Retrieval of Cloud-Cleared Atmospheric Temperature Profiles from Hyperspectral Infrared and Microwave Observations

by

William Joseph Blackwell

B.E.E., Georgia Institute of Technology (1994)
S.M., Massachusetts Institute of Technology (1995)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2002

Author .....
Department of Electrical Engineering and Computer Science
May 14, 2002

Certified by .....
David H. Staelin
Professor of Electrical Engineering
Thesis Supervisor

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

Retrieval of Cloud-Cleared Atmospheric Temperature Profiles from Hyperspectral Infrared and Microwave Observations

by

William Joseph Blackwell

Submitted to the Department of Electrical Engineering and Computer Science
on May 14, 2002, in partial fulfillment of the
requirements for the degree of
Doctor of Science

Abstract

This thesis addresses the problem of retrieving the temperature profile of the Earth's atmosphere from overhead infrared and microwave observations of spectral radiance in cloudy conditions. The contributions of the thesis are twofold: improvements in 1) microwave instrumentation and 2) hyperspectral signal processing and estimation algorithms.

The NPOESS Aircraft Sounder Testbed–Microwave (NAST-M) passive spectrometer was designed, fabricated and deployed. NAST-M provides accurate brightness temperature measurements in 16 channels near the oxygen absorption lines at 50-57 GHz and 118.75 GHz, permitting the first reliably accurate retrieval images of temperature profiles and precipitation structure in cloudy areas.

The correlation structure of the NPOESS Aircraft Sounder Testbed–Infrared (NAST-I) instrument noise was analyzed in the spectral and spatial domains using the Iterated Order-Noise (ION) algorithm [1] for two representative flights. Results indicate that vibration-induced noise was the dominant component, but that it could be significantly reduced by filtering in the spatial domain.

Novel multi-pixel cloud clearing and temperature profile retrieval algorithms were developed for simulated Atmospheric Infrared Sounder (AIRS) and Advanced Microwave Sounding Unit (AMSU) radiances using neural networks. RMS temperature profile retrieval errors of $\sim$0.5 K were obtained for all levels of the atmosphere from 0–15 km in clear air at a horizontal resolution of 2000 km^2 and a vertical resolution of 1 km. RMS radiance errors under cloudy conditions for altitudes from 0 to 10 kilometers ranged from 1.25 K to 0.1 K for radiance retrievals near 15 microns, and from 0.8 K to 0.05 K for radiance retrievals near 4 microns.

Validation of the simulation results with NAST observations was hampered by the lack of a statistically-diverse data set accompanied by cloud truth. An upper bound on cloud-clearing performance (NEΔT) was estimated to be approximately a factor of two worse than the simulation results accompanied by ground truth. An improvement of approximately 25 percent in RMS radiance cloud-clearing performance was realized by rejecting 20 percent of soundings based on a neural network-derived metric.

Thesis Supervisor: David H. Staelin
Title: Professor of Electrical Engineering

Acknowledgments

First and foremost, I would like to thank Prof. Staelin for his guidance and support over the course of this research. Thanks also to Jack Barrett for labor and guidance on NAST–M, Phil Rosenkranz for answering many questions and for improving the quality of this thesis through many suggestions, and Mike Schwartz for many helpful discussions (and many pieces of pæ).

Thanks to Al Gasiewski for introducing me to the field of atmospheric remote sensing at Georgia Tech and his encouragement over the years.

Thanks to Carlos Cabrera, Fred Chen, Junehee Lee, and Vince Leslie for all the good times and many helpful brainstorming sessions.

Thanks to the crew and pilots at NASA DFRC and Scaled Composites for all the help with NAST-M over the last five years. Sung Yung Lee (NASA-JPL), Bormin Huang (UW), and Dan Zhou (NASA-Langley) shared software and cheerfully answered all my questions on the AIRS/AMSU and NAST-I datasets.

Thanks to all the RLE guys for making life easier in a number of areas. Seth Hall was always quick with a thoughtful response to my many computer-related questions. Dave Foss, Mary Young, and Maxine Samuels always went above and beyond the call of duty to help me get out of numerous jams.

Thanks to Mom, Dad, Steve, and Liz for all the support and encouragement over the years. Finally, thanks to Megan for showing me how to use the Ab Roller and brightening every day of my life ever since. **iww**.

This research has been funded in part by a National Science Foundation Graduate Fellowship, the Research Laboratory of Electronics at MIT (under NASA contracts NAS5-31276 and NAS5-31376 and NASA grants NAG5-2524 and NAG5-7487) and the Lincoln Laboratory, MIT (under PO-BX-7601 and AF Prime Contract F19628-00-C-0002). Opinions, interpretations, conclusions, and recommendations are those of the author and are not necessarily endorsed by the United States government.

Permission was obtained from SPIE and IEEE to include material from the following publications:

Blackwell, et al., "Comparative Performance Analyses of Passive Microwave Systems for Tropospheric Sounding of Temperature and Water Vapor Profiles," *P*roc. SPIE, vol. 2812, pp. 472-478, Denver, CO, 1996.

Blackwell, et al., "Analysis of Cloud Impact on Infrared and Microwave Atmospheric Sounding Performance using NAST," accepted for publication in *P*roc. IGARSS '02, Toronto, Canada, 2002.

Blackwell, et al., "Cloud Flagging and Clearing Using High-Resolution Infrared and Microwave Sounding Data," accepted for publication in *P*roc. IGARSS '02, Toronto, Canada, 2002.

Blackwell, et al., "NPOESS Aircraft Sounder Testbed - Microwave (NAST-M): Instrument Description and Initial Flight Results," *I*EEE Trans. Geosci. Remote Sensing, vol. 39, no. 11, pp. 2444-2453, Nov. 2001.

