneous heading and velocity of the vehicles.

With a source likelihood map, the robot is implicitly constrained in its search for the source to the area in its map, and in the reactionary control of Kowadlo and Russell [46], a bounded map of the air flow was needed. Boundary constraints are acceptable for terrestrial applications when it is assumed that some previous information is available about the source being searched for. Planetary explorations may lack these constraints due to insufficient prior knowledge of the regions being searched. Any information about the Martian methane plume will have a resolution of kilometers, which does not sufficiently constrain the plume for the previously mentioned methods. Therefore, an egocentric, reactionary algorithm is beneficial, but is more difficult to develop.
2.4 Neural Networks

Artificial neural networks (herein known as neural networks) are mathematical structures that use sequences of nonlinear computations to represent a nonlinear function. Neural networks are a network of individual neurons that are modeled to resemble the neurons in the human brain. The rules for how the activations of neurons interact with one another depend on the structure of the neural network. Feed forward networks are made up of layers that only propagate the activation signals forward in the network. Recurrent networks allow for feedback, or backwards propagation of the activation signal.

2.4.1 Neuron Model

Neurons can be characterized as follows: they are connected to other neurons, they have a weight for each input signal from these connections, they have an activation energy and activation function, and they have an output otherwise known as a response. The activation energy of a neuron in the network is the sum of the weighted outputs of the neurons in the previous layer plus a bias. The output of the neuron is the activation energy put through an activation function as shown in the following equations. The model of a neuron can also be seen in Figure 2.6.

\[
\begin{align*}
    v_m &= \sum_{n=1}^{N} w_{mn} y_n + b_m \quad (2.3) \\
    y_m &= g(v_m) \quad (2.4)
\end{align*}
\]

where \(v_m\) is the activation energy of neuron \(m\), \(y_m\) is the output of the \(m^{th}\) neuron in the current layer, \(y_n\) is the output of the \(n^{th}\) neuron of \(N\) in the previous layer, \(w_{mn}\) is the weight between neuron \(n\) of the previous layer and neuron \(m\) of the current
Figure 2.6: Model of a Neuron: The output of a previous neuron and the zeroth signal are shown with their weight and bias respectively. The activation is the sum of these weighted signals. The output is the activation energy put through the activation function.

Layer, \( b_m \) is the bias of the \( m^{th} \) neuron and \( g(\cdot) \) is the activation function. The bias can be considered a weight that is applied to a ‘zeroth signal’ that is always equal to one \((y_0 = 1)\). Therefore the generalized version of Equation 2.3 is:

\[
v_m = \sum_{n=0}^{N} w_{mn} y_n \tag{2.5}
\]

### 2.4.2 Activation Functions

The activation function of a neural network transforms the activation energy of a neuron into the neuron’s output. Neurons in the human brain employ a Heaviside step function, where the neuron outputs a value of 1 if a threshold is reached, otherwise it outputs a value of 0. The sigmoid function seen in Equation 2.6 is a continuous substitute to the Heaviside function. The threshold of the sigmoid function is set to zero, but is implicitly trained through the optimization of the bias on each neuron. To increase the threshold of the activation function, the bias is simply reduced. The constant coefficient \( A \) can be adjusted to change the gradient of the activation function at the threshold. It can be set by the user or it can be optimized as part of the training algorithm.

\[
g(x) = \frac{1}{1 + \exp(-Ax)} \tag{2.6}
\]
Some artificial neural networks are better suited for bipolar output \([-1 1]\) than binary output \([0 1]\). Examples of bipolar activation functions are the hyperbolic tangent function or the bipolar sigmoid seen below.

$$g(x) = \frac{2}{1 + \exp(-Ax)} - 1$$  \hspace{1cm} (2.7)

### 2.4.3 Network Structure

The most common structure for a feed forward neural network is the multilayer perceptron (MLP) that has an input layer, hidden layer(s), and an output layer. It has been shown that one hidden layer of sufficient neurons is able to replicate any function [48]. Each hidden layer consists of an arbitrary number of neurons. The input layer consists of the inputs to the system, and the output layer consists of a number of neurons consistent with the number of desired outputs. Between each layer of neurons are connections as shown in Figure 2.7 that propagate the activation of the neurons from a given layer to all neurons of the next layer. Each of these connections is weighted.

Feed forward networks, including the MLP, are well suited for many classification problems where inputs are directly matched to target outputs. These tasks include speech recognition and image processing among others. Feed forward networks are not suited for tasks where knowledge of previous inputs or outputs influences the current output. In order to take advantage of previous inputs and the corresponding response, feedback is introduced into the structure of the neural network. Networks with this type of structure are known as recurrent neural networks (RNN). There are many differing structures for RNN, mostly defined by where the feedback is placed in the structure. Internal feedback is when a layer of neurons feeds back to itself whereas external feedback is when a layer feeds back to a different layer than itself. Of the training rules for RNNs, the most popular are back-propagation-through-time
Figure 2.7: The Multi-Layer Perceptron This is an example of a MLP with 4 inputs, 1 hidden layer with 2 neurons, and 3 outputs. A weight exists for each connection shown.

(BPTT) and real-time recurrent learning (RTRL) [49].

2.4.4 Real Time Recurrent Learning for Recurrent Networks

The back propagation through time (BPTT) algorithm involves ‘unfolding’ the network for the current time step and converting the recurrent network into a feed forward network with the previous activations of the neurons acting as input neurons. The unfolded network therefore grows for each time step in the training pattern. The folding increases the size of the network for long patterns in time, which adds complexity to the training. Instead, the real-time recurrent learning algorithm is used, which draws its name from the fact that the algorithm updates the weights in real time while still processing the signal. Whereas in the back propagation algorithm the error signal is backpropagated through the network to determine the local gradient, in RTRL the gradient information is forward propagated through time [50]. The algorithm is demonstrated for a basic network is next. The method can be generalized for any recurrent network.

The network is defined as a single neuron with self-feedback. The input is \( u(t) \) and
Figure 2.8: A Recurrent Neural Network with Internal Feedback Recurrent neural networks are similar to the MLP except that one or more layers in the network feedback. A network can either feedback into itself (like the one in this picture), or the outputs can feed into the hidden layer.

Figure 2.9: R1 RNN: A simple recurrent neural network with one neuron exhibiting feedback

the output is $y(t)$. The activation of the neuron before going through the activation function is $v(t)$. Finally, there are two weights: $w_{01}$ is the weight from the input to the neuron, and $w_{11}$ is the weight on the feedback. This network is shown in Figure 2.9.

The error function, $E$, for a desired output $d(t)$ is given as the instantaneous innovation squared.

$$E(t + 1) = \frac{1}{2} [d(t + 1) - y(t + 1)]^2$$ (2.8)
The update rule for each of the weights is

$$w_{mn}(t+1) = w_{mn}(t) - \eta \frac{\partial E(t+1)}{\partial w_{mn}}$$  \hspace{1cm} (2.9)

where $\eta$ is the arbitrary learning rate parameter. The dynamics of the network are shown explicitly below.

$$v(t+1) = w_{01} u(t) + w_{11} y(t)$$  \hspace{1cm} (2.10)

$$y(t+1) = g(v(t+1))$$  \hspace{1cm} (2.11)

The gradient information for a particular weight is defined as $P_{mn} = \frac{\partial y}{\partial w_{mn}}$.

$$\frac{\partial E}{\partial w_{mn}} = \frac{\partial E}{\partial y(t+1)} \frac{\partial y(t+1)}{\partial w_{mn}} = -[d(t+1) - y(t+1)] P_{mn}(t+1)$$  \hspace{1cm} (2.12)

The local gradient can be solved from Equations 2.10 and 2.11.

$$\frac{\partial y(t+1)}{\partial w_{mn}} = \frac{\partial y(t+1)}{\partial x(t+1)} \frac{\partial x(t+1)}{\partial w_{mn}}$$  \hspace{1cm} (2.13)

$$\frac{\partial y(t+1)}{\partial w_{mn}} = g'(v(t+1)) \frac{\partial x(t+1)}{\partial w_{mn}}$$  \hspace{1cm} (2.14)

$$\frac{\partial y(t+1)}{\partial w_{01}} = g'(v(t+1)) \left[ u(t) + w_{11} \frac{\partial y(t)}{\partial w_{01}} \right]$$  \hspace{1cm} (2.15)

$$\frac{\partial y(t+1)}{\partial w_{11}} = g'(v(t+1)) \left[ y(t) + w_{11} \frac{\partial y(t)}{\partial w_{11}} \right]$$  \hspace{1cm} (2.16)
\[ P_{01}(t+1) = g'(v(t+1)) \left[ u(t) + w_{11} P_{01}(t) \right] \] (2.17)

\[ P_{11}(t+1) = g'(v(t+1)) \left[ g(t) + w_{11} P_{11}(t) \right] \] (2.18)

From the above equations it is apparent that a recursive equation is formed for the gradient information of each neuron. Therefore, the gradient information of the neuron must be propagated forward as the RTRL algorithm is run. The algorithm starts with zero initial conditions and then gradients will converge to their actual values as the propagation forward in time occurs. The update rule in Equation 2.9 can then be applied. The RTRL algorithm is shown to be a degenerate form of the extended Kalman filter (EKF) for the recurrent neural network [51] as the back-propagation algorithm is the degenerate form of the EKF for the feed forward network case [52]. In both the Kalman filter and RTRL methods, calculating the gradients can be complex, especially in networks with many weights and many feedback connections. If the network exhibits special forms, then simplified methods for calculating the recursive gradient equations have been proposed.

### 2.4.5 Forced Learning for Recurrent Networks

Forced learning techniques have been found to improve the stability of training and decrease the amount of training time required to converge to the desired network. Forced learning involves updating the network activations towards the desired values instead of the feedback values as the weights of the network are also updated. Unlike the feed forward network where only the current input activations affect the network, all past inputs affect the activations of all neurons in a recurrent network due to feedback. Correcting the weights based on the RTRL method in non-forced learning punishes neurons whose activation is due to past inputs despite the fact that the
weight of the neuron have already been corrected. Forced learning, or partial forced learning, behaves such that corrections to the weights in the network are due to errors in the current step.

### 2.5 State Estimation

In this section, the classical Kalman filter for linear optimal state estimation is presented. As the dynamics of the neural network are non-linear, a non-linear extension of the Kalman filter must be used. Traditionally, this is the extended Kalman filter, which uses a first order linearized approximation of the dynamics through the Taylor series. The extended Kalman filter works well for the feed forward network, but the dynamics of the recurrent network usually proves to be too non-linear for the first order approximation to model accurately. Higher order filters may prove to handle these dynamics better. The work in this thesis uses the unscented Kalman filter for nonlinear estimation problems instead. The unscented Kalman filter (UKF) has the advantage of being able to handle non-Gaussian dynamics through the use of sigma-particles. Another advantage is that the gradients of the dynamics, usually represented as the Jacobian in the first order approximation, are calculated numerically rather than analytically. Not needing the Jacobian is advantageous as the Jacobian of the feedback network can be complicated to compute, especially in a large networks.

#### 2.5.1 Kalman Filter

Kalman filtering is a state estimation technique that is optimal for linear systems that only encounter Gaussian noise. For a linear system subjected to Gaussian noise with zero mean and finite variance (shown in equations 2.19 - 2.22), an optimal estimate of the state \( \hat{x} \) can be formed given measurements \( y \). The normal (Gaussian) distribution for a random noise is denoted \( N\{\mu, \sigma^2\} \) where \( \mu \) is the mean and \( \sigma^2 \) is the variance.
\[ x_{k+1} = F_k x_k + G_k u_k + \omega_k \]  \hspace{1cm} (2.19)

\[ y_k = H_k x_k + \nu_k \]  \hspace{1cm} (2.20)

\[ \omega_k \sim N\{0, Q_k\} \]  \hspace{1cm} (2.21)

\[ \nu_k \sim N\{0, R_k\} \]  \hspace{1cm} (2.22)

The Kalman filter has two distinct phases involved. In the first phase, both the state estimate and the state covariance are updated based on Equation 2.19. In the second phase, the state is updated based on measurements using Equation 2.20. The notation for the state estimate and the covariance at a particular time step is used as follows: the variable \(\hat{x}_{ij}\) denotes the estimate of the state \((x)\) at time step \(i\) using all measurements \((y)\) up to time step \(j\). The estimated state in the first phase of the Kalman filter will be denoted as \(\hat{x}_{i|i-1}\) and the state estimate after the measurement update phase will be denoted as \(\hat{x}_{ij}\). Similarly, \(P_{xx}^{ij}\) denotes the state covariance for the state at time step \(i\) using measurements up to time step \(j\).

The first phase of the Kalman filter can be summarized by the following equations that update the \textit{a priori} state and covariance estimates.

\[ \hat{x}_{k+1|k} = F_k \hat{x}_{k|k} + G_k u_k \]  \hspace{1cm} (2.23)

\[ P_{k+1|k} = F_k P_{k|k} F_k^T + Q_k \]  \hspace{1cm} (2.24)

The second phase of the Kalman filter is summarized in the equations below. The Kalman gain, \(K\), is calculated and used to update the \textit{a posteriori} state estimate based off the measurement, \(y\). The covariance is also updated.
\[ K_k = P_{k|k-1}^{xx} H_k^T (H_k P_{k|k-1}^{xx} H_k^T + R_k)^{-1} \]  \( (2.25) \)

\[ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - H_k \hat{x}_{k|k-1}) \]  \( (2.26) \)

\[ P_{k|k}^{xx} = (I - K_k H_k) P_{k|k-1}^{xx} (I - K_k H_k)^T + K_k R_k K_k^T \]  \( (2.27) \)

### 2.5.2 Unscented Kalman Filter

The Kalman filter requires a linear system with known dynamics, and Gaussian statistics of the noise involved in the system. Many extensions to the Kalman filter exist to deal with nonlinearities in the dynamics of real systems. The nonlinear state equations are:

\[ x_{k+1} = f(x_k, u_k, \omega_k) \]  \( (2.28) \)

\[ y_k = h(x_k, \nu_k) \]  \( (2.29) \)

\[ (2.30) \]

The unscented Kalman filter (UKF) does not involve explicit computation of the gradients as other approaches like the extended Kalman filter do. The UKF uses 2M sigma points to describe the statistics of the state gradients, where M is the length of the state vector. These sigma points are defined below, where \( P_{xx}^j \) is the jth row of the state covariance matrix for the state vector.

\[ \sigma_j = x + (\sqrt{MP_{xx}^j})^T \quad j = 1 \ldots M \]

\[ \sigma_j = x - (\sqrt{MP_{xx}^j})^T \quad j = M + 1 \ldots 2M \]  \( (2.31) \)
During the first phase of the Kalman filter, each sigma point is updated according to the dynamic model of the system. The estimate of the state and the covariance are determined by calculating the mean and the covariance in the statistics of the sigma points.

\[ \sigma_k = f(\sigma_{k-1}, u_{k-1}) \]  
\[ x_{k|k-1} = \frac{1}{2M} \sum \sigma^j_k \]  
\[ P_{xx}^{k|k-1} = \frac{1}{2M} \sum (\sigma^j_k - x_{k|k-1})(\sigma^j_k - x_{k|k-1})^T \]

The predicted measurement comes from the measurement model and is used with the real measurement to calculate the measurement covariance matrix and the state-measurement cross-covariance matrix.

\[ \hat{y}_k = h(\hat{x}_{k|k-1}) \]  
\[ P_{yy}^k = \frac{1}{2M} \sum [y^j_k - \hat{y}_k][y^j_k - \hat{y}_k]^T \]  
\[ P_{xy}^k = \frac{1}{2M} \sum (\sigma^j_k - x_{k|k-1})(y^j_k - \hat{y}_k)^T \]

The Kalman gain and the measurement update step in second phase are similar to the Kalman filter equations.

\[ K = P_{xy}^{-1} \]  
\[ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k[y_k - \hat{y}_k] \]  
\[ P_{k|k} = P_{k|k-1} + K_k P_{yy} K_k^T \]
2.6 Probability Theory

Source Likelihood Maps define the probability that the source is in a defined region of the map. The map is divided into discrete regions, with each region given a probability of containing the source. If the source is known to be in the map, then the sum of the probabilities of all regions will equal unity. Probability theory from [53] needed to derive the SLIM algorithm presented by Pang and Farrell [6] is reviewed below.

