Detection and Localization of Aerosol Releases from Sparse Sensor Measurements

by

Emily Beth Fox

Submitted to the Department of Electrical Engineering and Computer Science

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

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2005

Author .....
Department of Electrical Engineering and Computer Science
April 18, 2005

Certified by .....
Jerome J. Braun
MIT Lincoln Laboratory Technical Staff
Thesis Supervisor

Certified by .....
John W. Fisher III
Principal Research Scientist
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

This work was sponsored by the National Science Foundation under Grant No. 0329901 under Air Force Contract No. FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

Detection and Localization of Aerosol Releases from Sparse Sensor Measurements

by

Emily Beth Fox

Submitted to the Department of Electrical Engineering and Computer Science
on April 18, 2005, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

In this thesis we focus on addressing two aspects pertinent to biological release detection. The first is that of detecting and localizing an aerosolized particle release using a sparse array of sensors. The problem is challenging for several reasons. It is often the case that sensors are costly and consequently only a sparse deployment is possible. Additionally, while dynamic models can be formulated in many environmental conditions, the underlying model parameters may not be precisely known. The combination of these two issues impacts the effectiveness of inference approaches. We restrict ourselves to propagation models consisting of diffusion plus transport according to a Gaussian puff model. We derive optimal inference algorithms utilizing sparse sensor measurements, provided the model parametrization is known precisely. The primary assumptions are that the mean wind field is deterministically known and that the Gaussian puff model is valid. Under these assumptions, we characterize the change in performance of detection, time-to-detection and localization as a function of the number of sensors. We then examine some performance impacts when the underlying dynamical model deviates from the assumed model.

In addition to detecting an abrupt change in particles in an environment, it is also important to be able to classify the releases as not all contaminants are of interest. For this reason, the second aspect of addressed is feature extraction, a stage where sensor measurements are reduced to a set of pertinent features that can be used as an input to the classifier. Shift invariance of the feature set is critical and thus the Dual Tree Complex Wavelet Transform (DT CWT) is proposed as the wavelet feature domain.

Thesis Supervisor: Jerome J. Braun
Title: MIT Lincoln Laboratory Technical Staff

Thesis Supervisor: John W. Fisher III
Title: Principal Research Scientist

Acknowledgments

I would like to thank Dr. John W. Fisher for his tireless efforts in supervising of my thesis. His encouragement, patience, and ideas have been invaluable in my research.

I would also like to acknowledge Dr. Jerome Braun whose constructive criticism and advice were pivotal in shaping my work at Lincoln Laboratory.

In addition, I am deeply indebted to Professor Alan S. Willsky for his guidance throughout the course of this research, especially in regards to the abrupt change detection material. His extensive wealth of knowledge and enthusiasm for research problems, in addition to an extreme dedication to students, was of great importance to me.

Both Professor Kevin Amaratunga and Professor Gilbert Strang went well above and beyond their duties in helping me with the wavelet section of this thesis while I was taking their course on the material and I would like to thank them for their efforts. Likewise, Dr. Nick Kingsbury provided me with my introduction to wavelets back in 2002 and has been a continual source of help, support, and advice in this area.

Finally, this work would be in a much different state without the help I received from other students, especially Jason Williams and Alex Ihler. I am amazed and inspired by their selflessness in guiding and teaching younger students such as myself.

Contents

List of Figures

Chapter 1

Introduction

Inferring the state of an environment from a sparse set of imprecise sensor measurements is essential to many applications in a variety of fields. Questions arise as to how a distributed sensor network should be configured to monitor the given environment as well as how the sensor measurements should be combined in a meaningful and informative manner. Biological sensing is one of the areas in which these questions are particularly acute. There are a variety of reasons why detection and localization of biological releases are of interest ranging from biodefense applications, both civilian and military, to environmental monitoring in industrial settings. It is often the case that rapid detection is an important consideration of such an application.

Due to the cost, size, and other logistical issues, it frequently the case that only a sparse set of biological and particle sensors are deployed to monitor a given area. In addition, these sensors are sensitive to small local regions and as such yield only a myopic view of the environment when used in isolation. These issues complicate the monitoring problem and impact the ability to rapidly detect releases. However, in combination with a dynamical model, it is possible to combine measurements from several sensors with non-overlapping fields of view so as to improve detection, time-to-detection, and localization beyond that obtained using a single sensor. A fundamental property of this system is that the complex dynamics operate on a small scale while observations are only available at a coarse, macroscopic scale. This is similar to the types of problems faced in a wide variety of fields as different as meteorology and

epidemiology.

Detecting and localizing a biological release requires fusing information both temporally and spatially from multiple collocated and distributed dissimilar sensors including biological sensors, particle counters, and anemometers. In many biological sensing applications, detection and time-to-detection of releases are of primary importance, and localization is secondary. The goal of the sensor network, then, is to intelligently combine sensor measurements to first detect the release in a timely manner and secondarily to localize the release in space and time. The two goals are distinct in that reasonable detection may be obtained from early measurements while localization generally relies on measurements aggregated over a longer time period. Bayesian filtering approaches provide a natural framework for such problems given an appropriate state definition and model of system dynamics.

In this thesis, we examine two aspects pertinent to the biological release detection problem. The first topic analyzed is that of detecting and localizing aerosol releases using Bayesian filtering techniques. We restrict ourselves to scenarios where particle propagation is reasonably well modeled by diffusion plus transport. We derive optimal inference algorithms for the case where the model is perfectly known, characterizing performance for a set of sensor configurations with regard to probability of detection P_d, time to detection T_d, and localization in time and space. This is accomplished via a series of controlled experiments using synthetic data. We characterize both the improvement of estimation performance attained by combining multiple sensor measurements as well as the sensitivity in performance to mismatched models.

A classification stage is needed to aggregate various sensor measurements and determine the type of release both in terms of material and various other properties. This stage also provides redundancy in release detection. For this reason, the second aspect investigated in this thesis is that of feature selection and finding an appropriate basis which contains the sensor measurement information crucial to event classification. A goal of the feature extraction stage is to reduce the dimensionality of data presented to the classifier which is then trained to differentiate between background and various events of interest. The focus of this thesis is on extracting

information from the particle concentration measurements. We investigate the utility of the wavelet transform for both signal conditioning (i.e. denoising) and as a feature basis.

In this application, it is important to have the feature set associated with a given signal be the same or similar to that of the signal shifted in time so that both signals are classified in the same way. The Dual Tree Complex Wavelet Transform (DT CWT) [16, 17] provides approximate shift invariance. A denoising technique combining the DT CWT and Donoho's wavelet shrinkage [9, 8] is developed and compared with other denoising techniques. The proposed denoising method indicates significant benefits in terms of shift invariance and computational complexity as compared to the other methods. The DT CWT is then used as a feature basis for representing particle concentration time histories. A preliminary set of features are proposed. Although the denoising technique is developed as a separate unit, because both the signal conditioning stage and feature extraction stage rely on the same transformation, both can be done in a combined fashion.

By considering only the particle concentration measurements, the feature extraction and Bayesian filtering algorithms developed in this thesis aid in identifying the time and location of an aerosol release, but not in determining the type of material. Because not every contaminant is of concern, it is important to be able to discriminate between release materials. Thus, a classifier incorporates features from other sensors such as biological sensors. However, the design of a classifier is beyond the scope of this thesis.

The filter provides information about the time and location of releases while a classifier identifies and distinguishes between types of releases. A decision stage is necessary in order to fuse the information provided from the Bayesian filter with that from the classifier, but this issue is not addressed in this thesis and is a topic of future research.

Chapter 2 provides a summary of past research that has been conducted in the areas of multisensor data fusion as applied to the chemical and biological defense problem, wavelet features for discrimination tasks, as well as Bayesian Filtering approaches

to similar applications. In Chapter 3, a potential laboratory setup is described and the measurements from this setup are taken as the sensor outputs throughout the rest of the thesis. Release detection and localization algorithms using Bayesian filtering are developed in Chapter 4 while the issue of determining shift invariant wavelet features from time series measurements for classification purposes is explored in Chapter 5. Finally, Chapter 6 provides a discussion of the results presented and outlines potential future work.

Chapter 2

Background

2.1 Multisensor Data Fusion with Biological Sensors

Multisensor data fusion is a large field encompassing applications where disparate and overlapping or congruent information is combined in a manner so as to provide more information about the system.

2.1.1 Biological and chemical detection

Several researchers have addressed issues related specifically to biological or chemical release detection. Braun et. al. [4, 5, 3] present multisensor fusion and machine learning approaches to such detection problems. In particular, aerosol anomaly detection for subway biodefense is presented in [4, 5]. The overall architecture involves signal conditioning, feature extraction, Principal Component Analysis (PCA) analysis, and a machine learning stage. The signal conditioning is performed using a 12th order median filter. Among the features investigated were: the peak amplitude per particle size-band in a given time window, the standard deviations for each size-band, and the coefficients of a fitting polynomial. After performing PCA as a means to reduce dimensionality, the feature vector was passed to a set of two 3-layer feedforward MLP neural networks to perform the classification. Because the performance of the classi-

fier is dependent upon the quality of the selected features, in this thesis we investigate other features pertinent to bioaerosol detection and develop better signal conditioning by using wavelet transforms.

2.2 Feature Extraction Using Wavelets

By providing both time and frequency information about a signal, the wavelet transform domain forms a suitable basis in which to select features from time-varying measurements. In addition to providing a feature basis, the wavelet transform allows for signal conditioning.

2.2.1 Wavelet signal conditioning prior to feature selection

Conditioning a signal prior to feature selection is an important yet complicated step. By filtering the signal, one must insure that the critical information is not lost through oversmoothing, etc. Denoising a signal by transforming it to the wavelet domain and then thresholding coefficients eliminates noise while still preserving the frequency components and signal characteristics to a certain degree. After deciding on a threshold, there is the choice between hard and soft elimination of coefficients. Hard thresholding involves removing all coefficients below the threshold while soft thresholding tapers smaller coefficients to zero. The choice between soft and hard thresholding will have an impact on the quality of the features and the performance of the classifier. The theory behind various wavelet denoising techniques is outlined by Donoho [9, 8] and Birge and Massart [2].

Farooq and Datta [10] explore the differences between hard and soft thresholding denoising methods as a pre-processing stage before feature extraction and Automatic Speech Recognition (ASR) stages. Although both denoising techniques proved to be very helpful in the classification performance, word recognition based off of hard thresholded coefficients was better in a high SNR situation while soft thresholding was better at a low SNR. These results are specific for phenome recognition and similar studies would have to be conducted in the biological sensor network application.

2.2.2 Wavelets as a feature basis

With or without signal conditioning, the wavelet coefficients may be used as features. Yen and Lin [30] conduct a thorough analysis of wavelet packet feature extraction as a method for vibration monitoring. The framework involves performing a seven-level wavelet packet decomposition (WPD) for each signal segment, computing robust feature measures, selecting a subset of the features, and then passing the features as the input to a neural network. In order to overcome the translation dependent nature of the WPD, Yen and Lin calculate the wavelet packet node energy (i.e. the sum of the squared wavelet coefficients in a given subband). Although not perfectly shift invariant, this representation provides a more robust feature for classification. Discriminant analysis is then used to identify a robust set of features from the set of transform domain energies. The paper claims that although the Karhunen-Loeve (KL) expansion is optimal in terms of signal representation, it is not optimal in terms of class separability. It argues that mean-square error should not be the primary criterion in reducing the feature dimension, but rather classification accuracy which improves with class separability. Using Linear Discriminant Analysis (LDA), the discriminate power of each feature was examined and those containing little class separation information were discarded. The WPD based features yielded nearly 100% correct classification from the neural network. The idea of finding a best basis for phoneme recognition is also explored by Long and Datta [18]. Specifically, the Best-Basis Algorithm of Coifman and Wickerhauser [7] is applied to the problem of speech recognition. Intrator et. al. [12] also examine the issue of best basis pursuits in discrimination tasks using wavelet features.

In the application of biological sensor networks, a single-tree expansion could be used instead of a WPD if the majority of the pertinent information is held in the lower frequency subbands. This is explored in Chapter 5 Section 5.3.4. In addition, because the shift invariance of the subband energies of the Dual Tree Complex Wavelet Transform (DT CWT) exceeds that of a standard DWT [17], the DT CWT is explored as an alternative transform method. However, the idea of best basis pursuit is not

examined in this thesis because it is classifier dependent and the design of the classifier is beyond the scope of this thesis. Thus, best basis pursuit is a topic for future work.

Dual Tree Complex Wavelet Transform

The Discrete Wavelet Transform (DWT) is the most general and widely used wavelet transform algorithm. However, although the DWT is a good approach in terms of representation of the signal, it is not a shift invariant transform. Various strictly shift invariant transforms have been proposed, but at the cost of high redundancy and computational complexity. Kingsbury proposes what is called the Dual Tree Complex Wavelet Transform (DT CWT) as an *approximately* shift invariant transform with only limited redundancy and perfect reconstruction [17]. For this application, the large data sets make low redundancy an important issue.

Neumann and Steidl [22] explore the DT CWT as a feature basis for signal classification using a hard margin Support Vector Machine. If one wants to use real filter banks in the time domain, the DT CWT requires the use of specially designed filters which may not be best suited for a given application. Thus, Neumann and Steidl examine the DT CWT in the frequency domain where standard wavelet filters can be used. The classification performance of the various filters is then compared. Once again, this topic relies on the classifier and could be an area of future research.

2.3 Bayesian Filtering for Dynamic State Estimation

Given an appropriate model, state estimation can play an important role in the detection of aerosolized releases and can supplement the information about the system provided by the types of features described above. However, Bayesian filtering requires the modeling of both system dynamics and observational errors. Airflow and particle dispersion are challenging to accurately model from sparse measurements and the uncertainty model is difficult to obtain because there is no ground truth.

Typically, Kalman filtering [15] and other Bayesian filtering techniques have been applied to problems of state estimation in the tracking community, and some of these methods have been known since the 1960s. Only recently have Bayesian filtering techniques been applied to more imprecise fields such as weather modeling. Weather prediction is a significantly different application from indoor airflow modeling. However, the underlying measurements and lack of global knowledge is similar to both applications. In weather prediction balloons, buoys, and satellites provide information at only specific places and times, which is analogous to anemometer readings in the system investigated in this thesis. Mackenzie [19] reports the nonlinear dynamics of atmosphere creates a non-Gaussian model of observational errors. This implies that the standard Kalman filter must be modified to apply to weather prediction. Although many forecasters are experimenting with "ensemble forecasting", Miller suggests using a nonlinear filter [19]. While the conclusion is that more experimentation is needed in order to verify the applicability of Bayesian filtering to weather prediction, researchers in the field have made significant advances and shown promising results using these techniques.

Specific to chemical and biological detection, Nehorai et. al. [14, 31] have examined using a combination of maximum likelihood parameter estimation with the Generalized Likelihood Ratio (GLR) test to create a binary hypothesis testing problem of detecting a release from a moving source. This is done in a non-recursive fashion and the state dimension grows with time and spacial extent of the region being monitored. To aid in reducing computational complexity, a parametric moving path model is used. Similar work by Nehorai et. al. can be found in [13, 21].

In addition, researchers have considered other Bayesian approaches. Release detection in hyperspectral imagery has been considered by O'Donnell [24] exploiting properties of isolated pixel intensities and Nofsinger [23] uses ensemble Kalman filtering approaches for detecting chemical plume. Broad coverage of abrupt change detection and the GLR is covered by Basseville in [1].

Chapter 3

Application Details

3.1 Laboratory Setup

The algorithms proposed in this thesis are *not* constrained to a particular laboratory setup, however, for the sake of completeness a possible laboratory setup is addressed here. Conforming with the cost of biological and other sensors in general, the proposed setup consists of a number of clusters of sensors dispersed sparsely within an enclosed area. These multiple sensor clusters are often different from one another and are categorized as follows:

- Primary - consisting of many sensors, both sophisticated and simple, such as biological sensors, particle counters, and anemometers.
- Secondary - consisting of fewer sensors which provide measurements that are required to be available at more locations throughout the region (e.g. anemometer readings).

Figure 3-1 shows an arrangement or primary and secondary clusters.

3.2 Measurements Available

In order to accurately determine whether a biological release occurred, it is essential to have information about particle density and the biological nature of the particles.

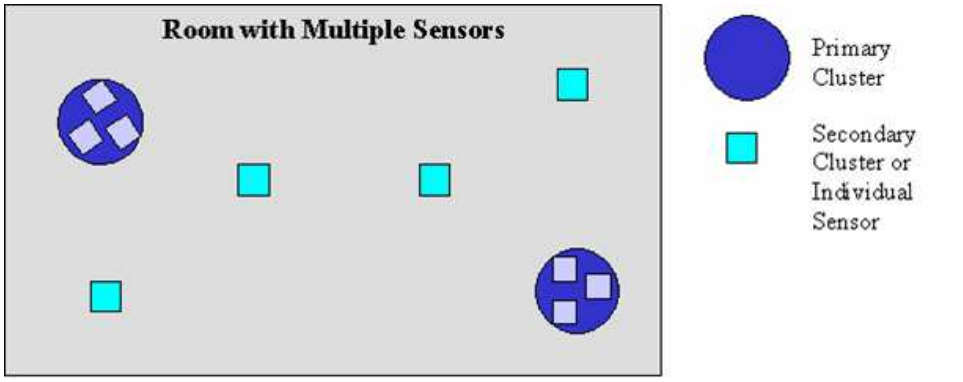


Figure 3-1: Example of possible room setup with two primary clusters and four secondary clusters. The secondary clusters may consist only of anemometers to measure wind velocity, or they may also contain a particle counter. Primary clusters will provide information about the biological content of the air.

This information is obtained by using commercially available particle counters that measure particle density based on size and biological sensors that determine whether there is biological matter in its locality by testing, for example, the fluorescence properties of the particles.

To connect the information provided by widely spaced primary clusters, one must have information about the wind field in addition to the particle concentration at each location. An array of anemometers provide information about the local x, y, and z components of wind velocity as well as temperature and humidity readings.

This setup provides us with sparse, discrete measurements of particle densities, airflow, and the biological nature of the particles. The goal of the project is to fuse this information in an appropriate manner so that a decision can be made about the state of the entire region at all times.

In this thesis, we will restrict ourselves to the measurements available from such a laboratory setup.

Chapter 4

Generalized Likelihood Ratio Test

4.1 Problem Formulation

The problem of detecting a localized release is illustrated in Figure 4-1. Particle sensors, indicated by black circles, are distributed sparsely over a region of interest. In this context, sparsity of the sensor configuration refers specifically to the coarse spacing of the sensors relative to the spatial extent of the release at time t_o. Particle concentrations at time $t > t_o$ are indicated by shaded contours. In a sparse sensor network, it will often be the case that there is a delay in observability of the event. That is to say that a release at time t_o may not be immediately registered by any of the sensors. The goal of the sensor network, then, is to intelligently combine sensor measurements distributed both temporally and spatially to first detect the release in a timely manner and secondarily to localize the release in space and time. The two goals are distinct in that reasonable detection may be obtained from early measurements while localization in both space and time generally relies on measurements aggregated over a longer time period.

In this thesis we specifically consider scenarios in which the dispersion of particles in the environment is well modeled by diffusion plus transport due to the presence of a wind field. With an appropriate definition of a state space, Bayesian filtering approaches provide a framework for such problems, presuming that the model parameters are available and inter-sensor communication is possible. In our particular

application it will be necessary to modify the standard equations to include a term which represents a localized event in time and space (i.e. the release) whose effects persist over time. The specific choice enables us to formulate the problem as a hybrid filtering and hypothesis testing problem resulting in certain efficiencies.

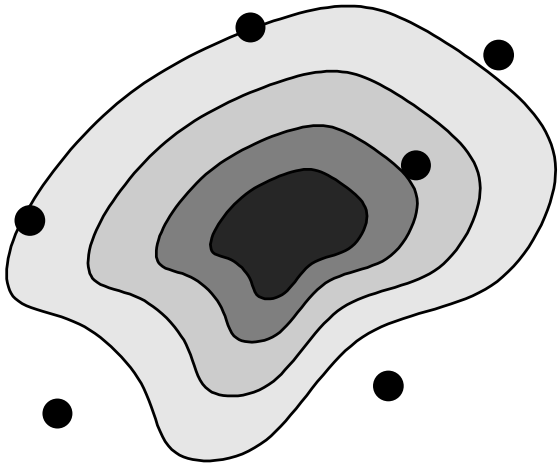


Figure 4-1: Sparse sensors, indicated by • symbols. Contours indicate particle density following a localized release at a single time instant.

Given a sparse sensor configuration and uncertainty regarding the underlying dynamics (e.g. rate of diffusion, knowledge of the wind field) it will generally be the case that the optimal inference approaches under an assumed parametrization will degrade in both detection and localization performance as compared to the matched case. Consequently, two primary issues that we will examine are:

- The utility of using a dynamical model for this multisensor application.
- Sensitivity of the inference procedure to a mismatch in the parametrization of the model.

In this chapter we examine these questions in some detail in the context of a specific dynamical model and a specific set of sensor configurations. Our results indicate that optimal inference on mismatched models yields reasonable detection performance at the cost of precise localization. Additionally, the methodology presented provides a framework within which to study these questions in more detail.

4.1.1 Adopting a Bayesian approach

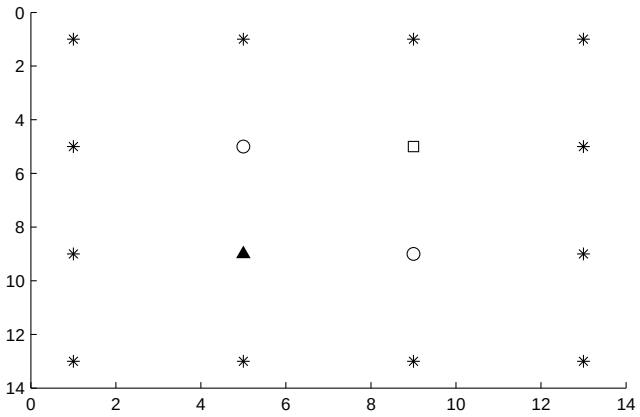


Figure 4-2: Sensor configurations for 1 (▲), 2 (▲ + □), 4 (▲ + □ + ∘), and 16 (▲ + ∘ + □ + ∗) sensors.

Figure 4-2 depicts the set of sensor configurations, consisting of 1, 2, 4, and 16 sensors, that we will consider in our experimental section. The sensors can be seen to lie on a regular grid although the approach and analysis do not require such regular configurations. The symmetry of the configurations does serve to simplify the presentation of some of the results. For our purposes, all releases shall occur in the center of the 16 sensor configuration.

Defining a state space and associated dynamical model enables the use of a Bayesian filtering approach. We divide the area of interest into a set of small regions or cells. The state of the system at time t is then the mean particle concentrations over one time step of the set of cells. The granularity of the cells is at a substantially finer resolution than the sensor placement. Specific settings will be discussed in Section 4.2.

4.1.2 Dynamic and measurement models

The standard Bayesian filtering problem, whether linear or nonlinear, Gaussian or non-Gaussian involves two main stages: prediction and update. In the prediction stage the dynamical model is used to predict the state distribution one time-step in

the future. In the update stage, current measurements are fused to update the state distribution. In order to formulate the filter equations, one must characterize the dynamical model, measurement model, process noise, and measurement noise.

Gaussian puff model

Accurately describing the spatial and temporal distribution of a contaminant released into an environment is a challenging problem because of the inherent randomness of the wind velocities in turbulent flow. Turbulent diffusion describes the behavior particles are supposed to follow within an airflow and often these particles can just be considered elements of the air itself. The term "turbulent diffusion" is commonly applied to the scenario of particles or gases dispersing in turbulent flow and is not the same as ordinary molecular diffusion. Rather, the concentration of a species, c, satisfies the continuity equation (see Equation 4.1. The derivations shown throughout this section can be found in [26].

$$\frac{\partial c}{\partial t} + \frac{\partial}{\partial x_j}(u_j c) = D\frac{\partial^2 c}{\partial x_j^2} + R(c,T) + S(\mathbf{x},t) \tag{4.1}$$

where u_j is the jth component of the wind velocity, D is the molecular diffusion of the species, R is the rate of generation of species by chemical reaction and is dependent on the temperature T, and S is the rate of addition of species at location $\mathbf{x} = (x_1, x_2)$ and time t.