Blackwell, et al., "NPOESS Aircraft Sounder Testbed-Microwave (NAST-M): Instrument Description and Initial Flight Results," *P*roc. IGARSS '00, Honolulu, Hawaii, 2000.

Blackwell, et al., "NPOESS Aircraft Sounder Testbed-Microwave (NAST-M): Results from CAMEX-3 and WINTEX," *P*roc. IGARSS '00, Honolulu, Hawaii, 2000.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

MEASUREMENTS of the state and composition of the Earth's surface and atmosphere have been made using passive microwave and infrared sensors for over fifty years[2]. Applications of these remote measurements are numerous, and encompass fields ranging from meteorology to oceanography and geology. For example, satellite measurements of atmospheric temperature are used to improve weather forecasting models, analyze climate change, and study the radiation budget of Earth.

Recent advances in airborne and spaceborne platforms have made atmospheric measurements possible on a global scale, and advances in sensor technologies have pushed the limits of achievable spatial and temporal resolution to unprecedented levels. These performance improvements, however, are not without concomitant data processing difficulties. The vast amount of data generated by present– and next–generation sounders must be transmitted and processed in a timely manner (usually real-time), which requires processing algorithms that are both efficient and robust. The latter requirement is often more challenging, as most processing methods are sensitive to noise originating both from the measurement system and from geophysical "clutter", such as surface variability, trace constituents, and clouds.

Clouds often introduce the largest source of error in atmospheric profiling measurements, and in the case of the temperature profile retrieval, can contribute errors of several degrees Kelvin [3]. The problem is compounded by the ubiquity of clouds – a global study from

1989-1993 found that, on average, clouds cover approximately 65 percent of the Earth's surface over land and approximately 80 percent over water [4].

Current weather prediction models require measurements of radiances characterizing the Earth's temperature profile to within small fractions of a degree Kelvin. While this level of accuracy is currently obtainable in clear air, significant instrumentation and data processing improvements are needed to achieve this accuracy in cloudy conditions. This thesis addresses both of these areas. The NPOESS Aircraft Sounding Testbed–Microwave (NAST-M) sounder was developed and deployed, providing the first high-resolution microwave data co-registered near two opaque oxygen lines and also co-registered with hyperspectral imagery near 3.6–16.1 μm. Signal processing and estimation techniques were derived and evaluated for retrieving the atmospheric temperature profile in the presence of clouds.

1.2 Problem statement and general approach

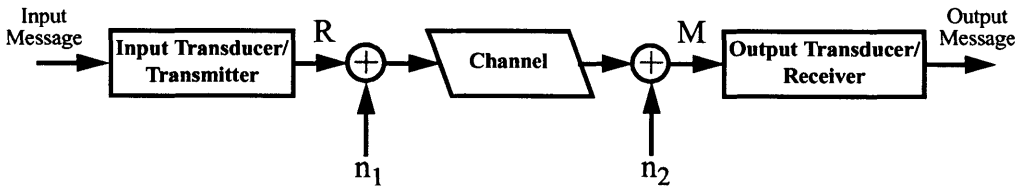

Figure 1-1: Canonical communication system.

The canonical communication system shown in Fig. 1-1 can be generalized to define the remote sensing retrieval problem in a familiar context. In this example, information is converted into electrical signals and transmitted over a channel. The signal is corrupted by noise processes (represented schematically in the figure by n_1) along the transmission path, distorted by the channel, and then corrupted by measurement error (represented schematically in the figure by n_2) at the receiver. In the simplest case, the noise processes are additive white Gaussian noise, and the channel is a linear time-invariant (LTI) system. It is not uncommon to encounter multiplicative noise processes and non-linear time-varying

channels. While there has been a great deal of research on signal processing techniques that recover information transmitted over an LTI channel in the presence of white Gaussian noise[5], the existence of multiplicative noise and/or nonlinear time-varying channels complicates the analysis.

A remote sensing system operates in much the same way as the communication system previously discussed. Information about the state and composition of the atmosphere is related to the radiation intensity measured by a sensor. This relation is described by Planck's radiation law and the equation of radiative transfer, to be discussed in Chapter 2. As a simple example, consider the task of retrieving the surface temperature of the Earth (the message) given a remote measurement of upwelling radiation intensity. The surface emits a radiance R that is a well-known function of its temperature, however this radiance must travel through the Earth's atmosphere to the sensor (analogous to a communication signal traveling through a channel). The atmosphere will change the radiance in a fashion that depends on a number of parameters, including the composition of the atmosphere, the amount of cloud cover present, and the frequency of the measured radiation, to name a few. The sensor will also introduce random errors during the course of measurement. The receiver in the remote sensing context is very similar to the receiver in the communication context; the desired signal, or information, must be extracted from a noisy measurement M.

This thesis addresses the specific problem of retrieving the temperature profile of the Earth's atmosphere in the presence of clouds as a function of altitude, given observations of electromagnetic radiation measured by a sensor at different frequencies. The exact probabilistic relationships of all the geophysical signals and contaminant signals (noise and geophysical clutter) will generally not be known, and must be estimated. The primary goals of the thesis are therefore twofold: first, information content analysis methods will be developed that characterize and reduce noise processes and/or geophysical clutter; second, linear and nonlinear estimators will be developed that produce estimates of the desired geophysical parameters.

1.3 Thesis outline

Chapter 2 summarizes the relevant physical issues in passive atmospheric sounding. The composition and thermal structure of the atmosphere is discussed, including a brief overview of cloud microphysics. Fundamentals of electromagnetic wave propagation are covered, including polarization and reflection/transmission at a boundary. The mechanisms of gaseous absorption are outlined, and the absorption coefficient and transmission function are defined. Scattering is then discussed, with a focus on Mie theory and the Rayleigh approximation. Simple radiative transfer concepts are introduced, including equilibrium radiation, weighting functions, and linearization methods. Finally, a survey of radio-frequency and optical instrumentation techniques is presented.