2.6.1 Basic Probability Theory

For any space $S$, there are multiple events that make up the space, $A_i$, $i = 1, 2, ..., N \in S$. Each of these events has some probability of occurring between zero likelihood and guaranteed likelihood. The sum of the probability of the events in the space must equal to unity.

\[
0 \leq f(A_i) \leq 1 \quad (2.41)
\]
\[
f(S) = 1 \quad (2.42)
\]

The probability of event $A$ occurring given that event $B$ has occurred is called the conditional probability and is denoted as $f(A|B)$.

\[
f(A|B) = \frac{f(A, B)}{f(B)} \quad (2.43)
\]

The probability of two events, $A$ and $B$ occurring is called the joint probability and is denoted as $f(A, B)$. If $A$ and $B$ are independent of each other then the following equations hold.
\[ f(A) = f(A|B) \]  
\[ f(A, B) = f(A)f(B) \] (2.44)  
\[ f(A, B) = f(A|B)f(B) \] (2.45)  

If the events are not independent then further techniques are needed to describe joint probabilities.

### 2.6.2 Chain Rule

The chain rule is used to show how joint probabilities can be transformed into conditional probabilities. If the formula for the conditional probability, Equation 2.43, is rearranged then the equation below is formed.

\[ f(A, B) = f(A|B)f(B) \] (2.46)

The above formula shows the transformation from a joint probability to a conditional probability. It can be extended to joint probabilities of any finite number of events.

\[ f(A_N, ..., A_1) = f(A_N|A_{N-1}, ..., A_1) \cdot f(A_{N-1}, ..., A_1) \] (2.47)

The chain rule can be used recursively to cast all joint probabilities into conditional probabilities. An example with four events is shown below.

\[ f(A_4, A_3, A_2, A_1) = f(A_4|A_3, A_2, A_1)f(A_3|A_2, A_1)f(A_2|A_1)f(A_1) \] (2.48)
2.6.3 Total Probability

The probability that an event \( A \) not occurring is called the complement of \( A \) and is denoted as \( \bar{A} \). The probability of an event and its complement is unity.

\[
f(B) + f(\bar{B}) = 1 \quad (2.49)
\]

Since the event \( B \) and its complement make the entirety of the space, then the entirety of \( A \) intersects with either \( B \) or its complement. Therefore, the probability of \( A \) can be expressed as the following.

\[
f(A) = f(A, B) + f(A, \bar{B}) \quad (2.50)
\]

Applying the chain rule on the right hand side of the above equation, the probability of \( A \) can be expressed as:

\[
f(A) = f(A|B)f(B) + f(A|\bar{B})f(\bar{B}) \quad (2.51)
\]

The above formula is the specific case of the Law of Total Probabilities when the space is divided into two regions, \( B \) and \( \bar{B} \). For the general case when the space is divided into \( N \) regions the law is stated as:

\[
f(A) = \sum_{k=1}^{N} f(A|B_k)f(B_k) \quad (2.52)
\]

2.6.4 Bayes Rule

Bayes rule is useful in computing ‘backwards’ conditional probabilities from evidence. To derive the expression from laws already discussed, consider the joint probability of two events, \( A \) and \( B \).
\[ f(A, B) = f(B, A) \tag{2.53} \]

The chain rule is applied to the joint probability distributions and then rearranged.

\[ f(A|B)f(B) = f(B|A)f(A) \tag{2.54} \]
\[ f(B|A) = \frac{f(A|B)f(B)}{f(A)} \tag{2.55} \]

Bayes law allows for a reverse conditional probability, given that the probability distributions of the two events independently are known. Often, the individual probability density function for the event \( A \) is unavailable. Using the law of total probability, Bayes’ Law is expressed in terms of conditional probabilities of \( A \) only.

\[ f(B|A) = \frac{f(A|B)f(B)}{\sum_{k=1}^{\infty} f(A|B_k)f(B_k)} \tag{2.56} \]
Chapter 3

Testing Environment

3.1 Martian Methane Plume Dispersion Model

A simulation of the fine structure of the methane plume was developed in order to test the algorithms for tracking Martian methane. Despite the desired characteristic of the rover control to be egocentric, the simulation of the plume can be bounded provided that the simulation area is large enough that the rover does not stray outside the boundaries. The Martian methane model is based on solving partial differential equations that describe the transfer of methane from a source over time. Validation of the model is important to ensure that the methods developed for source-localization are effective. Jones [54] adapted the statistical analysis that Chatwin [55] used to describe puffs of gas, and used them to describe atmospheric plumes. Jones’ experiments allowed him to quantify the properties of a ionized air plume by placing sensors at four discrete locations along the plume centerline. Farrell et al [56] use the same statistical scheme as Jones’ results to validate their filament-based model. This thesis will show the comparison of the new model’s results along side both of these previous works. Even though the plume dynamics are expected to slightly differ on Mars, the characteristics of the plume should be the similar as here on Earth. More importantly, the model should capture the characteristics of the plume that make
source-localization difficult.

The next section reviews the statistical methods and terms previous researchers have used to characterize both experimental data and odour models. The model is then presented in its entirety: the defining equations of state, the source reactions, the advection and turbulent diffusion models. Limitations placed on the model due to computation and the numerical solver are discussed. Finally, the results and statistical analysis are presented and compared to the previous works.

3.1.1 Previous Investigations and Statistical Methods

To date, many of the models of Martian methane plumes have concentrated on general circulation models and plumes on planetary scales (see Mischna et al [10]). A rover mission to Mars would not be covering the entire planet, but only a small region. *Curiosity*, which landed on Mars in August 2012, is expected to travel only 19 kilometers over its two year mission [57]. Furthermore, models of methane plume distributions estimate that measurements that are not directly in the plume path would need to be made within 100 meters of the plume axis to be discernible from background levels of methane [58]. A model of local methane plumes, as opposed to global scales, is needed to achieve the goal of creating a simulation environment for testing algorithms for a Martian Methane mission. In these local methane plumes, the general planetary circulation that is used as the forcing (advection) function is no longer applicable. A turbulent environment is more representative of the situation that a robotic rover would encounter in while seeking the methane source.

Many investigations on the structure of turbulent plumes exist. Well known characteristics of odour plumes include the tendency to meander with instances where local concentrations are much higher than the time mean [42]. Turbulence can be defined by its nature to create a rotational, diffusive and stochastic effect, where a
single particle would be observed to wander throughout an area of influence whose descriptive length is constantly increasing downwind [59]. Turbulence causes patches of ‘clean air’ in the plume where concentrations are indistinguishable from background levels. Intermittency is defined as the percentage of time where concentrations are below a defined threshold. Intermittency can be caused by both the meandering and the patchiness of the plume. In tests of several minutes duration, care must be taken to ensure that high values are descriptive of the plume and not the local wind pattern during that time [42].

In order to quantify the characteristics of the odour plume just mentioned, Jones [54] adapted the statistical methods used by Chatwin [55] to describe puffs of gas for plumes. Jones used these statistics to describe the results of his experiments with turbulent plumes. Farrel et al [56] used the same statistical methods to characterize their filament-based model of an odour plume and to compare their model to the experimental results of Jones. The model presented in this thesis will be subjected to the same statistical methods as these two authors to allow for a comparison between the presented model to both past experimental and simulated results.

**Long Term Statistics**

Time-averaged plumes are generally modeled as a Gaussian plume, of which there are many variations. Of the more popular models is one proposed by Sutton [60]. Since plume dynamics and the movement of a Mars rover are on short time scales, the Gaussian model does not capture enough detail for use as a simulation environment. However, the realism of the presented model can be shown if the long-term average of the model closely replicates the Sutton model. The Sutton model and contour map is shown in Equation 3.1 and Figure 3.1.
In equation 3.1, \( Q \) is the source strength in \([\text{kg/s}]\), \( u \) is the wind speed in \([\text{m/s}]\), \( S_y \) and \( S_z \) are the dispersion in the \( y \) and \( z \) direction respectively \([\text{m}^2/\text{s}]\). The common expressions for \( S_y \) and \( S_z \) are Equation 3.2, where \( C_y \) and \( C_z \) are constants with appropriate units and \( n \) is a unitless parameter that governs the shape of the plume. Often, \( n = 1 \) is the best match to nature.

\[
S_y = 0.5C_yx^{(2-n)/2} \\
S_z = 0.5C_zx^{(2-n)/2}
\] (3.2)

The model presented in this thesis only takes place in two dimensions. Thus, the model is solved for the 2D form as presented in Equation 3.3. The derivation of Equation 3.3 can be found in the Appendix.

\[
C(x, y) = \frac{Q}{\sqrt{2\pi S_y u}} \exp \left( -\frac{y^2}{2S_y^2} \right)
\] (3.3)

The concentration at a position \( \mathbf{p} \) and time \( t \) is denoted as \( C(\mathbf{p}, t) \). To compare the average concentration of the plume to the Sutton model, the initial period of time where odour is populating the simulation area is ignored. The expression for the mean concentration field is shown below with the time where the odour is well mixed denoted as \( t_m \) and the final time of the simulation, \( t_f \). Contour maps of the results will be compared to Figure 3.1.

\[
\bar{C}(\mathbf{p}) = \frac{1}{t_f - t_m} \int_{t_m}^{t_f} C(\mathbf{p}, \tau) \partial \tau
\] (3.4)
Figure 3.1: Sutton Model: A contour map of the Sutton model used for comparison to the model presented.

Amplitude Statistics

Jones’ experiment consisted of placing four sensors at discrete locations inside the atmospheric plume located in the nominal wind direction: 2, 5, 10 and 15 meters downwind from the source. Data was collected from these four locations throughout the entire experiment. The concentration information used for the statistical analysis are denoted as $\Gamma(p, t)$. Since there are many stochastic processes represented in the model multiple runs of the simulation are compiled together. Where distinction between multiple experiments is needed, the letter ‘r’ will represent the experiment as in $\Gamma^r(p, t)$. The mean concentration at a particular location is denoted as $\bar{\Gamma}(p)$.

$$\bar{\Gamma}(p) = \frac{1}{t_f - t_m} \int_{t_m}^{t_f} \Gamma(p, \tau) d\tau$$

(3.5)

Jones noted that the average concentration of the plume matched closely with the statistics when the integral was only calculated during the times when the sensor was in the plume (i.e. the concentration was above a defined threshold, $\gamma$).
Distance Downwind | 2 m | 5 m | 10 m | 15 m  
--- | --- | --- | --- | ---  
Jones \( \tilde{\Gamma} \) [nC m\(^{-3}\)] | 4.21 | 0.525 | 0.239 | 0.159  
| \( \hat{\Gamma} \) [nC m\(^{-3}\)] | 28.4 | 5.33 | 1.46 | 0.549  
Farrell \( \tilde{\Gamma} \) [molecules cm\(^{-3}\)] | 698.01 | 232.22 | 69.38 | 40.48  
| \( \hat{\Gamma} \) [molecules cm\(^{-3}\)] | 1494.0 | 655.4 | 237.3 | 136.9  

Table 3.1: General and Conditional Mean Data: Data collected by Jones during his experiments on an ionized-air plume and from Farrell et al on their filament-based plume model. In both cases the conditional mean is greater than the general mean due to the removal of zero-concentration events from the conditional mean. Also, in all four cases the mean decreases as the sensor’s position is further away from the plume.

The conditional case is denoted as \( \hat{\Gamma} \). Table 3.1 summarized the mean data from both Jones and from Farrell et al. As expected, the conditional means were greater than the general means for each location, which is accounted for the removal of zero-concentration time.

\[
\hat{\Gamma}(p) = \frac{1}{t_f-t_m} \int_{t_m}^{t_f} \Gamma(p, \tau) \partial \tau \quad \text{only integrated when } \Gamma(p) > \gamma \quad (3.6)
\]

Five additional parameters are used to describe the plume. The \( n^{th} \) central moment, denoted as \( M_n(p) \) (see Equation 3.7) for \( n = 2,3,4 \) respectively are the following: standard deviation (\( \sigma_\Gamma \)); skewness (\( S_\Gamma \)), which represents whether the distribution is weighted less than (negative) or greater than (positive) the mean; and kurtosis (\( K_\Gamma \)), which describes whether the distribution is uniform (low value) or peaked (high value). The moments are normalized so they can be compared across multiple experiments.

