EXTRACTING ATMOSPHERIC PROFILES FROM HYPERSPECTRAL DATA
USING PARTICLE FILTERS

by

Dustin Rawlings

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

_____           _____
Dr. Jacob Gunther                      Dr. Todd Moon
Major Professor                        Committee Member


_____           _____
Dr. Don Cripps                         Dr. Mark R. McLellan
Committee Member                       Vice President for Research and
                                       Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2013

# Abstract

Extracting Atmospheric Profiles from Hyperspectral Data Using Particle Filters

by

Dustin Rawlings, Master of Science

Utah State University, 2013

Major Professor: Dr. Jacob Gunther
Department: Electrical and Computer Engineering

Removing the effects of the atmosphere from remote sensing data requires accurate knowledge of the physical properties of the atmosphere during the time of measurement. There is a nonlinear relationship that maps atmospheric composition to emitted spectra, but it cannot be easily inverted. The time evolution of atmospheric composition is approximately Markovian, and can be estimated using hyperspectral measurements of the atmosphere with particle filters. The difficulties associated with particle filtering high-dimension data can be mitigated by incorporating future measurement data with the proposal density.

(61 pages)

# Public Abstract

Extracting Atmospheric Profiles from Hyperspectral Data Using Particle Filters

by

Dustin Rawlings, Master of Science

Utah State University, 2013

Major Professor: Dr. Jacob Gunther
Department: Electrical and Computer Engineering

Removing the effects of the atmosphere from remote sensing data requires accurate knowledge of the physical properties of the atmosphere during the time of measurement. There is a nonlinear relationship that maps atmospheric composition to emitted spectra, but it cannot be easily inverted. Inverting this relationship, however, would allow us to estimate atmospheric parameters by taking hyperspectral measurements of the light emitted from the atmosphere. The particle filter is a method whereby one can estimate a hidden system state based on measurements, without ever having to directly invert the measurement relationship.

Traditionally, particle filters do not perform well in high-dimensional systems. This thesis presents a modification to the particle filter algorithm which can significantly improve performance of atmospheric parameter estimation as well as other high-dimensional estimation problems.

"Lever vi inte i ett fritt land kanske? Får man inte gå hur man vill?"

Pippi Långstrump

# Acknowledgments

Gratitude to the many people that helped bring this body of work into existence, especially:

- Jacob Gunther,

- Todd Moon,

- Don Cripps,

- Gus Williams,

- Bill Smith,

- Melissa Yesalusky,

- Josh Hunsaker.

<div align="right">Dustin Rawlings</div>

# Contents

# List of Figures

# Acronyms

PDF       probability density function

ASSIST    Atmospheric Sounder Spectrometer for Infrared Spectral Technology

RTM      radiative transfer model

# Chapter 1

# Problem Statement

The earth's atmosphere can inhibit the performance of imaging systems that attempt to look through it [1]. Light passing through the atmosphere is scattered, absorbed, and emitted. The atmosphere also emits its own black-body radiation. All of these factors lead to the confusion of sensors trying to measure something embedded in the atmosphere, or something on the other side of it. For an aerial sensor, the target might be the surface of the earth. For a ground-based sensor, the target could be the farthest reaches of space.

Interference from the atmosphere can be compensated for to increase the fidelity of the measured data [2]. However, this interference is highly dependent upon the temporal composition of the atmosphere, meaning that proper compensation requires accurate knowledge of the atmosphere at the time that the measurement was made [1]. Remote sensing of atmospheric parameters can provide the information necessary to solve this problem.

## 1.1   Why Measure the Atmosphere Remotely?

The historic method for measuring atmospheric parameters is the weather balloon. By attaching meteorological equipment to a large balloon, features of the atmosphere can be measured as the balloon ascends, and this information can be relayed back to the surface via radio. There are many limitations to this approach. A weather balloon takes time to rise through the atmosphere, and rarely pursues a straight course as it ascends, being completely at the mercy of the wind. Therefore, an instantaneous measurement of a vertical column of atmosphere is simply not possible with weather balloons. Another limitation is the amount of data that can feasibly be collected. Imagine trying to monitor the atmosphere for changes every second for several days using balloons. Such an experiment would be very difficult to execute.

On top of restrictions to the frequency of measurements, it may be desired to measure the atmosphere in an inaccessible location, due to either geological or political barriers. Whereas launching a weather balloon from the ground in such an area is impossible, remotely viewing the area from a satellite might be an option. For all of these reasons, the ability to remotely measure the atmosphere opens many doors in the areas of meteorology, astronomy, and defense that were previously closed.

## 1.2  Hyperspectral Imaging

Hyperspectral imaging comes as an extension of multispectral imaging. A multispectral image is comprised of pixels, each of which represent radiance at a number of discrete wavelengths over a given area. Hyperspectral images contain radiance measurements at a much greater number of wavelengths, which are close enough together to accurately approximate a continuous radiance spectrum [3].

The Atmospheric Sounder Spectrometer for Infrared Spectral Technology II (ASSIST II) is a mid-wavelength/long-wavelength infrared hyperspectral sensor with spectral resolution of 0.7 $cm^{-1}$ over the range of 500 to 3000 $cm^{-1}$. This is a ground-based device designed to measure the down-welling radiance of the atmosphere. It is important to note that the ASSIST II is not a traditional hyperspectral imager, which combines a matrix of hyperspectral pixels into a single image, but rather a spectrometer which takes a single combined measurement of all light entering into the sensor. Figure 1.1 shows an example of a measurement taken with the ASSIST II.

## 1.3  Relationship Between Radiative Transfer and Atmospheric Parameters

Interesting features of the atmosphere that can be measured include temperature, water vapor mixing ratio, trace gas concentrations, and aerosol concentration. All of these factors have an effect on how light propagates through and is emitted from the atmosphere. A radiative transfer model (RTM) models the radiation intensity of light at specific wavelengths emitted from the atmosphere given a set of parameters (such as temperature, water vapor mixing ratio, etc.) [4]. The RTM must account for all sources of radiation that in-
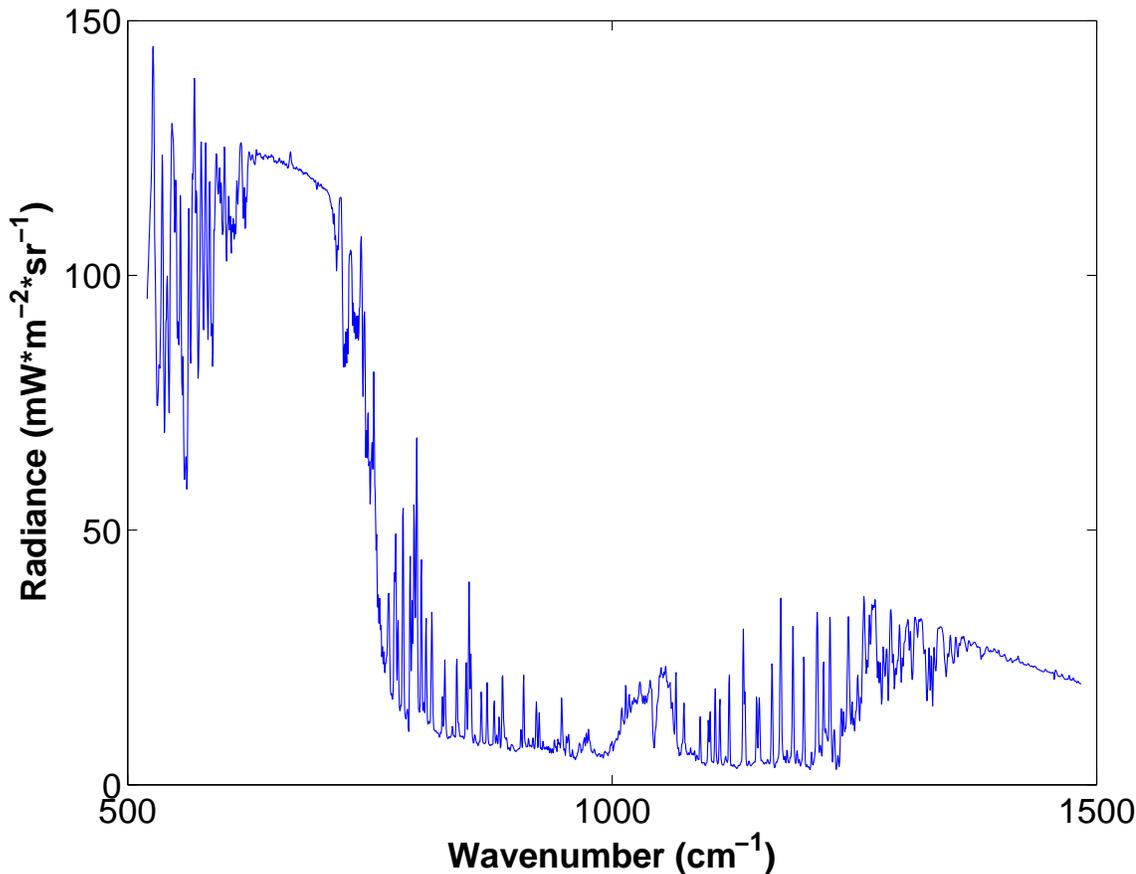
Fig. 1.1: Out-welling atmospheric spectra.

terface with the atmosphere. These radiative sources are pictorially described in figure 1.2. The RTM is known as a *forward model*, but what is truly desired for this application is the inverse of the RTM, which could model atmospheric parameters based on a measurement of radiation intensities at many different wavelengths. If this were possible, then remotely measuring parameters in the atmosphere could be achieved by measuring the light emitted from the atmosphere. Unfortunately, the RTM is nonlinear, and therefore not directly invertible; but methods have been developed over the past few decades which make good progress toward solving this problem [5,6].

## 1.4   Existing Methods for Inverting the Radiative Transfer Model

Fleming and Smith [5] have outlined several methods for inverting the RTM. Their
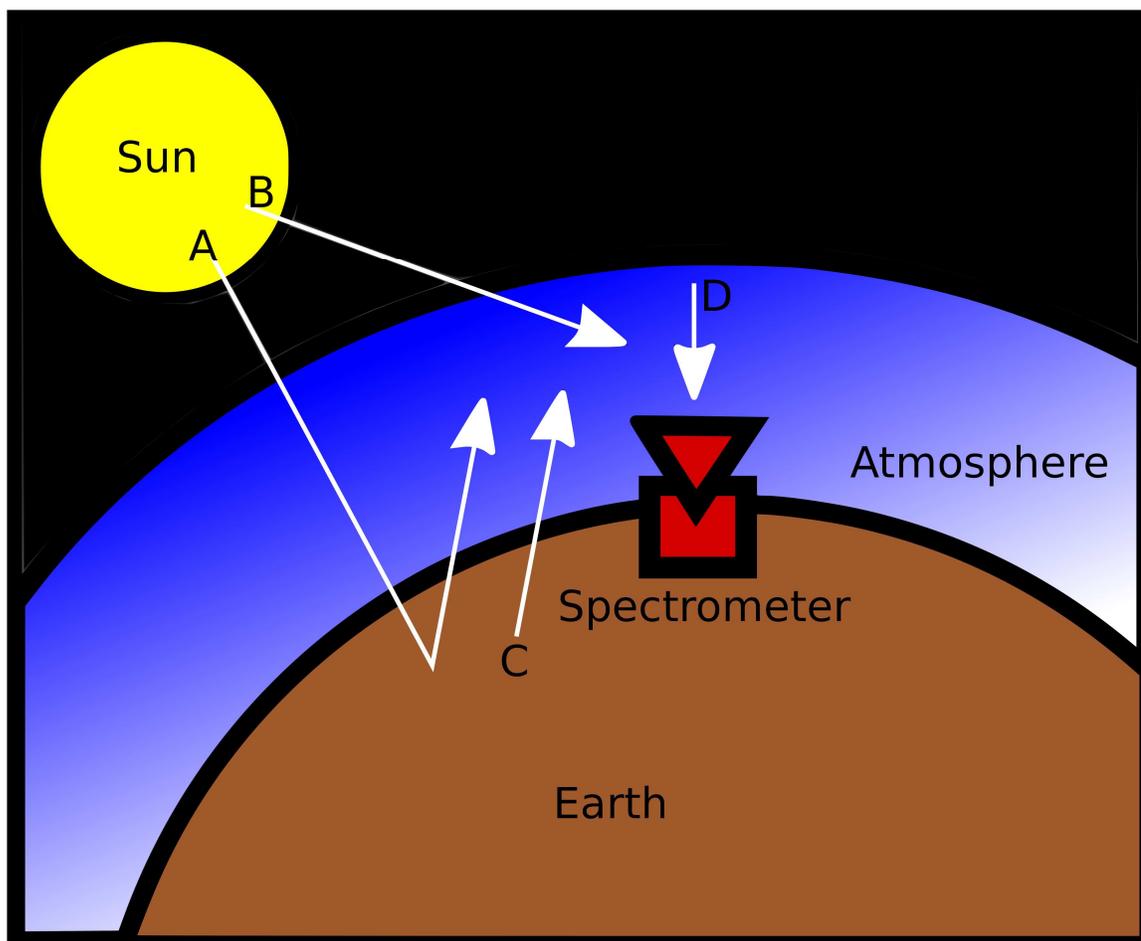
Fig. 1.2: Sources of radiation: (a) is light from the sun reflected by earth's surface and scattered in the atmosphere. (b) is light from the sun scattered in the atmosphere. (c) is blackbody emission from the earth scattered in the atmosphere. (d) is blackbody emission from the atmosphere itself.

study compares the performance of various iterative methods for nonlinear inversion. They were unable to identify a method that consistently outperformed all the others but mentioned that each method involves a tradeoff of some sort, for example the amount of necessary information and the importance of initialization parameters. Derivatives of the methods outlined by Flemming and Smith are still in use today, for example by Liu et al. in their work using super channels for the retrieval of atmospheric profiles [7].

## 1.5 Room for Improvement in Existing Methods

The methods in use by Liu et al. [7] for inverting the RTM overlook additional data that could be used to obtain better estimates of atmospheric parameters. It is believed that the strong temporal correlation that exists in the time sequence of these atmospheric parameters can be used to the advantage of an estimator. This fact is not exploited in iterative nonlinear inversion techniques. In addition, these existing methods can occasionally produce very inaccurate results, yet they provide no measure of confidence in the accuracy of their solution.

## 1.6 A New Method for Model Inversion: Particle Filtering

This thesis will detail the theory and implementation of a new method for inverting the radiative transfer model using a particle filter. The particle filter is well suited for problems that involve tracking the time evolution of a system state that has high temporal correlation [8]. It will be shown that this new method can capitalize on information contained in the problem that other methods are agnostic to, and that it can achieve better quality estimation of atmospheric profiles by including a metric of confidence in the given estimations.