Chapter 3 summarizes the signal processing and estimation techniques commonly used in hyperspectral data analysis. Two interpretations of the information content in a measurement are given and related: information theoretic (Shannon) and the degrees of freedom. Principal components analysis techniques (both nonlinear and linear) are introduced. Blind signal processing is briefly discussed. A Bayesian formulation of geophysical estimation theory is then discussed, including minimum mean-square error estimators. Non-Bayesian estimators are then covered, including the minimum-information estimator and the maximum-resolution estimator. Artificial neural networks are then introduced in an estimation context.

Chapter 4 details the NAST-M instrument and representative flight results from the Third Convection and Moisture Experiment (CAMEX-3) and the Winter Experiment (WINTEX). A brief summary of the NPOESS Aircraft Sounding Testbed–Infrared (NAST-I) is given. In Section 4.1, the radiometer specifications are discussed, including the local oscillator frequencies and passband characteristics. The scanning assembly and field-of-view specifications are then given. The construction of the calibration loads is summarized. Auxiliary systems, instrument control, and data handling are outlined. The calibration of NAST-M is discussed in Section 4.2. A weighted least-squares calibration procedure is outlined. A thermal gradient through the heated calibration load is modeled, and a correction is formulated. Antenna-beam spillover while viewing the internal calibration loads is

modeled, and a correction is formulated. In Section 4.3, the final calibration is checked by comparing observed brightness temperatures with a coincident overpass of the Advanced Microwave Sounding Unit (AMSU) instrument on the NOAA-15 satellite during WINTEX.

Chapter 5 provides a characterization of NAST-I system noise and methods for its reduction. Section 5.1 outlines the sources of noise in the NAST-I instrument and categorizes them as "additive" or "scene-dependent". A mathematical characterization of the dominant noise source (vibration-induced noise) is given. A summary of the flight data used for the analyses appears in Section 5.2. An analysis of the spectrally-uncorrelated component of the NAST-I system noise is given in Section 5.3. An estimate of the RMS variation of the spectrally-correlated component is also given. Section 5.4 provides a characterization of the spectral correlation structure of the spectrally-correlated noise component. Twenty-four representative channels are chosen for the analysis, and their properties are discussed. Section 5.4 provides a characterization of the spatial correlation structure of the spectrally-correlated noise component. Two NAST-I "pseudochannels" are formulated for the analysis.

Chapter 6 presents a methodology for the development and evaluation of estimators for retrieving the cloud-cleared temperature profile. Simulated radiances from the Atmospheric Infrared Sounder (AIRS), AMSU instruments are used to derive the estimators and test the performance. Section 6.2 summarizes the simulation procedure, including the characterization of the temperature profile set, the surface model, and the cloud model used. The single-pixel information content of clouds is examined in Section 6.3 using principal components analysis. Single-pixel temperature profile retrievals using neural networks are discussed in Section 6.4. The retrievals are then used to derive a single-pixel cloud impact metric (cloud flag). In Section 6.5, the single-pixel algorithms are extended to include spatial information present in multi-pixel "clusters". Multi-pixel cloud flags are derived. Multi-pixel statistical cloud-clearing algorithms are developed to retrieve clear-column radiances using regression and neural network estimators. Finally, the sensitivity to surface emissivity is examined in Section 6.6.

Chapter 7 presents NAST-I/M flight results from CAMEX-3 and WINTEX. NAST-M data from several overpasses of Hurricane Bonnie during CAMEX-3 are analyzed in Section 7.1. A two-band image of an eye overpass is shown, and a retrieval of the temperature

perturbation within an eye is given. Particle size information is presented using a metric derived from simulated data. The Multispectral Atmospheric Mapping Sensor (MAMS) used to provide high-resolution visible and infrared cloud imagery is summarized in Section 7.2. The construction of the CAMEX-3 cloudy radiance database is discussed in Section 7.3. The information content of cloud-induced radiance variations is examined in Section 7.4 using principal components analysis.

Chapter 8 evaluates some of the results of single-pixel results Chapter 6 using NAST observations of clouds over water from CAMEX-3. The information content of clouds observed in the CAMEX-3 measurements are compared with the simulation results in Section 8.1. The impact of NAST-observed clouds on simulated retrieval performance is evaluated in Section 8.2. A neural network estimator that estimates the magnitude of the NAST cloud-clearing error is developed in Section 8.3.

Chapter 9 summarizes the contributions of the thesis and suggests opportunities for further research.

Chapter 2

Physical Foundations of Atmospheric Sounding

THE following is a broad overview of relevant physical issues in passive atmospheric sounding and is intended to provide background and context to results presented later in the thesis. For additional details, the reader is referred to references on tropospheric remote sensing [6], atmospheric physics [7], atmospheric radiation [8], and electromagnetic wave propagation [9].

2.1 Overview of the Composition and Thermal Structure of the Earth's Atmosphere

The Earth's atmosphere extends over 100 km from its surface, and can roughly be categorized into four layers based on the thermal and chemical phenomena that occur within each layer. These layers are (in increasing altitude) the troposphere, the stratosphere, the mesosphere, and the thermosphere. The boundaries between each layer are usually not well-defined, but do show characteristic features. They are the tropopause, stratopause, and mesopause, respectively. The troposphere extends from the surface to an altitude of approximately 12 km and is characterized by a steady decrease in temperature with altitude. Almost all of the Earth's weather is created in the troposphere, and therefore it is the focus of

most atmospheric sounding research, including this thesis. The tropopause marks the region of the atmosphere where the temperature gradually changes from decreasing with altitude to increasing with altitude, and forms a somewhat nebulous boundary layer between the troposphere and the stratosphere. The stratosphere extends to an altitude of approximately 40 km, and is characterized by a sharp increase in temperature with altitude due to the absorption of ultraviolet radiation by ozone. High cirrus clouds sometimes form in the lower stratosphere, but for the most part there are no significant weather patterns in this layer. The mesosphere extends from approximately 40km to 80km, and is characterized by a decreasing temperature with altitude. Extremely low temperatures ($\sim -150°$ C) present at the top of the mesosphere sometimes allow the formation of noctilucent clouds, thought to be made of ice crystals clinging to dust particles. The transition from the mesosphere to the final thermosphere layer begins at an altitude of approximately 80 km. The thermosphere is characterized by warmer temperatures caused by the absorption of the sun's short-wave ultraviolet radiation. This radiation penetrates the upper atmosphere, stripping atoms of their electrons and giving them a positive charge. These electrically charged atoms build up to form a series of layers, often referred to as the ionosphere.