The peak-to-mean ratio (\( \hat{\Gamma}/\tilde{\Gamma} \)) is another parameter, where \( \hat{\Gamma}(p) = \max(\Gamma(p)) \). The last metric used is intermittency which is denoted as \( I \), and is the percentage of time where \( \Gamma(p,t) \leq \gamma \), where \( \gamma \) is the same threshold as defined previously.
Table 3.2: Amplitude Data: Unfiltered data collected by Jones during his experiments on an ionized-air plume and from Farrell et al for their filament-based plume model. The five parameters listed here are the standard deviation, the skewness, the kurtosis (the 2nd, 3rd and 4th moments respectively), the peak-to-mean ratio and the intermittency. Notice that the trends for each parameter with distance are similar between the model and the experiment. However, the model of Farrell does not match well the intermittency seen in Jones’ experiment. Creating the intermittency from the experiment as shown above (85% throughout) was a challenge faced during the design of the model presented in this thesis as well. It is worth noting that Farrell did not publish results for the 15 m position, probably because in Jones’ experiment, that sensor did not follow the trend set by the three closer sensors.

\[
M_n(p) = \frac{1}{(t_f-t_m) f_{t_m}^{t_f} \left[\tilde{\Gamma}(p) - \Gamma(p,\tau)\right]^n \partial\tau}^{1/n} \tilde{\Gamma}(p)
\]

(3.7)

Jones’ analysis of his experiment consisted of running the data collected from each sensor through a low pass filter to examine the effects on the five parameters mentioned above. In addition, he computed the results without filtering. Table 3.2 shows the results from Jones for all the positions and Farrell for three positions (Farrell only published results for the three positions shown) for the unfiltered case. The results for the filtered cases are not shown here but can be found in [54] and [56] for Jones and Farrell, respectively.
Temporal Statistics

When insects use odour plumes to seek out mates, generally they spend a lot of time flying inside the plume but where there is no detectable amount of odour. Indeed, many plumes are characterized as a group of puffs with a lot clean air in between. A useful metric for describing this structure is the distribution of the lengths of time where the sensor is measuring above a set threshold, and the distribution of time it takes in between these measurements. Equation 3.8 shows how the experiment can be broken down into discrete quantities of time inside a puff and time inside a gap between puffs. Timing begins only after the first detection event above threshold, hence the approximation in the equation. Jones presents two sets of temporal statistics with thresholds one order of magnitude difference. Although Jones’ results are not included within this thesis, qualitative comparisons between the results will be made in section 3.1.3.

\[
\sum_{i}^{N_p} t_{p,i} + \sum_{j}^{N_g} t_{g,j} \approx T
\]

(3.8)

3.1.2 Model Overview

This section will detail the model developed and the motivation for the decisions made in relation to the model. The concentration at a point in space and time will vary due to reactions that either release or destroy the substance, advection that carries the plume as a whole downwind, and molecular and turbulent diffusion that is responsible for changing the shape of the individual puffs of which the plume is made. The classic diffusion-advection-reaction (DAR) equation models these dynamics of the plume over time. Reaction refers to the locations where the chemical is created or destroyed. Wind which transport the chemical is dependent on the gradient of the chemical over the space. The wind will vary with both position and time and will lead to
some turbulent diffusion. Molecular diffusion is the natural ‘evening’ of concentration due to collisions at the microscopic levels. This process is dependent on the second derivative of the concentration over position. A concentration distribution over time with no reactive or advection would result in a uniform distribution as time reaches infinity. The DAR equation is shown as Equation 3.9, where $C$ is the concentration of methane and is a function of position and time ($C(p, t)$), $D_x$ and $D_y$ are the constant molecular diffusion constants in the x and y direction respectively, $u_x$ and $u_y$ are the instantaneous wind speeds in the x and y direction respectively ($u_x(p, t)$, $u_y(p, t)$). $Q$ is the rate of methane produced or destroyed at that point in space due to whichever reaction process (environmental or biological) occurs, ($Q(p, t)$).

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial x} \left( D_x \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left( D_y \frac{\partial C}{\partial y} \right) - u_x \frac{\partial C}{\partial x} - u_y \frac{\partial C}{\partial y} + Q \quad (3.9)$$

The partial differential equation is numerically integrated over time and a mesh to solve for the concentration at any given location in the simulation area. Boundary conditions on the solver are such that the concentration one-unit distance of the mesh inside the control boundary is equal to the concentration at the boundary. The boundary condition allows for the calculations of the necessary gradients. The wind vector $u$ contains all the wind contributions, and is made up of three components; $u = a + v + t$. These three components are the advection ($a$), which is a large scale process that is responsible for moving the entire plume; a medium scale function ($v$) that is responsible for mixing the individual puffs of methane, and a small scale turbulence ($t$) that is responsible for the individual motion of the methane puffs.

A number of simplifications have been made while solving the DAR equation which correspond to the Martian methane scenario.

- As a Martian rover is a surface mobility platform with no ability to change its vertical position, the model is in two dimensions only. As of consequence, the
simulation area is considered flat ground and no perturbations in the wind due to uneven ground or obstacles (including the vehicle) are taken into account.

• As discussed earlier, one of the anomalies in the methane measurements on Mars is the quick time scale at which methane is destroyed. Although there are many proposals for processes that could explain methane destruction on quicker time scales, the true nature of the destruction remains a mystery. In addition, the time scales of the simulation will be much smaller than even these relatively ‘quick’ methane lifetimes. Therefore, no destruction process is modeled. Instead, methane is carried out of the simulation environment due to the wind, but no methane is brought into the simulation area due to potential upwind sources. The imbalance acts as the sink and stabilizes the solver.

• The wind model to be presented does not necessarily follow conservation laws. It is not meant to replicate real wind patterns on Mars; however, it is meant to produce methane distributions whose statistics match those which would be expected for those situations.

Source Reaction

The methane source location and radius are defined in the simulation parameters. Discrete puffs of methane were produced at a given frequency. Both the time between puffs, and the strength of the puffs were perturbed by random noise. The strength of each puff and the time between puffs are perturbed by ±10% of the nominal strength. The duration of each puff is kept constant.

Wind Model

The wind model is made up of three parts: advection, vortex, and turbulence. Each of these represents a transportation scale; advection affects the plume as a whole,
vortex affects the puffs as individuals, and turbulence affects the particles of methane. The advection force varies its magnitude and direction both in time and position that causes the plume to form a sinuous shape. Vortexes are local, circular wind patterns that change the shape of the puffs and move them cross-wind simultaneously. Turbulence is discrete, stochastic noise added to the system.

The advection function is randomly formed by creating a three-dimensional surface called the advection potential, $Z$. Wind strength and direction following lines of constant altitude on the surface. Equation 3.10 shows how the two components of the advection vector, $a_x$ and $a_y$, are determined from $Z$.

$$a_x(x, y) = -\frac{\partial Z(x, y)}{\partial y} \quad a_y(x, y) = \frac{\partial Z(x, y)}{\partial x} \quad (3.10)$$

The advection potential consists of the superposition of a linear plane that represents the nominal wind and a ‘noisy’ surface that adds texture. The noisy surface is computed using the Fourier series seen in Equation 3.11. The parameters of the function ($A_n, B_n, C_n, D_n$) are varied by filtering white Gaussian noise through $H(s) = \frac{c}{s^2 + as + b}$. The term $L_x$ and $L_y$ describe the total length of the simulation area in the x and y direction respectively.

$$Z_f(x, y) = \sum_{n=1}^{N} \left( A_n \sin \frac{n\pi x}{L_x} + B_n \cos \frac{n\pi x}{L_x} + C_n \sin \frac{n\pi y}{L_y} + D_n \cos \frac{n\pi y}{L_y} \right) \quad (3.11)$$

Medium scale mixing of methane puffs is modeled by adding ‘vortex’ to the model. Vortexes are created by adding a Gaussian pulse to the advection potential $Z$ centered at the position where the vortex is located. The amplitude, diameter and position can be specified according to Equation 3.12. The vortex travels along the simulation area according to the advection, that is $\dot{p}_v = a$. 
\[ Z_v(x, y) = A \left( \frac{\exp \left( \frac{-8(x-x_0)}{d} \right)}{1 + \exp \left( \frac{-8(x-x_0)}{d} \right)} \right) \left( \frac{\exp \left( \frac{-8(y-y_0)}{d} \right)}{1 + \exp \left( \frac{-8(y-y_0)}{d} \right)} \right) \] (3.12)

Turbulence is added to the base wind in the form of a stochastic change in magnitude of the \( x \) and \( y \) wind directions. In order to prevent the model from diverging, feedback is used to drive the wind model towards the base wind. It is seen below that the second derivative of the turbulence is calculated by multiplying a gain by a matrix of random numbers on a normal distribution \((R)\), and then is corrected by the feedback of the previous full wind model and the current base wind model.

\[
\ddot{U}^{k+1}_{x,turb} = K_1 R^{k+1}_x + K_2 (U^k_x - U^{k+1}_{x,base}) \quad (3.13)
\]

\[
\ddot{U}^{k+1}_{y,turb} = K_1 R^{k+1}_y + K_2 (U^k_y - U^{k+1}_{y,base}) \quad (3.14)
\]

**Model Solver Limitations**

In the explicit form, the difference equation has a stability issue when the time step used is greater than a threshold. The threshold is a function of the coefficients of the equation. This section details the analysis of the stability of the difference equation and the determination of a time step that both satisfies stability while also minimizing computation during simulation runs. Stability of difference equation requires that the coefficients of each \( C^h_m \) term in the difference equation must be non-negative.

Rearranging the difference equation to get:
\[
\frac{\Delta x^2 \Delta y^2}{\Delta t} (C_{i,j}^{k+1}) = \left[ \frac{\Delta x^2 \Delta y^2}{\Delta t} - 2\Delta y^2 D_x - 2\Delta x^2 D_y - \Delta x \Delta y^2 u_x - \Delta x^2 \Delta y u_y \right] C_{i,j}^{k} \\
+ \Delta y^2 D_x C_{i+1,j}^{k} + \Delta x^2 D_y C_{i,j+1}^{k} + [\Delta y^2 D_x + \Delta x^2 \Delta y u_x] C_{i,j+1}^{k} \\
+ [\Delta x^2 D_y + \Delta y^2 \Delta x u_y] C_{i-1,j}^{k} + \Delta x^2 \Delta y^2 Q 
\] (3.15)

The value of the time step is determined by the \( C_{i,j}^{k} \) term. The coefficients of \( C_{i,j}^{k} \) are set to \( \geq 0 \).

\[
\left[ \frac{\Delta x^2 \Delta y^2}{\Delta t} - 2\Delta y^2 D_x - 2\Delta x^2 D_y - \Delta x \Delta y^2 u_x - 2\Delta x^2 \Delta y u_y \right] \geq 0 
\] (3.16)

\( \Delta x \) and \( \Delta y \) values are assumed to be equal (\( \Delta \)) since this will make processing the simulation results easier. In addition, the physical diffusion constants will be omni-directional in the 2D space so \( D_y, D_x \rightarrow D \). The total nominal wind vector is both \( u_x \) and \( u_y \), so it compensates for any reference frame coordinates of the vector, plus it helps to compensate for any turbulence or vortex contributions.

\[
\Delta t \leq \left[ \frac{\Delta^4}{4\Delta^2 D + 2\Delta^3 u_{\text{max}}} \right] 
\] (3.17)

The time step is chosen to be small enough as to satisfy Equation 3.17 above.

**Sensor Model**

The equations used to model the sensor and low-pass filter are given as Equation 3.18. Here, the internal state of the filter is \( f \) and the output of the sensor is \( y \). The bandwidth of the filter is \( \alpha \) and the threshold \( \gamma \) is the same as defined previously.
\[ \dot{f}(t) = -\alpha f(t) + \alpha \Gamma(t) \]
\[ y(t) = \begin{cases} f(t) & \text{if } f(t) \geq \gamma \\ 0 & \text{otherwise} \end{cases} \quad (3.18) \]

### 3.1.3 Model Analysis

Multiple simulations have been reviewed to ensure that the simulation performs similarly in different initial conditions and with different random number seeds. This section will discuss the results of one of these experiments that is representative of the entire group. At each instant during the simulation the plume is characterized as sinuous and patchy. The general mean concentration map of the entire simulation however, closely resembles a Gaussian plume model. Figure 3.2 compares the concentration maps between the instantaneous and the time-averaged cases. Figure 3.2a shows isopleth from the time averaged map and a Gaussian plume. The Sutton plume as defined in section 3.1.1 is used as the Gaussian plume for comparison. Values used are \( Q = 1, u = 1 \) and \( C_y = 0.025 \) respectively. It is seen that the Sutton model well-approximates the time averaged plume in the simulation.

Table 3.3 presents the general mean and the mean conditional on being inside the plume, comparable to the same data given by Jones and Farrell and summarized in Table 3.1. As would be expected the conditional means are greater than the general mean at each location. The means from the two previous results and the presented model are also plotted in Figure 3.3. The data is normalized to allow comparison between the two models and the experiment based on the results at the 2 m position. It is seen that the current model follows the downward trend of the other two plots, but does not match the experimental data as well as the model of Farrell et al,
Figure 3.2: Long Term Average Comparison: Results from the simulation with the plume source at (10,0). The results presented in (3.2a) are the mean over a ten minute period compared to Sutton’s Gaussian model. In (3.2b) a snapshot of the instantaneous plume is shown. Blue color represents no concentration where red denotes the highest concentration. It can be seen that the instantaneous plume contains several ‘puffs’ of concentration at high levels with large amounts of ‘clean’ space in between.

<table>
<thead>
<tr>
<th>Position</th>
<th>2 m</th>
<th>5 m</th>
<th>10 m</th>
<th>15 m</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bar{\Gamma}$ [kg/m$^3$]</td>
<td>5.776</td>
<td>2.729</td>
<td>1.801</td>
<td>1.487</td>
</tr>
<tr>
<td>$\tilde{\Gamma}$ [kg/m$^3$]</td>
<td>6.438</td>
<td>4.582</td>
<td>3.121</td>
<td>2.407</td>
</tr>
</tbody>
</table>

Table 3.3: General and Conditional Mean Data: Presented is the (general) mean of the simulated measurements of the model at four positions and the conditional ‘in the plume’ mean.

especially for the conditional mean. The discrepancy is a result of the fact that in both Jones’ experiment and Farrell’s model the intermittency for all positions were fairly similar, whereas in the current model the intermittency at 2 m is significantly lower than the other positions. Since this trend greatly affects both means, it is the likely source of the discrepancy.

The amplitude statistics are presented in Table 3.4 for comparison to the unfiltered values summarized in Table 3.2 in addition to the filtered values published in [54] and [56]. For consistency with the publications of Jones and Farrel et al, the results are
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>30 Hz</th>
<th>10 Hz</th>
<th>3 Hz</th>
<th>1 Hz</th>
<th>0.3 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Deviation</td>
<td>$\sigma_G$</td>
<td>0.551</td>
<td>0.551</td>
<td>0.549</td>
<td>0.544</td>
<td>0.496</td>
</tr>
<tr>
<td>Skewness</td>
<td>$S_G$</td>
<td>0.195</td>
<td>0.195</td>
<td>0.194</td>
<td>0.182</td>
<td>0.154</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>$K_G$</td>
<td>0.708</td>
<td>0.708</td>
<td>0.706</td>
<td>0.691</td>
<td>0.617</td>
</tr>
<tr>
<td>Peak-To-Mean Ratio</td>
<td>$\hat{\Gamma}/\bar{\Gamma}$</td>
<td>3.10</td>
<td>3.10</td>
<td>3.07</td>
<td>2.97</td>
<td>2.70</td>
</tr>
<tr>
<td>Intermittency</td>
<td>I(%)</td>
<td>10.5</td>
<td>10.5</td>
<td>10.2</td>
<td>7.5</td>
<td>1.2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>2 m Downwind</th>
<th>5 m Downwind</th>
<th>10 m Downwind</th>
<th>15 m Downwind</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Deviation</td>
<td>$\sigma_G$</td>
<td>0.655</td>
<td>0.655</td>
<td>0.655</td>
<td>0.655</td>
</tr>
<tr>
<td>Skewness</td>
<td>$S_G$</td>
<td>0.269</td>
<td>0.269</td>
<td>0.268</td>
<td>0.284</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>$K_G$</td>
<td>0.823</td>
<td>0.823</td>
<td>0.822</td>
<td>0.847</td>
</tr>
<tr>
<td>Peak-To-Mean Ratio</td>
<td>$\hat{\Gamma}/\bar{\Gamma}$</td>
<td>3.11</td>
<td>3.11</td>
<td>3.09</td>
<td>2.99</td>
</tr>
<tr>
<td>Intermittency</td>
<td>I(%)</td>
<td>41.2</td>
<td>41.2</td>
<td>40.7</td>
<td>35.2</td>
</tr>
</tbody>
</table>

Table 3.4: Statistical Data for Modeled Plume: The five statistical parameters; standard deviation, skewness, kurtosis, peak-to-mean ratio, and intermittency, are given for four sets of data describing four locations along the nominal wind position for a ten minute long simulation. The first three parameters are normalized with the conditional mean at that location, with thresholds 5% of the maximum levels seen at the 2 m position.
Figure 3.3: Comparison of Means: The figures above show the mean concentration over four positions for Jones’ experiment, Farrell et al’s simulation and the model presented in this thesis. All data sets are normalized by that set’s concentration at 2 m.

presented for the four positions and six filtering conditions. Trends for many of the parameters fit the trends of Jones’ experiment. These trends include a decrease of all parameters as the filter bandwidth decreases and increase of the skewness, kurtosis, and peak-to-mean ratio as the distance from the source increases.