In a perfectly known wind field which represents the wind velocities at all locations, there would not be any turbulent diffusion. Instead, the process would simply follow the advection equation,

$$\frac{\partial c}{\partial t} + \frac{\partial}{\partial x_j}(u_j c) = 0 \tag{4.2}$$

However, due to the randomness of the wind velocities, we can represent u as the sum of a deterministic and stochastic component, $\bar{u} + u'$. Because u' is a random variable and drives the particle concentration, the resulting concentration c can similarly be

divided into the sum $< c > +c'$ where the $< . >$ indicates averaging. The u' and c' terms are zero mean by definition so $< c' >= 0$. Substituting these variables into equation 4.2 yields,

$$\frac{\partial\langle c\rangle}{\partial t} + \frac{\partial}{\partial x_j}(\bar{u}_j\langle c\rangle) + \frac{\partial}{\partial x_j}(u_j'c') \;=\; D\frac{\partial^2\langle c\rangle}{\partial x_j^2} + R(\langle c\rangle + c', T) + S(\mathbf{x}, t) \qquad (4.3)$$

The most common method for relating the turbulent flux $\langle u_j'c'\rangle$ is based on what is called *mixing-length theory* or *K theory* (see [26]) and results in the following equation,

$$\langle u_j'c'\rangle \;=\; -\sum_k K_{jk}\frac{\partial\langle c\rangle}{\partial x_k} \qquad (4.4)$$

where K_{jk} is the *eddy diffusivity*. When the coordinate axes are aligned with the principle axes of the eddy diffusivity tensor $\{K_{jk}\}$, then the only nonzero elements are the K_{jj}. In this scenario, equation 4.4 becomes,

$$\langle u_j'c'\rangle \;=\; -K_{jj}\frac{\partial\langle c\rangle}{\partial x_j} \qquad (4.5)$$

Furthermore, it is common to invoke three assumptions:

1. A chemically inert species,

$$R = 0$$

2. Negligible molecular diffusion relative to turbulent diffusion,

$$\tfrac{\partial^2\langle c\rangle}{\partial x_j^2} \ll \tfrac{\partial}{\partial x_j}\langle u_j'c'\rangle$$

3. An incompressible atmosphere,

$$\tfrac{\partial\bar{u}_j}{\partial x_j} = 0$$

With the above assumptions and assuming Equation 4.5 holds, the atmospheric diffusion equation becomes,

$$\frac{\partial\langle c\rangle}{\partial t} + \bar{u}_j\frac{\partial}{\partial x_j}(\langle c\rangle) \;=\; \frac{\partial}{\partial x_j}\left(K_{jj}\frac{\partial\langle c\rangle}{\partial x_j}\right) + S(\mathbf{x}, t) \qquad (4.6)$$

Another convention is to represent a rapid release of material as an instantaneous point source. This idealization simplifies the solution of the partial differential equation. The assumptions are as follows:

1. $\langle c(x,y,0)\rangle = c_o\delta(x)\delta(y)$

2. $\langle c(x,y,t)\rangle = 0$ as $x,y,z \to \pm\infty$

Using all the preceding assumptions and defining the mean and initial concentrations as $\langle c(x,y,t)\rangle = c_x(x,t)c_y(y,t)$, $c_x(x,0) = c_o^{1/2}\delta(x)$, and $c_y(y,0) = c_o^{1/2}\delta(y)$, the set of atmospheric diffusion equations in 2D are,

$$\frac{\partial c_x}{\partial t} + \bar{u}\frac{\partial c_x}{\partial x} = K_{xx}\frac{\partial^2 c_x}{\partial x^2} \tag{4.7}$$

$$\frac{\partial c_y}{\partial t} + \bar{u}\frac{\partial c_y}{\partial y} = K_{yy}\frac{\partial^2 c_y}{\partial y^2} \tag{4.8}$$

The differential equations can be solved by using the Fourier transform. The derivation is shown below.

$$\begin{aligned} C(s,t) &= \mathcal{F}\{c_x(x,t)\} = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} c_x(x,t)e^{-jsx}dx \\ \frac{\partial C}{\partial t} + js\bar{u}C &= -s^2K_{xx}C \\ C(s,0) &= \frac{c_o^{\frac{1}{2}}}{\sqrt{2\pi}} \\ C(s,t) &= \frac{c_o^{\frac{1}{2}}}{\sqrt{2\pi}}e^{-(s^2K_{xx}+js\bar{u})t} \end{aligned}$$

$$
\begin{aligned}
c_x(x,t) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} C(s,t) e^{jsx} ds \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-(s^2 K_x x t - js(x-\bar{u}t))} \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\{\left(s(K_{xx}t)^{1/2} - \frac{j(x-\bar{u}t)^2}{2(K_{xx}t)^{1/2}}\right)^2 - \frac{(x-\bar{u}t)^2}{4K_{xx}t}\}} \\
&= \frac{c_o^{\frac{1}{2}}}{2(\pi K_{xx} t)^{\frac{1}{2}}} e^{\frac{-(x-\bar{u}t)^2}{4K_{xx}t}}
\end{aligned}
$$

The same solution holds for computing the concentration in the y-direction, replacing x with y, $\bar{u}$ with $\bar{v}$ and K_{xx} with K_{yy}. This results in the following equation for computing the mean concentration of a species at a position (x, y) after a time t given an initial concentration of c_o,

$$
< c(x,y,t) > = \frac{c_o}{4\pi t (K_{xx} K_{yy})^{\frac{1}{2}}} e^{-\frac{(x-\bar{u}t)^2}{4K_{xx}t} - \frac{(y-\bar{v}t)^2}{4K_{yy}t}} \tag{4.9}
$$

Because of the random nature of atmospheric motions, the distribution of a concentration can never be predicted with absolute certainty. Although there is an equation for turbulent diffusion, a single mathematical model cannot accurately compute concentrations over a wide range of conditions. The resulting Gaussian solution which involves diffusion plus transport by a wind field is valid in the situation where the release is embedded in a field in which turbulent eddies are small relative to the puff. See Figure 4-3(a).

Dynamic and measurement models

Equations 4.10 and 4.11 describe the dynamic model in the case where the mean wind field (the primary cause for transport) is known. In practice this quantity must be estimated from, for example, a set of separate anemometer measurements. This estimation process itself represents some complexity in determining how to fuse the wind field measurements. Furthermore, uncertainty in the wind field results in

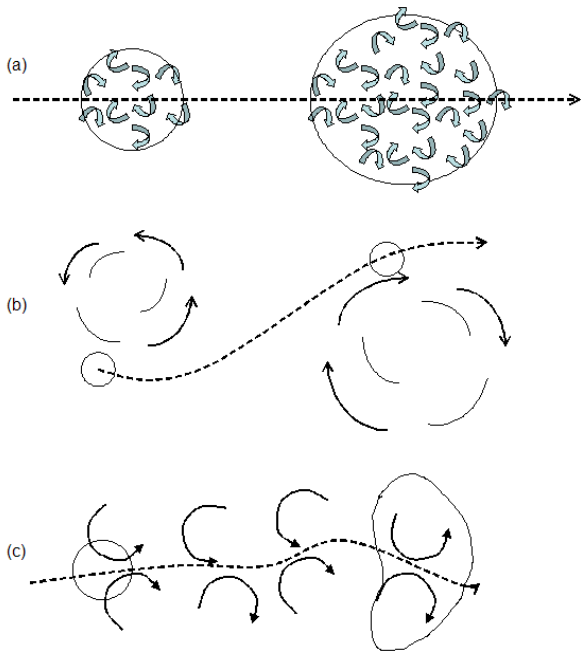


Figure 4-3: Dispersion of a puff of material when: (a) turbulent eddies are smaller than the puff, (b) turbulent eddies are larger than the puff, and (c) turbulent eddies are of comparable size to the puff.

a nonlinear dynamical system. In this chapter we develop the inference procedure assuming the mean wind field is known at each time step. In Section 4.2 we relax this assumption and investigate the impact of an incorrect mean wind field. The primary advantage of presuming a known wind field is that the dynamical model remains linear allowing for a computationally efficient inference procedure. At issue, addressed empirically, is the impact on estimation performance when this assumption is incorrect. More specifically, we address how much deviation from the assumed wind field in terms of bias and variance can be tolerated and still yield acceptable estimation performance.

The propagation model is a point source Gaussian puff model (see [26]) in 2D, thus there is no reflection or absorption of particles. Effectively, the formula results in Gaussian diffusion, the variance of which is determined by a diffusion coefficient, biased by the mean wind field. This Gaussian kernel is formed for every cell based on the given wind vector and distributes the particles accordingly. The total mean particle concentration mapping, $A(t; \mathcal{V}_w)$, is the the sum of the new concentration

maps from each cell. Note that this formulation still allows for time-varying wind fields. The system equations for the ambient case (i.e. no release) are given by:

$$x(t+1) = A(t;\mathcal{V}_w)x(t) + w(t) \tag{4.10}$$

$$y(t) = C(t)x(t) + v(t) \tag{4.11}$$

where $x(t)$ is the vector of cell particle concentrations, $A(t;\mathcal{V}_w)$ is the matrix of biased diffusion coefficients parameterized by the wind field $\mathcal{V}_w$, $w(t)$ is the process noise, $v(t)$ is the measurement noise, and $C(t)$ is a matrix relating the state to the sensor measurements $y(t)$. The process noise and measurement noise are assumed to be Gaussian and independent of each other over all time.

4.1.3 Modeling localized releases

While Equations 4.10 and 4.11 capture the ambient background conditions, they are not sufficient for reliably modeling a localized release. Releases are incorporated as an abrupt change to the dynamical system as in [29]. The advantage to this approach is that it allows one to simultaneously model a release as a localized event in time and space while capturing the persistent effects of the release over time. As we shall see, this leads to an efficient hypothesis testing approach. The equations presented in this section were rederived from [29].

Equations 4.10 and 4.11 are modified to account for an abrupt state change (a release in cell i at time ϕ) as follows:

$$x(t+1) = A(t)x(t) + w(t) + \beta f_i(t,\phi) \tag{4.12}$$

where $f_i(t,\phi)$ represents an impulse of particles in cell i at time ϕ, while β represents the strength of the release (e.g. number or volume of particles).

The assumed linearity of the system allows us to decompose Equations 4.12 and 4.11 into two terms. The first term $x_o(t)$ is related to the background process while the second $\gamma_i(t,\phi)$ captures the persistent effect of the release $f_i(t,\phi)$. This decomposition

is expressed as:

$$x(t) = x_0(t) + \beta\gamma_i(t,\phi) \tag{4.13}$$

$$y(t) = C(t)x_0(t) + v(t) + \beta C(t)\gamma_i(t,\phi) \tag{4.14}$$

The utility of the decomposition is that the problem reduces to one of detection in correlated noise (i.e. $x_o(t)$) performed in parallel for each cell. The detection formulations are greatly simplified by whitening the data and calculating the signature $\rho_i(t,\phi)$ of $\gamma_i(t,\phi)$ once it has been passed through the whitening filter yielding. Because the model is linear and the noise processes are Gaussian, a Kalman filter can be used to whiten the data. In this case, it yields the innovations sequence:

$$\nu(t) = \nu_0(t) + \beta\rho_i(t,\phi) \tag{4.15}$$

Due to the additive form of the model the filter is designed assuming there is no abrupt change. The standard Kalman filtering equations are as follows:

$$\begin{aligned}
\hat{x}(t|t) &= A(t)\hat{x}(t|t-1) + K(t)\nu(t) \\
\hat{x}(t+1|t) &= A(t)\hat{x}(t|t) \\
\nu(t) &= y(t) - C(t)\hat{x}(t|t-1) \\
P(t+1|t) &= A(t)\big(P(t|t-1) - \\
&\quad K(t)C(t)P(t|t-1)\big)A^T(t) + Q(t) \\
V(t) &= C(t)P(t|t-1)C^T(t) + R(t) \\
K(t) &= P(t|t-1)C^T(t)V^{-1}(t)
\end{aligned}$$

Where, $P(t+1|t)$ is the error covariance at time $t+1$ given sensor measurements up to time t, $K(t)$ is the Kalman gain, and $V(t)$ is the covariance of the residual $\nu(t)$. The state estimate of the whole system at any time is then the sum of the estimate due to $x_o(t)$ ($\hat{x}_0(t|t)$ and $\hat{x}_0(t|t-1)$), and the signature of $\gamma_i(t,\phi)$ ($\epsilon_i(t,\phi)$ and $\mu_i(t,\phi)$).

The Kalman filter state estimate can be decomposed similarly:

$$\hat{x}(t|t) = \hat{x}_0(t|t) + \epsilon_i(t,\phi) \tag{4.16}$$

$$\hat{x}(t|t-1) = \hat{x}_0(t|t-1) + \mu_i(t,\phi) \tag{4.17}$$

Taking $\beta = 1$, the state of $\gamma_i(t,\phi)$ evolves as in equation 4.18. The resulting estimates of the deterministic unknown $\gamma_i(t,\phi)$ are given by Equations 4.20-4.22.

$$\gamma_i(t+1) = A(t)\gamma_i(t) + f_i(t,\phi) \tag{4.18}$$

$$y_\gamma(t) = C(t)\gamma_i(t,\phi) \tag{4.19}$$

$$\mu_i(t+1,\phi) = A(t)\epsilon_i(t,\phi) \tag{4.20}$$

$$\begin{aligned}\epsilon_i(t,\phi) &= \mu_i(t,\phi) + K(t)[y_\gamma(t) - C(t)\mu_i(t,\phi)] \\ &= \mu_i(t,\phi) + \\ &\quad K(t)C(t)[\gamma_i(t,\phi) - \mu_i(t,\phi)]\end{aligned} \tag{4.21}$$

$$\begin{aligned}\rho_i(t,\phi) &= y_\gamma(t) - C(t)\mu_i(t,\phi) \\ &= C(t)[\gamma_i(t,\phi) - \mu_i(t,\phi)]\end{aligned} \tag{4.22}$$

where $\rho_i(t,\phi)$ is the desired signature of $\gamma_i(t,\phi)$ after it has been passed through the Kalman whitening filter.

4.1.4 Cell hypothesis testing framework

While the preceding framework has simplified the dynamical model we are still left with the task of testing various release hypotheses enumerated over space (cell index) and time. Additionally, the model is parameterized by the unknown release amount β. We address this issue using a Generalized Likelihood Ratio (GLR) approach allowing us to formulate and analyze a set of enumerated hypotheses which both detect and localize a release.

Referring to Equation 4.15 and assuming that β is known for the moment, we can construct an indexed set of hypotheses. The null hypothesis, H_0, indicates no release

while the alternative hypotheses, $H_{i,\phi}$, indicates an event in cell i occurring at time ϕ. Note that prior to time ϕ both the null and alternative hypotheses are identical.

$$H_0 \quad : \quad \nu(t) = \nu_0(t) \tag{4.23}$$

$$H_{i,\phi} \quad : \quad \nu(t) = \beta\rho_i(t,\phi) + \nu_0(t) \tag{4.24}$$

It is well known [28] that the sufficient statistics for this particular hypothesis test are the correlation between the signal $\rho_i(t,\phi)$ and the output $\nu(t)$,

$$d_i(t,\phi) \quad = \quad \sum_{\tau=\phi}^{t} \rho_i^T(\tau,\phi)V^{-1}(\tau)\nu(\tau) \tag{4.25}$$

and the energy of the signal

$$a_i(t,\phi) \quad = \quad \sum_{\tau=\phi}^{t} \rho_i^T(\tau,\phi)V^{-1}(\tau)\rho_i(\tau,\phi) \tag{4.26}$$

resulting in a log-likelihood of

$$l_i(t,\phi) \quad = \quad \beta d_i(t,\phi) - \frac{1}{2}\beta^2 a_i(t,\phi) \tag{4.27}$$

for the hypothesis $H_{i,\phi}$ versus H_0 given sensor measurements up to time t. The derivation of these standard results is shown Appendix A.

Under the assumption of there being one and only one release, the number of hypotheses grows linearly with time. The resulting hypothesis tree after three time steps is shown in Figure 4-4. Because of the growing size of the tree, in practice a sliding window is used to implement the inference algorithm. This is well justified by two related issues. The first is that it takes some time for releases to propagate to the nearest sensor. The second is that after a period of time, additional measurements do not contribute significantly to the probability of a detection in a given cell at a specific time. Therefore, there is little need to look further back in time. Specifically, the window enforces that only hypotheses of a release occurring a maximum of M

timesteps in the past are considered.

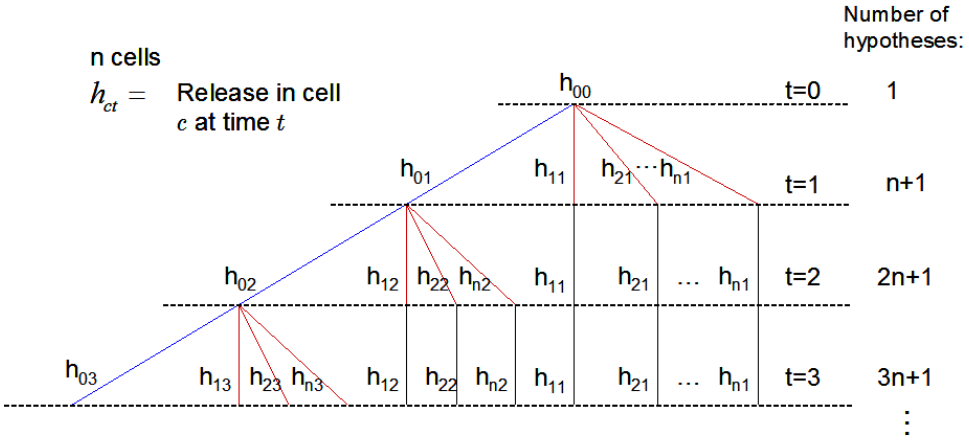


Figure 4-4: Hypothesis tree for a region with n cells at time step 3. Hypothesis h_{ct} represents a release in cell c at time t.

4.1.5 Estimating the release amount β

In a GLR approach [25, 28], one substitutes the Maximum-Likelihood esimate of β for each hypothesis. Conditioned on hypothesis $H_{i,\phi}$, $d_i(t,\phi)$ is Gaussian with mean $\beta a_i(t,\phi)$ and variance $a_i(t,\phi)$ (see Appendix A) giving an alternative expression for Equation 4.25 as

$$d_i(t,\phi) \quad = \quad \beta a_i(t,\phi) + \zeta_i(t,\phi) \tag{4.28}$$

where $\zeta_i(t,\phi)$ is a zero-mean Gaussian with variance $a_i(t,\phi)$. Consequently, the ML estimate of β is:

$$\hat{\beta}_i(t,\phi) \quad = \quad \frac{d_i(t,\phi)}{a_i(t,\phi)} \tag{4.29}$$

and the resulting log-likelihood ratio is:

$$l_i(t,\phi) \quad = \quad \hat{\beta}_i(t,\phi) d_i(t,\phi) - \frac{1}{2}\hat{\beta}_i(t,\phi)^2 a_i(t,\phi) \tag{4.30}$$

$$= \quad \frac{1}{2}\frac{d_i^2(t,\phi)}{a_i(t,\phi)} \tag{4.31}$$

If one takes $\hat{\phi}$ and $\hat{i}$ to be the values that maximize $l_i(t,\phi)$ and define

$$l_{\hat{i}}(t,\hat{\phi}) \triangleq \max_{i,\phi} l_i\,(t,\phi) \tag{4.32}$$

and denotes H_1 as the hypothesis that a release has occurred regardless of time or location, the likelihood ratio test for detection is simply

$$l_{\hat{i}}(t,\hat{\phi}) \;\; \gtrless \;\; \eta \tag{4.33}$$

where the value of η is set using Neyman-Pearson criterion to specify a fixed probability of false alarm, P_F.

4.1.6 Probability of detection

The probability distribution of an individual $l_i(t,\phi)$ conditioned on H_0, $p(l|H_0)$ is a chi-squared distribution with 1 degree of freedom (see Appendix A). Equations for the probability of false alarm and probability of detection (given a specific location, time, and amount) are

$$P_F = \int_{\eta}^{\infty} p(l|H_0)dl \tag{4.34}$$

$$P_D(\beta,\phi,i) = \int_{\eta}^{\infty} p(l|H_1,\beta,\phi,i)dl \tag{4.35}$$

In this case, hypotheses are being tested in parallel, so in practice due to the maximization function in Equation 4.32 the threshold is set empirically.

4.1.7 Bounds on probability of false alarm

Because the application considered in [29] is a M-ary hypothesis test rather than one of detecting *any* alternative hypothesis, there is no maximization over the likelihood ratio. In this section we extend the results presented in [29] by deriving bounds on the probability of false alarm in the case where we consider the detection problem described by Equation 4.32. With the likelihood ratio as defined in Equation 4.31,

the ratio test of Equation 4.33 can be rewritten as,

$$\max_{(i,\phi)} \left(\frac{|d_i(t,\phi)|}{\sqrt{a_i(t,\phi)}} \right) \gtrless \sqrt{\eta} \tag{4.36}$$

where $d_i(t,\phi)$ and $a_i(t,\phi)$ are as previously defined. Let $y_i(t,\phi) = \frac{d_i(t,\phi)}{\sqrt{a_i(t,\phi)}}$. The probability of false alarm, P_F, is given by,

$$\begin{aligned}
P_F &= Pr(\exists\{(y_i(t,\phi) > \sqrt{\eta}) \cup (-y_i(t,\phi) > \sqrt{\eta})\}|H_0) \\
&\leq Pr(\exists(y_i(t,\phi) > \sqrt{\eta})|H_0) + Pr(\exists(-y_i(t,\phi) > \sqrt{\eta})|H_0) \qquad (4.37) \\
P_F &\geq \max\{Pr(\exists(y_i(t,\phi) > \sqrt{\eta})|H_0), Pr(\exists(-y_i(t,\phi) > \sqrt{\eta})|H_0)\} \qquad (4.38) \\
\alpha \leq \; P_F &\leq \alpha + \delta \qquad (4.39)
\end{aligned}$$

where $\alpha = \max\{Pr(\exists y_i(t,\phi) > \sqrt{\eta}|H_0), Pr(\exists -y_i(t,\phi) > \sqrt{\eta}|H_0)\}$ and $\gamma = \min\{Pr(\exists y_i(t,\phi) > \sqrt{\eta}|H_0), Pr(\exists - y_i(t,\phi) > \sqrt{\eta}|H_0)\}$.

Each $d_i(t,\phi)$ has a Gaussian distribution, namely,

$$d_i(t,\phi) = \begin{cases} \mathcal{N}(0, a_i(t,\phi)) & : \quad H_0 \\ \mathcal{N}(\beta a_i(t,\phi), a_i(t,\phi)) & : \quad H_{i,\phi} \end{cases}$$

Because $a_i(t,\phi)$ is a deterministic quantity, $y_i(t,\phi)$ is also distributed as a Gaussian. Conditioned on the null hypothesis, the mean is given by $E[y_i(t,\phi)|H_0] = 0$. Similarly,

$$\begin{aligned}
y_i(t,\phi)y_j(t,\theta) &= \frac{d_i(t,\phi)d_j(t,\theta)}{\sqrt{a_i(t,\phi)a_j(t,\theta)}} \\
E[y_i(t,\phi)y_j(t,\theta)] &= \frac{a_{ij}(t,\phi,\theta)}{\sqrt{a_i(t,\phi)a_j(t,\theta)}} \\
a_{ij}(t,\phi,\theta) &= E[d_i(t,\phi)d_j(t,\theta)] = \sum_{\tau=\max(\phi,\theta)}^{t} \rho_i^T V^{-1} \rho_j \\
R &= cov(\underline{y}(t)) = [E[y_i(t,\phi)y_j(t,\theta)]]_{ij}
\end{aligned}$$

Thus, $\underline{y}$, conditioned on H_0, is distributed as $\mathcal{N}(\underline{0}, R)$.

Although the above bounds are valid, they are challenging to compute given that each $a_i(t, \phi)$ is unique to the wind field present between times ϕ and t. Therefore, the threshold used in this thesis is determined empirically through Monte Carlo simulations (sample size of 1500) of background conditions in the four and 16 sensor configurations.

4.1.8 Approximations

There are a few key approximations used to simplify the current problem. Firstly, in order to formulate this problem in terms of a Kalman filter, the mean wind field is taken as a known parameter so that the dynamic model is linear and both the dynamic uncertainty and measurement noise are assumed to be white Gaussian processes. Secondly, it is assumed that there is one and only one release. Also, this release is an impulse of particles at the center of a given cell. Another significant simplification is the fact that analysis of the environment is constrained to two dimensions rather than three.

The easiest alteration in these assumptions is to work in 3D. This would drastically increase the state dimension, but would be feasible with enough computational power.

Allowing a uniform release over some time frame rather than an impulse of particles would change the formulation by also having an unknown time at which the abrupt change ends. Having multiple releases of unknown and possibly different sizes would drastically increase the hypothesis space as well. Both of these relaxing of constraints would greatly complicate the formulation.

Because detection is the primary objective and localization in time and space is secondary, an appropriate question to ask is whether the simplified system (as outlined by the approximations above) can perform well in terms of detectability when the assumptions do not hold in the data but are still implemented in the inference procedure.

4.2 Empirical Results

We present a series of experiments in which we examine the utility of using multiple sensors. The first set of experiments utilize a known mean wind field, specifically no transport term, to characterize probability of detection, time to detection, and localization in time and space as a function of the number of sensors used. In the second set of experiments, a sequence of increasing wind fields are incorporated into the dynamical model, but not into the inference procedure. In the third set of experiments, noise is generated on the mean wind field provided to the simulation, but the inference procedure only uses mean wind field. The purpose of the second and third set of experiments is to examine the utility of the inference procedure as the model deviates from that assumed by the inference procedure. Detection probabilities for the 4 and 16 sensor configurations are examined and compared as a function of the wind field. In addition, localization performance is examined for these model mismatch scenarios.

4.2.1 Simulation parameters

In order to have symmetry and eliminate extraneous variables, we consider the case where the release is in the center of a room populated with sensors as given in Figure 4-2. The coordinates of the sensors shown in this figure are relative to a 13x13 cell area of interest. However, to reduce edge effects, the data is simulated over an area of 25x25 cells. A sensor is assumed to observe the particle concentration throughout the entire cell in which it is located with a measurement noise standard deviation of 10 particles per cell area.