## 1.7 Hampton University Experiment

To further research into inverting the radiative transfer model and to test the performance of the ASSIST II interferometer, Hampton University conducted a series of experiments to collect hyperspectral measurements of down-welling atmospheric radiation continuously over the course of several hours. This data was to be used in testing radiative transfer techniques and to attempt to recover a continuous measurement of atmospheric parameters for the corresponding time. The data set obtained from this experiment was used in the testing and development of the particle filtering solution described in this report.

# Chapter 2

# Experimental Setup

The data that are being used to for the testing and development of this new algorithm for radiative transfer model inversion was collected during an experiment conducted by Hampton University in April 2011. During the course of the experiment, 23.5 hours of down-welling spectral data were collected at a rate of approximately 25 samples per hour.

## 2.1  Hyperspectral Sensor

The ASSIST (Atmospheric Sounder Spectrometer for Infrared Spectral Technology) II is a mid-infrared $(500-3000 \text{ cm}^{-1})$ spectroradiometer. It was designed by LR Tech as a tool for atmospheric scientists to profile temperature and moisture content in the atmosphere. The device operates by performing Fourier Transform Spectroscopy on data collected by a Michelson interferometer.

### 2.1.1  Michelson Interferometers

A Michelson interferometer creates an interference pattern in light by first splitting a beam of light and then recombining the two beams after they have traveled different distances, so that the two beams will be out of phase [9]. The device is named after its inventor, Albert Abraham Michelson, and was used by him in the famous Michelson-Morley experiment of 1887. This experiment was a failed attempt to demonstrate the effect of "aether wind" on the speed of light.

A basic Michelson interferometer (depicted in figure 2.1) operates by shining a beam of light onto a 50/50 beam splitter, such as a half-silvered mirror. Both of the divergent beams are directed toward mirrors which reflect the beams back toward the beam splitter (now acting as a beam combiner). Because the two reflecting mirrors are different distances
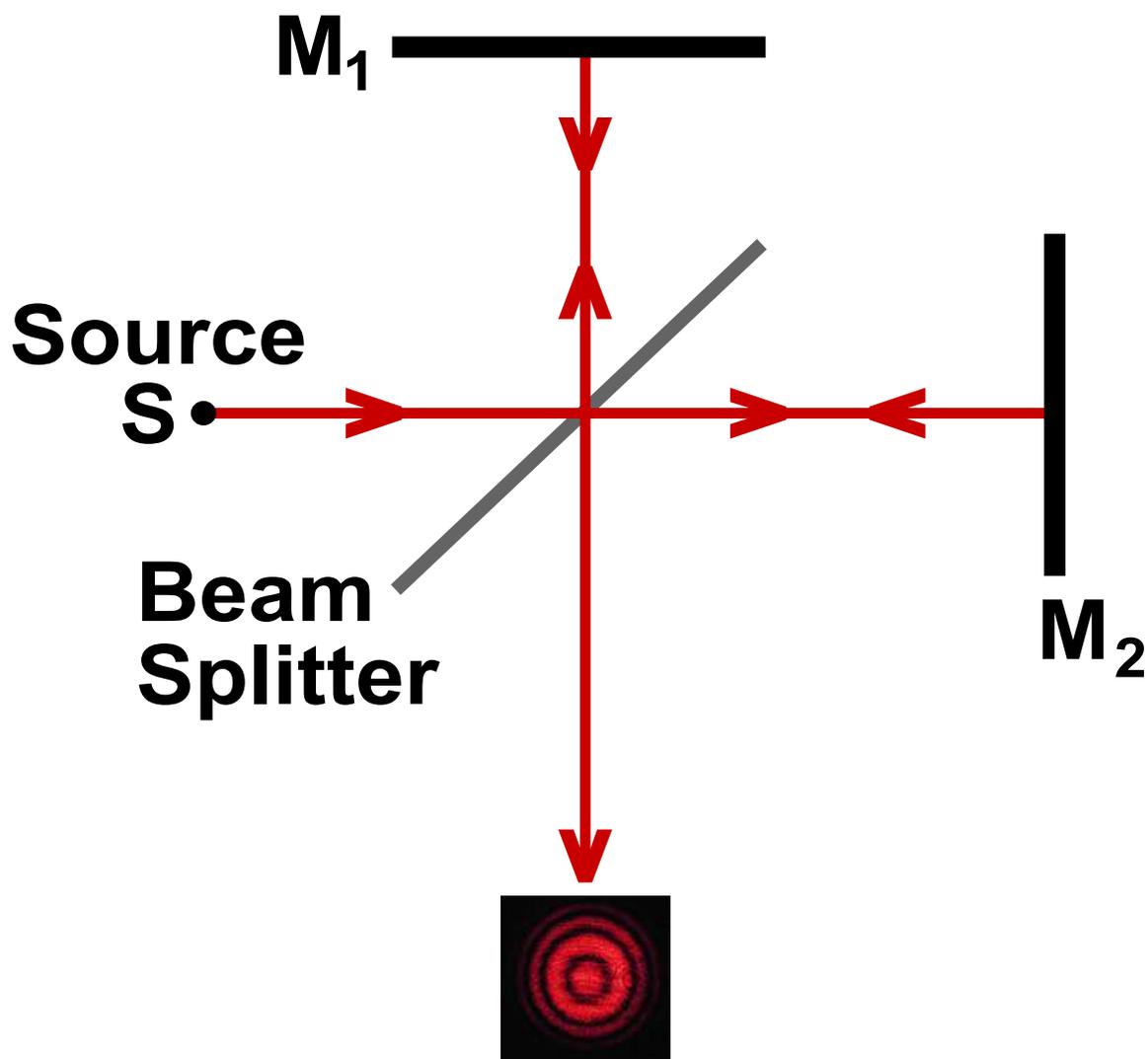
Fig. 2.1: Optical paths in a Michelson interferometer. Image credit: Stigmatella aurantiaca at English Wikipedia.

from the beam splitter, the two beams become out of phase with each other when they are recombined. A detector captures the interference pattern created by the out of phase beams after they are recombined.

When the interferometer is modified such that the position of one of the reflecting mirrors is configurable, it becomes possible to measure the intensity of a range of spectra of light entering the interferometer by a method called Fourier Transform Spectroscopy.

### 2.1.2 Fourier Transform Spectroscopy

A Michelson interferometer with a movable mirror allows a series of interferograms with varying time delay on one of the interfering beams to be measured. Spectral information is contained in the temporal coherence of the light, and can be retrieved through the Fourier Transform of these interferograms.

Let $I(p, \tilde{\nu})$ represent the intensity of light at a frequency $\tilde{\nu}$ for path length difference $p$ in the interferometer. This function can be expressed as

$$I(p, \tilde{\nu}) = I(\tilde{\nu})[1 + \cos(2\pi\tilde{\nu}p)],$$

$$I(p) = \int_0^\infty I(p, \tilde{\nu})d\tilde{\nu} = \int_0^\infty I(\tilde{\nu})[1 + \cos(2\pi\tilde{\nu}p)]d\tilde{\nu}.$$

This is a Fourier cosine transform of $I(\tilde{\nu})$. The inverse transform gives us

$$I(\tilde{\nu}) = 4 \int_0^\infty [I(p) - \frac{1}{2}I(p = 0)] \cos(2\pi\tilde{\nu}p)dp.$$

Therefore, the intensity of light at wavelength $\tilde{\nu}$ is a function of the intensity of light for path length difference $p$ as measured by the interferometer.

### 2.1.3 ASSIST II Specifications

The following specifications are provided by the maker of the ASSIST II, LR Tech:

- Spectral range: 500 to 5,000 cm$^{-1}$;

- Single plate KBr beamsplitter (self compensated);

- Nominal beam diameter at beam stop: 2.54 cm;

- Maximum beam divergence is 45 milliradians full angle;

- Spectral sampling: 0.5 cm$^{-1}$;

- HeNe laser metrology system (for digital sampling and mirror velocity control) mounted on pre-aligned assembly for easy replacement;

- Sampling rate (laser fringe rate: 32khz to 100 kHz, computer selectable);

- Throughput: 0.00805 sr cm$^2$;

- Start-up is with a white light zero path difference reference channels used to define ZPD position and sampling window; the white light is turned off during normal scanning;

- Purging capability of the interferometer enclosure;

- Dimensions: 12.4 inches x 13.2 inches x 7.8 inches high;

- Operating temperature range: 10° to 40° C.

## 2.2   Experiment Procedure

The experiment was carried out over a series of nine hour segments where the ASSIST interferometer was continually measuring data. Twice during the experiment a weather balloon was launched with a radiosonde attached to it to measure actual atmospheric parameters. The sensor does not function well in the rain so a rain sensor controlled a hatch door over the sensor to shield it during rain fall. During periods of rainfall, the data being collected by the sensor is incorrect (essentially just a measurement of the emissivity of the rain hatch door in front of it). This data is therefore tagged as bad data, and is not used in the atmospheric profile extraction process.

The purpose of the experiment was to perform a field test of the ASSIST II interferometer as a ground-based hyperspectral sensor and to collect a large amount of real world data to test extraction algorithms on. In the future the results from this experiment can help in developing algorithms for space-based hyperspectral data collection where the hyperspectral sensor is located on a satellite observing the surface of the earth. While ground-based remote sensing can still provide many advantages over weather balloon atmospheric measurement, space-based observation would provide the ability to observe atmospheric properties anywhere in the world.

## 2.3    Additional Data Available

In addition to the data collected during the experiment, statistical weather data from the area where the experiment was carried out is available to us. This data includes historical measurements and averages for temperature, water vapor mixing ratio, barometric pressure among many others. This data is important for the extraction algorithm by providing initialization parameters and covariances for the atmospheric profiles that we are trying to estimate.

## 2.4    Results

As a result of the experiment, two days worth of hyperspectral data were collected. The experiment was considered a success, the ASSIST II functioned properly and a large amount of high quality data is now available for use in testing extraction algorithms. Figure 2.2 depicts a 9-hour segment of the data collected during the experiment. Each column of pixels in the image represents a single measured spectra, similar to figure 1.1. While there is not much insight to be gained by this representation of the data, it is provided as an illustration to the subtle changes which take place in the radiance over the course of a day.
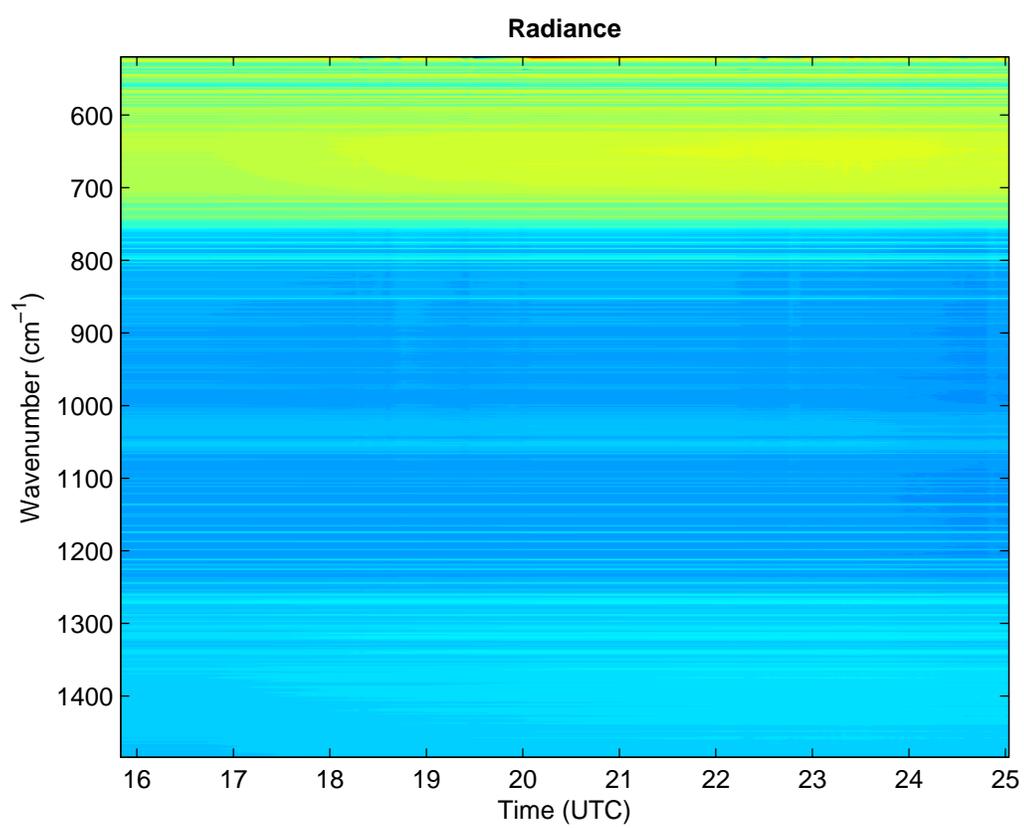
Fig. 2.2: Radiance data collected during the Hampton University.

# Chapter 3

# Radiative Transfer Models

## 3.1 Theoretical Background

A radiative transfer model describes the change in radiation intensity of light as it propagates through a medium [4]. The fundamental equation of radiative transfer at some wavelength $\nu$ is

$$\frac{1}{k_\nu}\frac{dI_\nu}{ds} = -I_\nu + \frac{j_\nu}{k_\nu}.$$

In this equation, $I_\nu$ is the radiation intensity along a ray path, $k_\nu$ is the local absorption, and $j_\nu$ is the volume emission. What this tells us is that as light propagates through a medium some of it will be absorbed by the medium at certain frequencies. The medium itself will also emit light at certain frequencies.

The radiance emitted from the atmosphere is dynamic, and is primarily affected by temperature, water vapor, aerosols, and trace gases. Because of this dependence, it is possible to develop a model for radiative transfer which can predict the out-welling radiance of an atmosphere with a given composition [7]. The inverse of this problem, calculating the composition of the atmosphere from measured radiance, is quite difficult.