The most obvious trend that the model did not replicate is the consistent levels of intermittency throughout the experiment. The two-meter downwind location had a much lower percentage of intermittency than the other three positions. In Jones’ experiment, levels of intermittency did not follow an exact trend and were fairly consistent. Intermittency increased 20% for the three data points given by Farrell et al, a range that was twice as wide compared to Jones’ data for the same distance covered. Once the plume reaches the five meter position, it is well mixed and the intermittency stays very consistent (a range of 3% over 10 meters). Even so, the intermittency was still much lower than that of both the experimental data and the model of Farrell et al. The next deviation from the experimental data is the tendency for the standard deviation to increase as distance from the source increases; the
opposite trend is seen in the experiment.

The final difference in results is the odd results Jones gets for the 15-meter downwind position. In his experiment, the skewness, kurtosis, and peak-to-mean ratio all tend to increase with distance downwind except for the final position (15-meters) in where it decreases. The standard deviation is the only parameter that doesn’t exhibit this reversal. Farrell et al do not acknowledge this strange behaviour in their analysis of their model, and instead do not publish their results or make comparison at the 15-meter downwind position; their results match the trends up to that point. The results of the current model show that they do not reverse trends at the 15 m point for any of the parameters. It is assumed that these are the same results as Farrell et al’s model, and although not preferable based on the experimental data, acceptable as a validation.

The short-scale analysis is summarized in the histograms of Figure 3.4 for two positions and two threshold levels. Jones published histograms for the 2-meter and 15-meter downwind positions; due to the low intermittency of the 2-meter position, the results were taken for the 5-meter position instead. In keeping consistent with Jones, the higher threshold is one order of magnitude higher than the lower threshold. In all cases, the puff lengths were generally much smaller than the lengths of time between puffs. The puff and gap statistics are the main characteristic of the results shown by Jones. In addition, the puff lengths at the higher threshold are much shorter than for the lower threshold, and the gap lengths exhibit the reverse trend. The puff return statistics for the 5-meter position do not exhibit the same separation between the two thresholds that the 15-meter position and the results of Jones have. However, the lengths at the high threshold extend into the higher range of durations that would be expected.
(a) Puff length statistics at 5 m downwind
(b) Puff length statistics at 15 m downwind
(c) Puff return time statistics at 5 m downwind
(d) Puff return time statistics at 15 m downwind

**Figure 3.4: Temporal Statistics:** The charts above give the distribution of times for the length of puffs (continuous concentrations above threshold) and the time for a puff to return (continuous time below threshold) at 5 m and 15 m downwind for two different thresholds. The red bars represent the lower threshold (0.5 kg/m³) and blue bars represent the higher threshold (5 kg/m³). As with Jones, the higher threshold is one order of magnitude higher than the lower threshold.
3.1.4 Conclusion

Presented is a model for simulating methane plumes on Mars. The model can be used to create scenarios for testing robotic algorithm’s effectiveness and efficiency at finding the source of the plume. Although the analysis presented here was done for a single source, multiple sources can be incorporated into the simulation with ease. Analysis of the presented model shows that it is suitable for simulating Martian methane plumes as its characteristics represent the challenges of the source localization problem. Although there are some discrepancies between the amplitude statistics for the current model and experimental data, the short-scale data shows that the plume exhibits the characteristics that make plume-tracing difficult, that is, the short time spent sensing concentration within large gap times. Furthermore, since the nature of the fine scale Martian methane plumes is currently unknown (and will remain so until a mission concept like the one presented earlier is deployed), this scheme represents the best environment for algorithm development. Therefore, there is no gain in continuing to tune the presented model at the current time.

3.2 Enceladus Plume Model

_Cassini_ measurements with the Ion and Neutral Mass Spectrometer (INMS) indicate a plume whose composition is mostly water (91±3%) with carbon dioxide (3.2±0.6%), nitrogen or carbon monoxide (4±1%) and methane (1.6±0.4%) [61]. Quantities less than 0.5% of ammonia were measured. The measurements were unable to determine which compound, nitrogen or carbon monoxide, was measured at the 28 dalton point (having a weight of 28 atomic units). Waite et al. [61] estimated the ejection rate of the plume to be 90 kg/s based on INMS reading. Measurement of the plume during a stellar occultation with the Ultraviolet Imaging Spectrograph (UVIS) lead Hansen et al. [62] to estimate the mass loading at a lower limit of 150 kg/s, that was confirmed
through modeling by Tian et al. [29]. Current estimates of the ejection rate of the plume are between 500 and 750 kg/s, varying as Enceladus orbits around Saturn.

Spencer et al. [20] were the first to report a thermal anomaly associated with Enceladus’ tiger stripes, measured with the Cassini Infrared Spectrometer (CIRS). Temperatures measured were much higher than for the rest of the moon, and above expected levels. Seven hot spots were recorded with the maximum temperature at 157 ± 24 K. The estimated total heat radiated from the south polar terrain was estimated to be 5.8±1.9 GW. Spencer et al. [20] estimated that the minimum distance from the surface to a liquid reservoir would be 40 meters if heat transfer from the interior to exterior was pure conduction. It is more likely that heat is transferred through convection from gas rising through the jets, increasing the potential depth of the liquid reservoir. Of the eight sources located by Spitale and Porco [19], four match identically to thermal hot spots, with two other sources being in the near vicinity of other hot spots.

The eight main locations where jets of material were observed throughout the close-range flybys of Cassini provide an initial target for a penetrator mission whose purpose is to penetrate into the subsurface ocean. High resolution photos of these areas during subsequent flybys revealed that these plume sources are actual characterized by numerous jets on the order of tens of meters in diameter spread over square kilometers. The diameter of the jet defines the targeting accuracy of the mission. In order to get the penetrator to have a targeting ability on the order of meters, the first assumption that needs to be made is that the target location is known. Constantly updating the estimate of the target location is necessary for the success of the mission.

Postberg et al. [63] has done extensive work on the composition of the plume and E-rings. One of the observations that they made was on the average size of particles that make up the plume. They noted that not only would it be expected that larger particles be closest to the moon, but also larger particles get further into the plume
Figure 3.5: Particle Size in Plume: Results published by Postberg et al. [63] showing the average particle size through the plume. Jets III VI and VII are seen as an artifact of the observations.

in the ‘jet stream’ of the plume’s sources. Figure 3.5 shows the E5 flyby plane and the artifacts of jets III, IV and VII can be seen through the average particle size. Information about the average particle size may complement well the concentration levels in our navigation algorithms. Modeling of the average particle size is not completed in this work, but may be a useful addition in the future.

A penetrator mission, where the vehicle was released from a main spacecraft during flyby, would take about 12 minutes to impact from time of release (See Appendix B). The time scale of the descent is much shorter than the orbital period of the moon (1.3 days). During the high speed descent, different cues as to the target location exist. An initial selected target of one of the eight source locations would provide course targeting to the geysers. Visual techniques could orient the penetrator with respect to the distinct tiger stripes. Visual orientation systems such as this have been developed for precision moon landers, and orients the spacecraft with respect to craters [64]. Once the penetrator falls closer to the planet, the plume will start
to decompose into its individual jets. Being able to target one of these jets for the final leg of the mission might allow the penetrator to impact at a location where deeper penetration will occur. Ideally, although probably not realistic, the penetrator impacts through the jet and comes into contact with the subsurface ocean.

In order for the model of the plume to be valuable, the focus is on the following characteristic:

- The model is representative of the plume structure on the small scale as well as the large scale. Where most models only concern themselves with representing the plume on the large scale (far away from the moon where the jets have diffused into a single plume) the concern of this mission is penetrating into one individual jet and therefore requires a small scale model. Therefore the model must consist of the multiple jets instead of single sources or only the eight sources noted by Spitale and Porco [19].

- There are numerous proposed jets throughout the tiger stripes, although theories are that the jets are more numerous and cluster in the eight regions seen by Spitale and Porco [19]. Because the exact structure of the plume is still unknown, a stochastic component of the model is included to provide for several scenarios. This means that multiple scenarios can be created with different combinations of location and strengths for the jets. Each scenario should result in realistic distribution and should match previous results.

- The plumes have been observed to deviate between individual Cassini measurements, and some correlation with the tidal stresses have been observed by Nimmo et al. [22]. Infrequent measurements of the plume at the same orbital position of Enceladus have limited the ability to confirm correlation. The distribution of jets over the tiger stripes and their relative strengths must be impacted by the desired true anomaly of the moon during the mission.
• The proposed penetrator mission has a time scale that is much shorter than the cycle length of the deviations of jet activity. A dynamic plume model would change insignificantly over the time frame of the mission. Therefore, a static plume model is sufficient.

3.2.1 Model Description

The plume of Enceladus is formed by the diffusion of multiple geyser jets erupting on the tiger stripes in the South Polar Terrain (SPT) of the moon. Eight sources of high ejection rates were located by Spitale and Porco [19]. Closer flybys provided higher resolution images of the SPT and showed that many more jets existed along all four tiger stripes. Many of the models in the literature disregard these minor sources and only use the eight well observed sources. These models have proved to be representative of the overall plume at altitudes where Cassini is conducting its flybys. However, for a descent mission that relies heavily on the fine structure of the plume, the model must contain multiple jets. The plume concentration is the sum of the effects of all jets.

For each flyby of Cassini, the plume has been observed to be in a different form. Attempting to create models that replicate the measurements of Cassini result in different emission rates for the sources for each pass, and imagery has failed to show consistent eruptions for each jet during every flyby. The activations of particular jets were shown to correlate with the expected tidal stresses of the region where the source is located. When the source is in tension the jets seem to be activated, but when the sources are in compression they tend to be stagnant.

As one of the goals of the model is to be able to produce a high number of different scenarios, the jets are randomly distributed along the tiger stripes defined by a probability distribution function (PDF). The PDF is created by defining the percentage of jets that are to be located at the eight well-observed sources; the remaining
are normally distributed about the tiger stripes. Each jet is given the same strength potential (the maximum ejection rate) and therefore the percentage of jets at the well-observed source locations is proportional to the percentage of ejection material coming from these same sources. Further variability is added by alternating the strengths of the jets as a function of the normal stresses acting on the tiger stripe at the position of the jet. Although the exact relationship between ejection rate and stresses is not solved, two proposals by Hurford et al. [65] were considered: ejection rates proportional to the normal stress when positive and the maximum ejection rate occurring at the instant the jet switches from compression to tension and then dying off. In both cases, a compression stress results in zero ejection from the jet.

The tidal stress model used to describe the stresses acting on an orbiting body due to the non-circular orbit of Enceladus is adopted from Nimmo et al. [22]. The model shown is a simplified model but matches well with the more developed SatStress created by Wahr et al. [66]. These tidal stresses as defined by Equations 3.19 - 3.21 are a function of the position on the moon and orbital position of the moon, represented by the colatitude ($\theta$), the longitude ($\phi$) and the mean anomaly ($m$). The colatitude is positive in the south direction (0 - $\pi$) and the longitude is positive in the west direction (0 - 2$\pi$). A list of the other parameters and the values used can be seen in Table 3.5.

\[
\sigma_{\theta\theta} = c_1 e(-\cos m[3l_2 \sin^2 \theta + c_2 \delta(3 \cos^2 \theta - 1)]) \\
+ .5[l_2(1 + \cos^2 \theta) + c_2 \delta \sin^2 \theta][-\cos (2\phi + m) + 7 \cos (2\phi - m)]) \quad (3.19)
\]

\[
\sigma_{\phi\phi} = c_1 e(-\cos m[-3l_2 \sin^2 \theta + c_2 \delta(3 \cos^2 \theta - 1)]) \\
+ .5[l_2(1 + \cos^2 \theta) + c_2 \delta \sin^2 \theta][-\cos (2\phi + m) + 7 \cos (2\phi - m)]) \quad (3.20)
\]

\[
\sigma_{\theta\phi} = c_1 e l_2 \cos \theta[\sin (2\phi + m) - 7 \sin (2\phi - m)] \quad (3.21)
\]
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Mass</td>
<td>M</td>
<td>$5.7 \times 10^{26}$ kg</td>
</tr>
<tr>
<td>Gravitational Constant</td>
<td>G</td>
<td>$6.67 \times 10^{-11}$ m$^3$/kg s$^{-2}$</td>
</tr>
<tr>
<td>Moon Surface Gravity</td>
<td>g</td>
<td>0.11 m/s$^2$</td>
</tr>
<tr>
<td>Moon Radius</td>
<td>R</td>
<td>252.1 km</td>
</tr>
<tr>
<td>Moon Semi-Major Axis</td>
<td>a</td>
<td>238000 km</td>
</tr>
<tr>
<td>Eccentricity</td>
<td>e</td>
<td>0.0047</td>
</tr>
<tr>
<td>Love Number</td>
<td>$h_2$</td>
<td>0.2</td>
</tr>
<tr>
<td>Love Number</td>
<td>$l_2$</td>
<td>0.04</td>
</tr>
<tr>
<td>Lamé Parameter</td>
<td>$\lambda$</td>
<td>6.8 MPa</td>
</tr>
<tr>
<td>Lamé Parameter</td>
<td>$\mu$</td>
<td>3.5 MPa</td>
</tr>
<tr>
<td>Parameter</td>
<td>$z$</td>
<td>$\frac{3GMR^2}{2a^3}$</td>
</tr>
<tr>
<td>Parameter</td>
<td>$c_1$</td>
<td>$\frac{2\mu}{gR}$</td>
</tr>
<tr>
<td>Parameter</td>
<td>$c_2$</td>
<td>$h_2 - 3l_2$</td>
</tr>
<tr>
<td>Parameter</td>
<td>$\delta$</td>
<td>$\frac{3\lambda+2\mu}{\lambda+2\mu}$</td>
</tr>
</tbody>
</table>

Table 3.5: Parameters in Stress Model: The various parameters for the Nimmo et al. [22] stress model and the parameters for Enceladus.