The dynamics of the particle concentration in every cell is propagated using a Gaussian kernel with a diffusion coefficient of 0.5. The process noise standard deviation is taken to be 100 particles per cell area. For Sections 4.2.2 and 4.2.3, we restrict ourselves to a pure diffusion model in order to avoid the effects of wind which could blow the particles towards or away from the nearest sensor. In the case where there is wind, the utility of having multiple sensors to better cover the region becomes

apparent. In Section 4.2.4, simulations were run for increasing wind biases. However, the wind bias in the truth model was not incorporated into the inference model which allows us to analyze this model inaccuracy. In Section 4.2.5, simulations were run for the model mismatch case of temporally and spatially white noise on the mean wind field of the simulation data, while the inference model only used the known mean wind field.

For every release amount and sensor configuration scenario, 100 Monte Carlo simulations were performed.

The size of the sliding window, M, was determined to be 14 time steps by analyzing the minimum time from the onset of the release by which the accrual of additional information was insignificant in the case of pure diffusion. In a sensor network which is distributed as a regular grid, like in the 16 sensor configuration used in this thesis, the pure diffusion scenario will be a worst case analysis because any wind would blow particles more rapidly towards a sensor.

The log-likelihood ratio threshhold was determined experimentally to achieve a false alarm rate of 0.01 on 1,500 Monte Carlo simulations under benign conditions (i.e. no release) for the four and 16 sensor configurations. The values are 9.55 and 10.85, respectively. These thresholds were used in all experiments conducted.

4.2.2 Detectability

Figure 4-8(a) shows the number of detections achieved out of 100 simulations as a function of release magnitude for various sensor configurations. The figure demonstrates that significant gains are made in detectability of small release amounts going from one to four sensors while there is only marginal improvement from four to 16. For a sufficient release amount, all sensor configurations are able to reliably detect.

It is important to note that in all configurations, the release location was central within the four sensors in Figure 4-2. Hence, the distance to the nearest sensor is identical in all cases. The reason for the insignificant increase between four and 16 sensors in Figure 4-8 is that under pure diffusion with a release in the center, at least four sensors will see the release in both configurations. These four sensors can provide

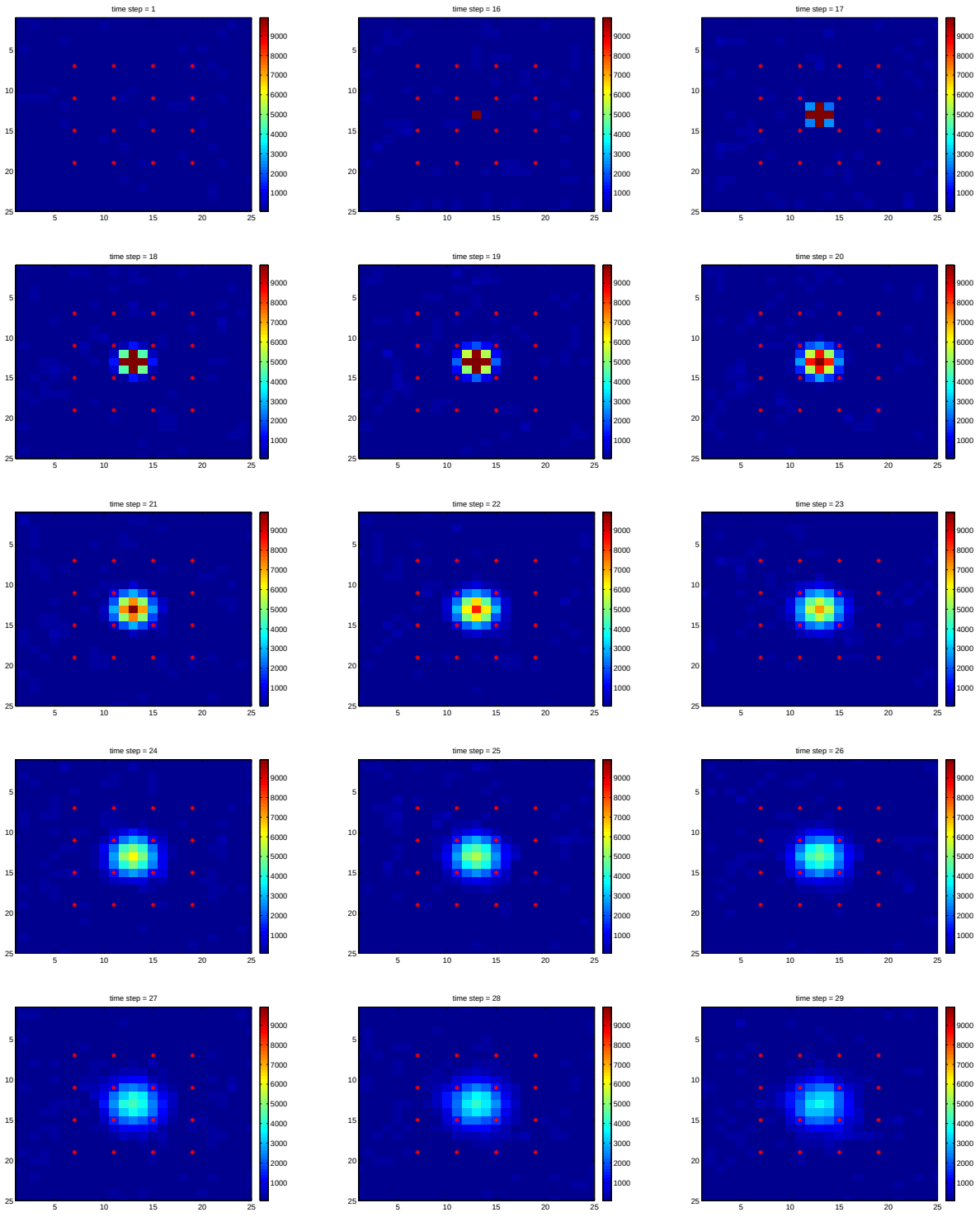


Figure 4-5: Screenshots of a simulation of an aerosol release. The red '*' indicates sensor locations in the 16 sensor configuration. Parameters: release amount = 1e5 particles; release time = 16; release location = (13,13); wx = 0; wy =0; standard deviation of dynamic uncertainty = 100 particles per cell area; standard deviation of measurement noise = 10 particles per cell area.

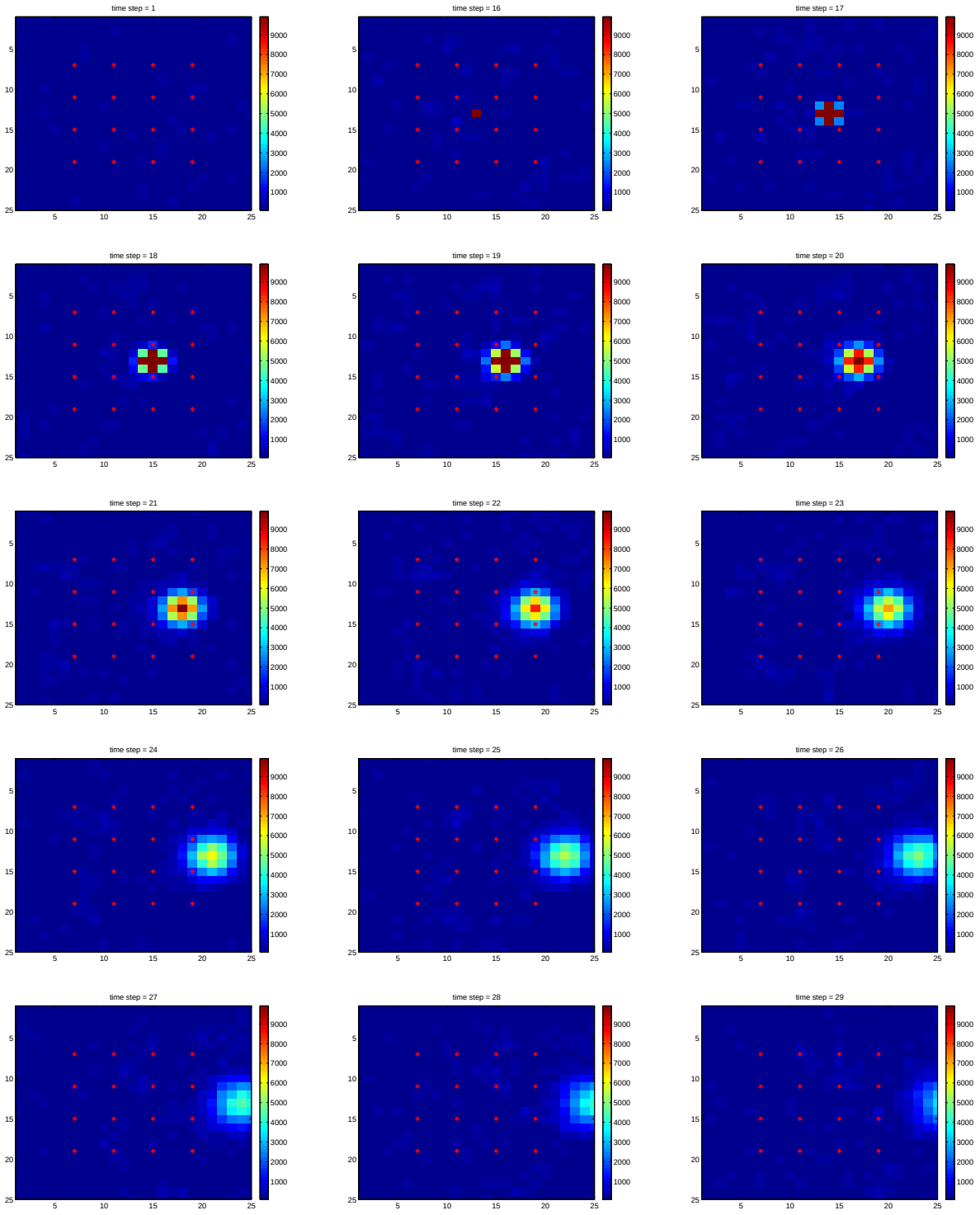


Figure 4-6: Screenshots of a simulation of an aerosol release with a wind bias. The red '*' indicates sensor locations in the 16 sensor configuration. Parameters: release amount = 1e5 particles; release time = 16; release location = (13,13); wx = 1; wy =0; standard deviation of dynamic uncertainty = 100 particles per cell area; standard deviation of measurement noise = 10 particles per cell area.

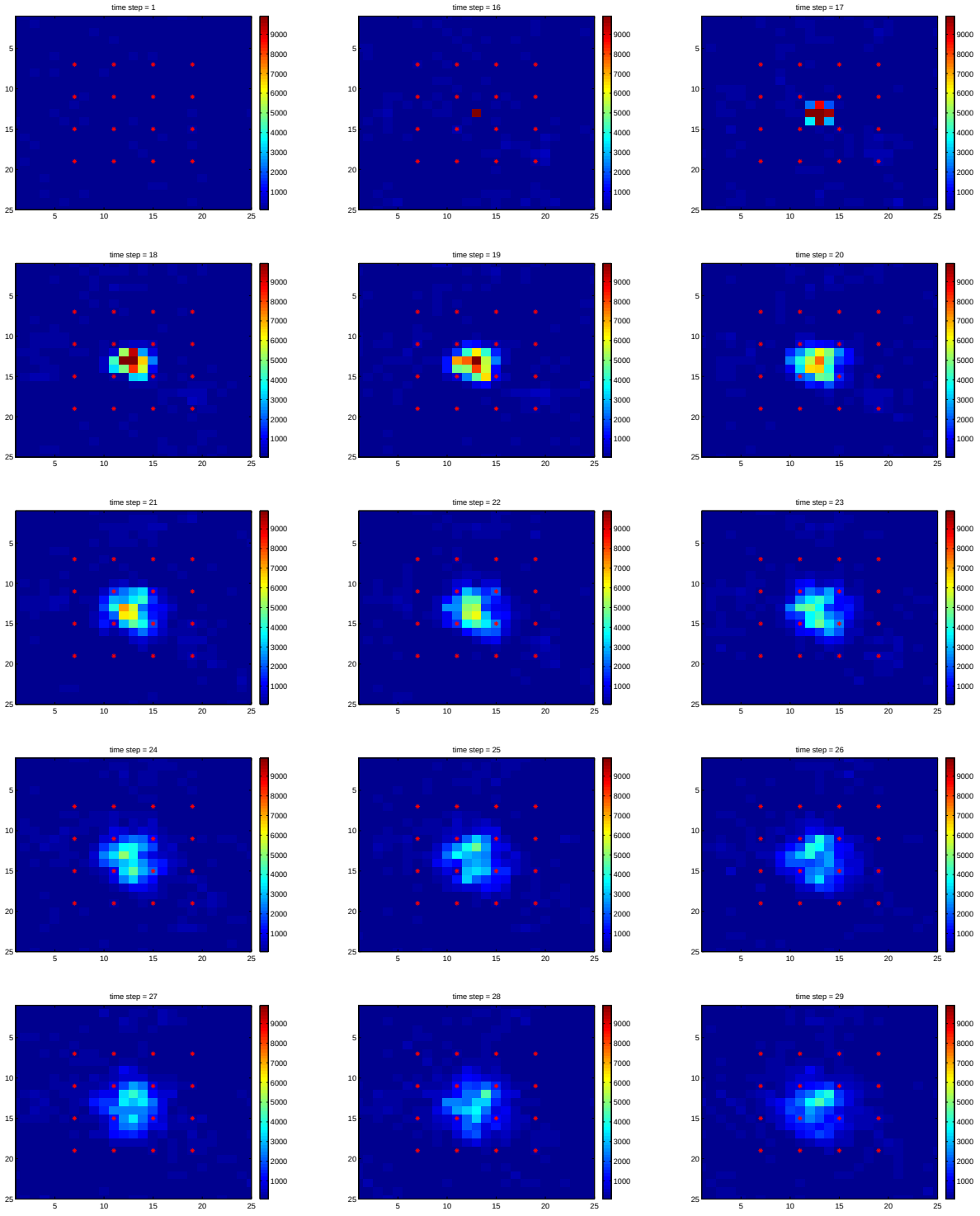
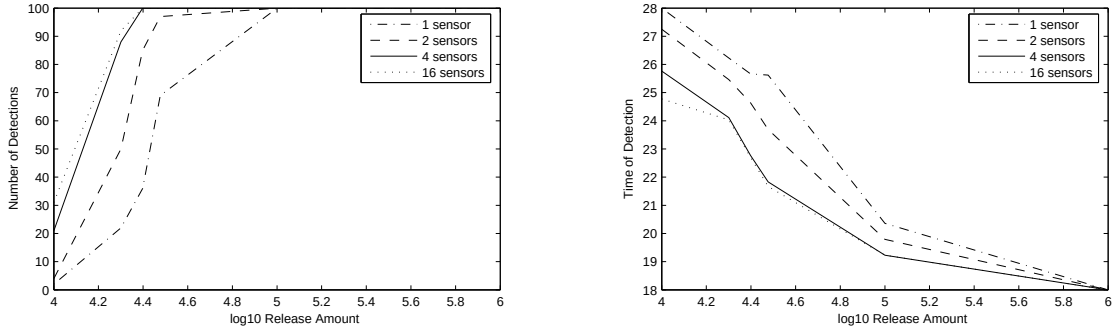


Figure 4-7: Screenshots of a simulation of an aerosol release with wind field noise. The red '*' indicates sensor locations in the 16 sensor configuration. Parameters: release amount = 1e5 particles; release time = 16; release location = (13,13); wx = 0; wy =0; standard deviation of dynamic uncertainty = 100 particles per cell area; standard deviation of measurement noise = 10 particles per cell area; standard deviation of wind vector uncertainty = 0.3162.

sufficient evidence of a release earlier and more often than one or two sensors which have not aggregated enough information. With a wind bias, it is likely that the 16 sensor configuration will have better performance if more sensors see the release than in the four sensor case. In addition, a release in a random location is likely to be further from the nearest sensor in sparser configurations. In such cases, the benefit of using multiple sensors will become clear.

The corresponding time to first detection for the same experiment is plotted in Figure 4-8(b). The figure shows the time of detection of a release occuring at time index 16. For a small release amount, significant performance is obtained using a larger number of sensors. The convergence for large release amounts is indicative of the time required for the release to propagate to the nearest sensor and is purely a function of the distance between the release and the sensor and thus sensor density.



(a) (b)

Figure 4-8: (a) Number of detections per 100 simulation data sets and (b) time to first detection versus release amount. Actual release time was 16 seconds.

4.2.3 Localization manifolds

In Figure 4-9 we examine one time slice of the hypothesis space (t=30). At any given time, there are multiple hypotheses of releases in a given cell, each differentiated by various times of releases. The maximum likelihood ratio of all hypotheses for each cell is plotted for various sensor configurations. Each hypothesis has an associated time of release and maximum likelihood release amount. The real sensor measurements provided to the inference algorithm for the four sensor case is plotted in the lower

right-hand figure. For each of the other sensor configurations, the measurements provided were a subset of those provided in the four sensor case.

The diagrams give an illustration of the degree of localizability achievable with different sensor configurations. In the one sensor case, the manifold is approximately circular indicating that the location of the release cannot be distinguished along a ring around the sensor. The two sensor and four sensor cases provide progressively better localization. As expected, two sensors have difficulty distinguishing along a line while four sensors are able to localize to a point. The degree of shading in the plots indicates the likelihood ratio value and hence the detection probability.

Note that the symmetry seen in Figure 4-9 is due to the combination of the release being in the center of the sensor configurations and the model being pure diffusion. Under wind or off center release locations, the localization manifolds would appear differently.

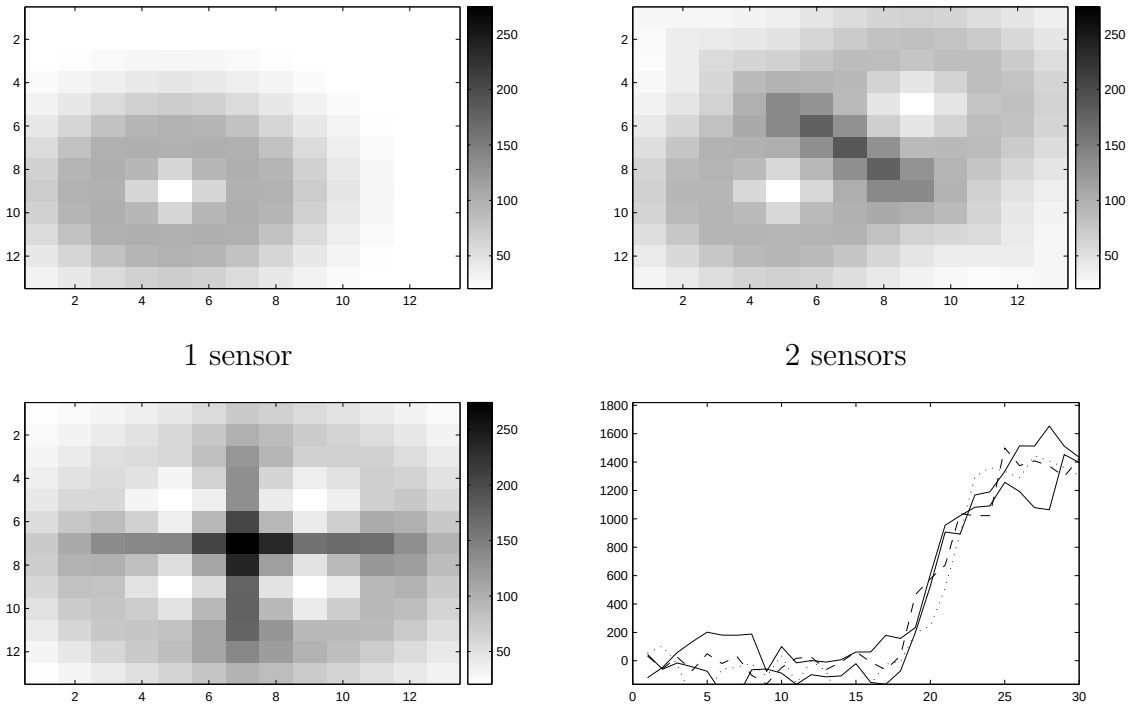
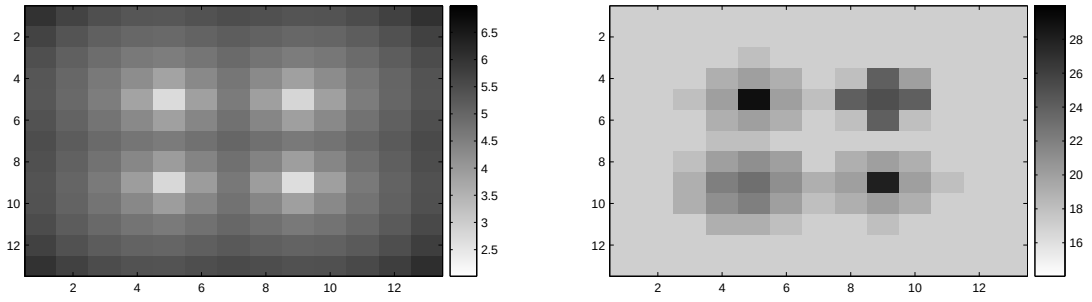


Figure 4-9: Maximum likelihood ratio at every cell in region of interest for 1,2, and 4 sensor configuration. The measurement sequence for the 4 sensor configuration sequence is also shown.



(a) (b)

Figure 4-10: (a) Map of ML estimate of the log release amount associated with maximum LR hypothesis, 4 sensors. (b) Map of time of release associated with maximum LR hypothesis, 4 sensors

Figure 4-10 shows the release amount and time of release manifolds corresponding to the 4 sensor configuration plot in Figure 4-9. That is to say, for each cell's maximum likelihood ratio hypothesis, the corresponding hypothesis time and release amount are plotted in Figure 4-10. As expected, hypothesized releases further from the true release location had to occur earlier in time and be of a larger magnitude in order to appear similar to the actual sensor measurements. Plot (b) hides some of this information because at the time slice examined, only hypotheses of a release a maximum of 14 time steps in the past are considered. Thus, an artificial floor on time of release is created.

Figure 4-11 shows the localization accuracy achieved using four and 16 sensors for various release amounts. The circles illustrate the mean and covariance of the hypothesized location of the detected release. The set of plots on the left are estimates produced at the time of first detection while the plots on the right are estimates after the system has accrued information until the detected hypothesis is "almost" outside the hypothesis window. We define "almost" to be a fixed lag parameter, n, which enforces that we stop accruing information when the window in which hypotheses of a release n time steps before the time corresponding to the initial best hypothesis are still considered. This is done because after the added time of gathering information, a hypothesis at an earlier time step may be better than the initially detected hypothesis. The main idea of this version of the algorithm is to alarm at the time of first detection

and then wait an allotted period of time until producing localization results. The acceptable amount of time to wait is application dependent. Thus, the plots in Figure 4-11 are indicative of the bounds on localization performance.

The trend indicates that larger release amounts can be localized better and regardless of release amount, 16 sensors outperform four sensors. The one and two sensor cases were excluded due to the poor localization results. For smaller release amounts, 16 sensors do not gain much by waiting to accrue additional information. However, both sensor configurations achieve significantly better localization in the case of larger release amounts. These results can be explained by the fact that the signal to noise ratio at later time steps or further from the release location is lower. Thus, the added information accrued by the outlying sensors in the 16 sensor case for small releases will be negligible.

In Figure 4-12, histograms of the time from release until detection are plotted for the various release amount scenarios for four and 16 sensors. Not surprisingly, for larger release amounts the mean and variance of the time to detection is smaller than that of small release amounts. Because the threshold in the 16 sensor case is different than in the four sensor configuration, we see in the 2.5e4 release amount scenario that it is possible for 16 sensors to not detect a release when 4 sensors can. This is purely a result of the fact that in order to maintain a $P_F = 0.01$, the likelihood ratio threshold is more conservative in the 16 sensor case and in some outlying cases this will result in lower detection though it is not the overall trend.

4.2.4 Effect of unmodeled wind bias

To analyze the effects of errors in the assumed known wind field, data produced under various wind biases was provided to the inference algorithm. The wind was always taken to be straight to the right (as defined in the coordinate system of Figure 4-6) such that it tends to transport the release between the two right-hand sensors in the four sensor configuration.