One radiative transfer model used to predict up-welling atmospheric radiance is the AIRS model defined by Strow et al. [4] Monochromatic radiance at a wavelength $\nu$ leaving the top of a nonscattering, clear atmosphere is described as

$$
\begin{aligned}
R_\nu \;=\; & \epsilon_\nu B_\nu(T_s)\tau_\nu(p_s \to 0, \theta_{sat}) \\
& + \int_{p_s}^{0} B_\nu(T(p))\frac{d\tau_\nu(p \to 0, \theta_{sat})}{dp}dp \\
& + F_\nu^d \rho_\nu^t \tau_\nu(p_s \to 0, \theta_{sat}) \\
& + \frac{H_\nu}{\sec(\theta_{sun})}\tau_\nu(0 \to p_s, \theta_{sun})\rho_\nu^s \tau_\nu(p_s \to 0, \theta_{sat}).
\end{aligned}
$$

Each term in this model can be described as follows:

- $\epsilon_\nu B_\nu(T_s)\tau_\nu(p_s \to 0, \theta_{sat})$ is the blackbody emission of the surface of the earth, $\epsilon_\nu$ is the surface emissivity, and $B_\nu(T_s)$ is the Planck function;

- $\int_{p_s}^0 B_\nu(T(p))\frac{d\tau_\nu(p \to 0, \theta_{sat})}{dp}dp$ is the atmospheric emission;

- $F_\nu^d \rho_\nu^t \tau_\nu(p_s \to 0, \theta_{sat})$ is the down-welling atmospheric emission reflected by the surface, where $F_\nu^d$ is the down-welling thermal flux and $\rho_\nu^t$ is the reflectance of this flux by the surface;

- $\frac{H_\nu}{\sec(\theta_{sun})}\tau_\nu(0 \to p_s, \theta_{sun})\rho_\nu^s \tau_\nu(p_s \to 0, \theta_{sat})$ is the reflected solar radiation, where $H_\nu$ is the solar irradiance incident at the top of the atmosphere and $\rho_\nu^s$ is the solar reflectance by the surface.

These terms also include the atmospheric layer-to-space transmittance $\tau_\nu(p_s \to 0, \theta_{sat})$, meaning the transmittance from some pressure $p_s$ to space (zero pressure) at an angle $\theta_{sat}$, the satellite zenith angle as measured along the ray from the surface to the satellite [4]. This model has much in common with other radiative transfer models, including the model that is used by Hampton University in their experiments, LBLCALC. The primary difference between the AIRS model and LBLCALC is perspective. While the AIRS model is designed to model up-welling radiance from a satellite, LBLCALC models down-welling radiance to a sensor on the ground. Since we are using Matlab code from Hampton University as a starting point to develop our particle filter solution, we will use LBLCALC for our radiative transfer model.

## 3.2   Model Limitations

Because of the summing effect of the atmosphere on radiative transfer, there is a many-to-one relationship between an atmospheric profile and emitted spectra as calculated by the radiative transfer model. This fact can pose a significant problem for any inversion technique, because based on the radiative transfer model, two vastly different atmospheric

profiles could potentially produce identical spectra, as well as an infinite number of different atmospheric profiles between them. Different extraction techniques take different approaches to resolving this ambiguity (usually relying on historical measurements to hopefully initialize the problem somewhere close to the solution [10]) but unfortunately, there is nothing to be done to completely resolve the problem. This non-invertible model is just that: non-invertible. No solution achieved using radiative transfer alone will be 100 percent reliable. The only hope for this approach is to be able draw boundaries around what is likely to be the truth, and what is unlikely.

## 3.3 Matlab Implementation

A Matlab implementation of the LBLCALC radiative transfer model was provided by Hampton University at the outset of this project. It is termed a "fast, line-by-line" algorithm for calculating radiative transfer because of the way it approximates the atmosphere as discrete layers with constant parameters instead of a continuum. During the discussion of this Matlab code, the radiative transfer model will be referred to by its Matlab function name, *fastlblcalc*.

The execution time of *fastlblcalc* in its original state was not suitable for use in a particle filter solution. Current extraction techniques that rely on iterative nonlinear inversion only need to calculate spectra with the RTM a handful of times, so long execution time of *fastlblcalc* was not much of an issue. It will be shown that using the RTM with a particle filter will result in many more calls to *fastlblcalc*, enough so to make execution time a problem. *fastlblcalc* takes as inputs a 60-dimensional vector representing a temperature profile, and a 60-dimensional vector representing a water vapor mixing ratio profile. *fastlblcalc* produces as an output an estimate of spectral emission from an atmosphere with the given parameters. This output is calculated based on the LBLCALC Radiative Transfer Model.

One goal of this project is to ensure that the execution time of the new extraction method is not significantly longer than that of the original algorithm. In order to talk about execution times in a meaningful way, a pair of units for time measurement will be introduced.

The length of time required to process the entire Hampton data set using the original extraction algorithm will be known as 1 extraction time. The length of time required for one call to the original *fastlblcalc* function will be called 1 *fastlblcalc* time. Since execution times vary based on the host machine, these units will help to compare execution times between old and new methods.

For reference, 1 *fastlblcalc* time is approximately 5 seconds on an AMD Athlon 64 2 GHz processor with 4 GB RAM. If this function were to be used as part of a particle filter, *fastlblcalc* would need to be called once per particle at each time step. There are 234 time steps in the Hampton University data set that will be used in development, meaning that a filter with 50 particles would make a total of 11,700 separate calls to *fastlblcalc*. With our reference time of 5 seconds per *fastlblcalc* time, this equates to over 16 hours of computation time. Since 1 extraction time is about 20 minutes on our reference machine, this equates to about 48 times longer than a single extraction time. This would represent an unacceptable increase in execution time, but fortunately *fastlblcalc* can be optimized to greatly increase its speed.

*fastlblcalc* is known as a line-by-line method for computing radiative transfer. For a discrete set of wavelengths, radiative transfer is approximated by modeling propagation of light through a number of atmospheric "layers" where each layer is assumed to have constant parameters such as temperature, mixing ratio, aerosols, and trace gases.

*fastlblcalc* is tuned to be compatible with the ASSIST II spectrometer, which measures 2000 wavelengths in the range of $(500-3000 \text{ cm}^{-1})$. This means that *fastlblcalc* will compute a radiative transfer prediction 2000 times, one for each wavelength measured by the ASSIST, during a single call to the function. Nearly all of the atmospheric parameters which factor in to the radiative transfer model are predetermined based on historical data from the area of measurement. The only parameters which can vary, as mentioned earlier, are temperature and water vapor mixing ratio.

The basic structure of the *fastlblcalc* algorithm can be described as a nested "for" loop:

1: **for** $i = 1 : 2000$ **do**                          ▷ For each desired wavelength

2:     **for** $j = 1 : 60$ **do**                              ▷ At each layer of the atmosphere

3:          RTM(i,params(j)); ▷ Model radiative transfer for given atmospheric parameters

4:     **end for**

5: **end for**

## 3.4   Optimizations

The approach for optimizing this function is twofold:

1. Look for segments of code which are written as inefficient loop operations that could benefit from Matlab's more optimized matrix-vector operations;

2. Find redundant calculations inside of loop statements that can be precalculated outside of the loop once instead of at each iteration.

Fortunately, there were several examples of each case which could be optimized. Some of the code inside of *fastlblcalc* was originally written in Fortran, and later ported to Matlab code. During the port, not much care was taken to take advantage of Matlab's capabilities in matrix-vector operations. As a result, several code loops were able to be reduced to much faster matrix-vector operations.

The greatest step in optimization came from eliminating redundant calculations. Such redundancies were identified at nearly every level of execution inside of *fastlblcalc*. To better illustrate these optimizations, a set of "execution levels" will be defined:

- Level 1: Not executed during *fastlblcalc*;

- Level 2: Executes once per call to *fastlblcalc*;

- Level 3: Executes 2000 times per call to *fastlblcalc*;

- Level 4: Executes 120,000 (2000 × 60) times per call to *fastlblcalc*.

Some examples of redundancies that were eliminated are: Identifying a setup routine at level 2 which could be moved to level 1; identifying calculations at level 3 which did not depend of wavelength, and could therefore be moved to level 2. The most dramatic

optimization comes from the discovery of large portions of code at level 4 which also did not depend on wavelength and could therefore be moved up to level 2.

### 3.5 Performance

The result of these optimizations was to reduce the execution time of *fastlblcalc* down from 1 *fastlblcalc* time to approximately 0.16 *fastlblcalc* time. This reduction brings our estimate of a 16-hour extraction time with 50 particles down to 2.5 hours. This is still significantly longer than 1 extraction time, (it is actually about 10 extraction times) however it does bring it much closer, and is more than fast enough for real-time data processing.

It will be shown in the following chapters that 50 particles may be more than is necessary, and the running time of the particle filter begins to decrease very quickly as the number of particles decrease.

Care was taken to ensure that the numerical accuracy of the results from *fastlblcalc* were not disturbed due to the optimization. By comparing results obtained by the original *fastlblcalc* to results from the optimized *fastlblcalc* on identical data sets, the mean squared error between the two was observed to be less than $10^{-14}$.

# Chapter 4

# Proposed Solution

## 4.1 Theoretical Background of Particle Filters

A particle filter is a Monte-Carlo method for modeling system state dynamics. It is similar in function to the Kalman filter. The Kalman filter can provide optimum state estimation under the constraints of linear state dynamics and Gaussian noise [11]. The particle filter, on the other hand, is not limited by these constraints, and is free to operate in nonlinear systems with non-Gaussian noise [12]. This is a tremendous advantage in the case of radiative transfer modeling, due to the nonlinear nature of the problem.

The standard particle filter assumes a system with state dynamics defined as:

$$x_{t+1} = F(x_t, n_t),$$

where $x_t$ is the system state at time $t$ and $n_t$ is a noise process. $x_t$ is therefore a stochastic function of the system state at the previous time. Also assumed in the particle filter is the notion that the system state $x_t$ is not directly observable. The only information available to the observer comes through what we will refer to as the measurement function:

$$y_t = G(x_t, \nu_t).$$

Each measurement $y_t$ is a stochastic function of the system state $x_t$. We will assume that G is not invertible, so that $x_t$ cannot be directly inferred from measurement $y_t$. The goal of the particle filter can be smoothing, state estimation, or prediction. For the purposes of this problem of estimating atmospheric parameters based on spectral measurements, system state estimation is our primary goal.

The particle filter algorithm begins by making a guess for the initial state of the system,

$x_0$. A prior distribution for $x_0$ is also assumed. How this probability density function (PDF) is found will be discussed in the next chapter. We define a random variable with this PDF from which we will make a sequence of random draws. Each random draw from the PDF is called a particle. The set of particles randomly drawn from the PDF can be thought of as a discrete approximation of the PDF, as depicted in figure 4.1. As the number of drawn particles increases, the approximation to the PDF gets better. The number of particles necessary to adequately characterize the PDF is generally at the discretion of the designer. Too many particles will needlessly increase the computational load of the algorithm, while too few particles will misrepresent features in the PDF and cause errors. Often the driving factor in the number of required particles is the dimensionality of the system state. In most cases, the number of particles required to characterize the PDF increases exponentially with the dimensionality of the system state variable.

Once we have our set of discrete samples of the PDF, we begin the iterative phase of the algorithm. First, each particle is propagated one time step into the future by means of the state dynamics function $F(x_t, n_t)$. The notation used for particles will be $x_t^i$, representing the $i$th particle at time $t$. This gives us the set of particles $x_{t+1}^i = F(x_t^i, n_t)$. We can now use the measurement available to us at time $t + 1$ to determine which of these particles are most likely to be an accurate representation of the system state $x_{t+1}$. Each particle $x_{t+1}^i$ is transformed into something comparable with the measurement $y_{t+1}$ through the measurement function $G(x_t, \nu_t)$. Each transformed particle $y_{t+1}^i = G(x_{t+1}^i, 0)$ is compared to the measurement

$$m_i = ||y_{t+1} - y_{t+1}^i||_2.$$

Note that the 2-norm is used as an example. Other ways of comparing $y_{t+1}$ to $y_{t+1}^i$ may be appropriate depending on the situation. The set of measurements $m$ are then transformed into a set of weighting values which serve to rank each particle based on how close the transformed particle is to the measurement. The weights take the form

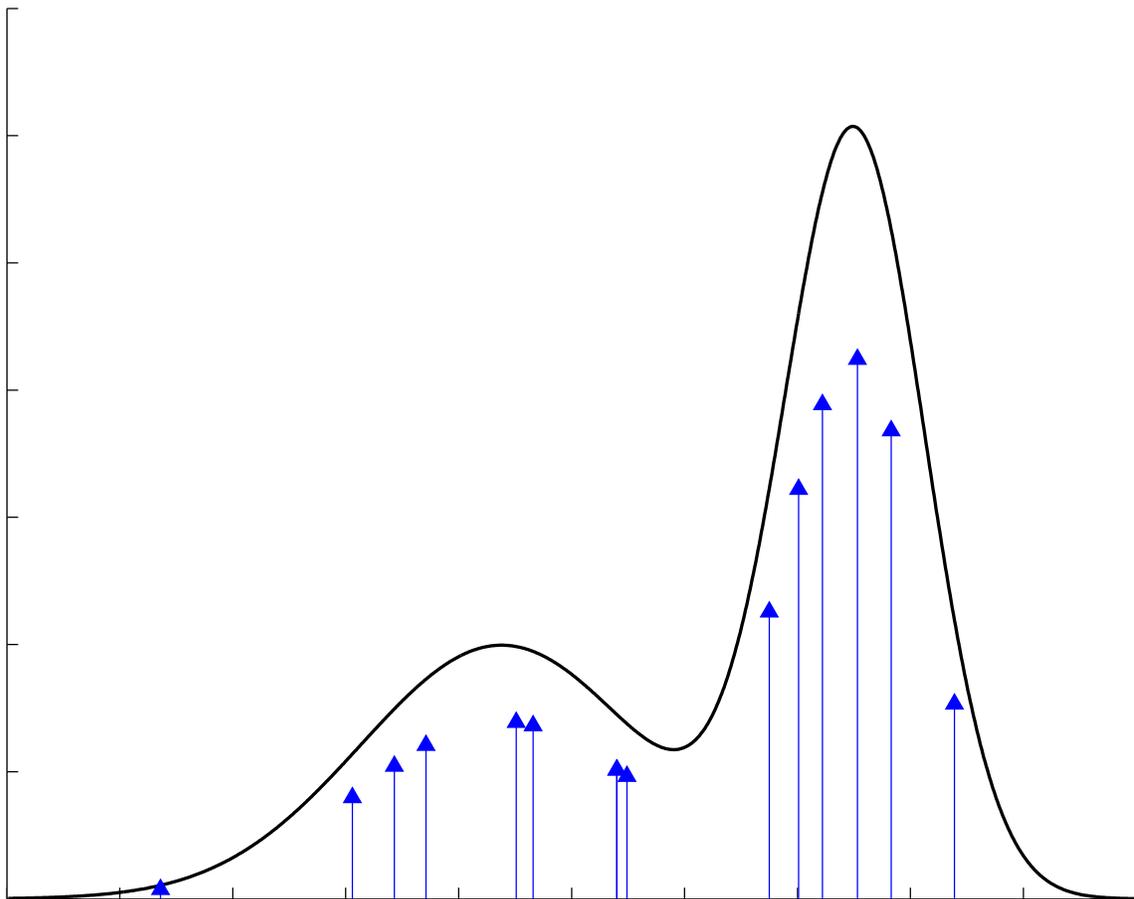$$w_i = \frac{1/m_i}{\sum_i 1/m_i}.$$

Fig. 4.1: A discrete set of particles approximating a PDF.