To determine whether the tiger stripes are in a state where eruptions may occur, the normal and shear stresses on the tiger stripes are resolved from the primary stresses above. The shear ($\tau$) and normal ($\sigma$) stresses are a function of the orientation of the fault line at the location in question (Equations 3.22 and 3.23 respectively). The angle between the fault and constant line of longitude is defined as $\beta$, and $\beta' = \beta - \pi/2$.

\[
\tau = .5(\sigma_{\phi\phi} - \sigma_{\theta\theta}) \sin 2\beta' + \sigma_{\theta\phi} \cos 2\beta' \tag{3.22}
\]
\[
\sigma = \sigma_{\theta\theta} \cos^2 \beta' + \sigma_{\phi\phi} \sin^2 \beta' + \sigma_{\theta\phi} \sin 2\beta' \tag{3.23}
\]

Jets are placed so that $M$ percentage of the total jets ($N$) are placed in proximity
Figure 3.6: Model of Tiger Stripes: The blue crosses represent the modeled boundary of the tiger stripes for comparison to Figure 2.3. The black dots indicate the center of Spitale and Porco’s eight identified main sources.

to the location of the eight source locations, with the remaining jets being uniformly distributed throughout the tiger stripes. Figure 3.6 shows the model of the tiger stripes with the eight plume sources marked. The source strength of the jets are given as a function of the normal stresses on the tiger stripe at the location where the jet is. The strength of each jet is then normalized such that the total mass discharge is between 500 and 750 kg/s. The exact value for the total discharge is a function of how many of the jets are in compression.

Jet models developed to replicate in-situ plume measurements by Cassini are of two main types. Saur et al. [32] developed a Gaussian model of the plume, and analyzes its ability to replicate the magnetic field observations by Cassini’s magnetometer during the E0, E1 and E2 flybys. He uses his model as both a single source located at the south pole, and a multi-source model using the eight sources located by Spitale and Porco [19]. Jia et al. [67] use a similar but different form of a Gaussian plume while also studying the magnetosphere. The Saur et al. [32] model is used again by Fleshman et al. [68] and Kriegel et al. [69] in studying the plasma interaction of Enceladus with magnetometer data. The other common form of the plume model is derived from Maxwellian velocity distribution at the source(s) [31, 33].
A Gaussian jet model is employed for the model, albeit in different form than the model proposed by Saur et al. [32]. Equation 3.24 shows the model for a single jet where \( Q \) is the jet emission rate, \( u_c \) is the characteristic velocity, \( k_c \) is the diffusion constant, and \( \Psi_0 \) is the size of the jet. The concentration is a function of position, that is described by the altitude \( (h) \) and the radius \( (R) \) from the vector normal to the surface at the jet location. The model is derived from plume models that propagate downwind. One of the main assumptions is that the majority of the particles in the plume eventually escape Enceladus’ sphere of influence and enter to the E-ring, only to be replaced by new particles.

\[
C = \frac{Q}{2\pi u_c \Psi_r} \exp \left( -\frac{R^2}{\Psi_r} \right) \tag{3.24}
\]

\[
\Psi_r = \Psi_0 + \frac{k_c h}{u_c} \tag{3.25}
\]

To ensure that the model is representative of Enceladus’ plume, a comparison is performed of the results to previous models that have been validated against Cassini instrument data. Figure 3.7 show the normal stresses acting on the eight source locations according to the model. The presented model has positive ejection rates for jets with positive (tensile) stresses and zero ejection rates for negative (compressive) stresses. Figure 3.7 notes the orbital position of the E0, E1, and E2 flybys by Cassini. Saur et al. [32] modeled the plume of Enceladus with only a single jet at the eight source locations. The ejection rates from these jets are shown in Table 3.6. A comparison of the results from Saur et al. [32] to the stresses seen in Figure 3.7 show that the expected sources (those experiencing tension) are indeed activated in Saur’s model. The presented model does not match well with the Dong et al. [33] model of the E3, E5, and E7 flyby model results. However, it can be seen that the E3 and E7 flybys are close in orbital position of Enceladus, and they have different source activation. However, the E3 and E5 flybys are much further away in orbital position
Figure 3.7: Stresses on Eight Source Regions The figure shows the stress levels on the eight source regions by Spitale and Porco [19]. Positive stresses correspond to tension loading while negative stresses correspond to compressive loading. It is expected that the jets in these regions would be contributing to the overall plume during the times where the stresses are tensile.

and the source activations are similar. It is concluded that the different jet models used between Dong et al. [33] and this thesis may explain the discrepancy.

3.2.2 Conclusion

A penetrator mission to the moon Enceladus is proposed. One of the key challenges in this proposal is targeting the jet sources to reach the subsurface ocean. To solve this problem the concept of VIFFing is presented, where the penetrator is able to thrust laterally during the free fall portion of the descent. A control system is needed to steer the spacecraft towards the geyser sources during this phase of the mission. To aid in the development and testing of such a system, a fine-scale model of the Enceladus plume is proposed. The model presented differs from others in the sense
Table 3.6: Previous Model Results: Shown are the results from Saur et al. [32] for flybys E0, E1, and E2 and from Dong et al. [33] for flybys E3, E5, and E7. These show the estimated activated sources. The estimated sources are compared to the stress model for the eight sources. The E0, E1, and E2 flybys correspond well with the model by [32] but do not compare well with Dong et al. [33]

<table>
<thead>
<tr>
<th>Flyby</th>
<th>MA [×10^8/s]</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>E0</td>
<td>264° [×10^8/s]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>240</td>
<td>190</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E1</td>
<td>142° [×10^8/s]</td>
<td>4.8</td>
<td>12.1</td>
<td>12.1</td>
<td>0</td>
<td>0</td>
<td>12.1</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>E2</td>
<td>98° [×10^8/s]</td>
<td>4.8</td>
<td>12.1</td>
<td>12.1</td>
<td>0</td>
<td>0</td>
<td>12.1</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>E3</td>
<td>290° [kg/s]</td>
<td>127.5</td>
<td>127.5</td>
<td>127.5</td>
<td>0</td>
<td>0</td>
<td>127.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E5</td>
<td>206° [kg/s]</td>
<td>262.5</td>
<td>262.5</td>
<td>262.5</td>
<td>0</td>
<td>0</td>
<td>262.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E7</td>
<td>283° [kg/s]</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>210</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

that it focuses on the small scale jets instead of the large, diffused plume. In addition, the ability to create a variety of realistic simulations is crucial to the testing of the system. Although the dynamics of the jets and plume are still not well understood, an algorithm can be tested for robustness to gain confidence in the mission in the face of the unknowns.
Chapter 4

Neural Control for Source Localization

Problem

The characteristics of localizing and navigating to an odour source in a turbulent plume have been shown to be complex and nontrivial. The more successful systems in the literature center around high-computational algorithms. In space missions, computation is limited and therefore it is desired to reduce the computational cost of the algorithms. In addition, there is a desire for an egocentric controller that is not limited to a search region. Neural networks are best known for mapping nonlinear functions through low computational networks. The benefits of using a neural network for control are in the low computation needed to perform potentially complex tasks. Once the network is optimized, called training, the neural network maps input sets to output sets. Training allows the network to learn a certain output behaviour in response to inputs from the environment.

The ‘cost’ of using neural control is in the training algorithms. The more complex the task being performed, the more neurons the network must have. The weight space grows exponentially with additional neurons, and their associated weights, and thus the optimization (training) problem grows as well. Despite the large effort taken to train neural networks, there are many benefits. Neural networks have been seen
to generalize the response to inputs it has not seen before, so long as the input is sufficiently similar to inputs it has seen during training. The ability to generalize is vital for the development of the controller, since the true odour plumes will not resemble the time averaged plume, nor will there be any correlation between the plume at two time instants. The network will have to generalize the behaviours it is trained to replicate in order to be able to achieve the task. This chapter will describe the efforts taken in teaching a recurrent neural network to react to time-averaged odour plumes as moths have been observed to act in nature, and the results for both static and dynamic plumes.

4.1 Controller Properties

This section outlines the details of designing and training the neural network controller. As stated in the previous section, the goal is to train the network to replicate the odour source seeking behaviours of moths. This section begins by reviewing the specific behaviours that are to be replicated. The development only replicates the most basic of behaviours due to complexity in training. Next the structure of the network, introducing the concept of a growing network, will be discussed. The model to which the network is to be optimized to is introduced, and the training sets which capture the desired behaviour are discussed. Lastly, the method for adjusting weights during training is discussed.

4.1.1 Desired Behaviour

The goal of the neural network is to replicate a simplified form of the behaviour seen in moths. As mentioned previously, there are two phases of plume-tracing: acquisition and following. The neural network is designed to achieve the second phase; that is, it will be designed to follow the plume once the plume is located. Moths have been seen
to exhibit two basic patterns: a slow casting while moving upwind for laminar plumes and casting followed by sudden burst forward upon detection in turbulent plumes. Since the casting pattern is featured in both of these behaviours it was chosen as the behaviour to teach to the neural network. The behaviour that will be used to train the network is as follows: while the vehicle is continuously moving forward at a slow speed, cast back and forth in the cross-wind direction ensuring that the vehicle stays within the odour plume, and converges to the source.

4.1.2 Structure of the Neural Network

The growing and pruning of neural networks has been explored for the MLP and other feed forward neural networks [70], however it has not yet been applied to recurrent neural networks. It is well known that when there are more neurons in the network, the network can replicate functions of higher order with lower errors. For this reason, the network has to be of a certain complexity in order to accurately replicate the behaviours being sought. The trade off with more complex structures is the higher dimensionality of the weight-space in which the training rule is attempting to solve. During optimization, false solutions due to phenomenon like local minima are more likely to occur due to these higher dimension spaces. For this reason, training begins on a network of lower complexity until the network is replicating the pattern to the best of its ability. Complexity is added to the network while the network is continued to be trained, taking advantage of the previous solution.

To take advantage of the previous solution, the structure of the network must not change the pathways from the input to the output, through the current weights. New weights may be added (starting at a value of zero before continuing training) to connect new neurons to previous neurons so long as they do not block these previous connections. Each time a new neuron is added to the layer, the associated weights to and from that neuron are added to the training rule, starting with a weighted value of
zero. The current trained weights act as the same network before the new neuron was added. One of the characteristics of this scheme is that as more neurons are added to the network there are naturally more weights connecting the neurons. For example, the second neuron add 4 weights \((w_{02}, w_{12}, w_{21}, w_{22})\) to the weight space. However, the third neuron will add 6 weights, and the number of additional weights will grow exponentially with each additional neuron.

4.2 Training

4.2.1 Training Sets

The desired response of the network is to have the vehicle act like a moth in a ribbon plume; slow forward speed casting back and forth and staying in the plume. The plume model used for training and initial testing is the 2D Sutton model presented in Equation 3.3 and shown again below:

\[
C(x, y) = \frac{Q}{\sqrt{2\pi S_y u}} \exp \left( -\frac{y^2}{2S_y^2} \right)
\]

\[
S_y = 0.5C_y x^{(2-n)/2}
\]

Figure 4.1 shows the centerline concentration for the 2D Sutton model with \(Q = 1\), \(C_y = 2\), and \(u = 1\), as a function of distance downwind. Since the concentration is below a value of 1 for all regions except very close to the source, the concentration value will be used as the input for the network. The output for the network is the lateral velocity that will be kept between zero and one for simplicity, as the activation function’s range is \([-1, 1]\).

As the moth gets closer to the source of the odour, the casting pattern shortens.
Figure 4.1: **Centerline Concentration**: This figure shows centering concentration for the 2D Sutton model with respect to distance downwind from the source.

To capture this effect, the casting length of the training set is reduced as the downwind distance is shortened. To non-dimensionalize the lateral distance, the standard deviation of the lateral concentration profile is used as the location measurement. Figure 4.2 shows the 2D Sutton model with the one-sigma bounds shown.

The individual training sets assumed that the vehicle was not moving upwind, and instead was only moving laterally within the plume. A ‘slice’ of the plume in the lateral direction was taken and the network was trained to cast between one standard deviation limits. The input to the network was the measured concentration, and the output was the lateral velocity. The network was trained on slices of the plume from many distances downwind. Training sets with different downwind positions was to ensure the network would learn to shorten its casting width in response moving closer to the source.