Figure 4-13 shows the probability of detection for the 4 and 16 sensor configurations as a function of increasing wind bias. The release amount for these experiments

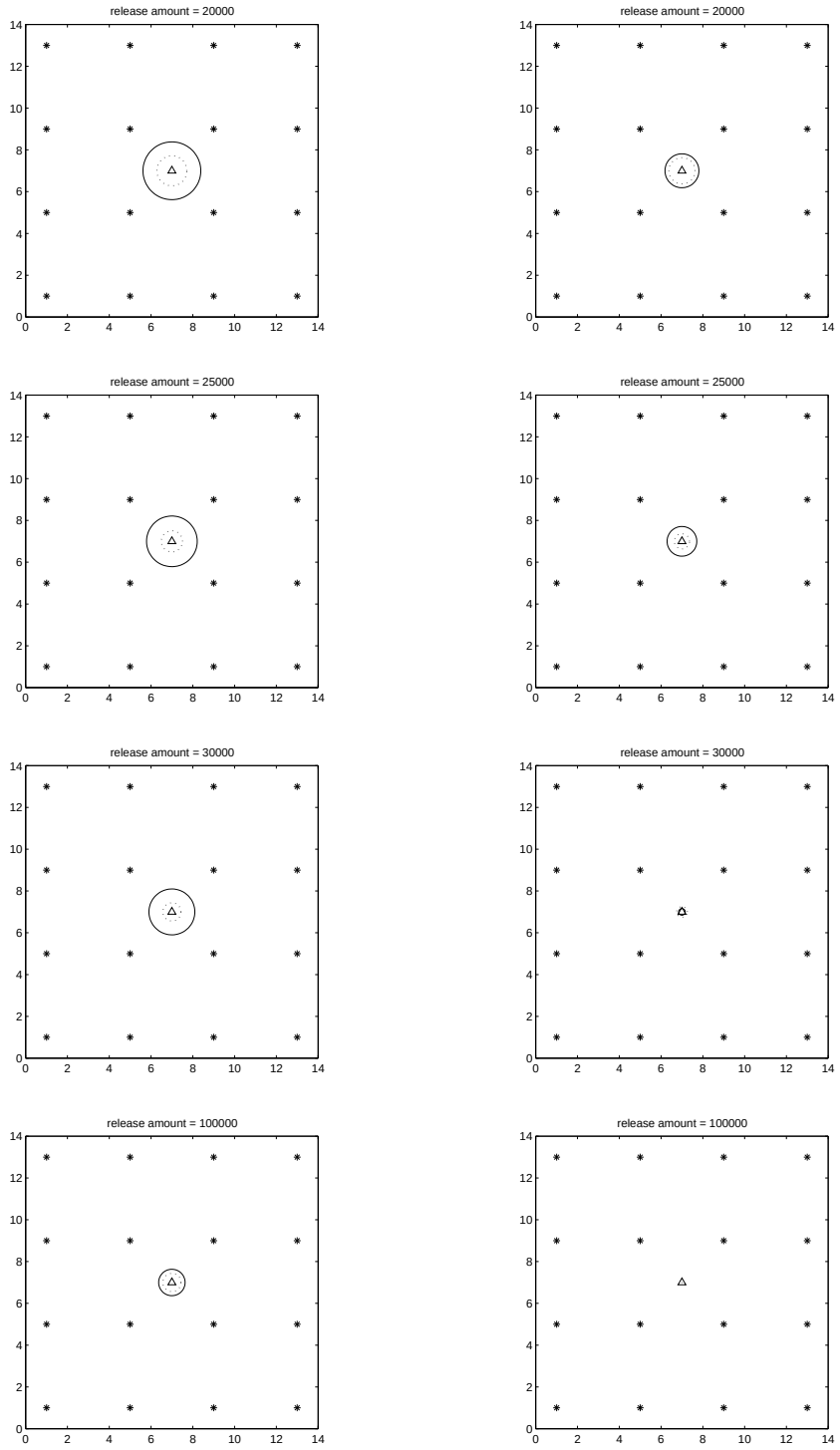


Figure 4-11: Mean and covariance of localization for various release amounts conditioned on first detection based on information (left) at time of first detection and (right) accrued until the hypothesis is almost out of the window. Dashed lines = 16 sensor configuration, solid lines = 4 sensor configuration.

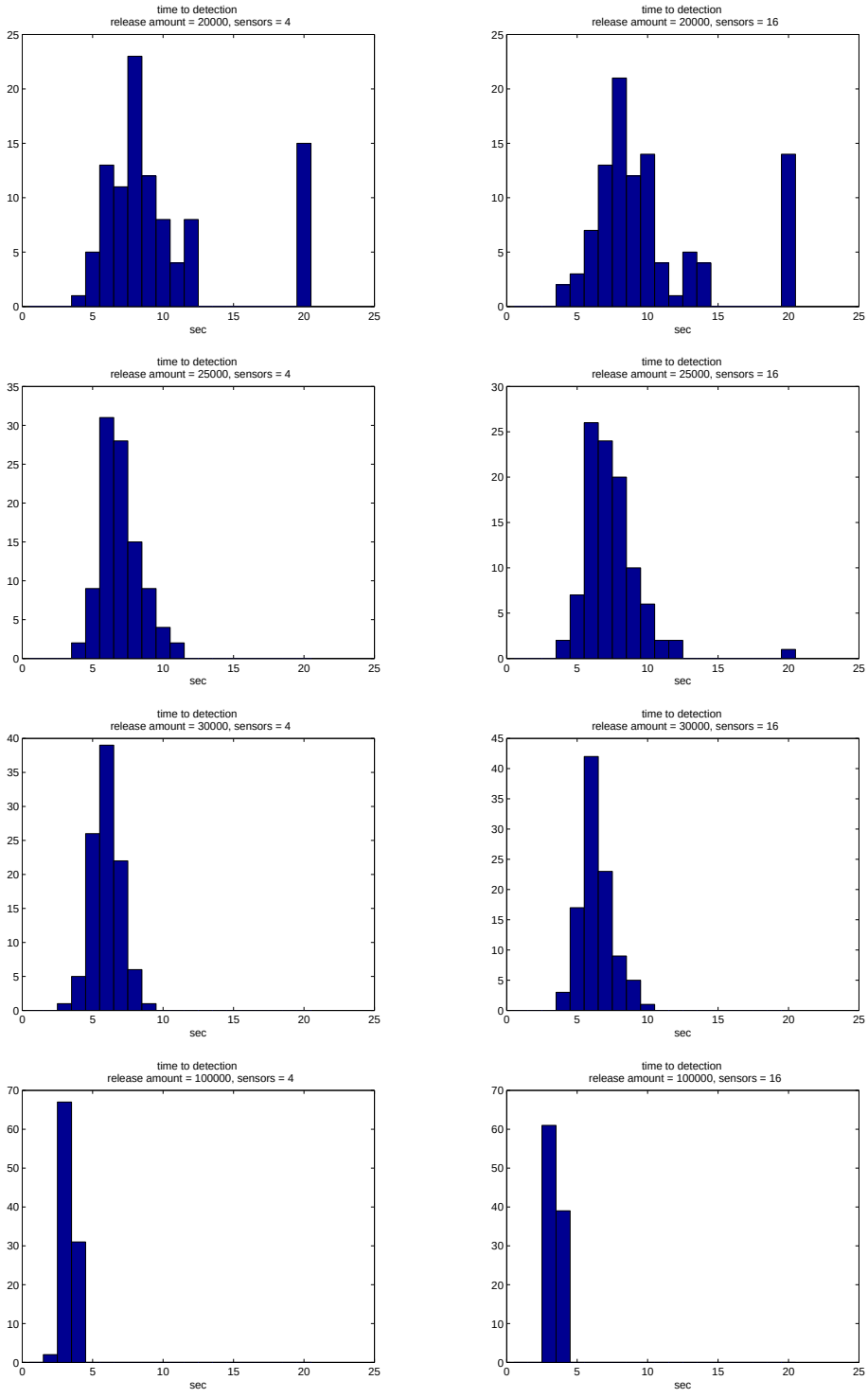


Figure 4-12: Time to detection histograms for various release amounts. (Left) 4 sensors and (right) 16 sensors. No detection was assigned a value of 20 seconds.

was set to 1e5 particles. Given that the dynamical model does not explain the dispersion of particles very well, we conclude that detection is explained by several factors. The first is that the release amount is sufficiently large that only a few sensors are necessary to detect the release. The second is that diffusion is sufficiently high that *some* sensors detect particles before the material is blown clear of the network. Finally the detected particle concentrations are sufficiently high over a short period of time to declare a detection under the pure diffusion model. It is likely that lower release amounts would yield lower detection probabilities.

The 16 sensor network clearly outperforms the four sensor configuration because as the strength of the wind bias increases, it becomes less likely that any of the four sensors see the release because transport is overwhelming diffusion. By the time the release has arrived at the added layer of sensors in the 16 sensor configuration, it has diffused enough that a set of sensors are able to detect the release. With higher wind biases, the performance of the 16 sensor network would degrade and another layer of sensors would be needed to adequately cover the region. See Figure 4-6 for a visualization of the above description.

Localization estimates for this data, as expected, degrade as the wind bias increases. As Figure 4-14 depicts, for a small wind bias, the localization performances are similar to those in the matched model case. For all larger wind biases, the 16 sensor configuration suffers in localization performance by accruing more information while the four sensor configuration marginally benefits by waiting. This is because in the 16 sensor configuration, more sensors are accruing mismatched information and thus the sensor network has less evidence for the correct release location. The plots of the two largest wind biases (the only ones where fours sensors outperform 16 sensors at first detection) are challenging to analyze since there are very few detections in the four sensor configuration (see Figure 4-13). However, we can examine the 16 sensor configuration which has full detection and see that the trend of worse localization with increased wind bias still holds. These results highlight the point that localization performance suffers more quickly than detection performance when the wind field is mismatched to the inference procedure.

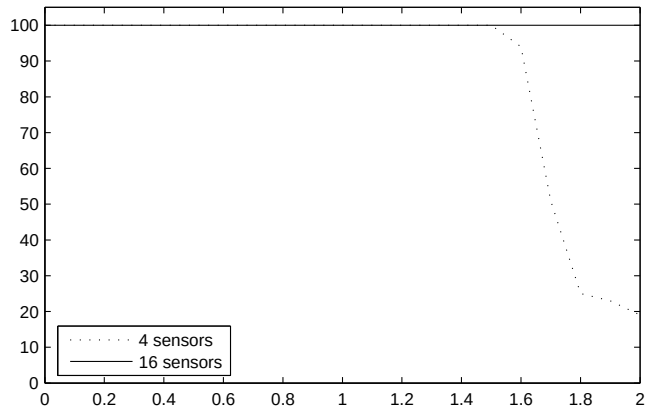


Figure 4-13: Probability of detection versus strength of x-direction wind bias

4.2.5 Effect of unmodeled wind variance

In addition to our model suffering from an unknown wind bias, another plausible error is an unmodeled variance on the perceived known wind field. To analyze this type of error, data was produced under various standard deviations of spatially and temporally white Gaussian noise on a known underlying mean wind field. The inference algorithm shared the knowledge of the mean wind field, but uncertainty on this field was not modeled.

For all variance values tested (0.01, 0.02, 0.04, 0.06, 0.08, 0.1, 0.2, 0.4, 0.6, 0.8, 1), there was a 100% detection rate based on 100 trials. All the trials were run with a release amount of 1e5 particles. As in the case of an unmodeled wind bias, it is likely that lower release amounts would yield lower detection probabilities for this scenario of model mismatch.

Localization estimates for these data sets are shown in Figure 4-15. Comparing the smallest wind variance plots with those of the corresponding 1e5 release amount plots of Figure 4-11, one can see the degradation in localization performance caused by the model mismatch. Performance degrades further with increased unmodeled uncertainty on the wind field. In all cases, accruing more information provides better localization results. As with the matched model, in almost all cases the 16 sensor configuration outperforms the 4 sensor configuration. Only in the high variance situation

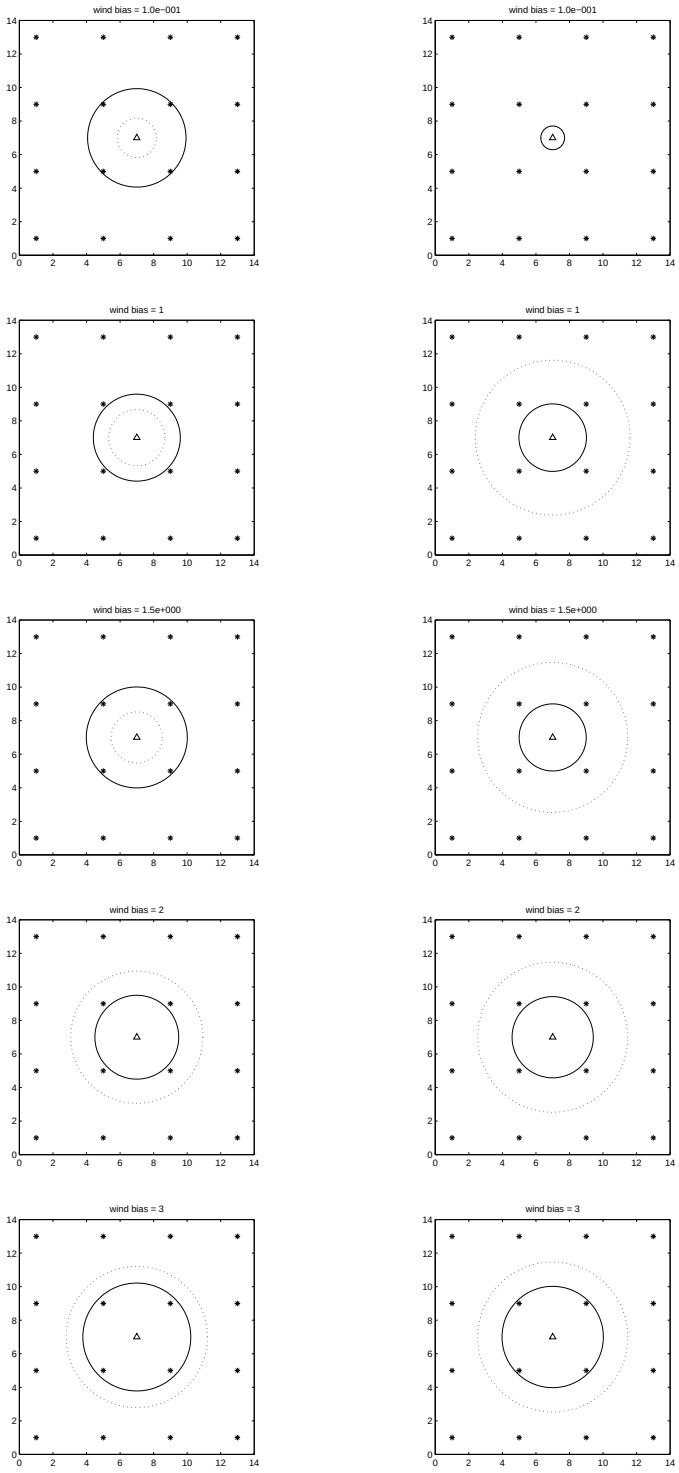


Figure 4-14: Mean and covariance of localization for various strengths of wind bias conditioned on first detection based on information (left) at time of first detection and (right) accrued until the hypothesis is almost out of the window. Dashed lines = 16 sensor configuration, solid lines = 4 sensor configuration.

where information is accrued does this not hold. Here, both sensor configurations perform approximately equally. Comparing these results with those from an unmodeled wind bias, we see that the trends for randomness on the wind field better mirror those of the nominal matched case. This is because the disturbances on the propagation of particle concentrations caused by noise on the wind field are more similar to the type of errors modeled in the dynamic equation than the effects caused by a bias on the wind field.

4.3 Discussion

We have presented a Bayesian state estimation approach to the problem of detection and localization of an airborne biological release. Our approach allows for integration of measurements from multiple sensors over time. We have demonstrated the utility of this formulation and characterized the performance of a set of sensor configurations with regard to detection, time to detection and localization performance. We have also investigated some aspects of model mismatch due to incorrect wind field assumptions. From the experiments conducted, we see that model mismatch impacts localization performance more than detectability.

The formulation presented provides a framework for answering questions such as the interaction between release amount, release location, sensor density, and sensor placement. The appropriateness of this formulation for a given application depends on the validity of the modeling assumptions we have made. These assumptions include that the mean wind field is deterministically known and that the dynamic model can be described as linear when conditioned on the wind field. The linearity of a diffusion plus transport dynamic model makes this framework suitable in many scenarios.

Our approach integrates sensor measurements at a centralized processor. By arranging sensor nodes into groups, our approach provides the basic building blocks for a distributed processing configuration. The issues of how this arrangement should be conducted are a topic of future research. In addition, it is clear that knowing the model parameters is a critical factor in these approaches. Methods from machine

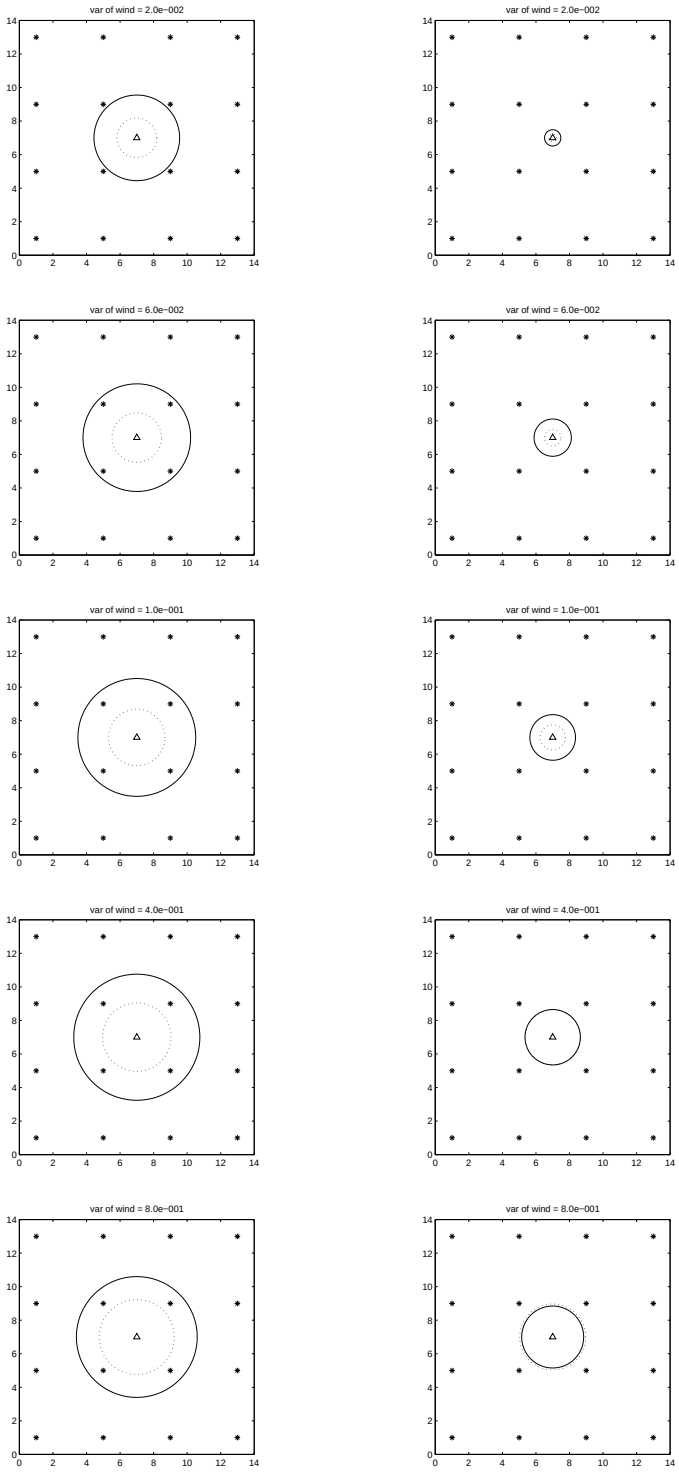


Figure 4-15: Mean and covariance of localization for various levels of wind field uncertainty conditioned on first detection based on information (left) at time of first detection and (right) accrued until the hypothesis is almost out of the window. Dashed lines = 16 sensor configuration, solid lines = 4 sensor configuration.

learning may provide approaches to learning the underlying model parameters so as to reduce sensitivity to mismatch.

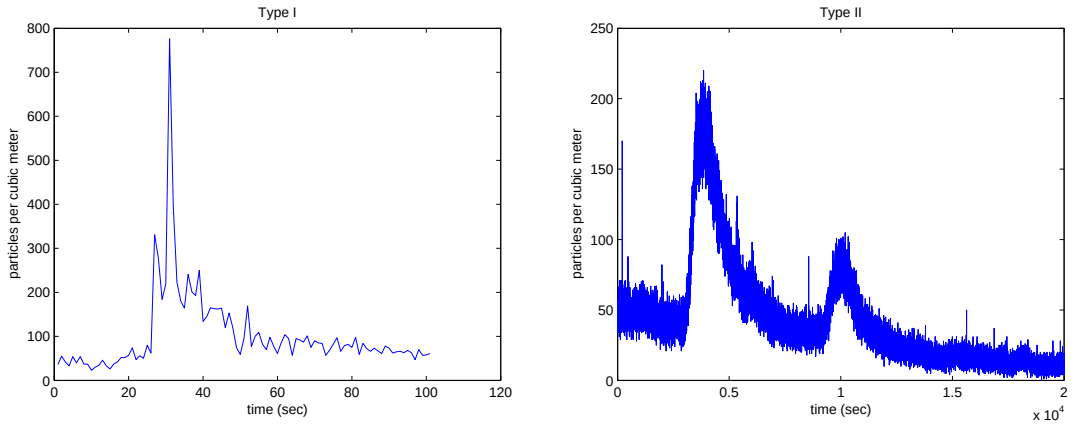
Chapter 5

Wavelet Feature Extraction

As a precursor to any classification of release and background events, it is important to extract features from the sensor measurements which are salient to the discrimination process. These features are derived from all sensor measurements. However, for the purpose of this thesis we only address the question of determining important aspects of particle concentration measurements.

It is desirable for the features to include information regarding the timing and rates of particle concentration changes. The wavelet domain can provide an appropriate basis in such a case because it provides both time and frequency information. This is in contrast with the Fourier domain which only provides frequency information and thus cannot be used to localize an event in time. In this thesis, the wavelet transform is used to reduce the dimensionality of the data through both signal conditioning ('denoising') and as a feature basis.

For the purpose of this thesis, we will analyze two types of signals and judge feature validity on the ability to distinguish between these events. One signal will be called 'Type I' and is characterized by having a fast, sharp peak of high amplitude. This signal was generated by a mock event in which a plume of dust was released. The other signal, 'Type II', is a low amplitude, long trend background event. This signal is characteristic of the type of background events present in the environment being monitored under benign conditions. See Figure 5-1 for time series plots of the two types of signals.



(a) (b)

Figure 5-1: Plots of (a) Type I and (b) Type II signals in amplitude versus time form. Note that the Type I plot has 100 samples while the Type II has 20,000.

In order to accurately classify signals in time using a wavelet domain feature basis, it is important for these features to be shift invariant. This means that a signal shifted in time should have approximately the same feature associated with it as the unshifted signal. Section 5.1 outlines a framework for the problem and discusses the analytical tools that will be used. Detailed theoretical information about the tools used is described in Section 5.2. The results are presented in Section 5.3 and Section 5.4 provides a brief conclusion and discussion of results.

5.1 Framework

5.1.1 The wavelet transform

A wavelet transform provides efficient localization of signal information in both time and frequency, which is very important in this application. This is in contrast with a Fourier series, for example, which only provides information about frequency. When examining signals versus time, we not only care about how quickly the signal changes, but also when the change occurs.

A central framework of a wavelet transform is multiresolution analysis [20]. The theory of multiresolution analysis is very similar to that of filter banks. At each level,

the *mother wavelet* is scaled and convolved with the signal being analyzed to give information in a certain frequency subband. The feature selection takes place in this transformed domain. Some coefficients or subbands hold more pertinent information than others and thus those are the ones analyzed as the features to represent the signal.

Whereas with the Fourier series sinusoids are the chosen basis functions, in wavelet analysis one can specify the desired properties of the signal decomposition and then derive the appropriate basis functions. This allows for efficient representation of signals in the best basis for the application. Best basis pursuit is a topic not explored in this thesis.

5.1.2 Dual Tree Complex Wavelet Transform

The Discrete Wavelet Transform (DWT) is the most general and widely used wavelet tranform algorithm. However, although the DWT is a good approach in terms of representation of the signal, it is not a shift invariant transform. Various strictly shift invariant transforms have been proposed, but at the cost of high redundancy and computational complexity. Kingsbury proposes what is called the Dual Tree Complex Wavelet Transform (DT CWT) as an *approximately* shift invariant transform with only limited redundancy and perfect reconstruction [17]. For this application, the large data sets make low redundancy an important issue. In addition, it may prove to be important to reconstruct a denoised version of the signal, so perfect reconstruction is also a significant benefit. For these reasons, the DT CWT transform is used as the chosen wavelet basis. The properties of the DT CWT transform are explored in more detail in Section 5.2.3.

5.1.3 Wavelet Feature Extraction

Wavelet transforms do not provide dimensionality reduction in and of themselves. Rather, once in the wavelet domain, certain coefficients must be selected or functions on the coefficients must be preformed to reduce the amount of information to only

that which is pertinent to the classification problem.

Denoising

Like any real data, the data from the sensors in this application are noisy. The wavelet coefficients that are primarily due to the noise and not the underlying signal tells us very little about the environment condition and can be eliminated without loss of information. Wavelet transforms provide a simple way in which to perform denoising. The wavelet transform compresses the l^2 energy of the signal into a few large coefficients. Because the noise is not amplified through this transform if the wavelet basis is orthonormal, it is possible to threshold the coefficients in the wavelet domain to eliminate those due to noise. The reconstruction of the signal from the thresholded coefficients provides an estimate of the true signal in the absence of noise. A shift invariant denoising technique is established by applying the denoising process to the DT CWT. The reduced set of coefficients after denoising form a starting point for the dimensionality reduction.

Subband energy

Because the general trend of the Type I and Type II signals vary significantly in frequency content, especially after the denoising process, the energy of coefficients in the subbands of the DT CWT decomposition vary greatly. Therefore, subband energy may be an appropriate shift invariant feature in terms of a class separability measure. Using the energy in a subband as opposed to sets of coefficients, for example, also drastically reduces the dimensionality of the feature vector.