2.1.1 Chemical composition of the atmosphere

The Earth's atmosphere is composed of a variety of gases. Each gas interacts characteristically with electromagnetic radiation of a given frequency. This relationship forms the physical basis by which the atmospheric temperature can be measured by observing radiation of different frequencies that has been emitted by and transmitted through the atmosphere.

The average fractional volumes of various species in the Earth's atmosphere are given in Table 2.1. Perhaps the most important gases in the atmosphere, from the point of view of their interaction with electromagnetic radiation, are water vapor, oxygen, carbon dioxide, and ozone. Oxygen and carbon dioxide are well-mixed in the atmosphere below approximately 100 km, and therefore frequencies near the resonances of these molecules are desirable for temperature sounding. The vertical distribution of ozone has a maximum number density near 25 km. Above 30 km, ozone is rapidly formed by photochemical

Table 2.1: Composition of the Earth's atmosphere [6].

Molecule	Volume Fraction†	Comments
N_2	0.7808	Photochemical dissociation high in the ionosphere; mixed at lower levels
O_2	0.2095	Photochemical dissociation above 95 km; mixed at lower levels
H_2O	< 0.04	Highly variable; photodissociates above 80 km
A	9.34×10^{-3}	Mixed up to 110 km; diffusive separation above
CO_2	3.45×10^{-4}	Slightly variable; mixed up to 100 km; dissociated above
CH_4	1.6×10^{-6}	Mixed in troposphere; dissociated in mesosphere
N_2O	3.5×10^{-7}	Slightly variable at surface; dissociated in stratosphere and mesosphere
CO	7×10^{-8}	Variable photochemical and combustion product
O_3	$\sim 10^{-8}$	Highly variable; photochemical origin
$CFCl_3$ and CF_2Cl_2	1–2 $\times 10^{-10}$	Industrial origin; mixed in troposphere, dissociated in stratosphere

†Fraction of lower tropospheric air

reactions from oxygen so that an equilibrium is established during the daylight hours. Below this level, ozone is created more slowly and is highly variable. Water vapor is especially important in the troposphere because of its role in the formation of clouds and precipitation and in transporting significant amounts of energy in the form of latent heat and infrared radiation. Water vapor is extremely variable near the surface–water vapor may account for up to 4 percent of a tropical atmosphere, while the abundance in a polar atmosphere may be more than an order of magnitude less.

2.1.2 Vertical distribution of pressure and density

Vertical variations in the pressure and density of the Earth's atmosphere are much greater than either the horizontal or temporal variations. It is therefore helpful to define a reference or "standard"[1] atmosphere which is a representation of the atmosphere as a function of height only. Below an altitude of 100 km, the atmospheric pressure and density are almost always within ±30 percent of the standard atmosphere.

Atmospheric density decreases with altitude due to the Earth's gravitational field. If a condition of static equilibrium is assumed, the relationship between density and pressure as a function of altitude may be expressed by the following differential equation:

$$dp = -g\rho\, dz, \tag{2.1}$$

where p and ρ are the pressure and density at altitude z measured vertically upward from the surface. The change in gravitational force with altitude is small enough over the relatively short extent of the atmosphere to be ignored. The ideal gas equation $PV = nRT$ can be used to relate the density of an ideal gas of molecular weight M_r to its temperature and pressure:

$$\rho = \frac{M_r p}{RT}, \tag{2.2}$$

[1]The horizontal and temporal variations of the Earth's atmosphere do vary substantially on a global and seasonal scale. A wide variety of "standard" atmospheres have been tabulated for various geographical regions and seasonal periods [10].

where R is the gas constant per mole, and T is the temperature (K). Equation 2.1 can then be expressed as

$$\frac{dp}{p} = -\frac{dz}{H}, \tag{2.3}$$

which can be integrated to find the pressure p at altitude z:

$$p = p_0 \exp\left\{-\int_0^z \frac{dz}{H}\right\}, \tag{2.4}$$

where p_0 is the surface pressure and $H = RT/M_r g$ is known as the scale height. The scale height is the increase in altitude necessary to reduce the pressure by a factor of e. In the troposphere, H typically varies between ~6 km at $T = 210$ K to ~8.5 km at $T = 290$ K.

2.1.3 Thermal structure of the atmosphere

The macroscopic thermal features of the atmosphere were outlined previously. We now examine features that occur on a finer vertical scale, with a focus on the lower troposphere. The bottom 1–2 km of the atmosphere exhibits the greatest thermal variability due to strong surface interactions and diurnal variations. At some latitudes, temperature inversions exist in the lowest 2–3 km of the atmosphere. Above 3 km, there is a systematic decrease of temperature with altitude that can be characterized by an adiabatic lapse rate, as follows. Continuing the assumption of the previous section that the atmosphere is in hydrostatic equilibrium, the first law of thermodynamics can be applied to a unit "parcel" of atmospheric mass:

$$dq = c_v\, dT + p\, dV, \tag{2.5}$$

where c_v is the specific heat at constant volume. Provided no heat enters or leaves the parcel (i.e., the process is adiabatic), the quantity dq is zero. Equation 2.5 can then be substituted into the differential form of the ideal gas law to yield:

$$\frac{dT}{dZ} = -\frac{g}{c_p} = -\Gamma, \tag{2.6}$$

where c_p is the specific heat at constant pressure and Γ is the lapse rate. Equation 2.6 shows that the change in temperature with altitude is constant, for constant c_p and g. Under typical tropospheric conditions, c_p varies slightly with altitude, and the dry adiabatic lapse rate in the troposphere is approximately 10 K/km. If the latent heat released by the condensation of rising moist air is considered, the average lapse is approximately 6.5 K/km.