Since recurrent neural networks have been shown to replicate sinusoidal patterns
Figure 4.2: Model: This figure shows 2D Sutton model as well as the boundaries for the desired motion of the vehicle as it travels upwind towards the source. The concentration of the methane is shown in color on a scale from 0 to 1 kg/m$^2$. The boundary of the model shown is 10 meters in both directions crosswind from the centerline of the plume, and 20 meters downwind of the source. The boundaries for the desired motion are shown as black boundaries moving downwind the source.
Figure 4.3: Training Set: This figure shows an example training set. The top graph shows the input concentration, the middle graph shows the desired response velocity, and the bottom graph shows the result of the vehicle casting.

with little effort, casting pattern is defined as a sinusoid, with the position limits equaling the standard deviation (\(\sigma\)) described above and the velocity limited by \(v_{\text{max}} = 1\). An example training set for a downwind distance of 5 m is shown in Figure 4.3. The desired trajectory in shown in Equations 4.1 and 4.2.

\[
y = -\sigma \cos \left( \frac{v_{\text{max}} t}{\sigma} \right) \tag{4.1}
\]

\[
v = v_{\text{max}} \sin \left( \frac{t}{\sigma} \right) \tag{4.2}
\]

4.2.2 Kalman Filter Training Rule

In large neural networks, calculating the gradient information needed to perform either real-time recurrent-learning or extended Kalman filter methods can be extensive. This is due to the highly nonlinear dynamics arising from the feedback in the network. The unscented Kalman filter (UKF) does not involve explicit computation of the gradients, and has been shown to be more accurate for systems with higher number of
variables (weights and neuron outputs) and nonlinearity in the system [71]. For the UKF learning rule, both the activations of each neuron, and the weights between each layer are included in the state vector as in Equation 4.3, where \( x \) is the state vector, \( u \) is a vector of the activations of all neurons, and \( w \) is a vector of all the weights in the network.

\[
x = \begin{bmatrix} u \\ w \end{bmatrix}
\] (4.3)

The UKF uses 2M sigma points, where M is the length of the state vector. These sigma points are defined by Equation 4.4, where \( P_{xx}^{ij} \) is the jth row of the covariance matrix for the state vector.

\[
\sigma^j = x + (\sqrt{MP_{xx}^{ij}})^T \quad j = 1 \ldots M \\
\sigma^j = x - (\sqrt{MP_{xx}^{ij}})^T \quad j = M + 1 \ldots 2M
\] (4.4)

The inputs are propagated through each variation of the network corresponding to the sigma points. The mean of all propagated networks is taken to be the true state at the next time step. The output values are taken to be the mean of all the outputs from the sigma-networks and both the covariance matrix for the outputs and the cross-covariance matrix are calculated for the states and the outputs.

\[
\sigma_{k+1} = f(\sigma_k, u_k)
\] (4.5)
\[
\begin{bmatrix}
    y_{1}^{k+1} \\
    y_{2}^{k+1} \\
    w_{01}^{k+1} \\
    w_{02}^{k+1} \\
    w_{11}^{k+1} \\
    w_{12}^{k+1} \\
    w_{21}^{k+1} \\
    w_{22}^{k+1} \\
    b_{1}^{k+1} \\
    b_{2}^{k+1}
\end{bmatrix} = \begin{bmatrix}
    g(u^k w_{01}^k + s_1^k w_{11}^k + s_2^k w_{21}^k + b_1^k) \\
    g(u^k w_{02}^k + s_1^k w_{12}^k + s_2^k w_{22}^k + b_2^k) \\
    w_0^k \\
    w_0^k \\
    w_1^k \\
    w_1^k \\
    w_2^k \\
    w_2^k \\
    b_1^k \\
    b_2^k
\end{bmatrix}
\]

\[x_{k|k-1} = \frac{1}{2M} \sum_{k} \sigma_{k}^{j}\quad (4.7)\]

\[P_{k|k-1}^{xx} = \frac{1}{2M} \sum_{k} [\sigma_{k}^{j} - x_{k|k-1}][\sigma_{k}^{j} - x_{k|k-1}]^{T}\quad (4.8)\]

\[\hat{y}_{k} = \frac{1}{2M} \sum_{k} y_{k}^{j}\quad (4.9)\]

\[P_{k}^{yy} = \frac{1}{2M} \sum_{k} [y_{k}^{j} - \hat{y}_{k}][y_{k}^{j} - \hat{y}_{k}]^{T}\quad (4.10)\]

\[P_{k}^{xy} = \frac{1}{2M} \sum_{k} [\sigma_{j}^{j} - x_{k|k-1}][y_{k}^{j} - \hat{y}_{k}]^{T}\quad (4.11)\]

The Kalman gain and covariance updates are a relation of the output-covariance and the cross-covariance. Because the activations are included in the state vector, forced learning implicitly takes place when the activations are updated by the innovations process.
4.3 Results

4.3.1 Gaussian Plume

The structure of a neural network greatly impacts the complexity of the problems it can solve. It also adds to the difficulty in solving for the optimal weights. Because the number of weights increases exponentially with additional neurons, the structure of the network is kept to a minimum number of neurons as possible while still achieving the task of casting up towards the source. A simple 1-layer, 2-neuron network with internal feedback is initially picked for this task. The very simple network shown was trained until it was able to navigate the odour plume. Figure 4.5 shows the vehicle navigating the odour plume using the neural network controller for its lateral position. The network was trained with no forward velocity, but the vehicle was seen to perform well for many non-zero forward speeds. The forward speed for the results shown in Figure 4.5 is 5% of the maximum lateral velocity, although the vehicle was able to successfully navigate the plume for forward velocities up to 100% the
Figure 4.5: Results of 2R Network: This figure shows a trial of the vehicle navigating through the 2D Sutton plume after training for a successful trial. The vehicle begins inside the plume and follows an isoline of constant concentration upwind, converging in the direction of the source. The casting pattern does not match the desired pattern. The vehicle does display the desired characteristic of a decreased lateral casting distance as it moves upwind.

maximum lateral velocity.

To test the robustness of the neural controller, the network was tested with the vehicle having different initial conditions. Figure 4.6 shows the positions where the network successfully guided the vehicle to the source and where it failed to do so. The vehicle was biased only being able to complete the task when beginning either at the centerline or in the negative y direction. The bias is due to the fact that all training sets had similar initial conditions. Those initial conditions were a position to the left side of the centerline of the plume. In an attempt to remove this bias from the results, a new neural network was trained with training sets with various initial conditions but did not converge to a better solution than presented here.

Despite the bias, it was seen that the vehicle traveled a similar path regardless
Figure 4.6: Showing Consistency of Results: This figure shows starting locations in the plume which results in successful behaviour and non-successful behaviours. White circles denote the positions of successful trials and black x’s denote locations where the vehicle failed to behave as desired. The vehicle successfully navigates in the direction of the source when the vehicle has a starting position inside the plume to the left of the centerline facing downwind. It fails to travel in the direction of the source otherwise. This need for certain initial conditions probably originates from the training sets having consistent initial conditions, however attempts to retrain with a training sets having a variety of initial conditions did not yield convergent results.
Figure 4.7: Showing Robustness of Results: This figure shows the fused paths of trials from multiple starting positions. The locations corresponding to white circles in Figure 4.6 are all seen to travel along the same corridor centered at a particular isoline of constant concentration. The starting locations corresponding to black x's in Figure 4.6 are seen to diverge out of the boundary being considered and do not return.

of its starting location. Figure 4.7 shows the results from multiple trials overlaid onto the same plot. There seems to be a single alley that the vehicle travels in all successful trials. The alley seems to follow an isoline of constant concentration in the plume. The alley also shrinks as the vehicle approaches the source, which is one of the desired characteristics of the network behaviour. It can be seen that there are two phases of the traverse for successful trials. The first part is reaching the alley, and the second is the casting back and forth through the alley in the direction of the source.

The network causes the vehicle to cast through the plume with the casting distance shrinking as the vehicle nears the source. The observed behaviour is similar to the desired behaviour, however the casting behaviour does not follow the center of the
plume, but rather an isoline that is biased to one side of the plume. The following of isolines was seen in experiments on genetic algorithms to solve the source localization problem [72].

4.3.2 Dynamic Plume

The neural network was tested on a dynamic plume to observe its suitability for real-world environments. The network was tested on the Martian methane model presented in Section 3.1. Figure 4.8 shows the path the vehicle took through the plume on top of the average plume concentration for the simulation. Although the time-averaged plume resembles the 2D Sutton plume that was used for training, the instantaneous plume that the vehicle encountered differed. Figure 4.9 shows the measurement and output values for the simulation. The input measurements are seen to differ from the input training set. Lastly, a snapshot of the simulation is shown in Figure 4.10. It can be observed that the instantaneous plume differs from the 2D Sutton model.

The vehicle was able to stay inside the plume as it traveled towards the odour source. It is seen that the casting motion is not as consistent as it was for the Sutton static plume. The vehicle does reach the source area, but it tends to diverge when it faces the high concentration values very close to the source. This means that the exact source location cannot be found, but rather an area bounding the location of the source can be found.

4.4 Conclusions

Trained recurrent neural networks emulate the behaviours of scent tracking animals and insects for the purpose of detecting sources of methane on Mars. The neural network has many benefits. Neural networks provide a computationally light algorithm
Figure 4.8: **Results in a Turbulent Plume**: The figure shows the path taken by the vehicle guided by the neural network through a turbulent plume. The wind is to the right, causing a Sutton plume to form with the origin at [5, 20]. The vehicle begins at the location marked by a white square. The path of the vehicle is shown as it travels towards the source up to the point where it diverges away from the source. The plume shown in the background is the time-averaged plume for the time of the simulation. It can be seen that the time average of the plume in the simulation looks similar to the Sutton model used in training. The instantaneous concentration measured by the vehicle during the simulations were not similar to Sutton, and can be characterized as a turbulent plume. See Figures 4.9 and 4.10 for the measurement inputs and an example of the instantaneous plume, respectively.
Figure 4.9: Data from Turbulent Plume Trial: The figure shows the input and output of the neural network while navigating through a turbulent plume.

Figure 4.10: Screen Shot of Turbulent Plume: A screen shot of the simulation of the neural network navigating through the turbulent plume is shown. The path is shown up to the point of the screenshot. The dynamic plume is shown behind the partial path.
for reproducing a desired behaviour in response to measurements. Neural networks tend to generalize their behaviour in situations where they have not been explicitly trained, and once trained they are easy to implement. This generalization is the main motivation for using neural networks; the hope being that the generalization would provide additional robustness to highly turbulent conditions. The downside to using neural networks is the amount of effort required to train them. Both structure and training rules will need to be adapted depending on the results, and additional inputs may be needed for convergence.

The network developed for this thesis was able to achieve the goal of casting towards the source, although only under starting conditions inside the plume to the left side of the centerline facing upwind. Furthermore, the network was able to navigate through a dynamic plume that shows the generalization of the network to similar, yet different, input patterns from what it was trained to. The neural network did have problems of converging directly to the source. In the static case, the vehicle consistently ended in the negative-y direction away from the source location. In the dynamic plume, the vehicle tended to diverge near the source.

Since the neural network maps an input to a target output, the system is purely reactive, whereas the source likelihood maps discussed earlier allow more creativity in the controller since the controller and source identification system are independent. Reactive vs. predictive systems are an important consideration when accounting for obstacle avoidance in addition to source finding. Furthermore, in more advanced systems, additional information other than wind speed and direction and local concentration information may be used. For example, using the rovers cameras will allow the system to rule out certain areas as unlikely sources of methane and may reduce searching time. Passing this information to the neural network to take advantage of it may be more difficult than in traditional likelihood map methods. As stated earlier, the neural network does not face the boundary limitations of the tradition
SLIM, however it should be possible to alter these boundaries during the traverse.

Future work includes continuing to train the networks to perform the tasks required. Should the networks be able to behave as desired in all situations, including navigation, control will be required to avoid obstacles. The neural field, which is a completely interconnected recurrent neural network, has been used for navigating obstacles to a target location [73]. It may be found that the neural field can be combined with the currently proposed system to implement both tasks simultaneously.
Chapter 5

Source Likelihood Map for Source Localization Problem

This chapter will outline the work to improve the accuracy of the Source Likelihood Map (SLIM) algorithm proposed by Pang and Farrell [6]. First, the original algorithm will be explained. The novel addition to the algorithm and the results of the amendment will be presented in section 5.2.

5.1 Algorithm Explanation

Using Bayesian methods to create a Source Likelihood Map (SLIM) were proposed by Pang and Farrell [6] as an upgrade to their work using Hidden Markov Methods [34]. The authors applied their algorithms to aquatic unmanned vehicles to search for chemicals being emitted from the ocean floor. The algorithm is based on a puff model that describes the motion of individual puffs in the plume rather than the plume in its entirety. The algorithm uses a binary chemical sensor and a forcing vector sensor. The binary chemical sensor measures whether the vehicle is currently inside a puff or not. The algorithm uses the information of when it is in puffs to backwards integrate where the puff originated from. Backwards integration allows
for the creation of a source likelihood map (SLIM). The benefit of this algorithm comes from using Bayesian probability methods to update the SLIM with successive detection or non-detection events.

5.1.1 Puff Probability Model

Consider a single puff of odour that is centered at a position $p$, exposed to wind that varies over time and position $u(p, t)$. The motion of the puff of odour could be described as a stochastic process, where only the wind part of the advection-diffusion equation is considered, with the process affected by a white noise $\eta$. The covariance, $\sigma$ of the white noise is shown by the autocorrelation function, $R$

$$\dot{p}_i = u(p, t) + \eta_1$$ \hspace{1cm} (5.1)

$$R_{\eta_1 \eta_1}(\tau) = \sigma_1^2 \delta(\tau)$$ \hspace{1cm} (5.2)

The change in position of a filament is only dependent on the wind and a random walk. If the wind function is known then the motion of the puff can be tracked. In addition, if the puff is detected at some position and time $p(t_2)$, then the location of the puff at some earlier time, $p(t_1)$ can be determined through backwards integration.

$$p(t) = \int_{t_1}^t u(p, \tau) \partial \tau + \int_{t_0}^t \eta_1(\tau) \partial \tau + p(t_1)$$ \hspace{1cm} (5.3)

$$p(t_1) = p(t_2) - \int_{t_1}^{t_2} u(p, \tau) \partial \tau - \int_{t_1}^{t_2} \eta_1(\tau) \partial \tau$$ \hspace{1cm} (5.4)

The wind profile for every location is unavailable but the wind history for the position of the vehicle at each time step can be measured and stored. The record of wind measurements is used to approximate the flow for the entire area as the recorded
wind velocity measured by the rover, plus some noise.

\[ \int_{t_1}^{t_2} u(p, \tau) \, d\tau = \int_{t_1}^{t_2} [u(p_r(\tau), \tau) + \eta_2] \, d\tau \quad (5.5) \]

Where \( u(p, \tau) \) describes the entire wind velocity map and \( u(p_r(\tau), \tau) \) represents the recorded history of the wind vector at the rover location, \( p_r \) only.

\[ p(t_1) = p(t_2) - \int_{t_1}^{t_2} u(p(\tau), \tau) \, d\tau - \int_{t_1}^{t_2} \eta_1(\tau) \, d\tau - \int_{t_1}^{t_2} \eta_2(\tau) \, d\tau \quad (5.6) \]

Integrating white noise increases the variance of the noise increases the variance of the normal distribution. Normal distributions are denoted \( N\{\mu, \sigma^2\} \) where \( \mu \) is the mean and \( \sigma^2 \) is the variance.

\[ W_i(t_1, t_2) = \int_{t_1}^{t_2} \eta_i \, d\tau \sim N\{0, (t_2 - t_1)\sigma_i^2\} \quad (5.7) \]

\[ W(t_1, t_2) = W_1(t_1, t_2) + W_2(t_1, t_2) \quad (5.8) \]

\[ f(W(t_1, t_2)) = N\{0, (t_2 - t_1)\sigma_1^2\} + N\{0, (t_2 - t_1)\sigma_2^2\} \quad (5.9) \]

\[ = N\{0, (t_2 - t_1)(\sigma_1^2 + \sigma_2^2)\} \quad (5.10) \]

\[ = N\{0, (t_2 - t_1)\sigma^2\} \quad (5.11) \]

Where the total covariance is \( \sigma^2 = \sigma_1^2 + \sigma_2^2 \). The term for the integral of the wind history is simplified.

\[ v(t_1, t_2) = \int_{t_1}^{t_2} u(p(\tau), \tau) \, d\tau \quad (5.12) \]
\[ \mathbf{p}(t_1) = \mathbf{p}(t_1, t_2) - \mathbf{v}(t_1, t_2) + W(t_1, t_2) \] 

CHANGEINSIGNOF W (5.13)

The position vector is separated into its two coordinates: \( x \) and \( y \). The coordinates of the position where the odour was released are denoted as \( t_1 \) as \((x_i, y_i)\) and the position where the odour was detected as \( t_2 \) as \((x_j, y_j)\). The change in velocity, \( v_x \) and \( v_y \), are the integrals of the wind history in the respective directions.

\[
x_i = x_j - v_x(t_1, t_2) + w_x(t_1, t_2) \quad \text{(5.14)}
\]
\[
y_i = y_j - v_y(t_1, t_2) + w_y(t_1, t_2) \quad \text{(5.15)}
\]

The noise terms have the following probability density function.

\[
f(w_x(t_1, t_2)) \sim N \{0, (t_2 - t_1)\sigma_x^2\} \quad \text{(5.16)}
\]
\[
f(w_y(t_1, t_2)) \sim N \{0, (t_2 - t_1)\sigma_y^2\} \quad \text{(5.17)}
\]

The probability of a position \((x, y)\) being the position of the release event \((x_k, y_k)\) is the bivariate normal distribution.

\[
f_{t_1, t_2}(x) \sim N \{x_j - v_x(t_1, t_2), (t_2 - t_1)\sigma_x^2\} \quad \text{(5.18)}
\]
\[
f_{t_1, t_2}(y) \sim N \{y_j - v_y(t_1, t_2), (t_2 - t_1)\sigma_y^2\} \quad \text{(5.19)}
\]
\[
f_{t_1, t_2}(p) = f_{t_1, t_2}(x) \times f_{t_1, t_2}(y) \quad \text{(5.20)}
\]
This section provides the backward integration model for determining where a puff was emitted given a detection and a wind history vector. The next section will describe the discretization of the map into regions and the creation of the SLIM.