Derivative as another feature

In addition to the subband energy, it may prove to be important in the classification stage to have information about the derivative of the signal. Wavelets can provide a useful way to perform the derivative by providing some smoothing which is important with the type of data being analyzed.

5.2 Methodological Details

The following section elaborates on the technical details of the tools previously described. Sections through 5.2.4 describe standard wavelet transform results which can be found in [6, 27] or the specific papers referenced.

5.2.1 Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) is very useful in compression and denoising and has proven to be superior over the Discrete Cosine Transform (DCT) and the Discrete Fourier Transform (DFT) in many applications. Performing a DWT transform on a signal is equivalent to passing the input signal through a bank of bandpass filters, each with an impulse response that is a scaled version of the *mother wavelet.* The output signals are decimated so that the number of input samples is equivalent to the number of output samples. This is the filter bank interpretation of the DWT. A more formal definition of the DWT is that any signal $g(t)$ can be decomposed into approximation coefficients $c(k)$ and detail coefficients $d(k)$ by using a basis consisting of a scaling function, $\phi(t)$, and wavelet, $\psi(t)$, according to:

$$\begin{aligned} g(t) &= \sum_k c_{jo}(k) 2^{\frac{jo}{2}} \varphi(2^{jo}t - k) + \sum_k \sum_{j=jo}^{\infty} d_j(k) 2^{\frac{j}{2}} \psi(2^j t - k) \\ &= \sum_k c_{jo}(k) \varphi_{jo,k}(t) + \sum_k \sum_{j=jo}^{\infty} d_j(k) \psi_{j,k}(t) \end{aligned} \tag{5.1}$$

The coefficients $c_j(k)$ and $d_j(k)$ can be found by taking the inner products:

$$c_j(k) = \langle g(t), \varphi_{j,k}(t) \rangle = \int g(t) \varphi_{j,k}(t) dt \tag{5.2}$$

$$d_j(k) = \langle g(t), \psi_{j,k}(t) \rangle = \int g(t) \psi_{j,k}(t) dt \tag{5.3}$$

These formulas create a multiresolution interpretation of the DWT where the entire space of real functions, L^2, is the sum of the projection of the signal onto an approximation space V_J and an infinite number of detail spaces, W_J [6, 27].

$$L^2 = V_0 \oplus W_0 \oplus W_1 \oplus \cdots = V_J \oplus W_J \oplus W_{J+1} \oplus \cdots$$

$$\cdot \quad V_{J+1} = V_J \oplus W_J$$

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset \cdots \subset L^2$$

In practice, only a finite decomposition is performed. That is to say, the decomposition will only represent a projection of the signal onto a given approximation space V_J :

$$g_J(t) = \sum_k c_{j0}(k)\varphi_{j0,k}(t) + \sum_k \sum_{j=j0}^{J} d_j(k)\psi_{j,k}(t) \tag{5.4}$$

In essence, a chosen wavelet is scaled and convolved with the input signal to determine how closely correlated the wavelet is with any section of the signal. By scaling the wavelet, one receives information about the frequency content and by shifting the wavelet one gains insight about time locality. The DWT uses dyadic sampling so that only a subset of scales and shifted positions of a continuous convolution are calculated.

The DWT can either be iterated as a single or dual tree. In the case of dual tree iteration, the transform is referred to as a wavelet packet. When iterated as a single tree, only the first level approximation signal is further decomposed rather than both the approximation and detail signal. If the pertinent information about the signal is held in the lower frequency subbands, it is common to use a single tree.

5.2.2 Perfect reconstruction

Although for the most part we are not concerned about perfect reconstruction (PR) in the case of wavelet coefficients as a feature basis, a reconstructed denoised signal may be useful in some other areas of the biodefense application. Below is a brief derivation of the PR conditions for a filter bank. Figure 5-2 shows a simple decomposition and reconstruction tree where H0 is the lowpass analysis filter, H1 is the highpass analysis filter, G0 is the lowpass synthesis filter, and G1 is the highpass synthesis filter.

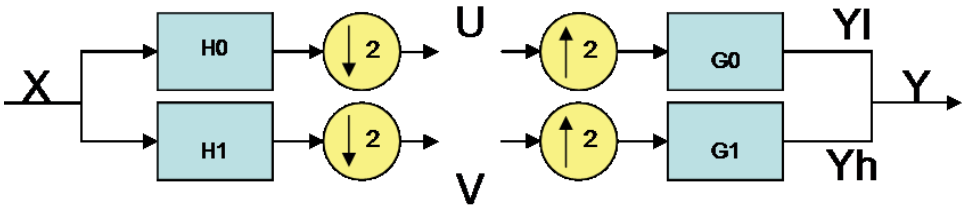


Figure 5-2: Single level decomposition of signal X. Y may not equal X even under PR conditions if U and V are processed in some way before the reconstruction stage.

Using standard linear system relationships, we obtain:

$$
\begin{aligned}
U(z) &= \frac{1}{2}\left[H_0(z^{\frac{1}{2}})X(z^{\frac{1}{2}}) + H_0(-z^{\frac{1}{2}})X(-z^{\frac{1}{2}})\right] \\
V(z) &= \frac{1}{2}\left[H_1(z^{\frac{1}{2}})X(z^{\frac{1}{2}}) + H_1(-z^{\frac{1}{2}})X(-z^{\frac{1}{2}})\right] \\
Y_l(z) &= G_0(z) * U(z^2) = \frac{1}{2}\left[H_0(z)X(z) + H_0(-z)X(-z)\right] G_0(z) \\
Y_h(z) &= G_1(z) * V(z^2) = \frac{1}{2}\left[H_1(z)X(z) + H_1(-z)X(-z)\right] G_1(z) \\
Y(z) &= Y_l(z) + Y_h(z) \\
&= \frac{1}{2}\left[H_0(z)X(z) + H_0(-z)X(-z)\right] G_0(z) + \frac{1}{2}\left[H_1(z)X(z) + H_1(-z)X(-z)\right] G_1(z) \\
&= \frac{1}{2}\left[H_0(z)G_0(z) + H_1(z)G_1(z)\right] X(z) + \frac{1}{2}\left[H_0(-z)G_0(z) + H_1(-z)G_1(z)\right] X(-z)
\end{aligned}
$$

For perfect reconstruction the aliasing terms must cancel which implies that the following must be true:

$$H_0(z)G_0(x) + H_1(z)G_1(z) = 2z^{-l} \tag{5.5}$$

$$H_0(-z)G_0(x) + H_1(-z)G_1(z) = 0 \tag{5.6}$$

5.2.3 Shift invariant transforms

The wavelet transform has become prevalent in signal compression and image coding in the past decade. However, only recently has the transform been applied to signal analysis and reconstruction. This is primarily due to the lack of shift invariance of the DWT. Shift invariance is especially important in this application because shifted versions of the same or similar signal should be classified the same as the unshifted

signal.

DWT lack of shift invariance

Even though the DWT allows for perfect reconstruction, it does not provide shift invariant reconstructions of the signal at each individual level. For example, if the input signal x to the system in Figure 5-2 was shifted by one sample, or equivalently the z-transform was multiplied by z^{-1}, the reconstruction of the lowpass part of y due to the shifted x is not a shifted version of the lowpass part of y due to the unshifted x. This is due to the downsampling at every level.

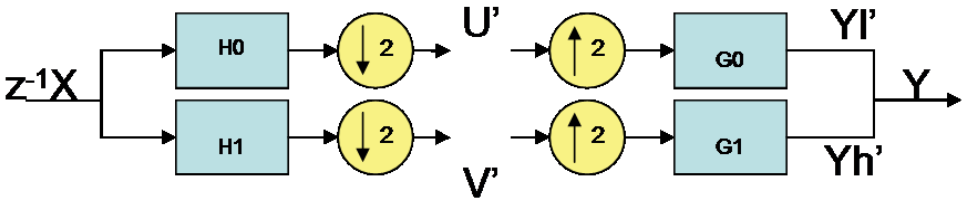


Figure 5-3: Single level decomposition of a shifted version of X. For the DWT, Yl' and Yh' are not shifted versions of Yl and Yh (from Figure 5-2)

Referring to Figure 5-3 we can show that Y_l' is not a shift of Y_l and Y_h' is not a shift of Y_h:

$$
\begin{aligned}
U(z)' &= \frac{1}{2}[z^{-\frac{1}{2}}H_0(z^{\frac{1}{2}})X(z^{\frac{1}{2}}) + (-z)^{\frac{1}{2}}H_0(-z^{\frac{1}{2}})X(-z^{\frac{1}{2}})] && (5.7)\\
V(z)' &= \frac{1}{2}[z^{-\frac{1}{2}}H_1(z^{\frac{1}{2}})X(z^{\frac{1}{2}}) + (-z)^{\frac{1}{2}}H_1(-z^{\frac{1}{2}})X(-z^{\frac{1}{2}})] && (5.8)\\
Y_l(z)' &= G_0(z) * \tilde{U}(z^2) \\
&= \frac{1}{2}z^{-1}[H_0(z)X(z) - H_0(-z)X(-z)]G_0(z) \\
&\neq z^{-1}Y_l(z) && (5.9)\\
Y_h(z)' &= G_1(z) * \tilde{V}(z^2) \\
&= \frac{1}{2}z^{-1}[H_1(z)X(z) - H_1(-z)X(-z)]G_1(z) \\
&\neq z^{-1}Y_h(z) && (5.10)
\end{aligned}
$$

As the formulas indicate, a negative sign in front of the $X(-z)$ term arises from the downsampling and upsampling of $(-z)^{-1}$.

The shift dependency of the subband reconstructions also implies that the energy of the subband coefficients will vary depending on input shift. Figure 5-4 shows the sensitivity of the DWT coefficient energies due to shifts on the input step relative to the sampling grid.

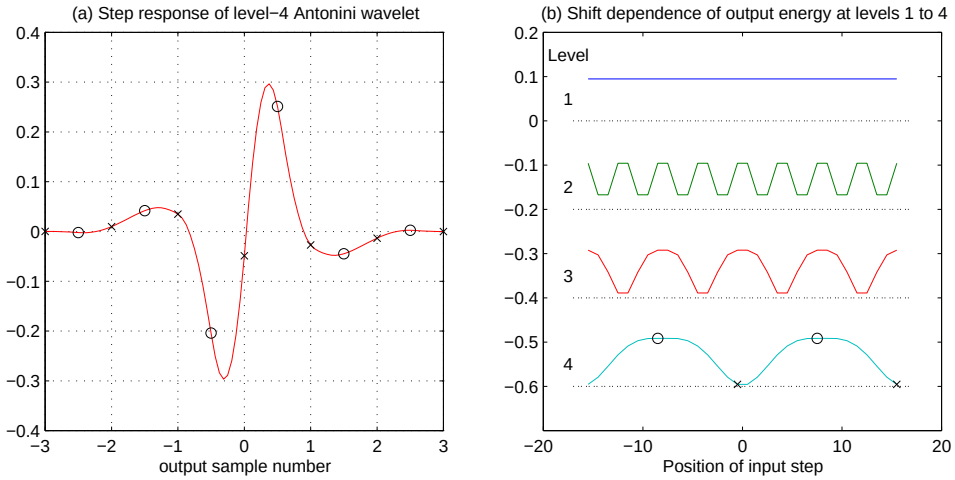


Figure 5-4: (a) Step response at level 4 of the Antonini (7,9)-tap wavelet; and (b) its shift dependence, showing the variation of energy in each level of wavelet coefficients, for a unit step input as the position of the input step is shifted. The zero lines (shown dotted) for each level have been offset for clarity. Figure taken from [16] with permission of the author.

Undecimated DWT

Since the original formulation of the DWT algorithm, various shift invariant transforms have been developed. The simplest method is to compute the DWT for all N shifts of a signal of length N. However, this requires storing N^2 elements and has a computational complexity of $O(N^2)$. Beylkin and others realized that there are only approximately *N*log(N)* different coefficient values [6] because the shifts by any odd number give the same coefficients as the shift by one and the shifts by an even number give the same coefficients as the unshifted signal [11]. This implies that computing the DWT and the shift of one at every level gives the total set of coefficients and has a computational complexity of *O(N*log(N))*.

Another mathematically identical formulation is that of the *algorithme a trous*. This algorithm is explained in the filterbank interpretation. Because the shift depen-

dency of the DWT arises from the downsampling, the *algorithme a trous* solves the shift invariance problem by using an undecimated version of the DWT. To imitate the sub-sampling, the low-pass filter is upsampled at every level by inserting zeros between the filter coefficients. Then, the detail coefficients are computed by taking the difference between two levels of approximation coefficients [11].

Complex Wavelet Transform

Another solution to the shift invariance problem is to use a Complex Wavelet Transform. The input to the first level is the real signal, but the output is complex. Every level thereafter, both the input and output are complex leading to a redundancy of only 2:1 in 1D (see Figure 5-5).

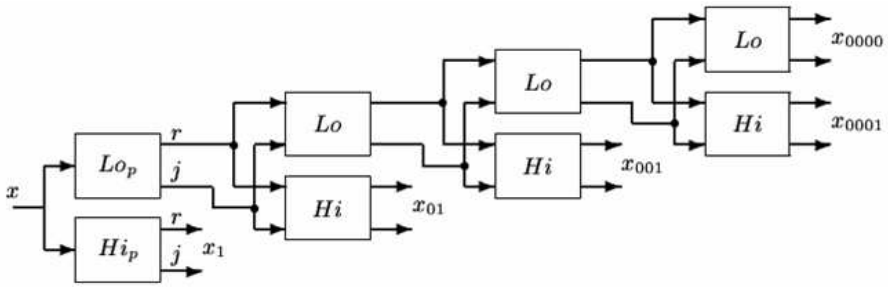


Figure 5-5: Block diagram of a 4 level Complex Wavelet Transform. The decimators have been left out for simplicity. Figure taken from [16] with permission of the author.

The *approximately* shift invariant properties of the Complex Wavelet Transform come from the fact that the magnitude of the step response varies slowly with input shift and only the phase changes rapidly [16]. This keeps the energy of each level of coefficients from varying greatly with input shift. See Figure 5-6.

Dual Tree Complex Wavelet Transform

Although the Complex Wavelet Transform provides approximate shift invariance, it does not allow for perfect reconstruction using short support filters in a single tree. When perfect reconstruction is possible, the filters have poor frequency selective properties [16]. For this reason, there is still motivation for a new shift invariant transform.

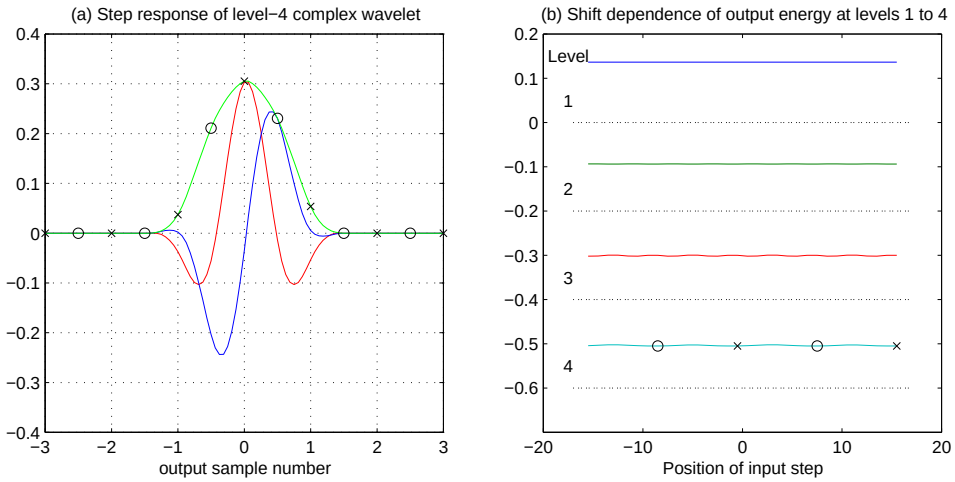


Figure 5-6: (a) Step response at level 4 of complex wavelet (a); and (b) its shift dependence, showing the variation of energy in each level of wavelet coefficient for a unit step input as the position of the input step is shifted. Figure taken from [16] with permission of the author.

One solution is the transform developed by Kingsbury called the Dual Tree Complex Wavelet Transform (DT CWT) and is approximately shift invariant with perfect reconstruction (PR) and limited redundancy [17]. The DT CWT is actually a *real* DWT with twice the sampling rate at each level of the tree. To accomplish this, the DT CWT uses two trees, a and b, for a typical single tree decomposition as shown in Figure 5-7. At the first level, the delays of the lowpass and highpass filters of tree b are one sample offset from those of tree a which ensures that after the downsampling tree a will have the opposite samples of tree b. Below level 1, in order to obtain uniform intervals between samples the filter must have a relative half sample delay. In the synthesis stage, the output from each tree is averaged to obtain the reconstructed signal.

A transform is deemed to be shift invariant if the signal reconstructed from any one level of coefficients is free of aliasing. In the case of the dual tree, this means that if both x001a and x001b, for example, reconstruct to form a signal that is free of aliasing as well as all other a/b subband pairs, the transform is shift invariant. To analyze the filter constraints we start by looking at the case in which only the coefficients from level m are retained. Using the noble identities we can push all

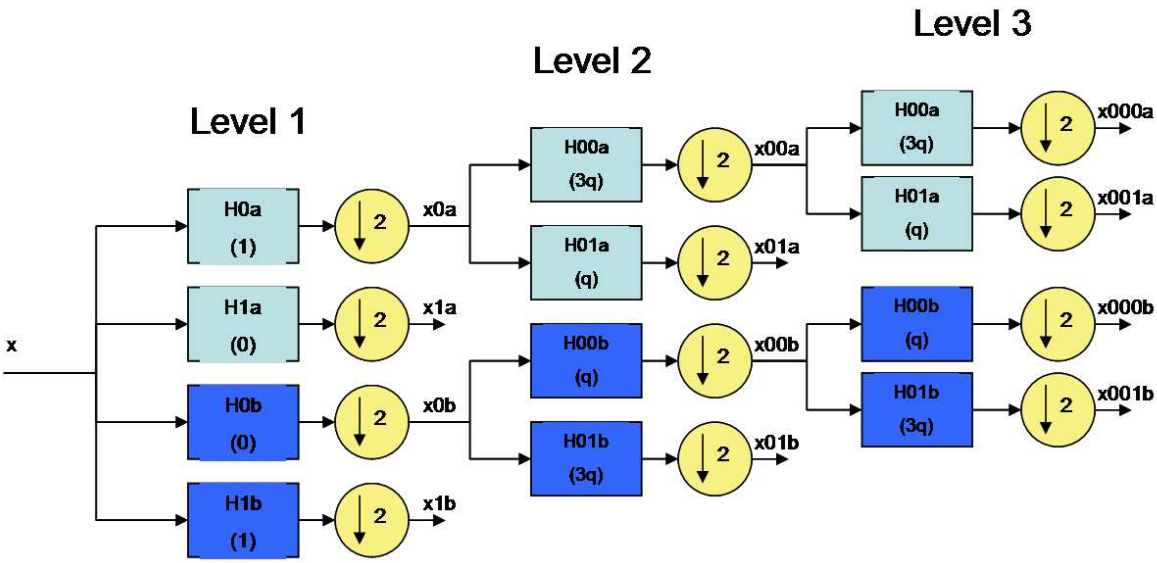


Figure 5-7: Analysis tree of the Dual Tree Complex Wavelet Transform (DT CWT). Each filter samples the signal in such a way as to create approximate shift invariance with limited redundancy

the downsampling and upsampling stages through the m filters and end up with an equivalent downsample or upsample of M where $M = 2^m$. The equivalent filters are given by Ha, Hb, Ga, and Gb where Ha and Hb are either both lowpass or both bandpass filters from tree a and tree b, respectively. See Figure 5-8.

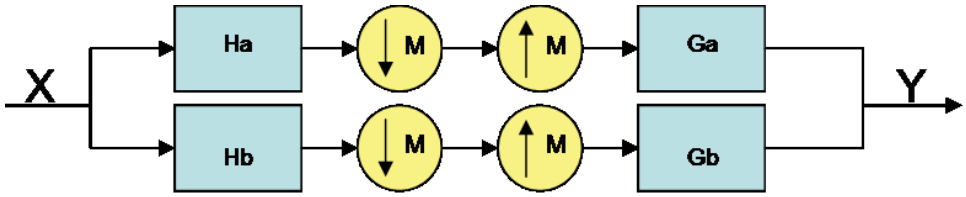


Figure 5-8: Reconstruction tree from only a subband of coefficients.

For the case in which we are looking only at x001a from Figure 5-7, Ha is given as follows:

$$H_a(z) = H_{0a}(z)H_{00a}(z^2)H_{001a}(z^4)$$

One can similarly construct Hb, Ga, and Gb. For the general case, the z-transform

of the reconstructed signal, Y(z), is given by:

$$Y(z) = Y_a(z) + Y_b(z) = \frac{1}{M}\sum_{k=0}^{M-1} X(W^k z)[H_a(W^k z)G_a(z) + H_b(W^k z)G_b(z)]$$

where $W = e^{\frac{j2\pi}{M}}$.

Only the X(z) component is shift invariant, so for approximate shift invariance we need:

$$H_a(W^k z)G_a(z) + H_b(W^k z)G_b(z) \approx 0 \qquad \forall k \neq 0$$

Referring to Figure 5-9, one sees that only the analysis lowpass filter shifts of k=1,-1 overlap with the synthesis filter. Kingsbury [17] derives a method for almost completely eliminating aliasing from the filter frequency responses. The solution he proposed to avoid this overlap is to cancel all odd shifts when the two components from tree a and b are summed. This leads to the following filter lowpass filter constraints:

$$H_b(z) = z^{\pm\frac{M}{2}} H_a(z) \tag{5.11}$$

$$G_b(z) = z^{\mp\frac{M}{2}} G_a(z) \tag{5.12}$$

$$H_b(W^k z)G_b(z) = (-1)^k H_a(W^k z)G_a(z) \tag{5.13}$$

The bandpass subband reconstruction is slightly more complicated. The upper passband of the k=-1,-2 bandpass analysis filter shifts overlap with the lower passband of the synthesis filter while the lower passband of the k=+1,+2 shifts overlap upper passband of the synthesis filter. The solution is to give tree b filters lower passbands of opposite polarity and tree $a$passbands of the same polarity. The bandpass constraints

result in the following formulations:

$$H_a(z) = P(z) + P^*(z) \tag{5.14}$$

$$H_b(z) = -j[P(z) - P^*(z)] \tag{5.15}$$

$$G_a(z) = Q(z) + Q^*(z) \tag{5.16}$$

$$G_b(z) = -j[Q(z) - Q^*(z)] \tag{5.17}$$

$$H_a(W^k z)G_a(z) + H_b(W^k z)D(z) = 2P(W^k z)Q(z) + 2P^8(W^k z)Q^*(z) \tag{5.18}$$

Where $P(z)$ and $Q(z)$ are complex filters with single passbands from $f_s/2M$ to f_s/M and negligible gain for negative frequencies. The bandpass filter responses for tree a and tree b should be treated as the real and imaginary parts, respectively, of the complex filters $P(z)$ for analysis and $Q(z)$ for synthesis.

Figure 5-9 shows the frequency responses of the analysis and synthesis lowpass and bandpass filters at level 3 for both the DWT and the DT CWT. One can clearly see the large aliasing of the DWT in plots (a) and (b) while the DT CWT only has minor aliasing in the bandpass filter as shown in plot (d).

Plots of both the scaling function and wavelet step and frequency responses are shown in Figure 5-10. When tree a and tree b are considered to be the real and imaginary components of a complex filter, the DT CWT looks very similar to the Complex Wavelet Transform. The fact that the magnitude of the filters vary gradually with input shift means that the energy of the level coefficients is approximately shift invariant.

From Figure 5-10 one can also see that unlike in the Complex Wavelet Transform case, the frequency selective properties of the filters are very good according to [16].