2.1.4 Cloud microphysics

Clouds affect the energy balance of the atmosphere through two mechanisms: 1) water cycle changes, including the release of latent heat through condensation and the removal of liquid water through precipitation, and 2) radiation budget changes, including the scattering, absorption, and emission of solar and terrestrial radiation. In future sections of the thesis, the microphysical properties of clouds (taken here to mean the size and shape of the particles and their volume concentration) will be used to characterize regions of precipitation by examining their interaction with microwave radiation. It is therefore useful to review several of the salient details of the microphysical structure of clouds and precipitation.

The microphysical properties of clouds depend highly on the size, shape, and phase of the water particles. Water droplets are typically smaller than 100 μm and are spherical. The distribution of water droplet concentration (the number of droplets per volume existing in a differential radius range dr) is reasonably approximated by analytic functions. A modified Gamma distribution is often used for this purpose. Table 2.2 gives average values of the number of particles (N_0), mean droplet radius (r_m), and cloud liquid water density (l) for a variety of clouds. Raindrops are generally nonspherical, resembling oblate spheroids with an aspect ratio (width-to-length ratio) that decreases as the drop size increases. One analytic function that is commonly used to relate raindrop size distributions to rainrate is the Marshall-Palmer distribution [12]. Ice crystals form in a wide variety of sizes and shapes. In addition to simple polyhedron forms, irregular crystals or combinations of simple shapes readily appear in nature.

Table 2.2: Representative dropsize concentrations (N_0), mean particle radius (r_m), and liquid water content (l) for several cloud types [11].

Cloud Type	N_0 (cm^{-3})	r_m (μm)	l (gm^{-3})
Stratus (ocean)	50	10	0.1–0.5
Stratus (land)	300–400	6	0.1–0.5
Fair-weather Cumulus	300–500	4	0.3
Maritime Cumulus	50	15	0.5
Cumulonimbus	70	20	2.5
Cumulus Congestus	60	24	2.0
Altostratus	200–400	5	0.6

2.2 Electromagnetic wave propagation

The thermal and compositional state of the atmosphere affects both the generation and propagation of electromagnetic (EM) waves. For now, we ignore the source of the EM waves and focus instead on their propagation through a homogeneous, lossless medium.

2.2.1 Maxwell's equations and the wave equation

In a source-free, homogeneous, and isotropic medium with permittivity ϵ and permeability μ, the spatial and temporal variation of electric and magnetic fields are related according to Maxwell's equations:

$$\nabla \times \vec{\mathbf{E}} = -\mu \frac{\partial}{\partial t} \vec{\mathbf{H}} \tag{2.7}$$

$$\nabla \times \vec{\mathbf{H}} = \epsilon \frac{\partial}{\partial t} \vec{\mathbf{E}} \tag{2.8}$$

$$\nabla \cdot \vec{\mathbf{E}} = 0 \tag{2.9}$$

$$\nabla \cdot \vec{\mathbf{H}} = 0. \tag{2.10}$$

A wave equation can be derived by taking the curl of Eq. 2.8 and substituting Eq. 2.9. After using the vector identity $\nabla \times (\nabla \times \vec{\mathbf{E}}) = \nabla(\nabla \cdot \vec{\mathbf{E}}) - \nabla^2 \vec{\mathbf{E}}$ and Eq. 2.10, we find:

$$\nabla^2 \vec{\mathbf{E}} = \mu\epsilon \frac{\partial^2}{\partial t^2} \vec{\mathbf{E}} \tag{2.11}$$

where the Laplacian operator ∇^2 in a rectangular coordinate system is

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}.$$

The wave equation (2.11) is a second-order partial differential equation of space and time coordinates x, y, z, and t. A simple solution to the wave equation is

$$\vec{\mathbf{E}}(\vec{\mathbf{r}}, t) = \vec{\mathbf{E}}_0 \cos(2\pi\nu t \pm \vec{\mathbf{k}} \cdot \vec{\mathbf{r}}) \tag{2.12}$$

where $\vec{\mathbf{k}} = \hat{\mathbf{x}}k_x + \hat{\mathbf{y}}k_y + \hat{\mathbf{z}}k_z$ and $\vec{\mathbf{r}} = \hat{\mathbf{x}}x + \hat{\mathbf{y}}y + \hat{\mathbf{z}}z$. Equation 2.12 represents two waves propagating in opposite directions in the $\vec{\mathbf{k}}$ direction with temporal phase $2\pi\nu t$ and spatial phase $\vec{\mathbf{k}} \cdot \vec{\mathbf{r}}$. A spectrum of values of frequency (ν) found in atmospheric remote sensing systems is shown in Fig. 2-1.

2.2.2 Polarization

The electric field vector of a uniform plane wave traveling in the $+z$ direction must lie in the xy-plane perpendicular to the z-axis. As time progresses, the tip of the electric field vector traces a curve in the xy-plane. It is the shape of this curve (linear, circular, or elliptical) that determines the polarization of the plane wave. If the curve is circular or elliptical, the tip may move in either a clockwise or counterclockwise direction. Two electromagnetic waves, identical in all respects except for their polarization state, can interact differently with matter. It is the very nature of these differences that are exploited in certain remote sensing methods.