With our newly found set of weighting values, we are almost ready to begin the iterative process again at the next time step. First, however, we must define a new PDF to characterize $p(x_{t+1}|x_t)$. The particle weights tell us which particles are more likely to be close to the true system state. In order for the particle filter to characterize the PDF, a higher probability in the PDF will translate into a higher number of particles in that area of the PDF. Therefore we randomly sample a new set of particles for the next iteration of the particle filter, and the pool that we will sample from is the previous set of particles. The likelihood of each particle being selected for inclusion in the next iteration is dependent upon its weight. Particles with large weights may be selected multiple times for inclusion in the next time step, while particles with low weights may not be included at all in the next round. This newly selected set of particles is now ready for the next round of itera-

tion, where they will be randomly propagated one time step into the future according to $x_{t+2}^i = F(x_{t+1}^i, n_t)$. At each time step, an estimate of the current state $x_t$ can be produced by taking the weighted average of all the particles, $\tilde{x}_t = \sum_i w_i x_t^i$.

It is important to note from this description of the particle filter algorithm that at no time is the measurement function $G(x_t, \nu_t)$ inverted. Because of this, the particle filter lends itself particularly well to problems with nonlinear or incomplete observations. Also of note is how discretely approximating the PDF through random samples means that there are not restrictions on the distribution as there would be with a Kalman filter.

One common problem with particle filters is the notion of degeneracy. Degeneracy happens when the majority of the weight is distributed among a relatively few number of particles. This is a situation to be avoided, because it indicates that there is a poor representation (few particles) of the statistically significant (high weight) areas of the PDF, and an over-representation (many particles) of statistically insignificant (low weight) areas of the PDF. A common way to combat filter degeneracy is to add a "resampling" step at the end of each iteration. The resampler only acts when it detects a weighting distribution that is too unbalanced. When the resampler acts, very low weight particles are dropped from the solution and high-weight particles are split into multiple particles. The effect of this is to achieve greater resolution on the approximation of the PDF in areas of interest. Without resampling, the low resolution in the approximation can fool the particle filter into moving off course and away from the true system state.

## 4.2   Adapting Extraction as a Particle Filter Problem

We approximate the atmosphere directly above a ground-based sensor looking upward as 60 layers with constant parameters at fixed altitudes. The parameters of interest in this application are temperature and water vapor mixing ratio. To estimate these parameters from spectral measurements, we construct a particle filter.

- Let $\mathbf{t}_k$ = temperature, $\mathbf{q}_k$ = water vapor mixing ratio; at time $k$

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{t}_k \\ \mathbf{q}_k \end{bmatrix} \qquad \begin{matrix} \mathbf{t}_k \in \mathbb{R}^{60} \\ \\ \mathbf{q}_k \in \mathbb{R}^{60} \end{matrix}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{n}_k.$$

- Let $\mathbf{y}_k$ be the observation at time $k$; $\mathbf{RTM}(\mathbf{x})$ is the radiative transfer model

$$\mathbf{y}_k = \mathbf{RTM}(\mathbf{x}_k) + \nu_k.$$

The random movement $\mathbf{n}_k$ drives the system forward based on the covariance of historical atmospheric measurements of the local area. This covariance is shown in figure 4.2. A more advanced model for atmospheric dynamics could potentially improve the performance of the particle filter significantly.

The probability density function (PDF) of the system state is approximated by a set of $N$ particles:

$$p(\mathbf{x}) \approx \sum_{i=1}^{N} w_i \delta(\mathbf{x} - \mathbf{x}^{(i)}).$$

## 4.3   Benefits of Particle Filter over Nonlinear Inversion

Using a particle filter to estimate atmospheric parameters can potentially achieve better results than existing iterative nonlinear inversion techniques by taking advantage of the highly correlated nature of atmospheric dynamics. With a good model for the time evolution of atmospheric parameters, the particle filter can track these parameters, and at the same time offer a confidence measure for the estimate based on the particles themselves. When there is uncertainty in the estimate, i.e. a flat-looking region of the system state PDF, the particles will naturally disperse to fill the space. Conversely, the particles will tend to coalesce around high-likelihood estimates. Therefore, the variance of the particles in the filter can be an effective measure of confidence in the estimate. This measure of confidence in the estimate is something not available in current RTM inversion methods, despite their inaccuracies.
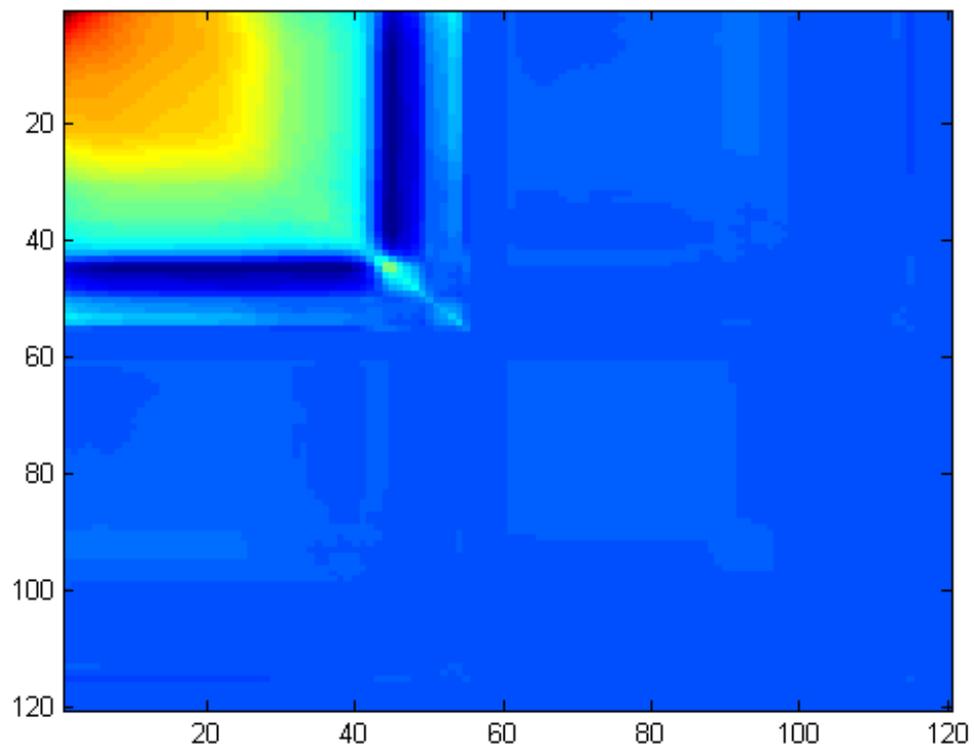
Fig. 4.2: Covariance of historical measurements of temperature and water vapor mixing ratio.

# Chapter 5

# Breaking the Curse of Dimensionality

## 5.1   Why Particle Filtering in High Dimensions Does Not Work

A particle filter represents a discrete approximation of a continuous probability density function. Each particle in the filter can be thought of as a scaled Kronecker delta function at a specific point in the PDF. At any time in the particle filtering process, an estimate of the system state can be obtained by taking the weighted average of all of the particles. This performs the same function as finding the expected value of the PDF which the particles are approximating. It can easily be seen that this method for state estimation would not lend itself well to a situation where the probability of the system state was highly multi-modal. For example, the expected value of a bi-modal PDF would likely be in the low probability region between the two modes, which is unlikely to be close to the actual system state. So, we see that although a unimodal noise model is not required for particle filters to function, some adaptation may be necessary for it to function well if multimodal noise is prevalent in the system.

We have said that particles in a particle filter approximate a continuous PDF. What should be obvious is that in order for the particle filter to accurately estimate the system state, the particles must be a *good* approximation to the PDF. Consider again a bi-modal PDF representing the probability of a system state. If there was only one particle (see figure 5.1) with which to approximate the PDF, where should it be? Certainly not at the expected value, because then the only choice for estimating the system state would lie directly in the center of an area that is highly unlikely to represent the system state. However, if the particle were placed in the center of one of the two modes, the particle may be very close to the true system state, or it may be even farther from the system state than before. Good approximation of high likelihood areas of the PDF are vital to accurate estimation, meaning

as many particles as can be computationally afforded.

As the number of particles available in the estimation becomes large, the approximation to the PDF approaches perfection. Unfortunately, dealing with a large number of particles becomes very computationally intensive. During the particle filtering algorithm, every particle must produce an observation estimate based on the measurement equation defined in the filter. In some cases the measurement equation may be very computationally simple, and filtering with thousands of particles might have little effect on the running time of the algorithm. In the case of this application, the measurement equation is the radiative transfer model, which is very computationally intensive, and even as many as 100 particles would make the extraction take too long to be palatable. Unfortunately, even the running time of the measurement equation turns out to be of little consequence where dimensionality is concerned. The number of measurements to be calculated increases linearly with the number of particles in the filter, but the number of particles needed for the filter to avoid collapse scales exponentially with the dimension of the system state.

To show that this is the case, consider a PDF representing the position of a target as uniformly distributed between zero and ten meters. If we impose as a constraint that there must be no more than one meter between particles in order to avoid filter collapse, then in the single-dimensional case a minimum of nine particles is needed. In the 2-dimensional case, where the target is now somewhere on a plane, the number of particles required to satisfy the constraint becomes nine squared. If the target were somewhere in a 3-dimensional volume, it would take nine cubed particles, etc.

As the number of dimensions increases, the number of required particles can quickly get out of hand. This is the reason that the standard particle filtering literature advises against particle filtering in many dimensions [13], and perhaps why particle filtering has found such enthusiastic response in the areas of target tracking in two or three dimensions. When faced with idea of trying to adapt a particle filter to avoid collapse in 120 dimensions, it does appear to be a daunting task.
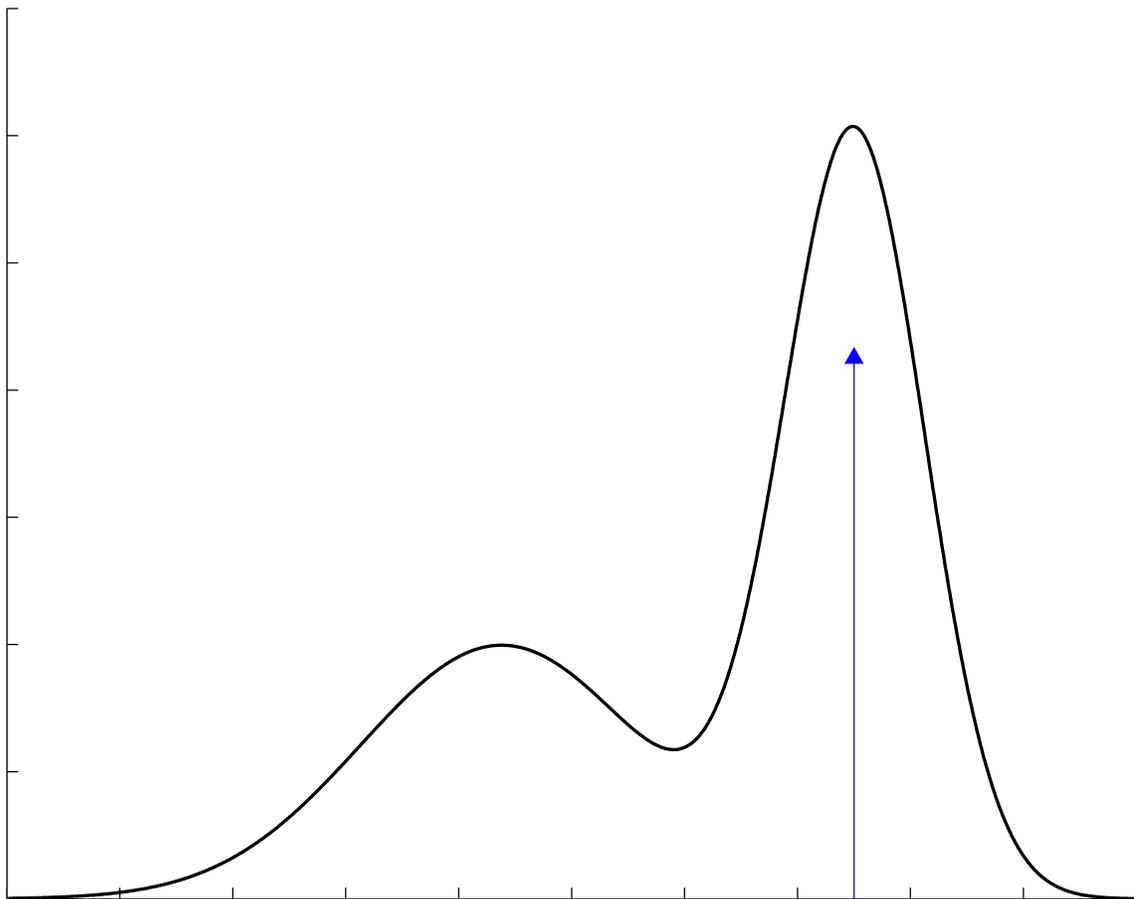
Fig. 5.1: A poor approximation for a bi-modal PDF.