5.1.2 Source Likelihood Maps from Detections

If a source likelihood map is divided into discrete regions, then the probability that the region $R_i$ contains the source emitting a puff at $t_1$ given the detection of a puff in region $R_j$ at $t_2$ is defined as $S_{ij}(t_1, t_2)$, which is the integral of $f_{t_1,t_2}(p)$ over the region $R_i$.

$$f_{t_1,t_2}(p) = \frac{e^{-\frac{(x-x_j-v_x(t_1,t_2))^2}{(2(t_2-t_1)\sigma_x^2)}} \times e^{-\frac{(y-y_j-v_y(t_1,t_2))^2}{(2(t_2-t_1)\sigma_y^2)}}}{\sqrt{2\pi(t_2-t_1)}\sigma_x\sigma_y}$$

$$S_{ij}(t_1, t_2) = \int_{x \in R_i} \int_{y \in R_i} e^{-\frac{(x-x_j-v_x(t_1,t_2))^2}{(2(t_2-t_1)\sigma_x^2)}} dx \times e^{-\frac{(y-y_j-v_y(t_1,t_2))^2}{(2(t_2-t_1)\sigma_y^2)}} dy$$

From here the vector of probabilities that describes the probabilities for all regions is the Source Likelihood Map (SLIM). For one detection, given a puff’s creation at time $t_1$ and its detection at time $t_2$, its origin can be predicted based on the detection location. If the creation time, the time when the puff is created at the source location, is unknown, then a range of possible times must be considered as the time the puff could have been created. The SLIM for a single detection event would then be the weighted average of all considered creation times between the beginning of the wind record at $t_0$ to the detection at $t_2$. 

$$f_{t_1,t_2}(p) = \frac{e^{-\frac{(x-x_j-v_x(t_1,t_2))^2}{(2(t_2-t_1)\sigma_x^2)}} \times e^{-\frac{(y-y_j-v_y(t_1,t_2))^2}{(2(t_2-t_1)\sigma_y^2)}}}{\sqrt{2\pi(t_2-t_1)}\sigma_x\sigma_y}$$

(5.21)
\[ B_{ij}(t_0, t_2) = \int_{t_0}^{t_2} S_{ij}(\tau, t_2)f(t_1 = \tau) \, d\tau \] (5.24)

If no information is available about the creation time distribution then the simple average is taken, as in [6]. A non-detection can also be used to create a SLIM. The probability of a non-event is unity minus the probability of the event.

The probability that a source emitting a puff from location \( R_i \) at time \( t_1 \) and is detected at a location \( R_j \) at \( t_2 \) is equal to \( \mu S_{ij}(t_1, t_2) \), where \( \mu \) is equal to the probability that the detector will sense the chemical given that the chemical is in the region. Therefore the SLIM created from a non-detection is:

\[ \bar{S}_{ij}(t_1, t_2) = 1 - \mu S_{ij}(t_1, t_2) \] (5.25)

\[ \gamma_{ij}(t_0, t_2) = \frac{1}{t_2 - t_0} \int_{t_0}^{t_2} \bar{S}_{ij}(\tau, t_2) \, d\tau \] (5.26)

### 5.1.3 Applying Successive Events

In order for the algorithm to be effective, the vehicle needs to be able to update the SLIM when additional information becomes available. This section discusses the method for updating the current SLIM when a new detection or non-detection event occurs. Let \( A_i \) stand for the event that there is a source located within the region \( R_i \). Let \( D_j(t_k) \) stand for a detection event happening in the region \( R_j \) at time \( t_k \). Then the probability that the source in a particular region based on a single measurement is \( B_{ij}(t_0, t_2) \).

\[ f(A_i|D_j(t_k)) = B_{ij}(t_0, t_k) \] (5.27)

To update the SLIM, the probability of the source being located in a region based
on all previous measurements is desired. The total SLIM made up of \( N \) measurements is denoted as \( \Pi \).

\[
\Pi = f(A_i|D_j(t_k); k = 1, 2, ..., N) \quad (5.28)
\]

\[
= f(A_i|D_j(t_N), D_j(t_k); k = 1, 2, ..., N - 1) \quad (5.29)
\]

To simplify the expression, define \( E_j(t_{N-1}) = \sum_{k=1}^{N-1} D_j(t_k) \). The definition of conditional probability is used to express the SLIM as joint probability distributions.

\[
\Pi = f(A_iD_j(t_N)E_j(t_{N-1})) / f(D_j(t_N))f(E_j(t_{N-1})) \quad (5.30)
\]

The chain rule is applied to the numerator.

\[
\Pi = f(D_j(t_N)E_j(t_{N-1})|A_i) f(A_i) / f(D_j(t_N))f(E_j(t_{N-1})) \quad (5.31)
\]

The current detection event and all previous measurements are assumed to be independent. Therefore, the probabilities can be separated.

\[
\Pi = f(D_j(t_N)|A_i)f(E_j(t_{N-1})|A_i) f(A_i) / f(D_j(t_N))f(E_j(t_{N-1})) \quad (5.32)
\]

\[
\Pi = f(A_i) f(D_j(t_N)|A_i)f(E_j(t_{N-1})|A_i) f(A_i) / f(A_i) f(D_j(t_N))f(E_j(t_{N-1})) \quad (5.33)
\]

Bayes Law is used to transform the conditional probabilities into the backwards probability.

\[
\Pi = \frac{1}{A_i} f(A_i|D_j(t_N)) f(A_i|E_j(t_{N-1})) \quad (5.34)
\]

The value for the new SLIM is the probability of the source being in the region
based on the current measurement multiplied by the probability of the source being in the region from the previous SLIM multiplied by a normalizing factor, $M$.

\[
f(A_i|E_j(t_{N-1})) = \Pi_{k-1}
\]

\[
f(A_i|D_j(t_N)) = B_{ij}(t_0, t_N)
\]

\[
\frac{1}{A_i} = M
\]

\[
\Pi_k = MB_{ij}(t_0, t_N)\Pi_{k-1}
\]

The process can be repeated to solve for the expression for updating the SLIM due to a non-detection event, which is shown below. See [6] for the entire derivation.

\[
\Pi_k = \frac{\Pi_{k-1}\gamma_{ij}(t_0, t_N)M}{\int \gamma_{ij}(t_0, t_2)}
\]

### 5.1.4 Individual Detections

Figure 5.1 shows the SLIM from a single detection event given different inputs. The first five plots show the SLIM from a single detection assuming the release time is known, for release times varying from one second to forty seconds. It can be seen that as the assumed release time is a long time from the detection time the uncertainty in the SLIM increases. The increase in uncertainty with time is due to the noise on the advection process. The result of the increase in variance as release time increases is that when the results are integrated over a uniform release time probability distribution, the total SLIM becomes biased to the vehicle location.
Figure 5.1: SLIM for Various Settings: Shown are six source likelihood maps based on a wind sensor history of a steady wind in the upwards direction and a single detection event. The first five plots show the SLIM produced if the release time ($\Delta T$) is assumed, and the last figure shows the SLIM for the case when the SLIM is averaged over all possible release times. It is seen that a single detection event is biased towards the position of the vehicle for the averaged SLIM case. The white square represents the vehicle location.
5.2 Improving Accuracy

This section will detail the efforts made in improving the accuracy of the above algorithm by using a non-uniform probability distribution for the release time in Equation 5.24. Instead of a uniform distribution, a normal distribution is used with the mean at a predicted release time and high standard deviation to account for errors. The section will first show the need for such an addition to improve accuracy. Next, the method for predicting the release time and formulating the probability distribution function are shown. Lastly, the results of the addition are shown.

5.2.1 Release Time Estimation

Until now, only the position of the puffs has been considered, while the concentration of the puffs has not been. As the position of the puff will change due to the advection processes taking place, the shape and distribution of the odour within the puff will change due to diffusion, of both the chemical and turbulent type. A puff-body coordinate system that moves with the center of the puff is used. The model of concentration distribution is shown by the diffusion equation.

\[
\frac{\partial C}{\partial t} = \frac{\partial}{\partial x} \left( D_x \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left( D_y \frac{\partial C}{\partial y} \right)
\]  

Equation 5.40 is simply the two-dimensional heat equation. Considering the initial conditions of a unit pulse, the solution is the Euclidean heat kernel.

\[
C(x, y, 0) = Q\delta(x)\delta(y)
\]  

\[
C(x, y, t) = \frac{Q}{4\pi t \sqrt{D_x D_y}} \exp \left[ - \left( \frac{x^2}{4D_x t} + \frac{y^2}{4D_y t} \right) \right]
\]

When a detection event occur, the concentration for the entire puff is recorded. If

\[
C(x, y, t) = \frac{Q}{4\pi t \sqrt{D_x D_y}} \exp \left[ - \left( \frac{x^2}{4D_x t} + \frac{y^2}{4D_y t} \right) \right]
\]
it is assume that the maximum concentration recorded is from the center of the puff and a prediction of the diffusion constants exists, then the time in which the puff has been diffusing can be predicted. It is assumed that the diffusion constant is equal in both axes.

\[ C(t) = \frac{Q}{4\pi t_{est}D} \quad (5.43) \]

From Equation 5.43 the release time can be estimated with knowledge of the diffusion constant, the source emission strength and the measured concentration. Figure 5.2 shows the corresponding concentration over time from Equation 5.43 and the simulation described in Chapter 3. Since an error in the concentration measurement can cause a potentially large error in the release time estimate, the standard deviation is proportional to the slope of the curve. The proportion chosen is 5% of the slope and is shown in Figure 5.3.

\[ \mu_{t_{est}} = \frac{Q}{4\pi DC} \quad (5.44) \]

\[ \sigma_{t_{est}} = 0.05 \frac{4\pi D\mu_{t_{est}}^2}{Q} \quad (5.45) \]

The final probability density function is written as.

\[ f(t_1 = \tau) = \frac{1}{\sqrt{2\pi}\sigma_{t_{est}}} \exp \left[ -\frac{(\tau - \mu_{t_{est}})^2}{2\sigma_{t_{est}}^2} \right] \quad (5.46) \]

### 5.2.2 Results

This section compares the original SLIM algorithm and the amended algorithm presented in the two previous sections. Because a path planner was not created as part
Figure 5.2: Estimating the Release Time The figure shows the estimated release time with relation to the concentration of the puff detected.

Figure 5.3: Standard Deviation of Estimate Release Time The figure shows the relationship with mean and standard deviation of the release time estimate.
of this work, the algorithms were tested in two simple vehicle paths. The first experiment involved a stationary vehicle 40 m downwind the source. The source emitted puffs at a constant frequency, and a constant wind pattern of 1 m/s was used for simplicity. The second experiment was performed under the same simulation conditions, but with the vehicle moving at a constant speed towards the source.

The major change that the amended algorithm provides is the SLIM created by the first detection event. Figures 5.4 and 5.5 show the SLIM after the first detection event for the original and amended algorithms respectively. The amended algorithm has an estimate that is accurate to the source location, but the variance of the estimate is high. The estimate of the original algorithm however is biased to be closer to the position of the vehicle and has a lower variance. The bias to the vehicle’s location comes from solving Equation 5.24 for $B$ with a uniform profile for the release time. Since the variance on the source increases with larger release times in Equation 5.23 the larger release times, and thus the further distances from the vehicle, are implicitly weighted lower. This makes sense from a probability view because the chances of detecting a puff from a source is more likely the closer to that source. However, when updating the SLIM this bias is increased further due to the multiplication of SLIM’s in the update rule.

Figures 5.6 and 5.7 show the concentration history of the vehicle as well as the error of the estimate and the entropy of the SLIM. Note that the initial error is due to the estimate being in the center of the search location when the SLIM is uniform. Even though the entropy of the SLIM decreases for the original algorithm, the estimate never converges to the source location due to the lack of information about the release time. The estimate of the source in the amended algorithm is fairly accurate after the first detection and converges on the source location with subsequent detections.

The results of the second experiment can be seen in Figures 5.8 and 5.9. The
Figure 5.4: Initial SLIM for Original Algorithm: This figure shows the SLIM after the first detection event experience by the original algorithm for the stationary vehicle. The source is shown by a black cross and the vehicle location is shown by a white square.

Figure 5.5: Initial SLIM for Amended Algorithm: This figure shows the SLIM after the first detection event experience by the amended algorithm for the stationary vehicle. The source is shown by a black cross and the vehicle location is shown by a white square.
Figure 5.6: Data from Stationary Experiment, Original Algorithm: This figure shows results for the stationary vehicle during the experiment for the original algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM.

Figure 5.7: Data from Stationary Experiment, Amended Algorithm: This figure shows results for the stationary vehicle during the experiment for the amended algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM.
estimate from the original algorithm does seem to converge on the source location but slower than the vehicle’s position. The entropy of the SLIM is fairly constant as the estimate changes with vehicle position. For the original algorithm to be effective, the path planner must ensure that the vehicle is always moving towards the source.

The amended algorithm initially shows results similar to the stationary experiment. The estimate of the source is fairly good after the first detection event and converges quickly to the source location. It appears that as the vehicle is moving towards the source, the estimate becomes slightly worse over time, even as the entropy is falling. The error in the source estimation probably due to the release time estimation being corrupted by the vehicles motion and perhaps becoming worse as the vehicle gets closer to the source with higher concentration values. However, the localization by the amended algorithm places less demand on the path planner then the original algorithm does because it is not reliant upon moving towards the source location.