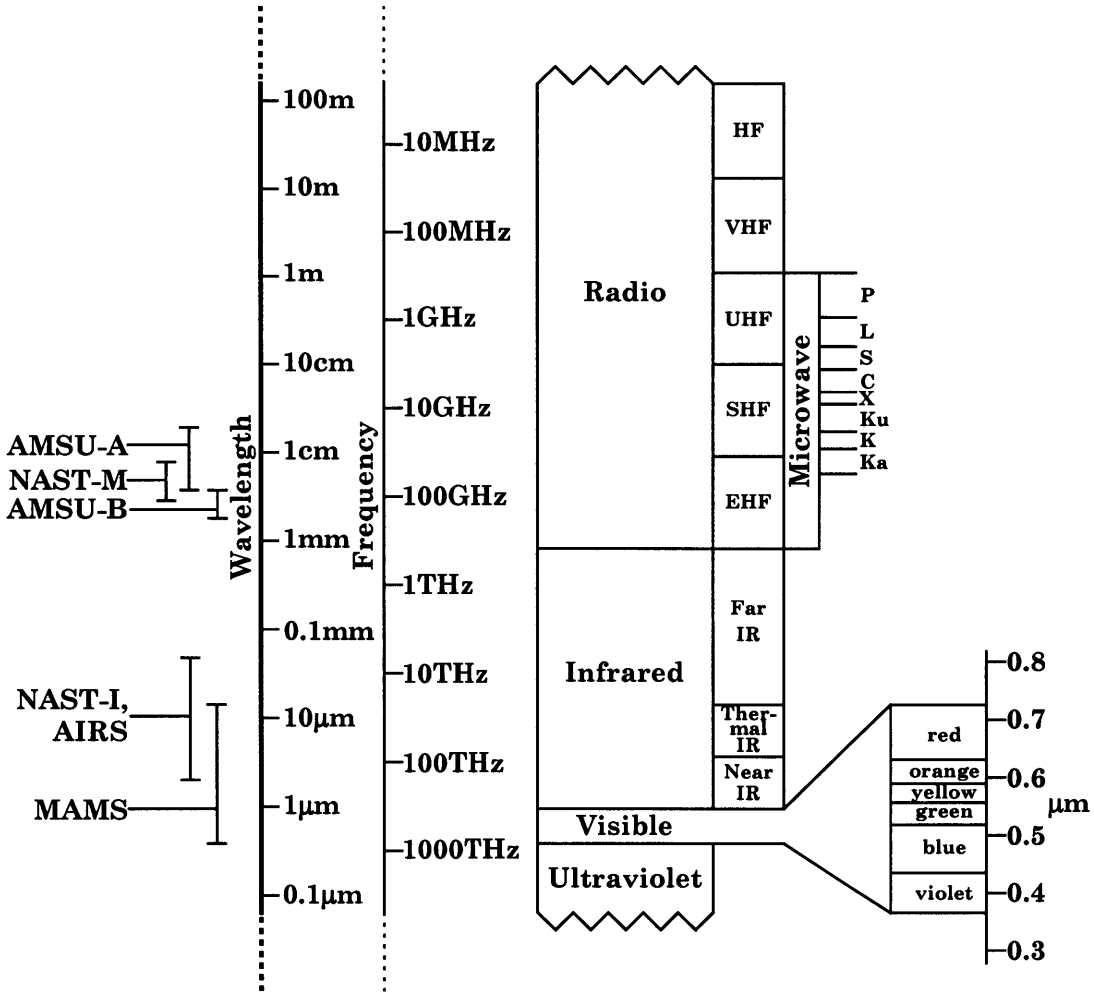

Figure 2-1: The electromagnetic spectrum. The diagram shows those parts of the electromagnetic spectrum which are important in remote sensing, together with the conventional names of the various regions of the spectrum. Also shown are wavelength regions of sensors used in the thesis: AIRS, AMSU, NAST-I, NAST-M, and MAMS. (Adapted from [6].)

Stokes parameters

A mathematical discussion of polarization can be facilitated by decomposing the $\vec{\mathbf{E}}$ vector into two components perpendicular to the direction of propagation, $\vec{\mathbf{k}}$, for a fixed point in space:

$$\vec{\mathbf{E}} = \hat{\mathbf{h}} E_h + \hat{\mathbf{v}} E_v = \hat{\mathbf{h}} e_h \cos(2\pi\nu - \varphi_h) + \hat{\mathbf{v}} e_v \cos(2\pi\nu - \varphi_v) \tag{2.13}$$

where $\vec{\mathbf{k}}$, $\hat{\mathbf{h}}$, and $\hat{\mathbf{v}}$ form an orthogonal system mutually perpendicular to one another. The four Stokes parameter may then be defined as follows:

$$I = \frac{1}{\eta}(e_h^2 + e_v^2) \tag{2.14}$$

$$Q = \frac{1}{\eta}(e_h^2 - e_v^2) \tag{2.15}$$

$$U = \frac{2}{\eta} e_h e_v \cos(\varphi) \tag{2.16}$$

$$V = \frac{2}{\eta} e_h e_v \sin(\varphi) \tag{2.17}$$

where φ is the phase difference $\varphi_h - \varphi_v$ and $\eta = \sqrt{\mu/\epsilon}$ is the characteristic impedance. In practice, it is often easier to measure the Stokes parameter rather than measure e_h, e_v, and φ directly.