Figure 5.2 shows the result of trying to track the temperature and water vapor mixing ratio of the atmosphere using the unmodified particle filter described in the previous chapter. The figure is a plot of error in the estimate compared to spectral measurements. This comparison is made through the RTM, which transforms the estimate into a spectral prediction, which can be compared directly to the observed spectra. Due to the high number of dimensions in the system state, the particle filter cannot track a minimal-error solution without an unfeasibly large number of particles. We will propose a modification to the particle filter algorithm which will help the filter track the system state with a limited number of particles. A discussion on the minimal-error curve in the figure can be found in Chapter 7.
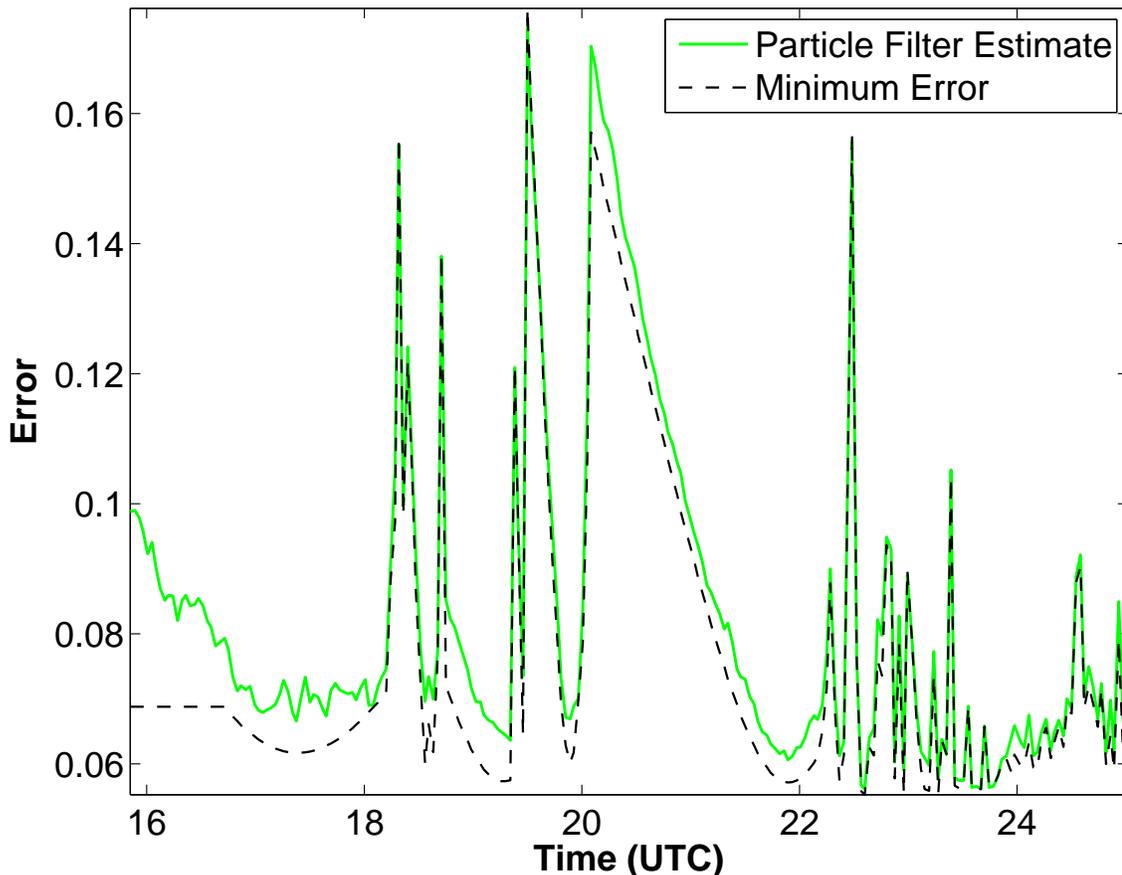
Fig. 5.2: Error of particle filter estimate, compared to the minimum error achievable.

## 5.2 Previous Work in High-Dimensional Particle Filters

Fortunately (or perhaps unfortunately, if using particle filters in high dimensions someday starts to get a lot of attention) we are not the first to be investigating this problem. P. J. van Leeuwen has published the results from his investigation into using particle filters for very high-dimensional geophysical systems [14]. He claims that information from future observations can be incorporated into the particle filtering algorithm in the form of a proposal density which serves to draw particles toward the observations. Because of this forcing of the particles toward the observations, the need for strong particle support over the entire PDF is diminished. The particles stay corralled close to the observation, so there is relatively good approximation in the area most likely to represent the true system state. The cost of this approach, however, is that there are very large areas of the PDF which

have zero particle support, and hence very poor approximation. As a result of this, this method may not hold up well to an environment rich in statistical outliers, as it depends on the system dynamics to be very predictable.

van Leeuwen makes bold claims in his work that with the inclusion of this proposal density, a very small number of particles would be able to approximate systems with thousands of dimensions without collapse. Also mentioned is that the implementation of the proposal density is flexible [15]; any rule which effectively draws the particles toward the observation and keeps them close enough to each other so as to all be weighted nearly equally should work.

## 5.3   A Surprisingly Effective Proposal Density

We recognize from van Leeuwen's work that there is flexibility in how we choose to draw the particles toward the observation. We also recognize that comparing particle locations, which reside in the system state space, to observations, which reside in the measurement space, is not trivial. We have already established that the radiative transfer model which transforms a vector from system state space into measurement space is highly nonlinear. In addition to the nonlinear relationship in the radiative transfer model, the measurement space has 2000 dimensions, while the system state space has only 120. All of this leads to a difficult time in determining what kind of movement of the particles in the system state space will correspond to them being more closely aligned with the observations in the measurement space.

van Leeuwen's suggestion for the proposal density requires that the measurement equation be linear, which is not the case with our radiative transfer model. Therefore, we must investigate just how "flexible" the choice of a proposal density is, to see if one can be adapted to work for our problem.

Since we recognize that comparing between measurement space and system state space is difficult, we rely on the measurement equation itself for the answer. At any given time in the particle filtering algorithm, each particle is assigned a weight corresponding to the likelihood that that particle is the actual system state. These weights are derived using

the measurement equation. By using the particle as an input atmospheric profile to the radiative transfer model, the model produces as an output a spectra which can be compared to what was actually measured with the interferometer. The closer that the model spectra is to the actual measured spectra, the more weight the particle will receive, as it is perceived that that particle is more likely to represent the true system state. Therefore, we recognize that drawing the particles toward the particle that possesses the highest weight is likely to correspond well to drawing the particles toward the observation. The process for this modification to the standard particle filter is summarized like so.

- In order to draw the particles toward a point in the system state space ($\mathbb{R}^{120}$) that corresponds to a point in the observation space ($\mathbb{R}^{2000}$), we recognize that drawing the particles toward the particle with the highest weight is likely to be drawing them toward the observation.

- Let $\mathbf{x}_k^{[i]}$ represent particle $i$ and $w_{i,k}$ represent the weight of particle $i$ at time $k$.

- Select the highest-weighted particle

$$j = \arg\max_i w_{i,k}.$$

- Draw the other particles some distance toward it

$$\mathbf{x}_{k+1}^{(i)} = \theta\mathbf{x}_k^{(i)} + (1-\theta)\mathbf{x}_k^{(j)} + \mathbf{n}_k \qquad\qquad 0 \leq \theta \leq 1.$$

- Adjust the weights

$$w_{i,k+1} = \frac{1}{||\mathbf{RTM}(\mathbf{x}_{k+1}^{(i)}) - \mathbf{y}_{k+1}||^2 \sum_{j=1}^{N} ||\mathbf{RTM}(\mathbf{x}_{k+1}^{(j)}) - \mathbf{y}_{k+1}||^2}.$$

This proposal density is a very simple and intuitive approach, but it will be shown that it is indeed an effective one. Figures 5.3 and 5.4 give a graphical representation of this process.
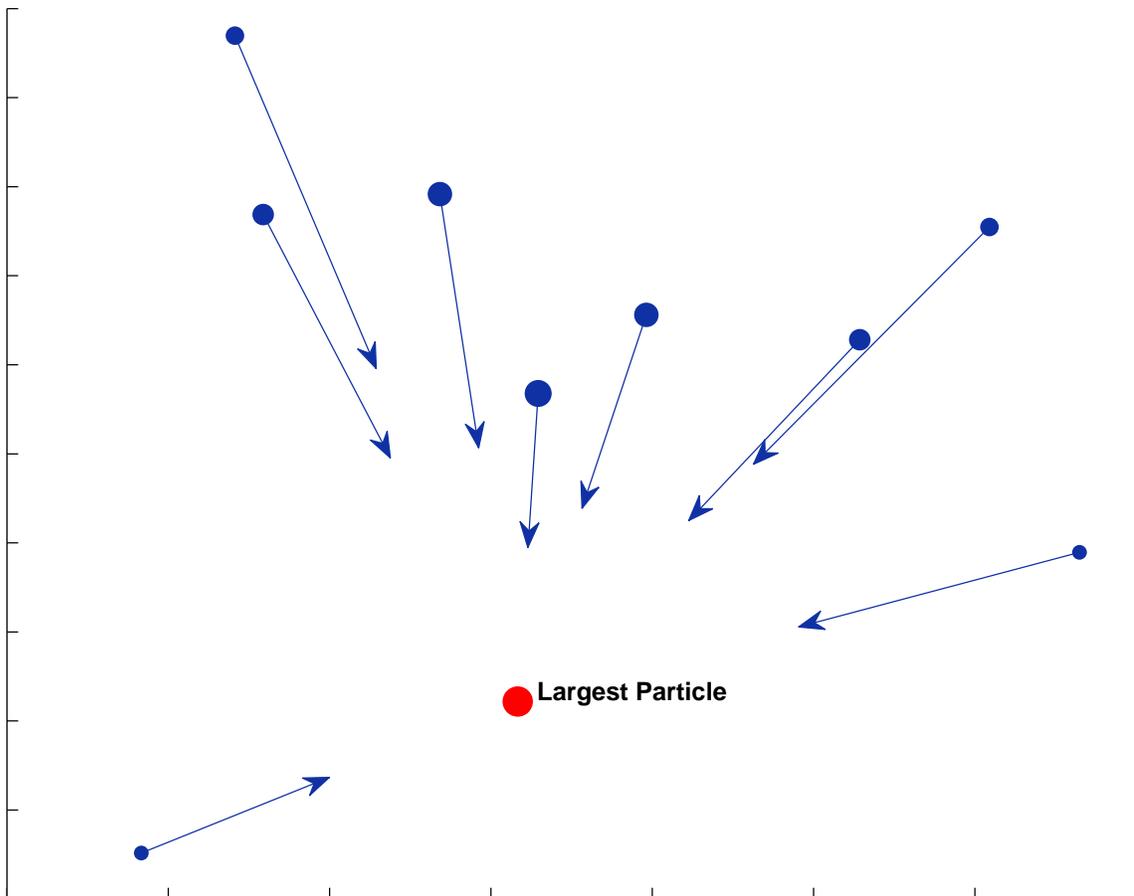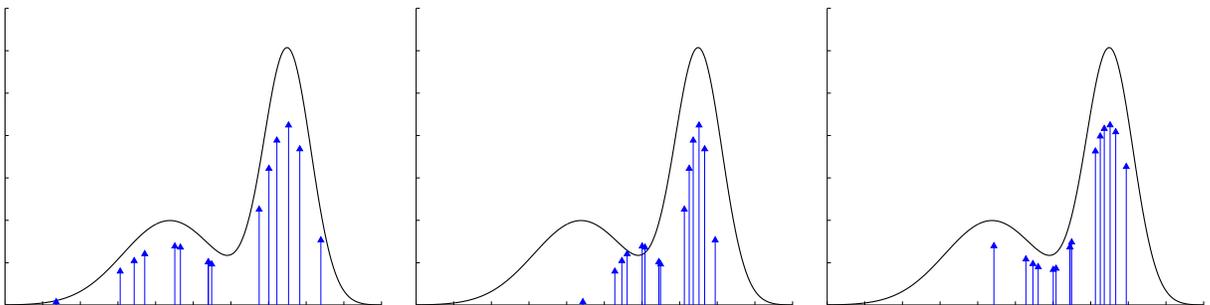
Fig. 5.3: Drawing particles toward the observation.



Fig. 5.4: PDF approximation improves in high-likelihood areas by drawing the particles close together.

# Chapter 6

# Working With Ambiguous Measurements

## 6.1   Ill-Posed Problems

There is a significant limitation to the notion of inferring atmospheric parameters from hyperspectral measurements. While the radiative transfer model can predict observed spectra based on atmospheric parameters reasonably well, mapping an observed spectra back to a set of atmospheric parameters is not unique. In fact, there is an infinite set of atmospheric parameters that map to an arbitrarily observed spectra according to the radiative transfer model. This makes ranking atmospheric parameters according to best model fit impossible.

Consider the two temperature profiles shown in figure 6.1, and an associated observed spectra. Our goal might be to determine which of these temperature profiles is more likely to represent the actual temperature of the atmosphere at the time this spectra was observed, according to our radiative transfer model. This would prove to be a very difficult task. Both of these profiles match the observation equally well, according to the radiative transfer model. Better, in fact, than the actual temperature profile of the atmosphere at the time the spectra was measured based on weather balloon measurement. It turns out that there is a continuum of temperature profiles between these two that all give comparable fit to the same measured spectra. In terms of a particle filter, these continua of equal fit represent deep grooves that the particles become trapped in while exploring the solution space, but these grooves tend to lead the particles away from the true solution rather than toward it.

The question becomes: Is there really sufficient information contained in spectral emission of the atmosphere to infer anything about atmospheric parameters? Can we have any certainty about a solution arrived to by means of the radiative transfer model when we know that the model cannot distinguish between the truth and a myriad of impostors?
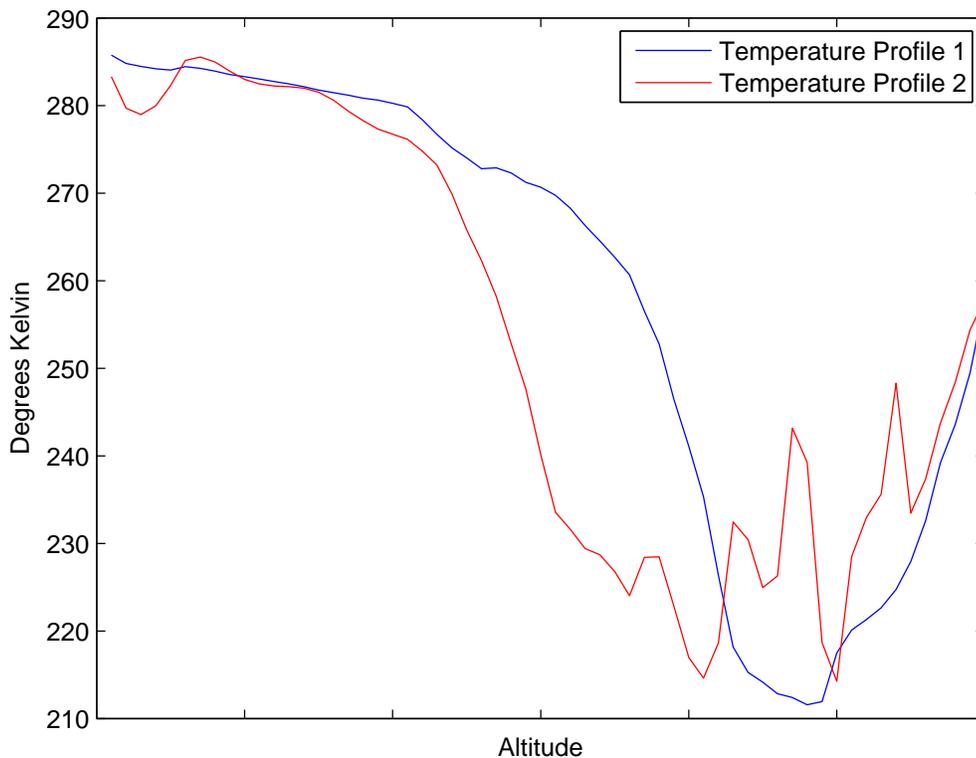
Fig. 6.1: Two temperature profiles that produce nearly-equal spectra in the radiative transfer model.

It is obvious that something besides model fit needs to influence weight assignment to particles. If model fit were the only metric to judge important particles, then the particles would be more or less free to roam along the continuous space where model fit is approximately equal, and they would invariably wander away from the true solution. Instead, we need a weighting scheme that is influenced by multiple metrics, model fit being one of them, so that when faced with multiple particles with comparable model fit, the filter can use additional information to select the particles that are truly more likely to accurately represent the system state.