5.3 Conclusion

The SLIM algorithm proposed by Pang and Farrell [6] provides for combining the estimates of multiple detection and non-detection events to locate a chemical source given detection measurements and forcing function measurements. The algorithm converges quickly to a precise solution where other algorithms have large regions of uncertainty in the estimate. However, the accuracy of the estimate needs improving. Presented is a simple addition to the algorithm that allows for more accurate estimates of the source location.

The estimation of the release time is dependent upon having good estimates for the turbulent diffusion coefficient as well as the source emission strength. Without these values, the release time cannot be accurately estimated. In practice, these terms
Figure 5.8: Data from Dynamic Experiment, Original Algorithm: This figure shows results for the dynamic vehicle during the experiment for the original algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM.
Figure 5.9: Data from Dynamic Experiment, Amended Algorithm: This figure shows results for the dynamic vehicle during the experiment for the amended algorithm. The data shows the input concentration to the vehicle, the error in the estimate of the source, and the entropy of the SLIM. It is worth noting that after the estimation error approaches zero during initial puff detections, the estimation error increases slightly as the vehicle approaches closer. This is due to the fact that the estimation of the release time is more sensitive at higher concentrations (see figure 5.2)
may be hard to obtain. One potential method to obtain these values is to attempt to fit time-averaged measurements of the plume to a 2D Sutton model, estimating the values $Q$, $u$, and $C_y$. It may be that the values obtained from this fitting correspond to the true values, and can be used in the algorithm presented. This method has not been explored due to time constraints on the project.
Chapter 6

Conclusion

This thesis shows the analogy between autonomous science for planetary exploration missions and the odour source localization problem, and analyzes different potential algorithms that could be used to solve this problem. Testing environments for two potential missions that utilize source localization have been presented and tested for consistency to the real-life environment. Neural networks acting as the controller were explored but found to be difficult to incorporate practically into this type of mission. Bayesian probability maps of the source were investigated and an addition to the algorithm was presented to improve its accuracy.

6.1 Summary of Contributions

1. The creation of a Martian methane plume model for testing vehicle control in turbulent conditions. The model parameters can easily be changed for conditioning the plume as desired. The model has been shown to display the characteristics that make the odour source localization problem difficult and compares well to previous models and experimental data.

2. The creation of a fine-scale model of Enceladus’ icy jets and the plume that they form. The model is a finer scale than previous models and accounts for
changes in the plume due to the geophysical stresses acting on the moon. The model agrees with the limited data available about the environment.

3. The design and simulation of a neural network controller trained to respond to odour plumes in a way that models a simplified form of moth behaviour. The controller is tested both in static and dynamic plumes. The limitations of neural networks for control that become apparent from the experiments are noted.

4. An addition to the algorithm by Pang and Farrell [6] that improves accuracy by predicting the release time of odour puffs. Results are presented that support the need for such an addition and indicate that the algorithm works well.

6.2 Future Works

Work towards finding computational efficient and accurate algorithms to the chemical source localization problem is still ongoing. This thesis looked at two widely varying techniques for solving the problem but there are many future works that can be completed. Also, as with any extraterrestrial environment model, new information about the environments on Mars and Enceladus will continue with current and new missions to these celestial bodies. The models presented in this thesis may need to be updated in the future to account for the new data gain by these missions. Specific future works include:

1. Tracking Cassini’s path in the vicinity of Enceladus and compare the measurements to the simulated data. Alter the PDF of the jet locations and strengths so that they do not just match former models but match the data.

2. There is still work to be done on training neural networks on the ability to cast through a dynamic turbulent plume. More advanced techniques of training
the neural network may provide better results. For example, using genetic algorithms with a fitness functions that gives preference to the moth behavior; or training the network in real time based on physical observations of a moth in a laboratory.

3. Neural fields have been used for local path planning through obstacle fields. A more practical mobile robot would be able to navigate to an odour source while also avoiding obstacles. Combining both of these reactive behaviours into one neural system would strengthen the case for neural control in space vehicles.

4. Predicting the release time of a plume is dependent upon knowing the diffusion constant for the chemical in question and knowing the ejection strength of the source. Chemical diffusion can be predicted based on the chemistry of the system, but the turbulent diffusion and source emission strength need to be predicted by other means. Taking time averaged values of the plume and different locations may allow for predicting the time-averaged values for the wind speed and direction, diffusion, and source strength values in the 2D Sutton model. This solution was not able to be addressed in the time of this study but could be explored in the future.

5. The source likelihood map method for localizing odour sources separates the localization from the control phase of the mission. Where reactive controllers locate the source by following the plume to the location, SLIM techniques do not need to travel to the source to locate it. The decoupling of the traverse and localization provides a need for a guidance/control system that decides the trajectory of the vehicle. The vehicle should move in such a way that it is likely for a detection event to occur. The development of such control laws combined with the SLIM localization method would improve the benefits of both algorithms.
6. Algorithms have been developed and tested for use in a rover control method. Testing these algorithms in the Enceladus environment for a penetrator would be the next step in showing that the odour source localization analogy can be applied to multiple planetary exploration missions and not just to the Martian methane mission.
Appendices
Appendix A

2D Sutton Model

The Sutton Model as presented in Equation 3.1 and 3.2 is shown below.

\[
C(x, y, z) = \frac{Q}{2\pi S_y S_z u} \exp \left( - \left( \frac{y^2}{2S_y^2} + \frac{z^2}{2S_z^2} \right) \right)
\]  \hspace{1cm} (A.1)

\[
Sy = 0.5C_y x^{(2-n)/2}
\]

\[
Sz = 0.5C_z x^{(2-n)/2}
\]  \hspace{1cm} (A.2)

To acquire the two-dimensional form of the model, integrate over the entire z-axis.

\[
C(x, y) = \int_0^\infty C(x, y, z) \, dz
\]

\[
= \frac{Q}{2\pi S_y S_z u} \exp \left( - \frac{y^2}{2S_y^2} \right) \int_0^\infty \exp \left( - \frac{z^2}{2S_z^2} \right) \, dz
\]  \hspace{1cm} (A.3)

The integral in the above expression is defined as \( I \). The integral is then squared and a substitution of variables takes place.
\[ I = \int_0^\infty \exp \left( -\frac{z^2}{2S_z^2} \right) \, dz \quad \text{(A.4)} \]

\[ I^2 = \int_0^\infty \exp \left( -\frac{z^2}{2S_z^2} \right) \, dz \int_0^\infty \exp \left( -\frac{w^2}{2S_z^2} \right) \, dw \quad \text{(A.5)} \]

\[ = \int_0^\infty \int_0^\infty \exp \left( -\frac{z^2 + w^2}{2S_z^2} \right) \, dz \, dw \quad \text{(A.6)} \]

The coordinate system is changed from Cartesian to polar.

\[ r^2 = z^2 + w^2 \quad \text{(A.7)} \]

\[ \partial z \, \partial w = r \, \partial r \, \partial \theta \quad \text{(A.8)} \]

\[ I^2 = \int_0^{2\pi} \int_0^\infty \exp \left( -\frac{r^2}{2S_z^2} \right) \, r \, \partial r \, \partial \theta \quad \text{(A.9)} \]

\[ = \int_0^{2\pi} \left[ -S_z^2 \exp \left( -\frac{r}{2S_z^2} \right) \right]_0^\infty \, \partial \theta \quad \text{(A.10)} \]

\[ = \int_0^{2\pi} S_z^2 \, \partial \theta \quad \text{(A.11)} \]

\[ = 2\pi S_z^2 \quad \text{(A.12)} \]

\[ I = \sqrt{2\pi} S_z \quad \text{(A.13)} \]

This provides the final form of the 2D Sutton Model.

\[ C(x, y) = \frac{Q}{\sqrt{2\pi} S_y u} \exp \left( -\frac{y^2}{2S_y^2} \right) \quad \text{(A.14)} \]

\[ S_y = 0.5 C_y x^{(2-n)/2} \quad \text{(A.15)} \]
Appendix B

Penetrator Design and Descent Analysis

Enceladus’ plume has changed form over the many flybys Cassini has made past the moon. In fact, of the many models made to represent the plumes structure([32, 33, 67]), none have been able to replicate the Cassini measurements for any multiple flybys. Tidal stressing has been shown to play a role in the activation of the jets over time. Hurford et al. [74] came up with a model that predicted which part of the tiger stripes would be in tension and compression throughout Enceladus’ orbit of Saturn. Spitale and Porco [19] compared the sightings of the different jets to the stress in the jet’s region and confirmed that the jets were seen with much more regularity when the region was in tension then when it was in compression. The authors note a particular example of a case where Source I was in optimal viewing conditions and was unable to be located; the region of Source I (the source viewed most actively of all the sources) was in compression at the time.

A penetrator mission to Enceladus would consist of three separate phases. The first phases would be the traverse from Earth to Saturn, and would complete as the composite makes its flyby of Enceladus. Phases two would consist of the penetrator reducing its speed and altitude to appropriate levels, ending with the velocity entirely in the direction of the moon. Phases three consists of the penetrator VIFFing (vectoring in forward flight) where the penetrator is free-falling and can only change its
### Table B.1: Select Flyby Data

<table>
<thead>
<tr>
<th>Flyby Name</th>
<th>Date</th>
<th>Speed (km/s)</th>
<th>Altitude (km)</th>
</tr>
</thead>
<tbody>
<tr>
<td>E0</td>
<td>2005/02/17</td>
<td>6.7</td>
<td>1260</td>
</tr>
<tr>
<td>E1</td>
<td>2005/03/09</td>
<td>6.6</td>
<td>497</td>
</tr>
<tr>
<td>E2</td>
<td>2005/07/14</td>
<td>8.2</td>
<td>166</td>
</tr>
<tr>
<td>E7</td>
<td>2009/11/02</td>
<td>7.7</td>
<td>99</td>
</tr>
<tr>
<td>E9</td>
<td>2010/04/28</td>
<td>6.5</td>
<td>99</td>
</tr>
<tr>
<td>E10</td>
<td>2010/05/18</td>
<td>6.5</td>
<td>197</td>
</tr>
</tbody>
</table>

Shown are the speed and closest altitude of select Cassini encounters with Enceladus. [75]

velocity in the lateral directions. It is in this last phase where the penetrator makes its course corrections to locate the target the target jet.

This chapter shows initial mission analysis showing the design of the penetrator and the analysis of the final phase of the mission during free fall.

### B.1 Penetrator Design

The second phase of the penetrator mission occurs when the composite spacecraft including the main spacecraft and the penetrator are encountering Enceladus. Table B.1 shows select encounter data from Cassini fly-bys of the moon. From the data, it is conceivable that fly-by can be constructed for low-speed passing ($\approx 6.5$ km/s) at multiple altitudes.

In order for the mission to be successful, the penetrator must survive the impact into the moon’s icy surface. Even if the target is impacted and the penetrator plunges into one of the jets, there is no guarantee that the jet is perfectly vertical through to the liquid reservoir. Higher speed impacts result in greater depth of penetration, and are therefore preferable so long as the spacecraft electronics and instruments can survive. Generally, penetrators have been shown to be able to withstand the impulse
of a 300 m/s impact. Therefore, the initial mission study will assume an impact velocity of 300 m/s.

The depth of penetration is largely affected by the penetrator design as well as the impact speed. Young [76] developed penetration equations at Sandia National Laboratories. For penetration into ice the depth is described in Equation B.1.

\[ D = 0.000234SN(W/A)^{0.06}(V - 100)\ln(50 + 0.06W^2) \] (B.1)

The variable \( D \) is depth in ft, \( S(ice) = 4.5 \pm 0.25 \) describes the permeability of the surface, \( N = 0.25L_v/d + 0.56 \) describes the geometry of the nosecone, \( V \) is velocity in fps, \( W \) is the weight in pounds, and \( A \) is the impact area in square inches. For most nosecones, a 2.5:1 \( L_v:d \) ratio is expected. Figure B.1 shows the penetration depth for different masses and geometries of penetrators.

The purpose of using a penetrator as a lander for planetary exploration are two-fold. First, the vehicle penetrates into the subsurface during landing, which allows for measurements. Secondly, they can be designed to be light and cheap to deploy as a secondary payload. For this reason, a total mass budget of 50 kg is assumed for the penetrator mission. To determine what proportion of that weight is penetrator mass the mass fo the fuel must be computed. The Tsiolkovsky rocket equation shown below relates the mass fraction to the total \( \Delta v \) needed for the mission.

\[ \Delta v = -I_{sp}g_0 \ln \left( \frac{m_f}{m_0} \right) \] (B.2)

For initial analysis it is assumed that the flyby speed, 6.5 km/s, will be equal to the \( \Delta V \) required. For an efficient rocket, a specific impulse of 300 s can be obtained. A structure for fuel tanks will be required, and a mass about 5\% the fuel mass is assumed.
Figure B.1: Penetration Depths for Different Penetrator Designs: This chart shows the depth of penetration for different Weight/Diameter Combinations for a 300 m/s impact into ice.
\[ M_{frac} = 1 - \frac{1}{\exp\left(\frac{\Delta v}{I_{sp}g_0}\right)} \]  

(B.3)

\[ M_{fuel} = M_{frac} M_{tot} \]  

(B.4)

\[ M_{str} = 0.05 M_{fuel} \]  

(B.5)

\[ M_{pen} = M_{tot} - M_{fuel} - M_{str} \]  

(B.6)

For a 50 kg mission a penetrator would be 3.5 kg of the mass budget. If the penetrator has a diameter of 4 cm, penetration capability up to 4 meters can be realized with a nosecone with 2.5:1 or higher \( L_n : d \) ratio. It is noted that although length is not mentioned here because it does not play a significant role in the penetration depth, it will have be considered later for launch loads, etc.

### B.2 Mission Analysis

The final phase of the mission is the free fall descent of the penetrator. Throughout the free fall portion of the mission the gravity accelerating the penetrator towards the moon is:

\[ \ddot{r} = \frac{\mu \dot{r}}{|r|^3} \]  

(B.7)

Using conservation of energy, the initial downward velocity for free fall starting at a set altitude can be determined such that the impact occurs at 300 m/s.

\[ \frac{V_0^2}{2} + \frac{\mu E}{r_0} = \frac{V_f^2}{2} + \frac{\mu E}{r_E} \]  

(B.8)

The variable \( V_f \) is the final velocity of 300 m/s, \( r_E \) is the radius of Enceladus at
Figure B.2: Initial Altitude vs Velocity: The plot shows the combination of initial altitude and speed which result in an impact velocity of 300 m/s.

Figure B.3: Initial Altitude vs Time: The plot shows the combination of initial altitude the time in free fall given that the impact velocity will be 300 m/s.
252.1 km, $r_0$ is the initial radius $r_0 = h + r_E$. Figures B.2 show the combinations of initial altitude and velocity of the free fall portion of the mission from Equation B.8. Figure B.3 shows the time in free fall for the different initial altitudes and corresponding initial speed. For a 200 km initial altitude, a 12 minute free fall would occur.
Bibliography