2.2.3 Reflection and transmission at a planar boundary

Electromagnetic radiation from the sun or the cosmic background can reflect off cloud tops and the surface of the Earth. The characterization of the transmitted and reflected components of radiation is necessary to develop cloud and surface models. Consider a linearly-polarized plane wave propagating in free space along direction $\vec{\mathbf{k}}_i$ that is incident upon a planar dielectric material with index of refraction $n = c\sqrt{\mu\epsilon}$ at an incidence angle of θ_i. The electric fields for the incident, reflected, and transmitted waves can be expressed

as (assuming a TE wave propagating in the xz plane):

$$\begin{aligned} \vec{\mathbf{E}}_i &= \hat{\mathbf{y}} E_i e^{-j\vec{\mathbf{k}}_i \cdot \vec{\mathbf{r}}} \\ \vec{\mathbf{E}}_r &= \hat{\mathbf{y}} \Gamma E_r e^{-j\vec{\mathbf{k}}_r \cdot \vec{\mathbf{r}}} \\ \vec{\mathbf{E}}_t &= \hat{\mathbf{y}} \mathcal{T} E_t e^{-j\vec{\mathbf{k}}_t \cdot \vec{\mathbf{r}}}, \end{aligned} \tag{2.18}$$

where Γ and $\mathcal{T}$ are the complex reflection and transmission coefficients, respectively. The tangential components of the net electric field must vanish at the boundary, requiring the tangential components of all three $\vec{\mathbf{k}}$ vectors to be equal along the boundary. The tangential components of the $\vec{\mathbf{k}}$ vectors can be expressed in terms of the angles of incidence, reflection, and transmission to yield

$$k_i \sin\theta_i = k_r \sin\theta_r = k_t \sin\theta_t, \tag{2.19}$$

where $k_i = k_r = \omega\sqrt{\mu_i \epsilon_i}$ is the magnitude of the propagation vectors $\vec{\mathbf{k}}_i$ and $\vec{\mathbf{k}}_r$. The magnitude of the transmitted wave vector is $k_t = \omega\sqrt{\mu_t \epsilon_t}$, which is in general not equal to k_i. Substitution into Eq. 2.19 gives the reflection law

$$\theta_r = \theta_i, \tag{2.20}$$

and Snell's law

$$\frac{\sin\theta_t}{\sin\theta_i} = \frac{k_i}{k_t} = \frac{n_i}{n_t}. \tag{2.21}$$

Given $\vec{\mathbf{k}}_r$ and $\vec{\mathbf{k}}_t$, the complex reflection and transmission coefficients can be found by supplementing the boundary condition for continuity of the electric field with a similar equation for the tangential magnetic field. For TE waves,

$$\Gamma_{TE} = \frac{\eta_t \cos\theta_i - \eta_i \cos\theta_t}{\eta_t \cos\theta_i + \eta_i \cos\theta_t} \tag{2.22}$$

$$\mathcal{T}_{TE} = \frac{2\eta_t \cos\theta_i}{\eta_t \cos\theta_i + \eta_i \cos\theta_t} \tag{2.23}$$

and for TM waves,

$$\Gamma_{TM} = \frac{\eta_i \cos\theta_i - \eta_t \cos\theta_t}{\eta_i \cos\theta_i + \eta_t \cos\theta_t} \tag{2.24}$$

$$\mathcal{T}_{TM} = \frac{2\eta_i \cos\theta_i}{\eta_i \cos\theta_i + \eta_t \cos\theta_t}. \tag{2.25}$$

As an important consequence of the preceding equations, unpolarized radiation incident upon a planar dielectric surface can become partially or totally polarized on reflection. For example, a portion of the unpolarized microwave radiation emitted by the atmosphere is reflected by the ocean surface and another portion is absorbed and re-emitted by the ocean surface. However, the TE and TM components of the emitted radiation are different when viewed from oblique angles, a characteristic that can be used to discriminate surface water from rainfall.

2.3 Absorption of electromagnetic waves by atmospheric gases

A knowledge of the mechanisms of electromagnetic radiation interaction with matter, as well as some of the fundamental properties of matter itself, is necessary to infer and interpret information about the atmosphere. In the following two sections, the interactions are described on a microscopic (molecular) and macroscopic (particle) level.

2.3.1 Mechanisms of molecular absorption

The total internal energy of an isolated molecule consists of three types of energy states,

$$\mathcal{E} = \mathcal{E}_e + \mathcal{E}_v + \mathcal{E}_r \tag{2.26}$$

where $\mathcal{E}_e$ = electronic energy, $\mathcal{E}_v$ = vibrational energy, and $\mathcal{E}_r$ = rotational energy. Rotational energy is associated with rotational motions of the atoms of the molecule about its center of mass, and vibrational energy is associated with vibrational motions of the atoms about their equilibrium positions. Radiation is absorbed (or emitted) when a transition takes place from a lower (or higher) energy state to a higher (or lower) energy state. The

frequency (ν) of the absorbed (or emitted) photon is given by the Bohr frequency condition,

$$\nu = \frac{\mathcal{E}_h - \mathcal{E}_l}{h}, \tag{2.27}$$

where h is Planck's constant and $\mathcal{E}_h$ and $\mathcal{E}_l$ are the internal energies of the higher and lower molecular states, respectively. The absorption spectrum due to a single transition is called an absorption line. Absorption by molecules in the mid- and near-infrared occur by vibration (although a mixture of vibrations and rotations are usually induced at these frequencies). In the microwave and far infrared, rotational transitions are the dominant mechanism of energy transfer.

2.3.2 Line shapes

Based on Eq. 2.27, the absorption (or emission) spectrum of an isolated, unperturbed, stationary molecule consists of sharply-defined frequency lines corresponding to transitions between quantized energy levels of the molecule. Atmospheric gases, however, consist of molecules that are in constant motion, interacting and colliding with one another. These disturbances cause the absorption lines to broaden. The two most important sources of line broadening are Doppler (thermal) broadening and pressure (collision) broadening, which is dominant for most frequencies up to an altitude of approximately 40km [13].