One additional metric that can be employed is a measure of how much a given atmospheric profile resembles other profiles observed empirically for a given geographic area. The idea is that for any particle we can assign it a measure of how "believable" it is, based on

previous observation of that atmospheric parameter. For example, there may be a particle that fits the model very well, but represents an atmospheric profile that is physically very unlikely to actually occur in nature. This particle should receive a lower weight than a particle that also fits the model well, but represents an atmospheric profile that very often occurs in nature, since that particle is more likely to represent the truth.

There is more than one approach to determining this measure of particle viability. One approach would be to model each atmospheric parameter as a random variable based on historical observations of the area. Particles that fall close to the mean would receive more weight than particles that fall far from it. There are disadvantages to this approach, however. If atmospheric parameters vary too much, then the mean of historical observations of that parameter could be meaningless (pun intended). In fact, the mean of all observations could itself represent an atmospheric profile that is statistically unlikely to occur in nature. Care would need to be taken to not make faulty assumptions about the distribution of the random variable such as unimodality, etc.

A possible alternative to modeling parameters as random variables is the idea that actual atmospheric parameters can be closely approximated by a sparse combination of instances of that parameter observed previously. For example, if we have a collection of temperature profiles for a given area measured by weather balloons, it is theorized that common temperatures can be easily synthesized by a sparse combination of these observations, while uncommon temperature profiles cannot be synthesized sparsely. Therefore, a metric of believability for a given profile can be obtained based upon how accurately that profile can be approximated by a sparse combination of previous observations of the atmospheric parameter.

There is more than one way to incorporate this sparse metric into the particle filter. The most direct way would be to make a sparse approximation to every particle at each time step, and assign weights based on the fit of that approximation. The weights assigned to the particles would also be influenced by model fit, as discussed previously, so the two metrics both influence the final weight of each particle.

## 6.2   Need for Accurate Dynamics

In the case of radiative transfer model inversion, there is a continuum of possible atmospheric parameters which all map to the same spectra. Because of this, it is only natural that when the particles in the filter exist close to the true state of the atmosphere, several of them will lie in or close to this continuum. All particles in this continuum will receive equal weights in the particle filtering algorithm, meaning that once the particles come in contact with the continuum, they are free to explore it in its entirety without penalty. This continuum of equal spectra extends well beyond the realm of physical possibility for atmospheric parameters, which could be easy to identify when the particles reach a state that just could not possibly be true. However, in order to reach such absurd locations, the particles would have had to travel a space of possible (but increasingly less likely) states first. Being able to detect when the particles stopped tracking the true solution and began to wander away from it can be difficult.

This is not a unique problem in particle filtering. Particle filters can still be very useful in cases where the measurement contains incomplete data, even though it gives rise to this very problem. Consider as an example a particle filter tracking a moving target on a 2-dimensional polar coordinate space, where the only measurement available is the angle of the target; the magnitude is unknown. With the combination of an imprecise model for the dynamics of the target, a restricted number of particles, or perhaps just a large amount of measurement noise, it is possible for the particle filter's estimate of the target position to diverge sharply from the actual position, even while making estimates that agree very certainly with the measurements.

This problem can often be avoided with a tighter system state dynamics model. The certainty of the model will propel the particles in the right direction and make up for missing information. However, in the case where the measurement does not contain complete information, you may never be able to avoid the need for occasional recalibration. The reason is that once you have diverged from the true system state, but are instead wandering through a space that the model is indifferent to, you may never reconverge with the true

solution again. Even more, if reconvergence did happen, there would be no way to detect it without a measurement with complete data.

In the case of radiative transfer model inversion, a relatively good way to obtain a complete measurement is by way of a radiosonde and a weather balloon. This is not ideal, since the dependence on weather balloon measurements is exactly what this method is supposed to eliminate the need for. However, it may be possible to "reset" the particle filter using confident forecast data and historical measurements for the area.

# Chapter 7

# Results

This chapter details the results of the extraction algorithm based on the modified particle filter that we derived previously. Figure 7.1 gives an in-process view of the extraction algorithm, with individual particles depicted in green and the INLI estimate for the given time step in red as a reference.

## 7.1  Summary of Innovations

The code for implementing the radiative transfer model in Matlab has been significantly optimized so that the running time for calculating each spectral estimation is reduced by an order of magnitude.

A proposal density was incorporated into the particle filter which draws the particles towards the particle most likely to be representative of the system state.

Using the covariance of other extractions and historical weather data from the area available to us the weights of the particles are not solely dependent on model fit but also upon a likelihood based on resemblance to typical weather for the area.

## 7.2  Comparison to Iterative Nonlinear Inversion

One of the fundamental problems in developing methods for inverting the radiative transfer models is not ever being able to say with certainty whether or not any estimate of atmospheric parameters is correct. We simply do not have a true measurement to compare against. The closest thing to a true measurement we have available to us is weather balloon data. As discussed previously, weather balloon measurement is only as reliable as the straightness of its ascension route; but since it is the best we can do, we have little choice but to hold it up as the standard to compare atmospheric parameter estimates
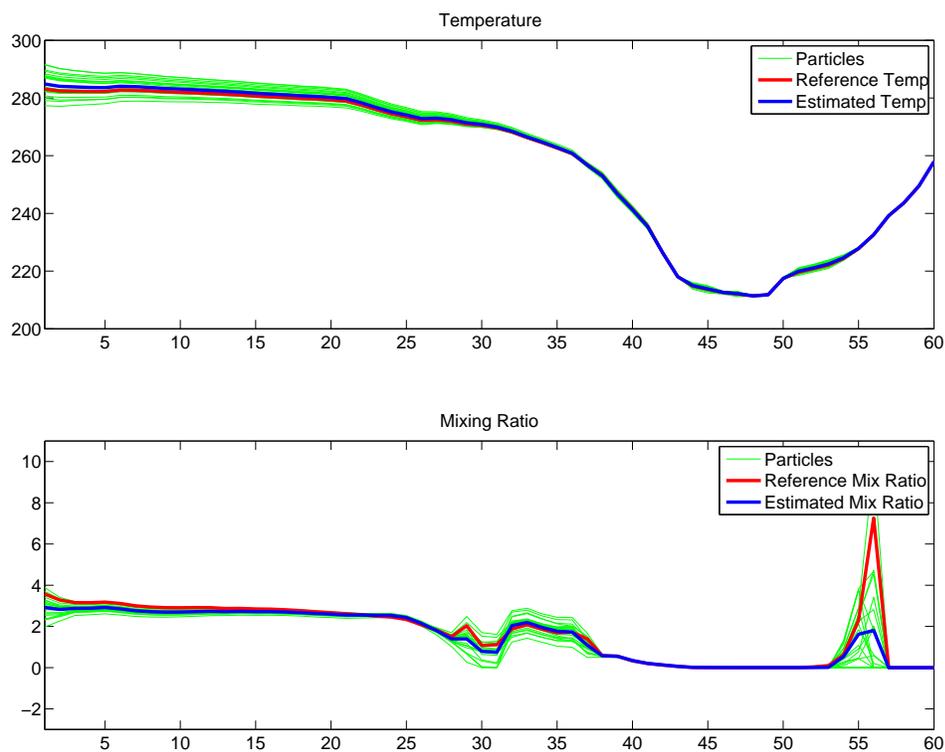
Fig. 7.1: Snapshot of the estimation process showing particle locations.

against. Unfortunately, we have only two weather balloon measurements during the entire hyperspectral data set, and so our judgment of the performance of an algorithm will rest principally upon the estimates at those two particular time steps.

Figures 7.2 and 7.3 show estimates of temperature for the entire nine hour data set using iterative nonlinear inversion and particle filtering, respectively. Figures 7.4 and 7.5 show the same for water vapor mixing ratio. Differences between the estimates are apparent, for example the particle filter detected a slight drop in temperature between hours 21 and 23, while the iterative nonlinear inversion method detected only a single peak in temperature during the day. Which estimate is correct? It is difficult to say.
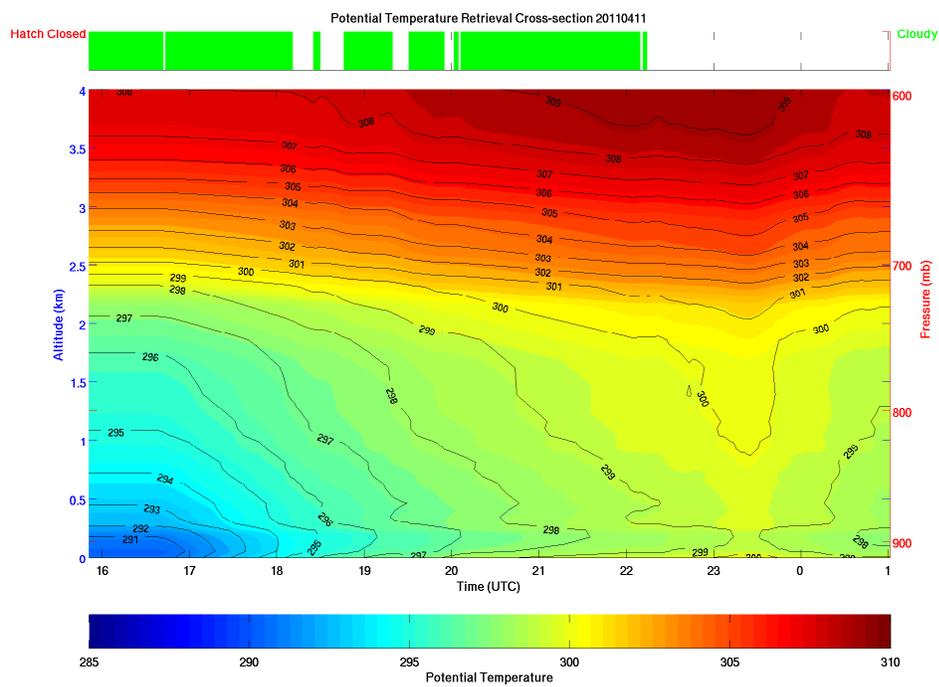
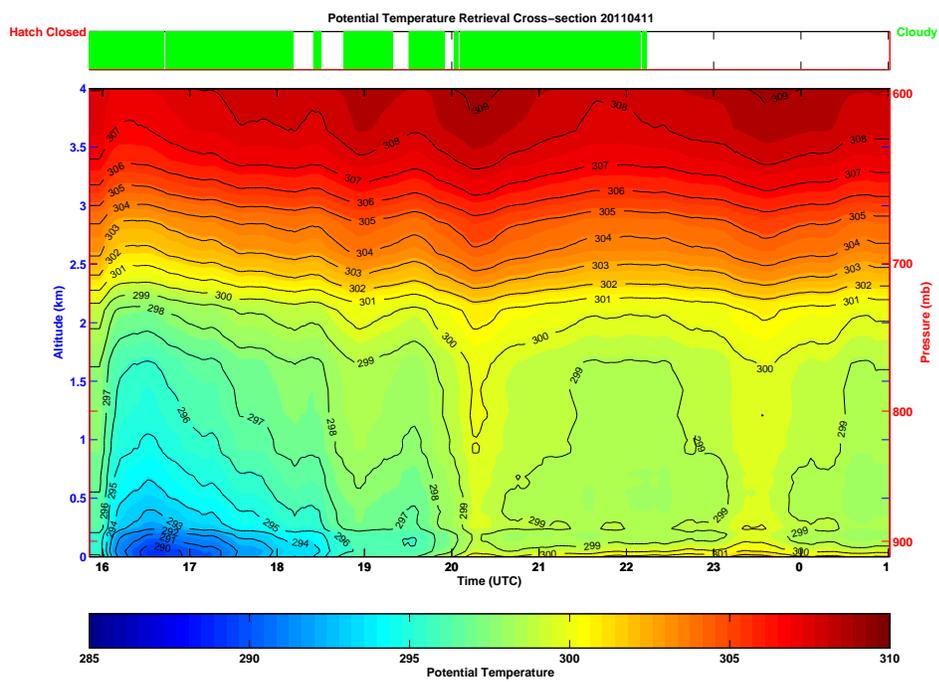Fig. 7.2: Temperature retrieval using iterative nonlinear inversion.



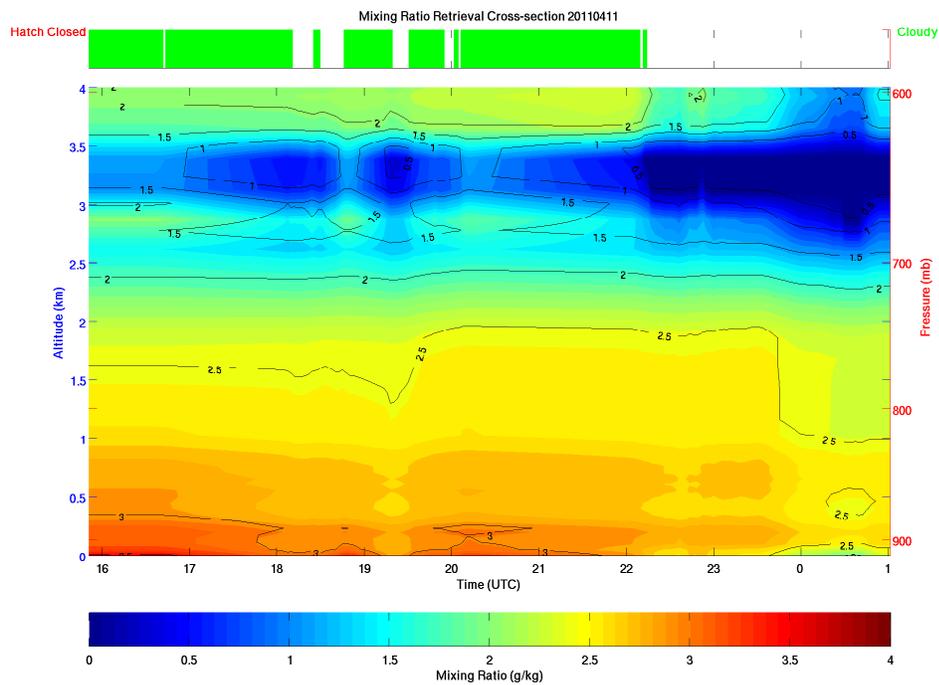Fig. 7.3: Temperature retrieval using the particle filter.