The plots in Figure 5-11 show the vast improvement of the DT CWT over the DWT in terms of shift invariance. A 4-level decomposition was performed on a series of 16 shifted step functions. Each plot under "wavelets" is the reconstruction from the detail (bandpass) coefficients at that level. The plot under "scaling function" is the reconstruction from the approximation (lowpass) coefficients. The DT CWT reconstructions differ only slightly between the shifted input signals whereas

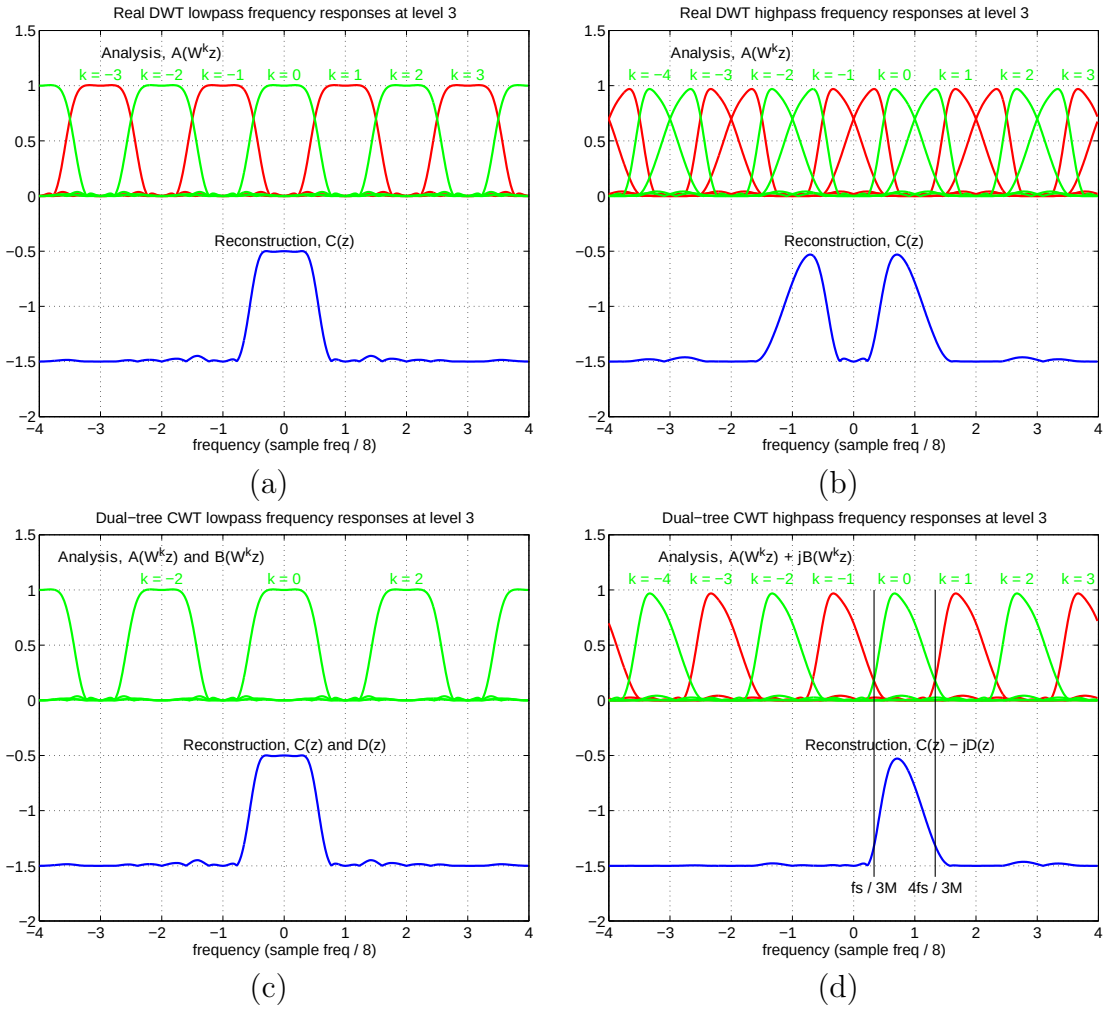


Figure 5-9: Frequency responses of analysis and reconstruction filters (from Table 1) at level 3, showing aliasing terms. Plots (a) and (b) show the responses of the lowpass and bandpass filters in a single wavelet tree, while (c) and (d) show the equivalent responses for the dual tree and demonstrate the reduced overlap of the reconstruction filters with the frequency-shifted (aliased) analysis filters. The horizontal axes are in units of f_s/M where $M = 8$ for level 3. The reconstruction responses are offset vertically by -1.5 to avoid confusion. Figure taken from [17] with permission of the author.

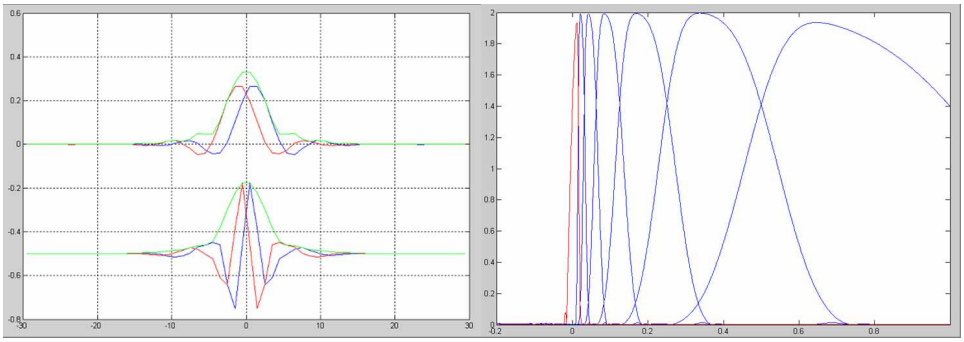


Figure 5-10: (a) Impulse responses of tree a (red) and tree b (blue) level 2 scaling function (top) and wavelet (bottom). (b) Frequency response of the level 6 scaling function (red) and level 1 to 6 wavelets (blue).

the DWT reconstructions demonstrate great variation. It is important to note that the coefficients themselves are not shift invariant. Rather, the shift invariance of this transform comes from combining the two tree at every subband. This leads to the approximate shift invariance of the subband reconstructions. Although the coefficients are not shift invariant, the magnitudes of the coefficients vary gradually with input shift because of the complex nature of the filters and thus the energy of a given level of coefficients is approximately shift invariant as in the Complex Wavelet Transform case.

5.2.4 Denoising algorithms

There are many wavelet denoising algorithms, but regardless of the technique used, the basic process involves converting the signal to the wavelet domain, thresholding coefficients in some manner, and then converting back to the time domain. By eliminating most of the wavelet coefficients solely due to noise, an estimate of the signal in the absence of noise is created.

For any wavelet basis W and signal x with additive noise z, the transformed signal can be written as:

$$Wy = Wx + \sigma Wz$$

If the basis W is orthonormal and the noise is white noise in the time domain, then it will remain white noise in the wavelet domain. Also, the wavelet transform compresses

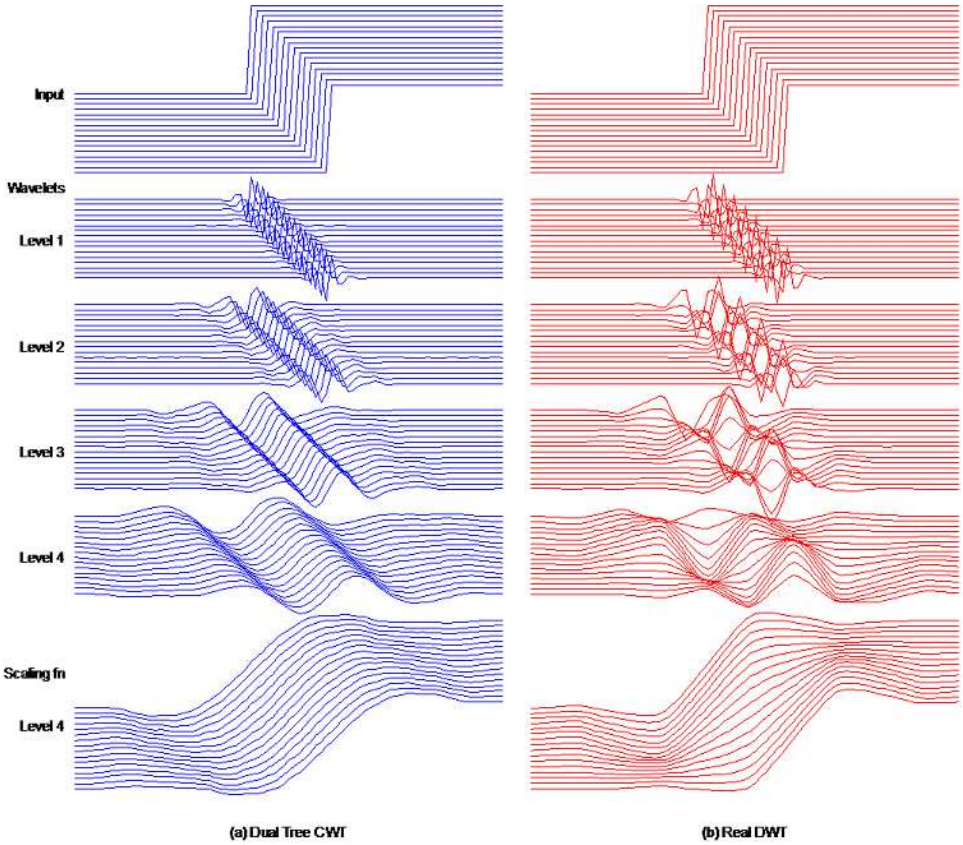


Figure 5-11: Comparison of the approximate shift invariance of the DT CWT (left) versus the DWT (right) by plotting the wavelet and scaling function components of a 4 level decomposition due to 16 shifted step responses. Figure regenerated according to [16]

the l^2 energy of the signal into a few large coefficients. Assuming that the signal to noise ratio (SNR) is large enough, the signal coefficients will stick up above the noise. This leads to the conclusion that simply thresholding the coefficients in the wavelet domain will eliminate the noise. Reconstructing the signal from the thresholded coefficients creates an estimate of the original uncorrupted signal.

There are two ways in which to threshold coefficients: hard and soft. Hard thresholding involves merely setting all coefficients with a magnitude less than the threshold to zero. However, this thresholding method causes artifacts in the reconstructed signal which are created by the abrupt change in coefficient magnitudes from the threshold to zero. In applications such as image compression, visual artifacts are significant and to be avoided. The idea behind soft thresholding is that once the smaller magnitude

coefficients have been set to zero, the remaining coefficients are scaled so that they gradually taper to zero. See Figure 5-12 for visualization and formulation of hard and soft thresholding.

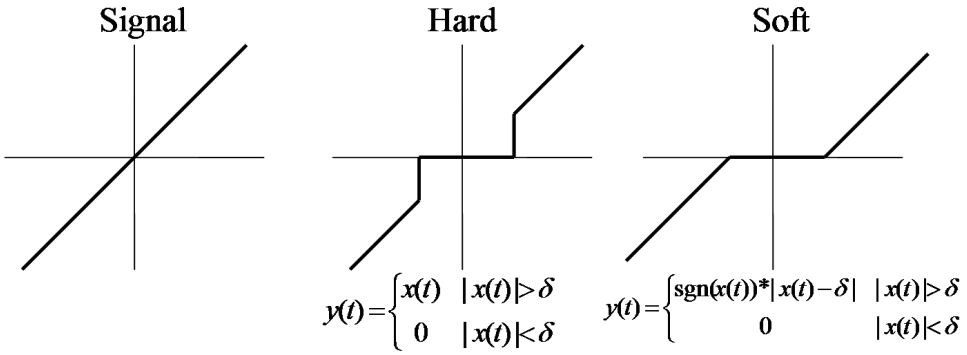


Figure 5-12: Input-output plots of hard and soft thresholding of wavelet coefficients. The x-axis corresponds to the size of the input coefficient and the y-axis corresponds to the resulting output size.

Because the primary concern in this application is with the selection of pertinent wavelet coefficients and not with the visualization of the reconstructed signal, it may not necessarily be the case that soft thresholding is the appropriate method. Experimentation with classification results can be used to determine which method is better. However, when the reconstructed signal is used for taking the derivative, it is better to have the smoother reconstruction given by soft thresholding. For this reason, it is likely that the denoising will take place over two stages, one in which hard thresholding is used to create a feature basis followed by another stage where the coefficients are tapered to zero to imitate soft thresholding. These coefficients are the ones used for creating the derivative and appropriate features are selected at this stage as well. The combined selected features are used as the feature vector. See Figure 5-13 for a block diagram.

Wavelet Shrinkage

One of the most prominently used methods for determining the denoising threshold is that of Donoho and Johnstone [9]. There are two ways to implement the thresholding. The threshold can either be a set value for all the levels of decomposition or a level-

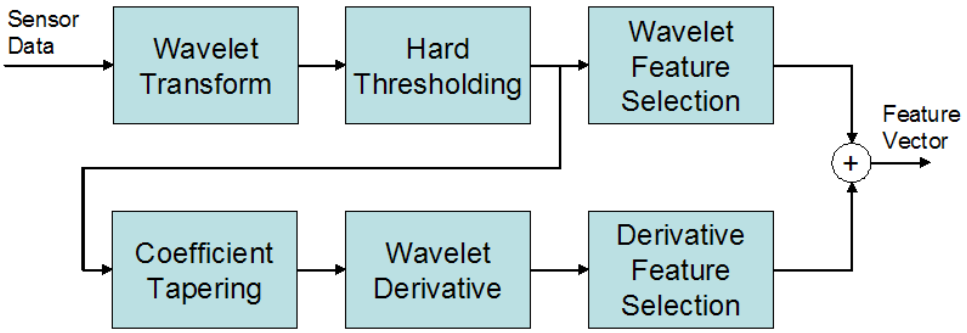


Figure 5-13: Detailed block diagram of possible feature selection process with both wavelet denoising and wavelet derivative.

dependent value can be used. The thresholds δ are given by:

$$\delta = \sigma\sqrt{2log(n)} \tag{5.19}$$

$$\delta_j = 2\sigma\frac{\sqrt{2log(n)}}{2^{J-j}\sqrt{n}} \tag{5.20}$$

Where $\sigma = \frac{MAD}{0.6745}$, MAD is the median absoluted deviation of the level 1 detail coefficients, and $n = 2^{J+1}$. δ_j is the level-dependent threshold where J is the number of levels in the decomposition and j is the current level. Donoho gives the name "Wavelet Shrinkage" to this denoising technique.

5.2.5 Implementing Wavelet Shrinkage with the DT CWT

It is fairly straightforward to apply the Wavelet Shrinkage algorithm to Kingsbury's DT CWT. If the DT CWT coefficients are written in the form: $Y(k) = a(k) + j * b(k)$, and one assumes that the shrinkage is applied equally to both the real and complex parts of the DT CWT coefficients, then the new coefficients are derived as follows. The k^{th} complex coefficient is converted into a single real coefficient by converting to magnitude form. This new set of coefficients is then analyzed as in the DWT case. That is to say, coefficients with a magnitude less than the threshold are set to zero while the remaining are rescaled so that they taper to zero. The factor by which each component of the k^{th} complex coefficient was scaled is given by the ratio of the magnitude thresholded coefficient over the original magnitude. In formulas, this is

done as follows:

$$
\begin{aligned}
C(k) &= \sqrt{a(k)^2 + b(k)^2} && (5.21)\\
C'(k) &= \sqrt{(n(k)a(k))^2 + (n(k)b(k))^2} \\
&= n(k)C(k) && (5.22)\\
Y'(k) &= n(k)a(k) + j * n(k)b(k) && (5.23)
\end{aligned}
$$

In these equations, $a(k)$ is the k^{th} coefficient from tree a, $b(k)$ is the k^{th} coefficient from tree b, $C(k)$ is the magnitude of the original k^{th} coefficient, $C'(k)$ is the magnitude of the thresholded coefficient, $n(k)$ is the k^{th} scaling factor, and $Y'(k)$ is the new complex coefficient.

5.2.6 Derivatives using wavelets

In addition to properties of the wavelet coefficients (e.g. subband energy), it may be important to have information about the derivative of the signal for classification of the Type I and Type II signals. However, there is still enough small variations in the denoised signal that is not pertinent to the overall rate of change information to make a pointwise difference infeasible. Using a wavelet to compute the derivative will incorporate more samples into the derivative at any point through smoothing which allows for better performance with the type of data in this application. The results are similar to that of creating a high order polynomial fit of the data and performing the derivative on the polynomial. Because the wavelet transform is already computed for other purposes in this application, the added complexity of performing a polynomial fit is not justified.

The basic formulas for performing the derivative are outlined below. The projection of a function $f(t)$ onto the V_J space can be represented in terms of approximation coefficients c_k and the scaling function $\phi(t)$. The derivative of $f^J(t)$ is $f^{J'}(t)$ and is also in V_J. However, the resulting equation involves the derivative of the wavelet which is not necessarily very smooth. A better approach is to assume that the deriv-

ative $f^{J'}(t)$ is parameterized by a new set of coefficients a_k, commonly referred to as the connection coefficients, and the scaling function. These new coefficients are given by the convolution of the old coefficients with a function $\Omega(l)$. In the biorthogonal case where the analysis scaling functions $\phi'(\tau)$ are orthogonal to the synthesis wavelet $\psi(t)$ and the analysis wavelets $\tilde{\psi}(t)$ are orthogonal to the synthesis scaling functions $\phi(t)$, the equations are as follows:

$$
\begin{aligned}
\Omega(l) &= \int \phi'(t)\tilde{\phi}(\tau + l)d\tau & a_k &= \int f^{J'}(t)\tilde{\phi}(2^J t - k)dt \\
f^J(t) &= \sum_k c_k \phi(2^J t - k) & &= \int \sum_m c_m 2^J \phi'(2^J t - m)\tilde{\phi}(2^J t - k)dt \\
f^{J'}(t) &= \sum_k c_k 2^J \phi'(2^J t - k) & &= \sum_m c_m \int 2^J \phi'(2^J t - m)\tilde{\phi}(2^J t - k)dt \\
&= \sum_k a_k \phi(2^J t - k) & &= \sum_m \int 2^J \phi'(\tau)\tilde{\phi}(\tau + m - k)\frac{d\tau}{2^J} \\
&= \sum_k \sum_m c_m \Omega(k - m)\phi(2^J t - k) & &= \sum_m c_m \Omega(k - m)
\end{aligned}
$$

As these formulas indicate, it is not necessary to reconstruct the signal and then take the derivative. Rather, the derivative can be computed directly from the old coefficients given the scaling function and its derivative. This allows for greater computational savings and blends in well with the framework of the wavelet feature extraction presented in this thesis.

5.3 Results

The methodology described was applied to the proposed framework and achieved promising results that will have be tested when the feature vector is passed to the classifier (which is beyond the scope of this thesis). We first analyze the practicality of wavelet denoising as compared to other methods such as median or lowpass filtering. Results for the shift invariance of te developed DT CWT denoising technique are then presented. After the signal is denoised, the subband energy of the denoised wavelet coefficients are used as a feature basis. For discrimination between type I and Type II

signals, the efficacy of this feature set is clear. Finally, the wavelet derivative results are presented. Although the properties of the signal derivative are accurate, it is unclear whether these features will improve discrimination performance.

5.3.1 Denoising

As a primary step in analyzing the utility of wavelet denoising, standard Matlab Wavelet Toolbox routines were run and compared to other filtering methods. The denoising using wavelets proved to be a vast improvement over other techniques such as median and lowpass filtering (see Figure 5-14).

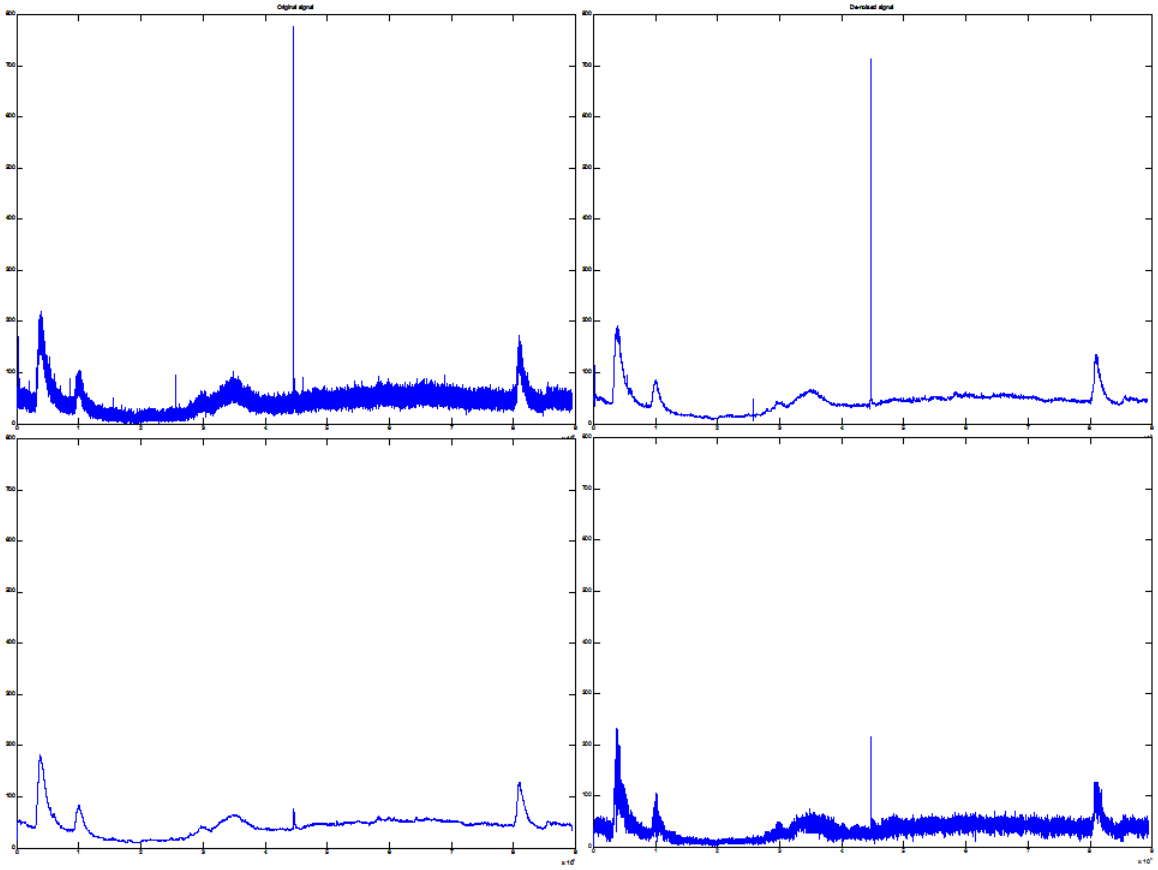


Figure 5-14: Original noisy data set containing both Type I and Type II events (top left); wavelet denoised signal (top right); median filtered signal (bottom left); and lowpass filtered signal (bottom right).

The large spike seen in the original data is a Type I event. Because this is one of the events of interest, it is crucial that the filtering method does not eliminate any

important characteristics of the event such as amplitude. Therefore, the median and lowpass filtering techniques can readily be discarded. When examining a detailed view of the plots, one also notices that the wavelet denoising preserves the jagged characteristics of the signal while the median filter suffers from quantization and the lowpass filter only leaves the low frequency content which hardly resembles the original signal (see Figure 5-15).

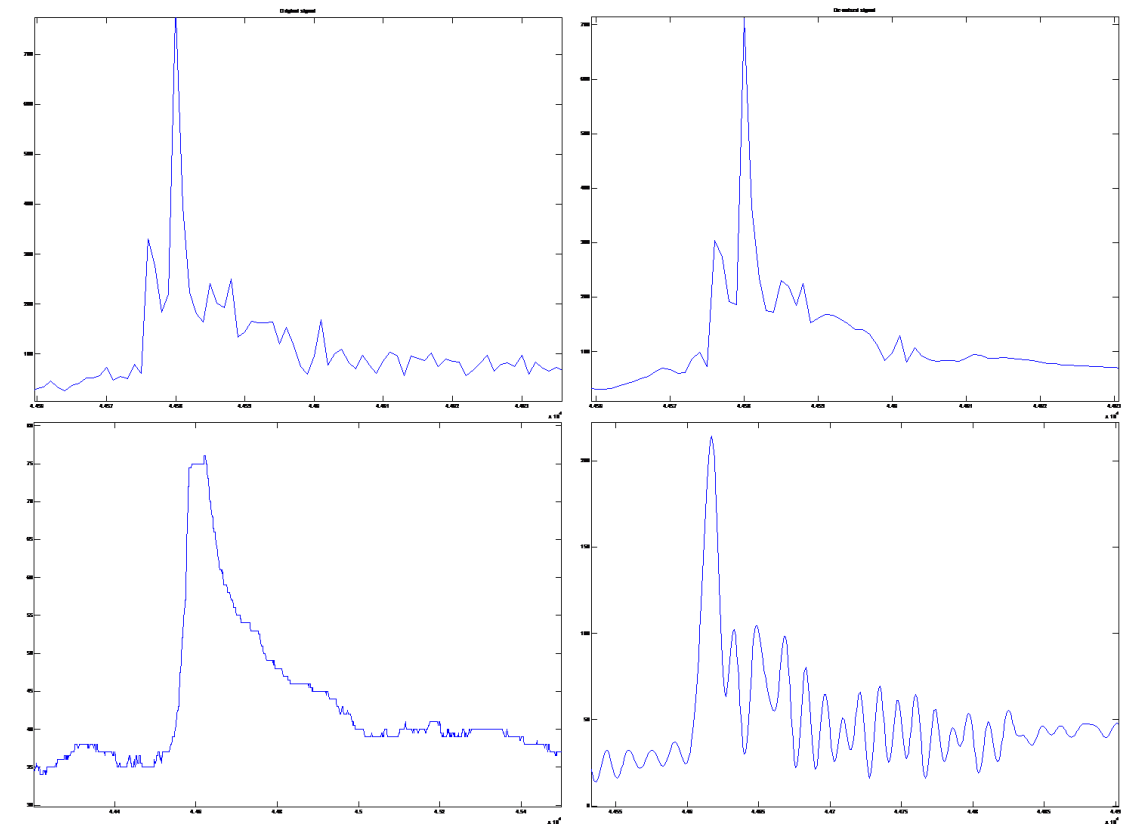


Figure 5-15: Zoom in on: original noisy data set containing both Type I and Type II events (top left); wavelet denoised signal (top right); median filtered signal (bottom left); and lowpass filtered signal (bottom right). Note that each figure has a different x,y-axis scaling. Refer to Figure 5-14 for information on relative amplitudes.