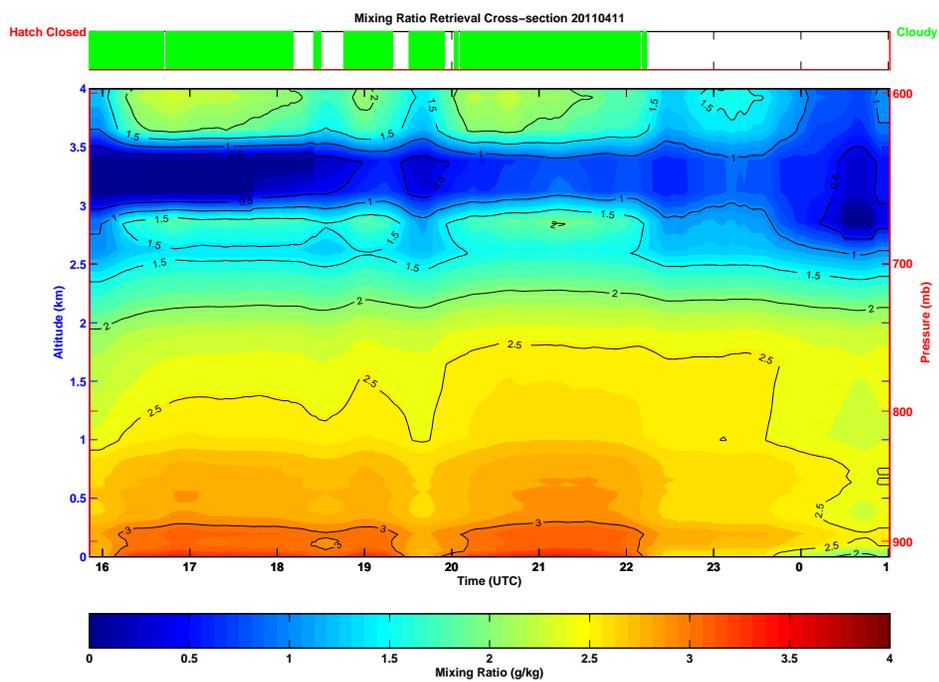Fig. 7.4: Water vapor mixing ratio retrieval using iterative nonlinear inversion.



Fig. 7.5: Water vapor mixing ratio retrieval using the particle filter.

### 7.2.1 A Performance Metric

While constructing the particle filter for this problem, we decided to weight the particles based on a comparison of the RTM output spectra from each particle to the actual spectra observed by the ASSIST II. This comparison can also give us a cursory look into the performance of an estimator. By taking each estimate of the atmospheric parameters and comparing the RTM output spectra for that estimate to the observed spectra at that time step, we can get a metric for how closely the estimates agree with the observations. We use the following for the comparisons

$$\text{Error}_k = \frac{||\mathbf{RTM}(\bar{\mathbf{x}}_k) - \mathbf{y}_k||^2}{||\mathbf{y}_k||^2},$$

where $\text{Error}_k$ is the mismatch of our estimate to the observation at time $k$, $\bar{\mathbf{x}}_k$ is the estimate at time $k$, and $\mathbf{y}_k$ is the observation at time $k$.

We have already discussed the many-to-one relationship between atmospheric profiles and observed spectra. For this reason, the comparison just described is not enough to determine whether or not an estimate of atmospheric parameters is correct, only whether or not it agrees with the observations. Figure 7.6 shows the result of this comparison for both the particle filter and iterative nonlinear inversion methods. Both methods track the same curve, meaning that they both agree equally well with the observations, despite differences in the actual estimates themselves. The error curve itself also has some interesting features to it, periods of time where both methods for some reason performed worse in their agreement with the observations. Figure 7.7 shows that the features of this curve are not coincidental. As the number of particles used in the particle filter increases, the error decreases until it converges to this shape and cannot do any better. A possible explanation for the interesting features of the error curve might be deficiencies in the observed data, or in the radiative transfer model. In either case, we believe that the error curve in figure 7.6 possibly represents a "minimum error" of sorts because two radically different approaches both converged to it.
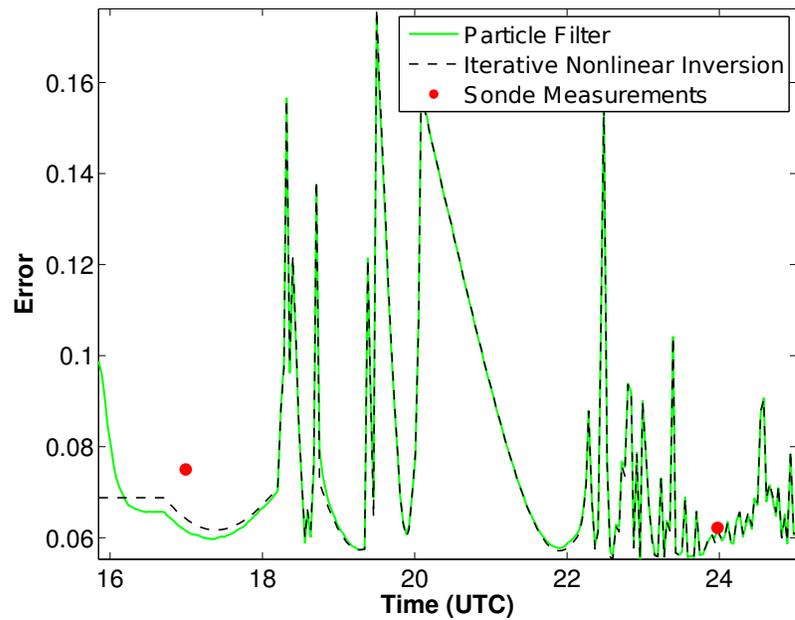
Fig. 7.6: State estimation error from the particle filter matches that from an iterative nonlinear inversion technique. Twice during spectral data collection the atmospheric parameters were measured using radiosondes attached to weather balloons.
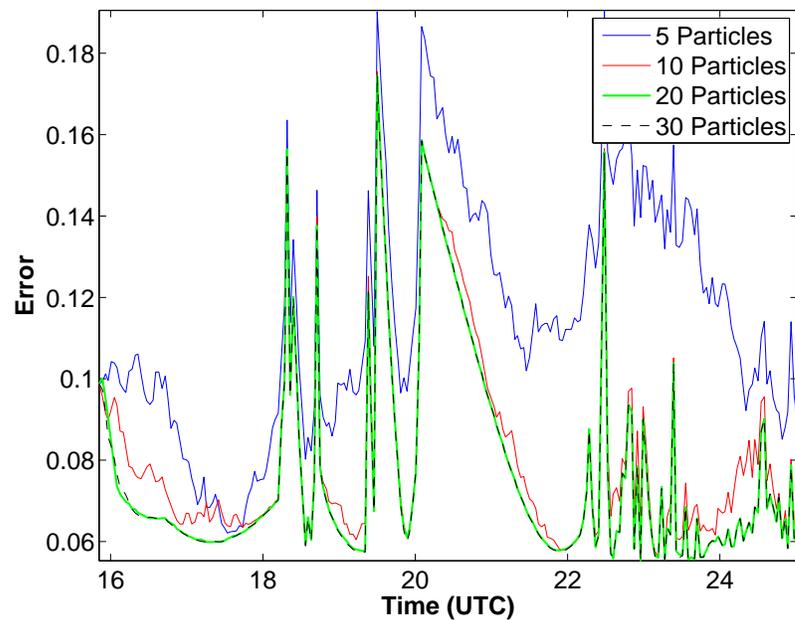


Fig. 7.7: State estimation error diminishes as the number of particles increase. Minimum error is achieved at approximately 20 particles.

### 7.2.2 Comparison to Weather Balloon Measurements

Figures 7.8 and 7.9 show the comparison of both the particle filter and iterative non-linear inversion estimates of temperature to weather balloon measurements. Temperature is the most dominant factor in the radiative transfer model, and it also tends to move smoothly. Because of these two factors, the temperature estimates of the particles tend to be tightly grouped with a high degree of confidence in the estimate. It can be seen from the figures that both methods track the weather balloon measurement well, although figure 7.9 shows the iterative nonlinear inversion method with a better estimate of the balloon measurement than the particle filter.
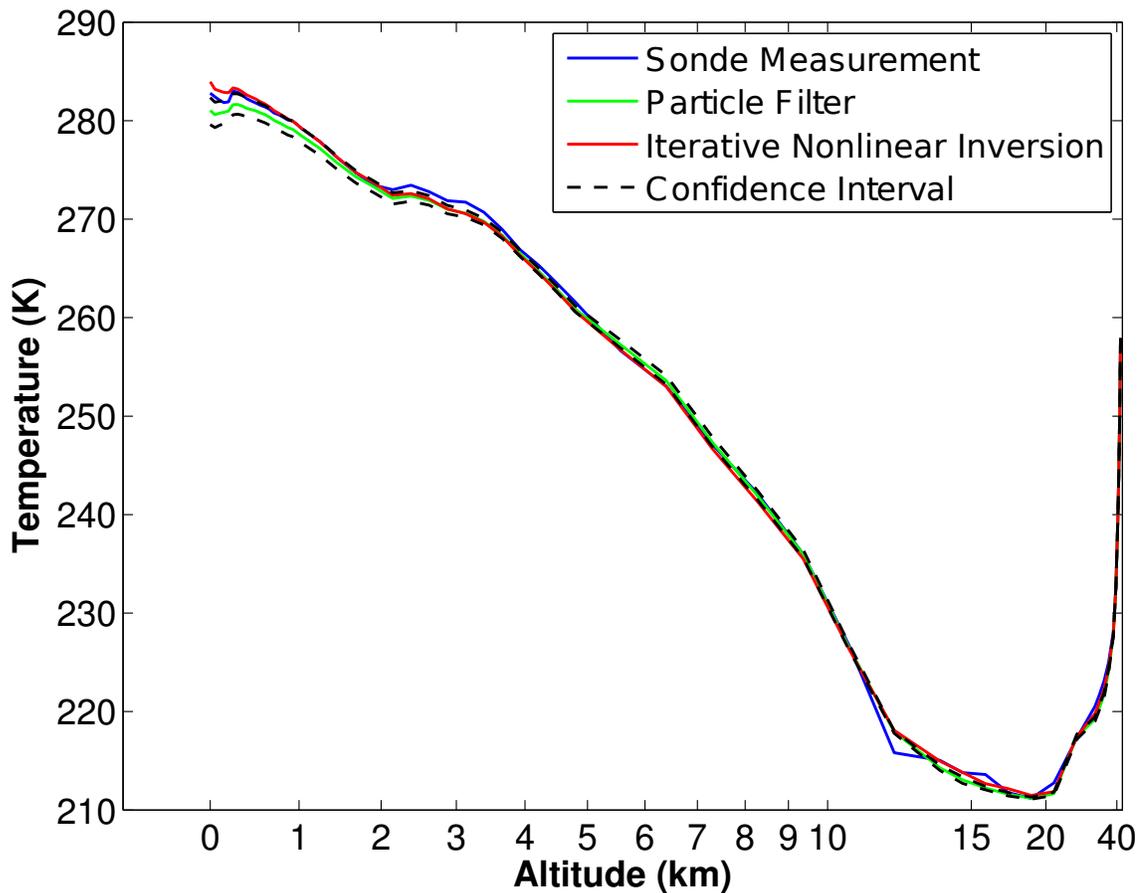


Fig. 7.8: Comparison of particle filter estimation of temperature to sonde measurement number one.
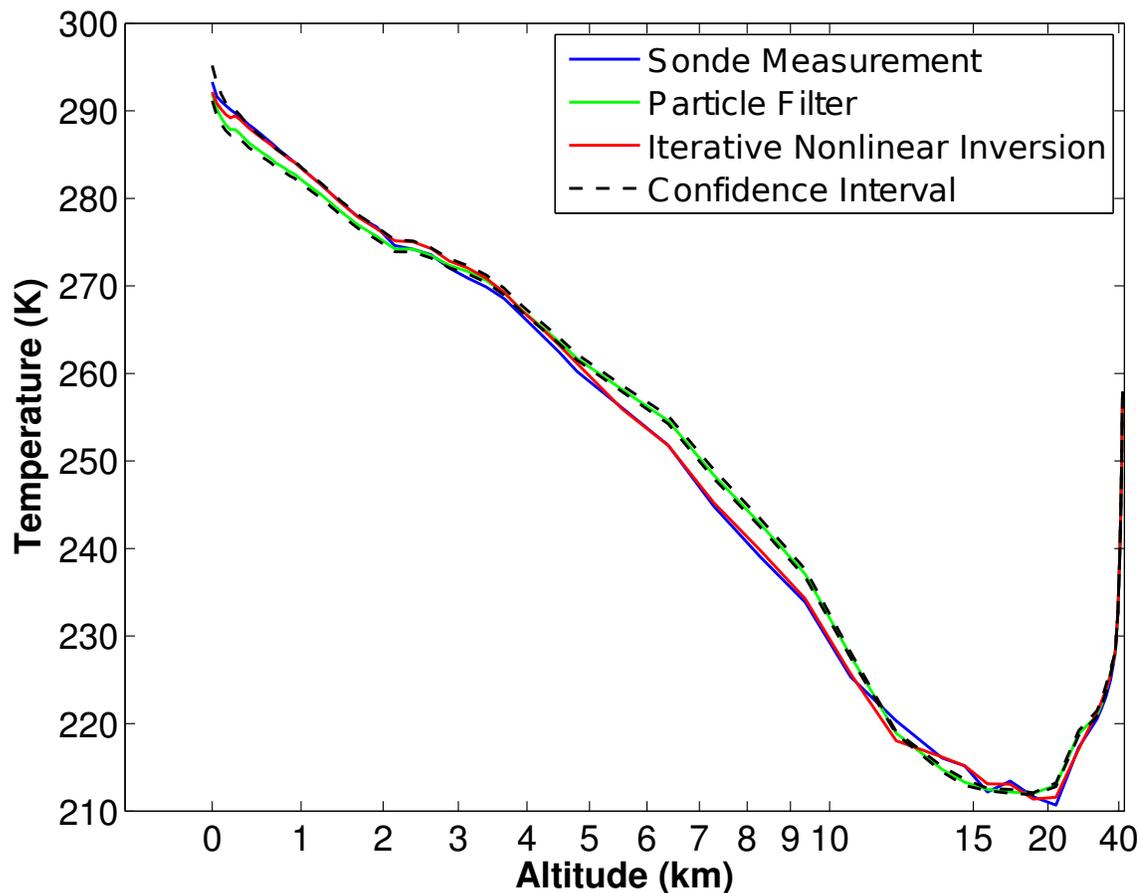
Fig. 7.9: Comparison of particle filter estimation of temperature to sonde measurement number two.

Water vapor mixing ratio estimates are more dynamic, as shown in figures 7.10 and 7.11. Figure 7.10 shows the iterative nonlinear inversion estimate to track closer to the weather balloon measurement most of the time, however at 40km it estimates a large spike in water vapor mixing ratio which isn't observed by the balloon. The particle filter's estimate at 40km is much more conservative. Figure 7.11 does not really depict a clear winner.