5.3.2 Shift invariant denoising

The method described for applying Wavelet Shrinkage to the DT CWT gave excellent results in terms of creating shift invariant denoised reconstructions. Take, for example, the Type II signal and a shifted version of this signal as shown in Figure

5-16. The shifted signal is created by reflecting the first 950 samples of the signal and cropping an equivalent amount at the end. The DWT and DT CWT denoised versions are shown in Figure 5-17. While it may be difficult to see from the figures, the denoised version of the shifted signal does not have all the same characteristics of the unshifted version. The DT CWT denoising is a vast improvement. Whereas the ∞-norm of the error between the shifted and unshifted signals ($||e||_\infty = \sup_t |e(t)|$) is 9.6328 for the DWT case, this number is reduced to 0.7830 in the DT CWT case. The 2-norm of the error ($||e||_2 = (\sum_t e(t)^2)^{\frac{1}{2}}$) over the time segment is 95.78 for the DWT denoising and only 12.28 for the DT CWT denoising. Figure 5-18 shows DT

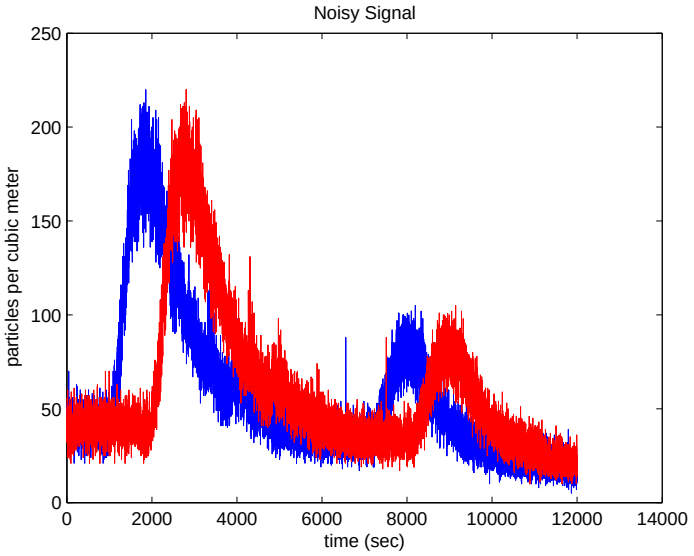
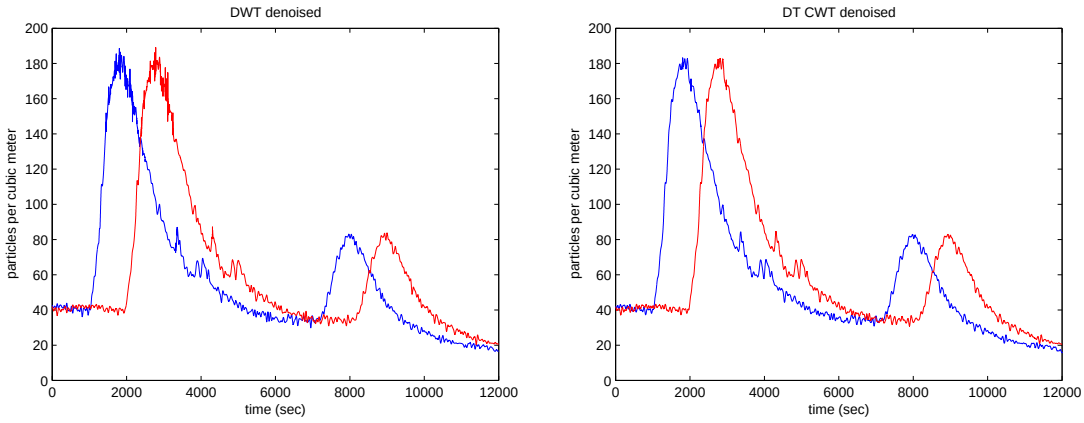


Figure 5-16: Plot of Type II event (blue) and the same signal shifted by 950 samples with the first 950 samples created by reflection (red)

CWT denoising applied to the Type I signal and the whole data set.

5.3.3 Subband energy analysis

In order for classification of the Type I and Type II signals to be successful, the features selected have to be both shift invariant and significantly different for each signal. One of the shift invariant features of the DT CWT is the energy of the coefficients at every level of the decomposition (see Figure 5-6). Because the coefficient content of the different levels varies greatly for the Type I and Type II signal (see



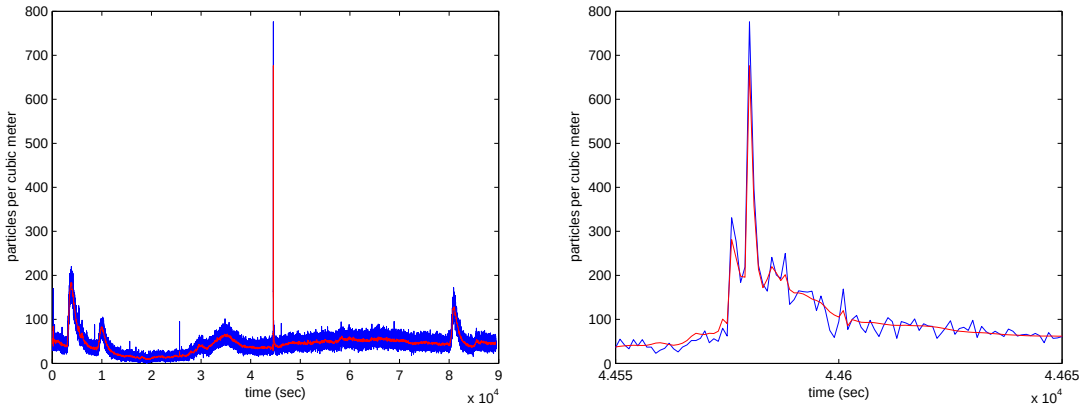
(a) (b)

Figure 5-17: Denoising of signal and shifted signal using (a) DWT and (b) DT CWT. ∞-norm of error: DWT = 9.6328, DT CWT = 0.7830. 2-norm of error: DWT: 95.78, DT CWT = 12.28.

Figures 5-19 and 5-20), this appears to be an informative feature with respect to discrimination between these signals.

5.3.4 Best tree analysis

Matlab's routines 'wpdec' and 'besttree' were used to analyze whether a wavelet packet would contain more pertinent information about the signal than the standard decomposition. The results indicate that the wavelet packet is useful in the entropy-of-coefficients sense. That is to say, the combination of wavelet levels that hold the coefficients with the most information in the wavelet packet decomposition are different from those given by the standard single tree (i.e. iteration of the lowpass channel). However, there is only a difference for the Type I signal. See Figure 4.8 for a visualization of the "best trees" for the Type I and Type II signals. The problem of providing the best energy levels as a feature basis to the machine learning stage is beyond the scope of this thesis.



(a) (b)

Figure 5-18: DT CWT denoising applied to (a) the whole signal and (b) zoomed in on the Type I event. The denoised signal is shown in red and the original noisy signal in red.

5.3.5 Derivatives

The final stage in the feature extraction framework presented is to perform the derivative on the denoised signal. One can adjust a parameter, i to change the number of sample points per integer for the scaling function $\phi(t)$. Thus, setting $i = 0$ gives the scaling function and wavelet values at integer points. Stem plots of $\phi(t)$ for values $i = 1, 2, 3, 4$ are shown in Figure 5-22.

Figure 5-23 shows the wavelet derivative applied to the DT CWT denoised signal as well as a zoom in on the Type II signal for a value of $i = 8$.

As expected, the wavelet derivative goes positive on the rising edge of the Type II signal, falls to zero when the signal peaks, and drops negative on the falling edge. Also note that the peak of the derivative is larger than the trough because the Type II signal has a faster rate of increase than decay. The results are encouraging, but the actual features obtained from the signal's derivative and the classification performance benefits gained are unclear and an area of future research.

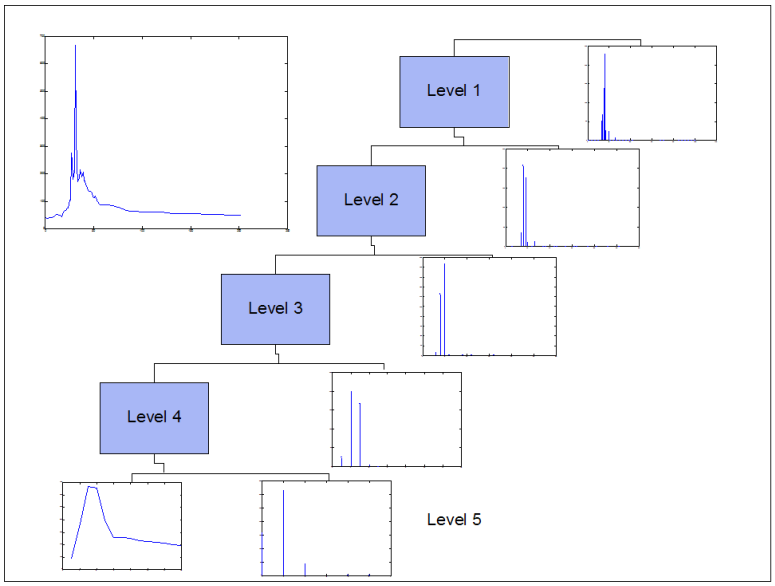


Figure 5-19: Magnitude of DT CWT wavelet coefficients for a 5 level decomposition of a Type I signal. The 5th level has both the approximation and detail coefficients plotted whereas the other levels only have the detail coefficients. The denoised reconstruction of the Type I signal is shown in the upper lefthand corner.

5.4 Discussion

Examining the results derived from applying the theory described in the methodology section to the proposed framework, it is clear that the Dual Tree Complex Wavelet Transform (DT CWT) provides an efficient way to compute shift-invariant wavelet features (e.g. subband energy). In addition, through application of Donoho's Wavelet Shrinkage to the DT CWT, shift-invariant denoised signals were reconstructed. Finally, the denoised coefficients were used to compute the derivative of the signal using scaling functions and connection coefficients and the results proved to give a fairly accurate representation of the rate of change of the signal.

It is important to note that this chapter outlined a theoretically justified proposal of features, but establishing their performance in terms of classification was beyond the scope of this thesis and remains to be investigated.

Another area of future work is in determining how the algorithm should be applied in real time. A sliding window approach has been developed, but the shape and length of the chosen window are open questions.

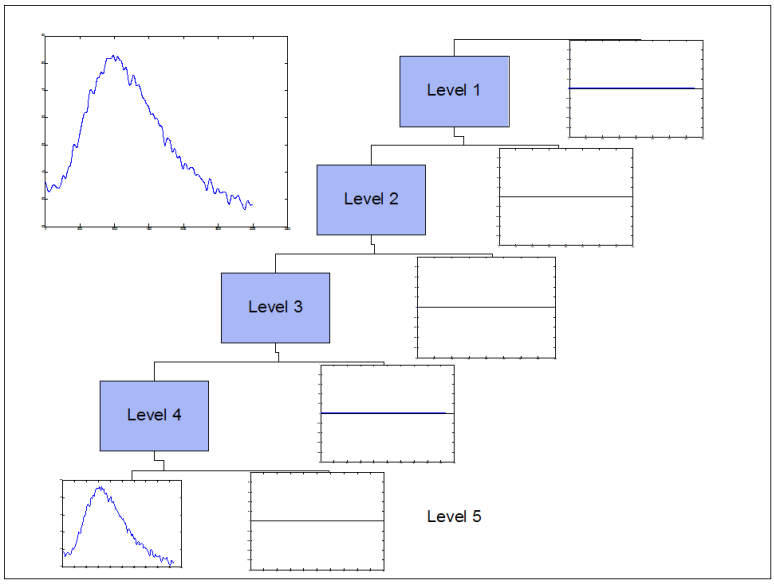
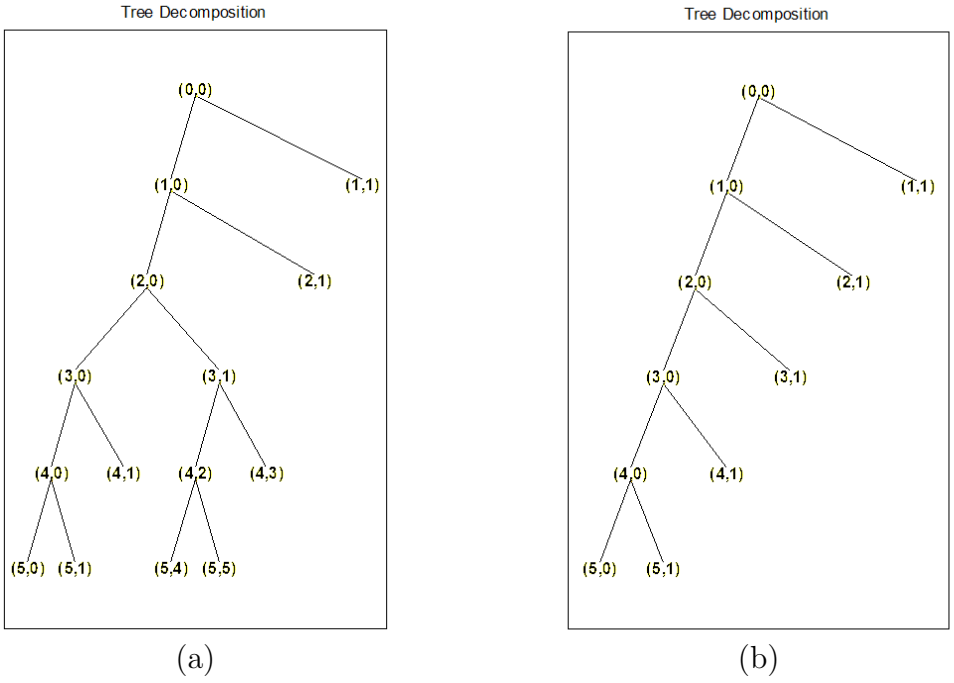
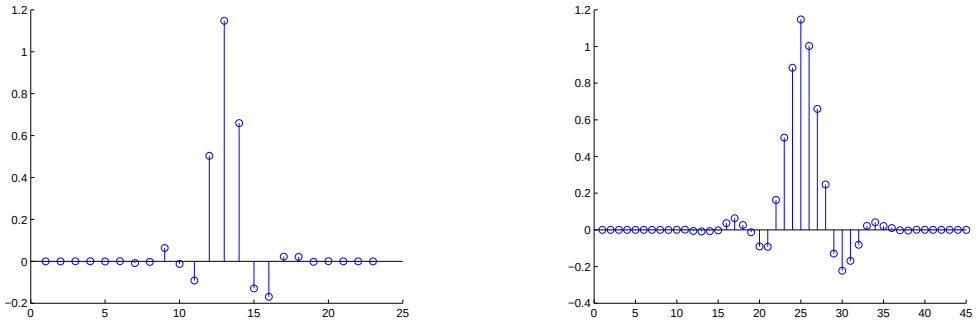


Figure 5-20: Magnitude of DT CWT wavelet coefficients for a 5 level decomposition of a Type II signal. The 5th level has both the approximation and detail coefficients plotted whereas the other levels only have the detail coefficients. The denoised reconstruction of the Type II signal is shown in the upper lefthand corner.

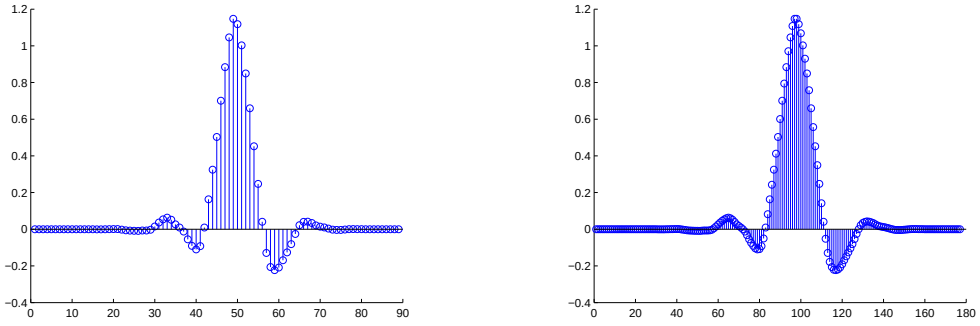


(a) (b)

Figure 5-21: Best tree diagrams for analyzing the Type I and Type II signals according to Matlab's 'besttree' algorithm using Shannon entropy.

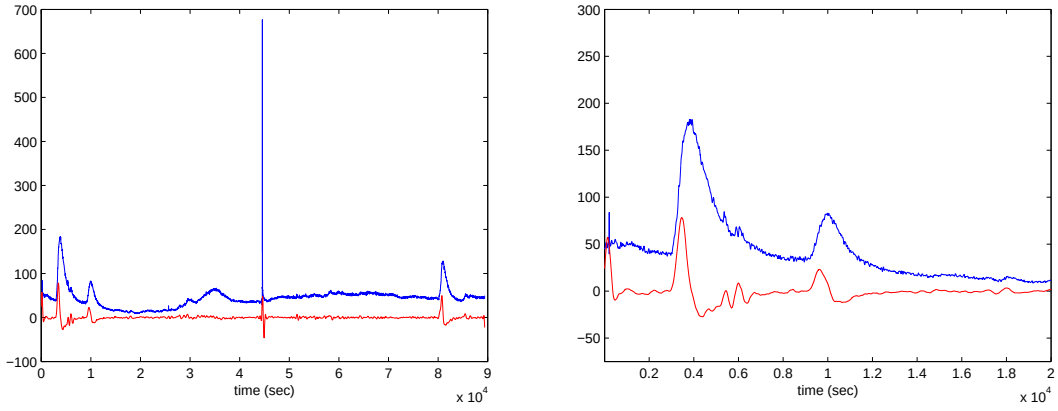


(a) (b)



(b) (b)

Figure 5-22: Stem plots of derivative scaling function for smoothing values $i = 1, 2, 3, 4$, where i is a discretization parameter indicating 2^i points per integer step.



(a) (b)

Figure 5-23: DT CWT denoising applied to (a) the whole signal and (b) zoomed in on the Type I event. The denoised signal is shown in red and the original noisy signal in red. Units: original signal = $\frac{particles}{m^3}$, derivative = $\frac{\frac{particles}{m^3}}{sec}$.

Chapter 6

Conclusion

One application of the analysis presented here is detection, localization, and characterization of a bio-aerosol release from sparse sensor measurements. In this thesis we have examined two aspects pertinent to this application. The first section addressed the problem of detecting and localizing a particulate release using a dynamical model and nonlinear filtering. The second section of the thesis involved investigating feature extraction from sensor data relevant to biological release detection. Specifically, we analyzed the suitability of wavelet feature extraction in reducing the dimension of particle concentration data set while still maintaining the important signal characteristics. These features would be part of the input to a classifier which would distinguish between release events.

6.1 Bayesian Filtering

We have presented a Bayesian state estimation approach to the problem of detecting and localizing a release. Our approach allows for integration of measurements from multiple sensors over time. This method involves a dynamic equation which incorporates an abrupt change of unknown magnitude at an unknown time in an unknown cell. A set of hypotheses, restricted by a fixed window size, are generated and a Generalized Likelihood Ratio (GLR) test compares the hypotheses using Kalman whitened data and the maximum likelihood estimate of the release amount.

We have demonstrated the utility of this formulation and characterized the performance of a set of sensor configurations with regard to detection, time to detection and localization performance. We have also investigated some aspects of model mismatch due to incorrect wind field assumptions. From the experiments conducted, we see that model mismatch impacts localization performance more than detectability.

The formulation presented provides a framework for investigating questions such as the interaction between release amount, release location, sensor density, and sensor placement. The appropriateness of this formulation for a given application depends on the validity of the modeling assumptions we have made. These assumptions include that the mean wind field is deterministically known and that the dynamic model can be described as linear when conditioned on the wind field. The linearity of a diffusion plus transport dynamic model makes this framework suitable in many scenarios.

6.2 Wavelet Feature Extraction

The classifier stage uses functions of the sensor data to categorize events. In addition to distinguishing between various release materials, this stage provides redundancy in discriminating between background and release events in the sense that release events that are not well modeled by the Bayesian filter can potentially be captured by this classifier stage. For this stage to be successful, one needs to limit the dimension of the input to the classifier by using feature extraction. We have presented a method for extracting information from the particle concentration measurements through the use of an approximately shift invariant wavelet transform, namely the Dual Tree Complex Wavelet Transform (DT CWT).

The results presented show that the DT CWT provides an efficient way to compute approximately shift-invariant wavelet features (e.g. subband energy). Through application of Wavelet Shrinkage to the DT CWT, we were also able to create approximately shift-invariant denoised signal reconstructions. We showed the shift-invariance performance gains of this method as compared to using Wavelet Shrinkage with the standard Discrete Wavelet Transform (DWT). Finally, the denoised wavelet coeffi-

cients were used to compute the derivative of the signal using scaling functions and connection coefficients and the results proved to give a fairly accurate representation of the rate of change of the signal.

It is important to note that while this thesis proposed theoretically justified features, the utility of any features would have to be verified empirically.

6.3 Future Work

There are many open research questions in the areas addressed in this thesis. Towards the top of the agenda would be experimental verification of the results presented. This includes testing the proposed features with a classifier and running the Bayesian filtering algorithm on real sensor data. The latter poses a challenge due to the fact that the sensors measure particle concentration in particles per cubic meter which does not map well to a 2D problem. One could either map the 3D measurements to 2D or extend the given inference formulation to 3D. In order to perform tests to determine the applicability of these algorithms, a number of data sets with actual events with some ground truth would have to be available.

The topic of choosing an optimal basis should be pursued. This would include investigating the issues addressed in Chapter 2 such as choosing an appropriate DT CWT wavelet and determining the subband energies necessary in terms of class separability.

Another area of future research is determining the specifics of the sliding window in which the wavelet decomposition is computed. Both the shape and length of the window must be determined.

In terms of the Bayesian filtering stage, the topics of further modeling system dynamics and implementing the algorithm with large scale sensor networks are areas of future research. Ultimately, the goal is to model only as much of the physics as is necessary for inference. Methods from machine learning may provide approaches to learning the underlying model parameters so as to reduce sensitivity to mismatch.

The approach described in this thesis relies heavily on knowledge of the mean wind

field. This field is determined by interpolating sparse anemometer readings using a weighted average scheme based on the cell's distance from the sensors. More sophisticated wind field determination methods could significantly improve the applicability of this approach to real scenarios.

Finally, an inherently difficult aspect of this problem is the fact that a large network of distributed sensors may need to perform detection in a decentralized manner. When the sensors in question are wireless, the limited bandwidth and power available to each sensor may pose a problem. Our approach integrates sensor measurements at a centralized processor. By arranging sensor nodes into groups, our approach provides the basic building blocks for a distributed processing configuration. However, an important area of future research is developing methods by which to effectively summarize and share the information pertinent to inference and detection among the distributed sensors in a communication constrained environment.

Appendix A

Derivations of Sufficient Statistics

In this section, we derive the various results presented in Chapter 4 about the sufficient statistics for hypothesis testing. We refer to H_0 as the null hypothesis indicating that there has not been a release. $H_{i,\phi}$ is termed an alternative hypothesis and represents a release in cell i at time ϕ. The innovations process is defined under the null and alternative hypotheses as follows:

$$\begin{aligned} H_0 &: \nu(t) = \nu_0(t) \\ H_{i,phi} &: \nu(t) = \nu_0(t) + \beta\rho_i(t,\phi) \end{aligned}$$

We note that ν_0 is a white Guassian process so $E[\nu_0] = 0$. In addition, $\rho_i(t,\phi)$ is a deterministic unknown so $E[\beta\rho_i(t,\phi)] = \beta\rho_i(t,\phi)$.

In order to derive the sufficient statistics, it is useful to normalize $\nu_0(t)$ to create a stationary noise process. In the form of the equations as above,

$$K_{\nu_0\nu_0}(t,\tau) = V(t)\delta(t-\tau)$$

Because $V(t)$ is a positive semidefinite matrix, we can decompose it into $V(t) =$

$V^{\frac{T}{2}}(t)V^{\frac{1}{2}}(t)$. Thus, let us perform the following transformation:

$$\begin{aligned} \zeta(t) &= V^{-\frac{1}{2}}(t)\nu(t) \\ w(t) &= V^{-\frac{1}{2}}(t)\nu_0(t) \\ s_i(t,\phi) &= V^{-\frac{1}{2}}(t)\rho_i(t,\phi) \end{aligned}$$

Now, the noise process $w(t)$ is stationary with identity covariance.

$$\begin{aligned} K_{ww}(t,\tau) &= E[w(t)w^T(\tau)] \\ &= E[V^{-\frac{1}{2}}(t)\nu_0(t)\nu_0^T(\tau)V^{-\frac{T}{2}}(\tau)] \\ &= V^{-\frac{1}{2}}(t)E[\nu_0(t)\nu_0^T(\tau)]V^{-\frac{T}{2}}(\tau) \\ &= V^{-\frac{1}{2}}(\tau)V(t)\delta(t-\tau)V^{-\frac{T}{2}}(\tau) = I \end{aligned}$$

The sufficient statistics are derived by projecting the observations onto an orthogonal basis where only a finite number of components have to be analyzed. We choose a complete orthonormal (CON) set of basis functions, namely $\{\phi_n(t)\}_{n=1}^{\infty}$, where

$$\sum_{\tau=\phi}^{t} \phi_i^T(\tau)\phi_j(\tau) = \begin{cases} 0 & : \quad i \neq j \\ 1 & : \quad i = j \end{cases}$$

We project the signals onto this orthonormal basis as follows:

$$\begin{aligned} \zeta(t) &= \sum_{i=1}^{\infty} \zeta_i \phi_i(t) \\ w(t) &= \sum_{i=1}^{\infty} w_i \phi_i(t) \\ s_i(t,\phi) &= \sum_{j=1}^{M} \alpha_{ij}\phi_j(t) \end{aligned}$$

There are a finite number, M, of deterministic signals $s_i(t,\phi)$, and thus we need at most M orthogonal functions to span the set $\{s_i(t,\phi)\}_{i=1}^{M}$. We now rewrite our

hypotheses in this basis.

$$\begin{aligned}
H_0 \quad &: \quad \zeta_i = w_i \\
H_{i,\phi} \quad &: \quad \zeta_1 = \beta\alpha_{i1} + w_1 \\
&\qquad\quad \zeta_2 = \beta\alpha_{i2} + w_2 \\
&\qquad\quad \vdots \\
&\qquad\quad \zeta_M = \beta\alpha_{iM} + w_M \\
i > M, &\qquad \zeta_i = w_i
\end{aligned}$$

We notice that the statistics for all terms $i > M$ of the projection of the normalized observation, $\zeta(t) \rightarrow \zeta_i$, are the same for both hypotheses. Thus, they do not add to the evidence of a given hypothesis and are deemed *indifferent statistics*. The problem is now reduced to a standard detection formulation.

$$\begin{aligned}
\underline{\alpha}_i &= [\alpha_{i1} \quad \alpha_{i2} \quad \ldots \quad \alpha_{iM}]^T \\
\underline{\zeta} &= [\zeta_1 \quad \zeta_2 \quad \ldots \quad \zeta_M]^T \\
\underline{\zeta} &\sim \mathcal{N}(\beta\underline{\alpha}, I)
\end{aligned}$$

Let $H_0 = H_{0,\phi}$ and assume that the hypothesis indexed by (i, ϕ) are mapped to the set of integers $1...M$ where M is the total number of hypotheses enumerated at any time.

$$\begin{aligned}
&\textit{choose} \qquad H_{\hat{i},\hat{\phi}} \\
&(\hat{i}, \hat{\phi}) \;=\; \arg\min_{(i,\phi)} \sum_{(i,\phi)=0}^{M-1} c_{(j,\theta)(i,\phi)} Pr[H_{i,\phi}|\underline{\zeta}]
\end{aligned}$$

The Minimum Probability of Error Criterion (MPE), which is used in this thesis, determines the costs as,

$$c_{(j,\theta)(i,\phi)} = \begin{cases} 0 & : \quad (i,\phi) \neq (j,\theta) \\ 1 & : \quad (i,\phi) = (j,\theta) \end{cases}$$

The Maximum A-Posteriori (MAP) rule then states,

$$\begin{aligned} \textit{choose} \qquad & H_{\hat{i},\hat{\phi}} \\ (\hat{i},\hat{\phi}) &= \arg\min_{(i,\phi)} 1 - Pr[H_{i,\phi}|\underline{\zeta}] \\ &= \arg\max_{(i,\phi)} Pr[H_{i,\phi}|\underline{\zeta}] \end{aligned}$$

We use Bayes rule to rewrite the term of interest as follows:

$$\begin{aligned} \textit{choose} \qquad & H_{\hat{i},\hat{\phi}} \\ (\hat{i},\hat{\phi}) &= \arg\max_{(i,\phi)} \frac{p_{\underline{\zeta}|H}(\underline{\zeta}|H_{i,\phi})P_{i,\phi}}{p_{\underline{\zeta}}(\underline{\zeta})} \\ &= \arg\max_{(i,\phi)} \ln p_{\underline{\zeta}|H}(\underline{\zeta}|H_{i,\phi}) + \ln P_{i,\phi} - \ln p_{\underline{\zeta}}(\underline{\zeta}) \\ &= \arg\max_{(i,\phi)} \ln p_{\underline{\zeta}|H}(\underline{\zeta}|H_{i,\phi}) + \ln P_{i,\phi} \end{aligned}$$

We are allowed to equivalently maximize over the log probability because the log probability is a sum of convex functions and is therefore convex. Also assuming uniform priors (i.e. $P_i = \frac{1}{M}, \forall i$), we are left with Maximum Likelihood (ML) rule.

$$\begin{aligned} \textit{choose} \qquad & H_{\hat{i},\hat{\phi}} \\ (\hat{i},\hat{\phi}) &= \arg\max_{(i,\phi)} \ln p_{\underline{\zeta}|H}(\underline{\zeta}|H_{i,\phi}) \end{aligned}$$

In this case, where $\underline{\zeta} \sim \mathcal{N}(\beta\underline{\alpha}, I)$, we have,

$$\begin{aligned}
\textit{choose} \quad & H_{\hat{i},\hat{\phi}} \\
(\hat{i},\hat{\phi}) &= \arg\max_{(i,\phi)} -\frac{M}{2}\ln(2\pi) - \frac{1}{2}(\underline{\zeta} - \beta\underline{\alpha}_i)^T(\underline{\zeta} - \beta\underline{\alpha}_i) \\
&= \arg\max_{(i,\phi)} -\frac{1}{2}(\underline{\zeta} - \beta\underline{\alpha}_i)^T(\underline{\zeta} - \beta\underline{\alpha}_i) \\
&= \arg\max_{(i,\phi)} -\frac{1}{2}\underline{\zeta}^T\underline{\zeta} + \beta\underline{\alpha}_i^T\underline{\zeta} - \beta^2\frac{1}{2}\underline{\alpha}_i^T\underline{\alpha}_i \\
&= \arg\max_{(i,\phi)} \beta\underline{\alpha}_i^T\underline{\zeta} - \beta^2\frac{1}{2}\underline{\alpha}_i^T\underline{\alpha}_i \\
&= \arg\max_{(i,\phi)} \beta\sum_{j=1}^{M}\alpha_{ij}\zeta_j - \beta^2\frac{1}{2}\sum_{j=1}^{M}\alpha_{ij}^2
\end{aligned}$$

We now examine the proposed sufficient statistics projected onto the CON basis.

$$\begin{aligned}
d_i(t,\phi) &= \sum_{\tau=\phi}^{t}\rho_i^T(\tau,\phi)V^{-1}(\tau)\nu(\tau) \\
&= \sum_{\tau=\phi}^{t}(V^{-\frac{1}{2}}(\tau)\rho_i(\tau,\phi))^T(V^{-\frac{1}{2}}(\tau)\nu(\tau)) \\
&= \sum_{\tau=\phi}^{t}s_i(\tau,\phi)^T\zeta(\tau) \\
&= \sum_{\tau=\phi}^{t}\sum_{j=1}^{M}\alpha_{ij}\phi_j^T(\tau)\sum_{k=1}^{\infty}\zeta_k\phi_k(\tau) \\
&= \sum_{j=1}^{M}\sum_{k=1}^{\infty}\alpha_{ij}\zeta_k\sum_{\tau=\phi}^{t}\phi_j^T(\tau)\phi_k(\tau) \\
&= \sum_{j=1}^{M}\alpha_{ij}\zeta_j
\end{aligned}$$

Similarly,

$$\begin{aligned}
a_i(t,\phi) &= \sum_{\tau=\phi}^{t} \rho_i^T(\tau,\phi) V^{-1}(\tau) \rho_i(\tau,\phi) \\
&= \sum_{\tau=\phi}^{t} (V^{-\frac{1}{2}}(\tau)\rho_i(\tau,\phi))^T (V^{-\frac{1}{2}}(\tau)\rho_i(\tau,\phi)) \\
&= \sum_{\tau=\phi}^{t} s_i(\tau,\phi)^T s_i(\tau,\phi) \\
&= \sum_{\tau=\phi}^{t} \sum_{j=1}^{M} \alpha_{ij} \phi_j^T(\tau) \sum_{k=1}^{\infty} \alpha_{ik} \phi_k(\tau) \\
&= \sum_{j=1}^{M} \sum_{k=1}^{\infty} \alpha_{ij}\alpha_{ik} \sum_{\tau=\phi}^{t} \phi_j^T(\tau)\phi_k(\tau) \\
&= \sum_{j=1}^{M} \alpha_{ij}^2
\end{aligned}$$

Thus, we see that the hypothesis testing rule can be written as,

$$\begin{aligned}
\textit{choose} \quad & H_{\hat{i},\hat{\phi}} \\
(\hat{i},\hat{\phi}) &= \arg\max_{(i,\phi)} l_i(t,\phi) \\
&= \arg\max_{(i,\phi)} \beta d_i(t,\phi) - \beta^2 \frac{1}{2} a_i(t,\phi)
\end{aligned}$$

Therefore, $a_i(t,\phi)$ and $d_i(t,\phi)$, as stated in Chapter 4, are sufficient statistics for the given hypothesis testing problem.

A.1 Derivation of likelihood ratio distributions

We now derive the fact that the likelihood ratios are distributed as chi-squared distributions.

$$\begin{aligned}
l_i(t,\phi) &= \frac{d_I^2(t,\phi)}{a_i(t,\phi)} \\
d_i(t,\phi) &= \sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\nu(\tau) \\
E[d_i(t,\phi)|H_0] &= \sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)E[\nu(\tau)|H_0] = 0 \\
E[d_i(t,\phi)|H_i] &= \sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\beta\rho_i(\tau,\phi) = \beta a_i(t,\phi) \\
var(d_i(t,\phi)) &= var(\sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\nu_0(\tau) + \sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\beta a_i(t,\phi))
\end{aligned}$$

Because the second term in the sum, $\sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\beta a_i(t,\phi))$, is a deterministic unknown, it does not contribute to the variance of $d_i(t,\phi)$. In addition, we use that $E[\sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\nu_0(\tau)] = \sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)E[\nu_0(\tau)] = 0$ and $E[\nu_0(\tau)\nu_0(\sigma)] = V(\tau)\delta(\tau-\sigma)$ in the following:

$$\begin{aligned}
var(d_i(t,\phi)) &= var(\sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\nu_0(\tau)) \\
&= cov(\sum_{\tau=\phi}^{t} \rho_i(\tau,\phi)V^{-1}(\tau)\nu_0(\tau), \sum_{\sigma=\phi}^{s} \rho_i(\sigma,\phi)V^{-1}(\sigma)\nu_0(\sigma)) \\
&= \sum_{\tau}\sum_{\sigma} \rho_i(\tau,\phi)V^{-1}(\tau)E[\nu_0(\tau)\nu_0(\sigma)]V^{-1}(\sigma)\rho_i^T(\sigma,\phi) \\
&= \sum_{\tau} \rho_i(\tau,\phi)V^{-1}(\tau)V(\tau)V^{-1}(\tau)\rho_i^T(\tau,\phi) \\
&= \sum_{\tau} \rho_i(\tau,\phi)V^{-1}(\tau)\rho_i^T(\tau,\phi) \\
&= a_i(t,\phi)
\end{aligned}$$

Let $X_0 \sim \mathcal{N}(0, a_i(t,\phi))$ and $X_i \sim \mathcal{N}(\beta a_i(t,\phi), a_i(t,\phi))$. Then,

$$\begin{aligned} H_0 &: l_i(t,\phi) = \frac{1}{a_i(t,\phi)} X_0^2 \\ H_1 &: l_i(t,\phi) = \frac{1}{a_i(t,\phi)} X_i^2 \end{aligned}$$

Thus, the likelihood ratios are distributed as chi-squared distributions with one degree of freedom. Under the alternative hypothesis, there is a noncentrality term of $\beta^2 a_i(t,\phi)$,

$$\begin{aligned} E[l_i(t,\phi)|H_0] &= \frac{1}{a_i(t,\phi)} E[X_0^2] \\ &= \frac{1}{a_i(t,\phi)} (var(X_0^2) + E[X_0]^2) = 1 \\ E[l_i(t,\phi)|H_i] &= \frac{1}{a_i(t,\phi)} E[X_i^2] \\ &= \frac{1}{a_i(t,\phi)} (var(X_i^2) + E[X_i]^2) = 1 + \beta^2 a_i(t,\phi) \end{aligned}$$

Appendix B

Pseudocode

B.1 Simulation

In this section, we outline how to implement a simulator to create state histories of particle concentrations. It is assumed that the simulator is passed a structure containing the mean wind field for all times over which the simulation is run. In addition, the release value, c, and it's corresponding (x,y) position, (rx,ry), the state noise covariance, Q, wind noise covariance, Rw, x and y diffusion coefficients, diffx and diffy, and the dimensions of the region, mapsize, are passed to the simulator. The output is the particle concentration and wind vector history. In this pseudocode, propmat is a function that determines the propagation matrix from the room dimensions and wind field.

Note: Throughout the pseudocode, '→' means draw a sample from this distribution.

```
function [x, wx, wy] = simdata(c,rx,ry,Q,Rw,diffx,diffy,mapsize)


r_vec = [0 ... 0 c 0 ... 0]'  %determined by rx,ry
if r_time == 1
    <x(0)> = r_vec
    x(0) -> N(<x(0)>,Q)
```

```
    for t = 1:stoptime
        wx(t) -> N(<wx(t)>,Rw))
        wy(t) -> N(<wy(t)>,Rw))
        A(t) = propmat(mapsize,wx(t),diffx,wy(t),diffy)
        x(t+1) -> N(A(t)*x(t),Q)
    end
else
    <x(0)> = 0
    x(0) -> N(<x(0)>,Q)
    for t = 1:r_time-1
        wx(t) -> N(<wx(t)>,Rw))
        wy(t) -> N(<wy(t)>,Rw))
        A(t) = propmat(mapsize,wx(t),diffx,wy(t),diffy)
        x(t+1) -> N(A(t)*x(t),Q)
    end
    wx(r_time) -> N(<wx(r_time)>,Rw))
    wy(r_time) -> N(<wy(r_time)>,Rw))
    A(r_time) = propmat(mapsize,wx(r_time),diffx,wy(r_time),diffy)
    x(r_time+1) -> N(A(r_time)*x(r_time),Q)
    for t = r_time+1:stoptime
        wx(t) -> N(<wx(t)>,Rw))
        wy(t) -> N(<wy(t)>,Rw))
        A(t) = propmat(mapsize,wx(t),diffx,wy(t),diffy)
        x(t+1) -> N(A(t)*x(t)+r_vec,Q)
    end
end
```

B.2 Inference Procedures

B.2.1 Initialization

We initialize our state estimate by sampling from a Gaussian distribution whose mean is the true state at the first time step and whose covariance is the steady state error covariance matrix, P0. P0 can be determined by solving the discrete time algebraic Ricatti equation using a propagation matrix, A(t), for the pure diffusion case. In code,

```
x0 -> N(x(:,1),P0)
```

where x(:,1) indicates taking the simulation particle concentrations in all cells at time 1.

B.2.2 Recursively obtaining windows of GLR statistics

The following code is run recursively and is designed to work with either real-time data or simulations. In the case of real-time data, it is assumed that the measurements are passed in a sensors structure and that this data includes information about the particle concentration measurements, wind field measurements, and sensor positions. In the case of synthetic data, we generate sensor measurements from the particle concentration histories in the sensor cells by sampling from a Gaussian distribution defined by a measurement noise covariance matrix R.

```
y -> = N(x(sensor_cells,:),R)
```

Here, x(sensor_cells,:) indicates taking the simulation particle concentrations over all time from only the cells with sensors.

The function runGLR takes in the previous statistics structure, stats_prev, the size of the hypothesis window, windowsize, the number of cells, N, the standard state-space matrices B and C (in this case B is just the zero matrix), the measurement noise covariance, R, the dynamic noise covariance, Q, the state, x(t), and the error covariance, P(t). The term sensmap is a matrix with 1's in the cells that have sensors

and 0's elsewhere. Assuming that the sensors remain stationary (i.e. sensmap is constant), the C matrix is not time dependent. We initialize the statistics, stats_prev, to zero and pass x0 and P0 as the first x(t) and P(t), respectively. We then iterate this algorithm for all time. The outputs at every iteration are the new statistics, state, error covariance, and likelihood ratios for the current hypothesis window (LR).

```
function [LR stats x(t+1) P(t+1)]=runGLR(t,stats_prev,sensmap,
                                          windowsize,N,B,C,R,Q,x(t),P(t))

% get previous stats:
a_prev = stats_prev.a
d_prev = stats_prev.d
gamma_prev = stats_prev.gamma
mu_prev = stats_prev.mu

%%%% Get measurements and model parameters %%%%
[y(t) wx(t) wy(t)] = getnewmeas(sensors,t)
% interpolates wind field using anemometer readings:
[windfieldx(t),windfieldy(t)] = interpwindfield(sensmap,wx(t),wy(t))
A(t) = propmat(mapsize,windfieldx(t),diffx,windfieldy(t),diffy)

%%%% Running GLR %%%%
% v(t) = y(t)-C*x(t|t-1):
nu(t) = y(t)-C*x(t)
% error covariance of nu: V(t) = C*P(t|t-1)*C'+R:
V(t) = C*P(t)*C'+R
K(t) = P(t)*C'*inv(V(t))

% Time-varying Kalman filter output = xhat(t+1|t), P(t+1|t)
[x(t+1) P(t+1)] = TVKalman(A,B,C,R,Q,y(t),x(t),P(t),K(t))
```

```
k=1
for phi = t:-1:t-windowsize+1
    if phi==t   % if the hypothesize release time is the current time
        % for every cell being monitored, hypothesize a release at this time:
        for i=1:N
            % make a release in cell i at beginning of current time step:
            gamma0 = [0 ... 0 1 0 ... 0]'
            % initialize states to 0 because it is a new hypothesis:
            mu0 = [0 0 ... 0]'
            d0 = 0
            a0 = 0
            % stat(k,i) means put info
            % in appropriate row relative to the windowsize (working
            % from the current time backwards) and column indicates
            % cell of release.
            % LR,a,d are structures that maintain information for a
            % full window while gamma and mu only hold info from one
            % timestep to the next.
            [LR(k,i) a(k,i) d(k,i) mu(k,i) gamma(k,i)] = GLR(t,phi,i,
                                                            gamma0,mu0,d0,
                                                            a0,nu0,invV,N,
                                                            A,C,K)
        end
    else    % if we are propagating a release hypothesized at a prior time step
        for i=1:N
            % we reference the (k+1)th row element because the hypothesis
            % corresponding to this cell was indexed one row down
            % in the previous stats window:
            gamma0 = gamma_prev(k+1,i)
            mu0 = mu_prev(k+1,i)
```

```
            d0 = d_prev(k+1,i)
            a0 = a_prev(k+1,i)
            [LR(k,i) a(k,i) d(k,i) mu(k,i) gamma(k,i)] = GLR(t,phi,i,
                                                            gamma0,mu0,d0,
                                                            a0,nu0,invV,N,
                                                            A,C,K)
        end
    end
    k = k-1
end

stats.mu = mu
stats.gamma = gamma
stats.a = a
stats.d = d
```

B.2.3 GLR

The function GLR computes the sufficient statistics, di and ai, and the likelihood ratios, ai, for every hypothesis Hi. The terms mu and gamma are necessary for computing the statistics and must be passed from one time step to the next, though a history of these terms is not stored. The function f creates a release vector at the appropriate time.

```
function [LRi ai di mu gamma] = GLR(t,phi,i,gamma0,mu0,d0,a0,nu0,invV,N,A,C,K)

%%%% Innovations %%%%
% rho_i(t,phi) = C*[gamma_i(t,phi)-mu_i(t,phi)]:
rho0 = C*(gamma0-mu0)
% nu(t) = nu0(t) + rho0(t)
nu = nu0 + rho0
```

```
%%%% Sufficient Statistics %%%%
% d_i(t,phi) = d_i(t-1,phi) + rho_i(t,phi)'*inv(V(t))*nu(t)
di = d0 + rho0'*invV*nu
% a_i(t,phi) = a_i(t-1,phi) + rho_i(t,phi)'*inv(V(t))*rho_i(t,phi)
ai = a0 + rho0'*invV*rho0

%%%% Likelihood Ratio %%%%
% LRi(t,phi) = (1/2)*(di(t,phi)^2)/ai(t,phi)
if ai==0 && di==0
    LRi = 0
else
    LRi = 0.5*(di^2)/ai
end

%%%% Update %%%%
% eps_i(t,phi) = mu_i(t,phi)+K(t)*C*(gamma_i(t,phi)-mu_i(t,phi)):
eps = mu0 + K*rho0

%%%% Prediction %%%%
% mu_i(t+1,phi) = A(t)*eps_i(t,phi):
mu = A*eps

%%%% State Propagation %%%%
% gamma_i(t+1,phi) = A(t)*gamma_i(t,phi)+f_i(t,phi):
gamma = A*gamma0 + f(t,phi,i,N)
```

Bibliography

[1] M. Basseville and I.V. Nikiforov. *Detection of Abrupt Changes: Theory and Applications*. Prentice Hall, Englewood Cliffs, NJ, 1993.

[2] L. Birge and P. Massart. *Festschrift for Lucien Le Cam: From model selection to adaptive estimation*. Springer, New York, 1997.

[3] J.J. Braun, Y. Glina, D. Stein, and E. Fox. Multisensor information fusion for biological sensor networks and cbrn detection. In *Proc. Conf. Science and Technology for Chem-Bio Information Systems*, Williamsburg, VA, October 2004.

[4] J.J. Braun, Y. Glina, and J.K. Su. Urban biodefense oriented aerosol anomaly detection using sensor data fusion. In *Proc. of 1st Joint Conf. Battle Management for Nuclear, Chemical, Biological, and Radiological Defense*, November 2002.

[5] J.J. Braun, Y. Glina, J.K. Su, and T.J. Dasey. Computational intelligence in biological sensing. *Proceedings of SPIE*, 5416, 2004.

[6] C.S. Burrus, R.A. Gopinath, and H. Guo. *Introduction to Wavelets and Wavelet Transforms: A Primer*. Prentice Hall, New Jersey, 1998.

[7] R.R. Coifman and M.V. Wickerhauser. Entropy based algorithms for best-basis selection. *IEEE Trans. Inform. Theory*, 38(2):713–718, March 1992.

[8] D.L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inform. Theory*, 41(3):613–627, May 1995.

[9] D.L. Donoho and I.M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

[10] O. Farooq and S. Datta. A novel wavelet based pre-processing for robust features in asr. In *Communication Systems, Networks and Digital Signal Processing Symposium*, Staffordshire, England, July 2002.

[11] A. Gyaourova, C. Kamath, and I.K. Fodor. Undecimated wavelet transforms for image de-noising. Technical Report UCRL-ID-150931, Lawrence Livermore National Laboratory, 2002.

[12] N. Intrator, Q.Q. Huynh, Y.S. Wong, and B.H.K. Lee. Wavelet feature extraction for discrimination tasks. In *Proceedings of The 1997 Canadian Workshop on Information Theory*, Toronto, Canada, June 1997.

[13] A. Jeremic and A. Nehorai. Design of chemical sensor arrays for monitoring disposal sites on the ocean floor. *IEEE Journal of Oceanic Engineering*, 23(4):334–343, October 1998.

[14] A. Jeremic and A. Nehorai. Landmine detection and localization using chemical sensor array processing. *IEEE Transactions on Signal Processing*, 48(5):1295–1305, May 2000.

[15] R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[16] N.G. Kingsbury. Image processing with complex wavelets. *Phil. Trans. Royal Society London A*, 357:2543–2560, September 1999.

[17] N.G. Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. *Journal of Applied and Computational Harmonic Analysis*, 10(3):234–253, May 2001.

[18] C.J. Long and S. Datta. Wavelet based feature extraction for phoneme recognition. In *International Conference on Spoken Language Processing*, Philadelphi, PA, October 1996.

[19] D. Mackenzie. Ensemble kalman filters bring weather models up to date. *SIAM News*, 36(8), October 2003.

[20] S.G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pat. Anal. Mach. Intell.*, 11(7):674–693, July 1989.

[21] A. Nehorai, B. Porat, and E. Paldi. Detection and localization of vapor-emitting sources. *IEEE Transactions on Signal Processing*, 43(1):243–253, January 1995.

[22] J. Neumann and G. Steidl. Dual-tree complex wavelet transform in the frequency domain and an application to signal classification. Technical report, University of Mannheim, September 2003.

[23] G.T. Nofsinger and K.W. Smith. Plume source detection using a process query system. In *Defense and Security Symposium 2004 Proceedings*, Bellingham, Washington, April 2004. SPIE.

[24] E.M. ODonnell, D.W. Messinger, C.N. Salvaggio, and J.R. Schott. Identification and detection of gaseous effluents from hyperspectral imagery using invariant algorithms. *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery X, Proceedings of SPIE*, 5425:573–582, April 2004.

[25] L.L. Scharf. *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Addison-Wesley, Reading, MA, 1991.

[26] J.N. Seinfeld and S.N. Pandis. *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change*. John Wiley & Sons, New Jersey, 1997.

[27] G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Wellesley, MA, 1996.

[28] H.L. Van Trees. *Detection, Estimation and Modulation Theory, Part I*. Wiley, New York, 1968.

[29] A.S. Willsky and H.L. Jones. A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Transactions on Automatic Control*, 21(1):108–112, February 1976.

[30] G.G. Yen and K. Lin. Wavelet packet feature extraction for vibration monitoring. *IEEE Transactions on Industrial Electronics*, 47(3):650–667, June 2000.

[31] T. Zhao and A. Nehorai. Detection and localization of a moving biochemical source in a semi-infinite medium. In *Proc. 3rd IEEE Sensor Array and Multichannel Signal Processing Workshop*, Sitges, Spain, July 2004. IEEE.