### 7.2.3    What Does It Mean?

The fact that a 120-dimensional system can be tracked with minimal error between the state estimates and the observation with only 20 particles is extremely encouraging. It shows that drawing the particles toward the observations is possible and that it can prevent
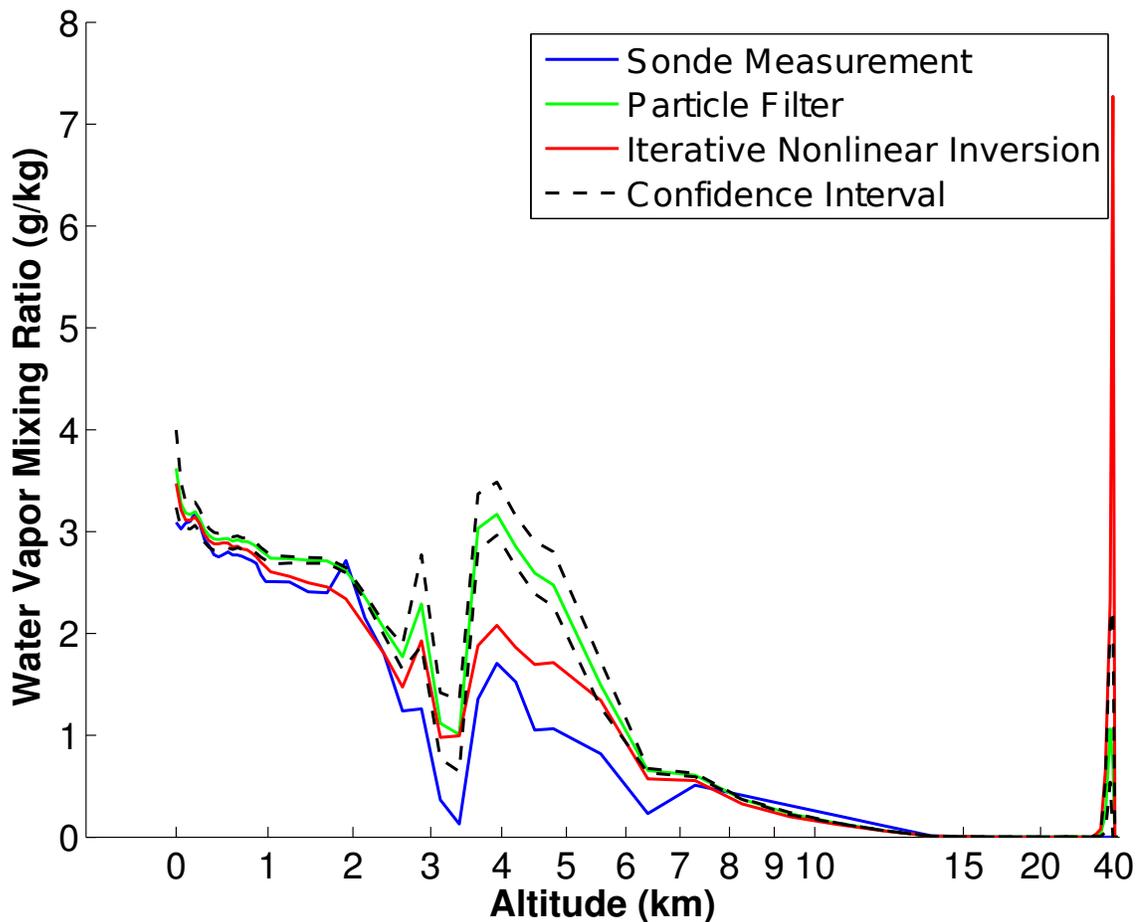
Fig. 7.10: Comparison of particle filter estimation of water vapor mixing ratio to sonde measurement number one.

filter degeneracy in high-dimensional systems even with a small number of particles. It is also encouraging that the estimates of the particle filter are close to measurements made by weather balloons, and comparable to estimates made by other methods, even when using the simplest possible model for atmospheric dynamics. It is believed that a more sophisticated model for atmospheric dynamics can greatly improve the quality of the estimate, and reduce even further the tendency for the particle filter to wander away from the true solution while maintaining minimal error between estimate and observation.

## 7.3   Computation Time

The running time of the particle filtering algorithm is slower than the iterative nonlinear
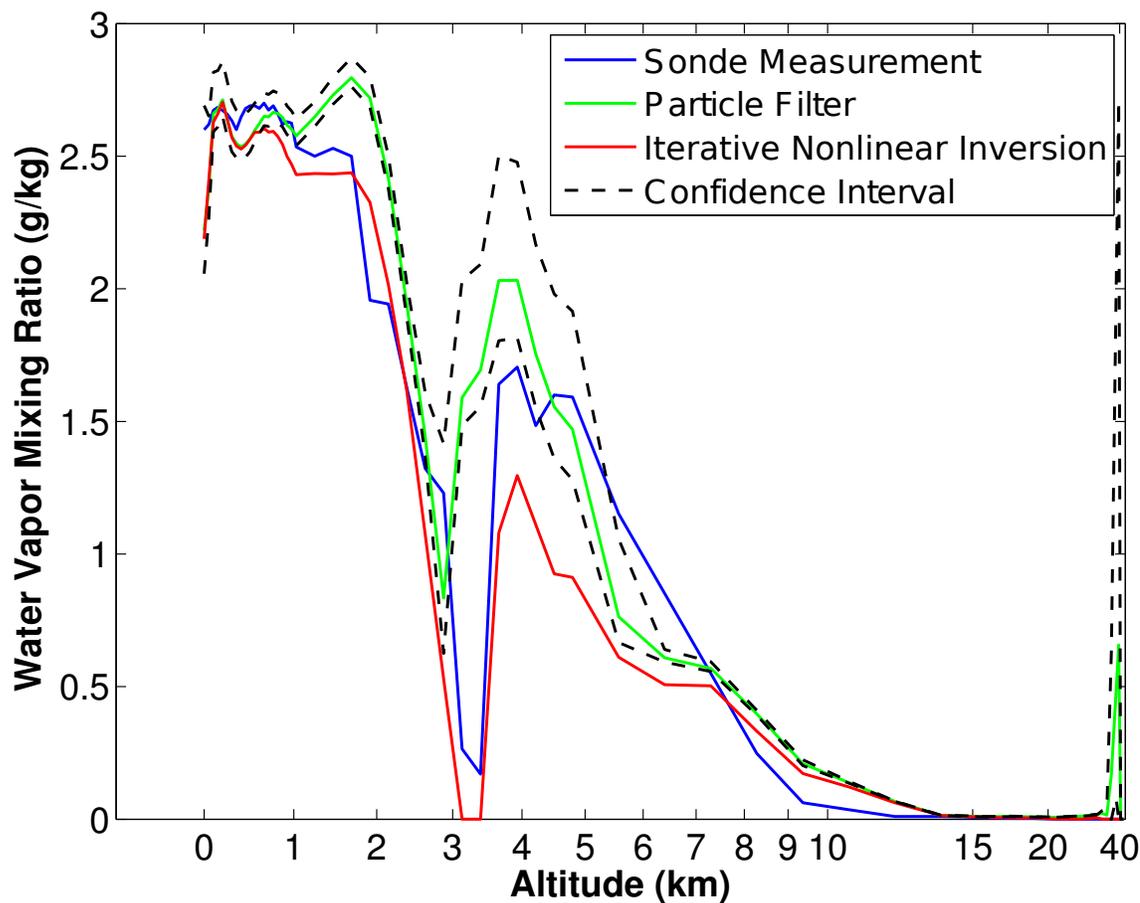
Fig. 7.11: Comparison of particle filter estimation of water vapor mixing ratio to sonde measurement number two.

inversion algorithm implemented at Hampton University. There is room for optimization inside the particle filtering algorithm itself which could serve to bring the running time down to something comparable to other methods but it is unlikely that this particle filtering solution will ever be faster than iterative nonlinear inversion due to the very few times that iterative nonlinear inversion must calculate radiative transfer compared to the particle filter.

# Chapter 8

# Conclusions

Particle filtering in high dimensions is a topic that will likely continue to be studied for some time. The advantages of particle filtering such as not needing to invert the observation equation, flexibility in modeling dynamics, and the ability to work in situations with non-Gaussian noise are extremely enticing. People will continue to look for ways to implement particle filters in high-dimensional spaces with relatively few particles. This body of work has demonstrated that particle filtering in high dimensions is possible with a modest amount of particles, although it can have a tendency to exaggerate other deficiencies in particle filtering related to the many to one relationship of a nonlinear observation model.

The main concern when particle filtering in high dimensions is that an unwieldy number of particles would be necessary to have a good enough approximation of the PDF of the system state in order to track the dynamics of the system. This concern is especially pertinent in systems whose dynamics tend to include occasional "decisions" where the system state might turn one direction or another. The PDF of the system state at the time step directly after such a decision could be thought of as multi-modal, with a certain probability associated with each decision. There is a "volume" of sorts at each of the modes in the PDF which needs to be represented by a number of particles. When working with fewer than an ideal number of particles to begin with, splitting the group between modes in the PDF will almost certainly cause collapse of the filter.

Degeneracy is a problem in all but the very simplest of particle filters, and has been a topic of discussion in the particle filtering literature almost since the idea of a particle filter was first proposed. A small number of particles will collect all of the weight in the filter, leaving the majority of the particles with a very small weight, and representing areas of the PDF that have very low probability. Ideally, we want particle weights to be approximately

equal, meaning that there is very dense representation of the PDF in its most likely areas. Resampling has been introduced as a technique to avoid filter degeneracy, and has proven to be effective in relatively low dimensional problems. However, in the 120-dimensional space of this problem, traditional resampling was not enough to keep the particle filter from wandering away from the system state, nor to keep the particles from wandering away from each other.

Particle filtering in high dimensions can work well with a strong and dependable model for system state dynamics. Perhaps that is a bit obvious. For instance, if the system state dynamics were perfect and deterministic without noise, why even bother to measure at all? However, based on the results from this work, it is believed that even in a noisy environment, dimensionality can be overcome if the state dynamics are well defined.

By implementing a modification to the particle filtering algorithm that draws particles toward the most likely observation we achieve good representation of the PDF around the estimate of the system state. The result of this modification is that the particles maintained nearly equal weights during the entire extraction and tracked a system state with minimal error compared to the observations. There are sacrifices which are made by taking this approach. By forcing the particles to keep a tight approximation around the estimated system state support of the PDF in less likely areas becomes virtually nonexistent. Also due to the many to one relationship between the observations and possible system states there is high likelihood that the particle filter will track an incorrect system state that happens to also have minimal error with the observations. In fact, as observed with weather balloon measurements, there is often an incorrect system state which matches the observation via the radiative transfer model better than the true system state.

Looking forward both for this problem and for high-dimensional particle filtering in general, there is work to be done to show what advantages more reliable dynamics models can bring to the table. For this problem the historical weather data combined with a smarter model for atmospheric dynamics could provide great improvement to the performance of the particle filter. As it stands the particle filter extracts atmospheric profiles from hyper-

spectral data with comparable results to iterative nonlinear inversion techniques. One of the things that particle filter has to offer in addition to state estimation is a measure of confidence in the estimated solution. Iterative nonlinear inversion has a serious shortcoming in that it can not sense when its estimations are wildly inaccurate. While the particle filter can also occasionally give inaccurate estimates, it brings with it a measure of confidence which can alert the operator of the potential for an inaccurate estimation.

# References

[1] D. Gu, A. Gillespie, A. Kahle, and F. Palluconi, "Autonomous atmospheric compensation (AAC) of high resolution hyperspectral thermal infrared remote-sensing imagery," *Geoscience and Remote Sensing, IEEE Transactions*, vol. 38, no. 6, pp. 2557–2570, Nov. 2000.

[2] J. P. Kerekes and J. E. Baum, "Hyperspectral imaging system modeling," *Lincoln Laboratory Journal*, 2003.

[3] D. Landgrebe, "Hyperspectral image data analysis," *Signal Processing Magazine, IEEE*, vol. 19, no. 1, pp. 17–28, Jan. 2002.

[4] L. Strow, S. Hannon, S. De Souza-Machado, H. Motteler, and D. Tobin, "An overview of the AIRS radiative transfer model," *Geoscience and Remote Sensing, IEEE Transactions*, vol. 41, no. 2, pp. 303–313, Feb. 2003.

[5] H. E. Fleming and W. L. Smith, "Inversion techniques for remote sensing of atmospheric temperature profiles," *5th Symposium on Temperature*, pp. 2239–2250, June 1971.

[6] P. Basili, S. Bonafoni, P. Ciotti, F. Marzano, G. d'Auria, and N. Pierdicca, "Retrieving atmospheric temperature profiles by microwave radiometry using a priori information on atmospheric spatial-temporal evolution," *Geoscience and Remote Sensing, IEEE Transactions*, vol. 39, no. 9, pp. 1896–1905, Sept. 2001.

[7] X. Liu, D. K. Zhou, A. M. Larar, W. L. Smith, P. Schluessel, S. M. Newman, J. P. Taylor, and W. Wu, "Retrieval of atmospheric profiles and cloud properties from IASI spectra using super-channels," *Atmospheric Chemistry & Physics Discussions*, vol. 9, pp. 8683–8736, Apr. 2009.

[8] P. Djuric, J. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. Bugallo, and J. Miguez, "Particle filtering," *Signal Processing Magazine, IEEE*, vol. 20, no. 5, pp. 19–38, Sept. 2003.

[9] C.-H. Cheng, "IIR filter model for the generalized Michelson interferometer," *Laser Electro-Optics Society Summer Topical Meetings*, pp. 197–198, July 2005.

[10] J. Susskind, C. Barnet, and J. Blaisdell, "Retrieval of atmospheric and surface parameters from AIRS/AMSU/HSB data in the presence of clouds," *Geoscience and Remote Sensing, IEEE Transactions*, vol. 41, no. 2, pp. 390–409, Feb. 2003.

[11] I. Selin, "The Kalman filter and nonlinear estimates of multivariate normal processes," *Automatic Control, IEEE Transactions*, vol. 9, no. 3, p. 319, July 1964.

[12] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, June 1999.

[13] P. Quang, C. Musso, and F. Le Gland, "An insight into the issue of dimensionality in particle filtering," *Information Fusion (FUSION), 2010 13th Conference*, pp. 1–8, July 2010.

[14] P. J. van Leeuwen, "Efficient nonlinear data assimilation for oceanic models of intermediate complexity," *Statistical Signal Processing Workshop (SSP), IEEE*, pp. 345–348, June 2011.

[15] P. J. van Leeuwen, "Nonlinear data assimilation in geosciences: an extremely efficient particle filter," *Quarterly Journal of the Royal Meteorological Society*, vol. 136, no. 653, pp. 1991–1999, 2010. [Online]. Available: http://dx.doi.org/10.1002/qj.699.