2.3.3 Absorption coefficients and transmission functions

Line shape $f(\nu - \nu_0)$, line position (ν_0), and line strength (S) mathematically define the absorption coefficient:

$$\kappa_\nu = S\, f(\nu - \nu_0). \tag{2.28}$$

The line strength of a specific atmospheric gas is governed by the number of absorbing molecules of that gas per unit volume, the temperature of the gas, and the molecular parameters associated with that transition.

Absorption of radiation by gases in the Earth's atmosphere is described in terms of transmission functions (or simply, transmittance). Lambert's law states that the change in

radiance intensity along a path ds is proportional to the amount of matter along the path:

$$dR_\nu = -\kappa_\nu R_\nu \, ds \tag{2.29}$$

where κ_ν is the volume absorption coefficient. Integration of Lambert's law along the path connected by s_1 and s_2 yields

$$R_\nu(s_2) = \mathcal{T}_\nu(s_1, s_2) R_\nu(s_1) \tag{2.30}$$

where $\mathcal{T}_\nu(s_1, s_2)$ is the monochromatic transmittance defined as

$$\mathcal{T}_\nu(s_1, s_2) = e^{-\int_{s_1}^{s_2} \kappa_\nu \, ds}. \tag{2.31}$$

The optical path (or thickness)[2] between s_1 and s_2 is defined as

$$\tau_\nu(s_1, s_2) = \int_{s_1}^{s_2} \kappa_\nu \, ds. \tag{2.32}$$

The absorption coefficient, transmittance, and optical path form the mathematical basis for the subject of Section 2.5 – radiative transfer. In practice, these quantities are not monochromatic, but band-averaged over some spectral response function of the instrument.

2.3.4 The atmospheric absorption spectra

The atmospheric absorption spectrum for microwave frequencies is shown in Fig. 2-2. Notable features include the water vapor absorption lines centered at 22.235, 183.31, and 325.15 GHz (lines at 380.20 and 448.00 GHz are difficult to identify on the plot) and oxygen absorption lines near 60, 118.75, 368.50, 424.76, and 487.25 GHz. The atmospheric absorption spectrum for infrared wavelengths between 3.6 and 16.1 μm is shown in Fig. 2-3. Notable features include the water vapor absorption lines near 6–7 μm, ozone absorption lines near 10 μm, and carbon dioxide absorption lines near 4.3–4.6 μm and 13-15 μm.

[2]The related quantities optical depth and opacity will be defined later.

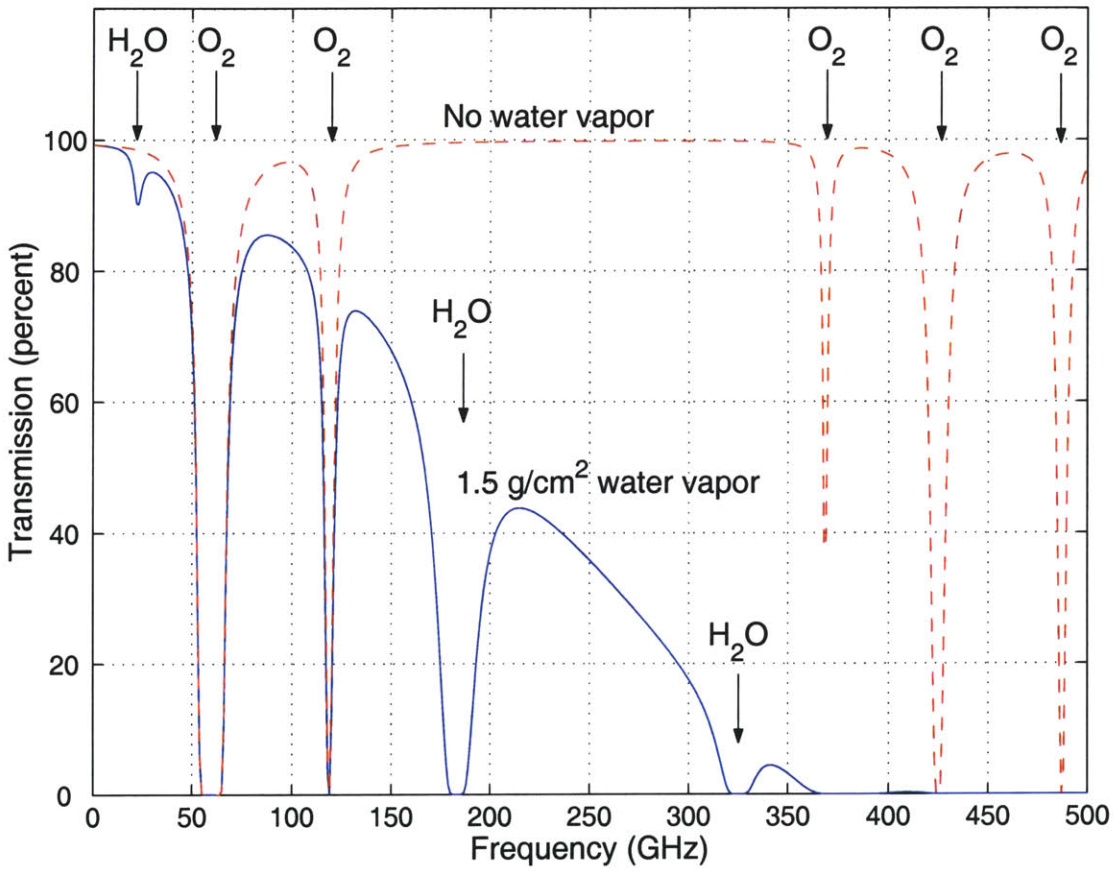

Figure 2-2: The microwave absorption spectrum. Two calculations for the percent transmission (nadir view) using the 1976 Standard Atmosphere are shown, one assuming no water vapor and one assuming 1.5 g/cm^2.

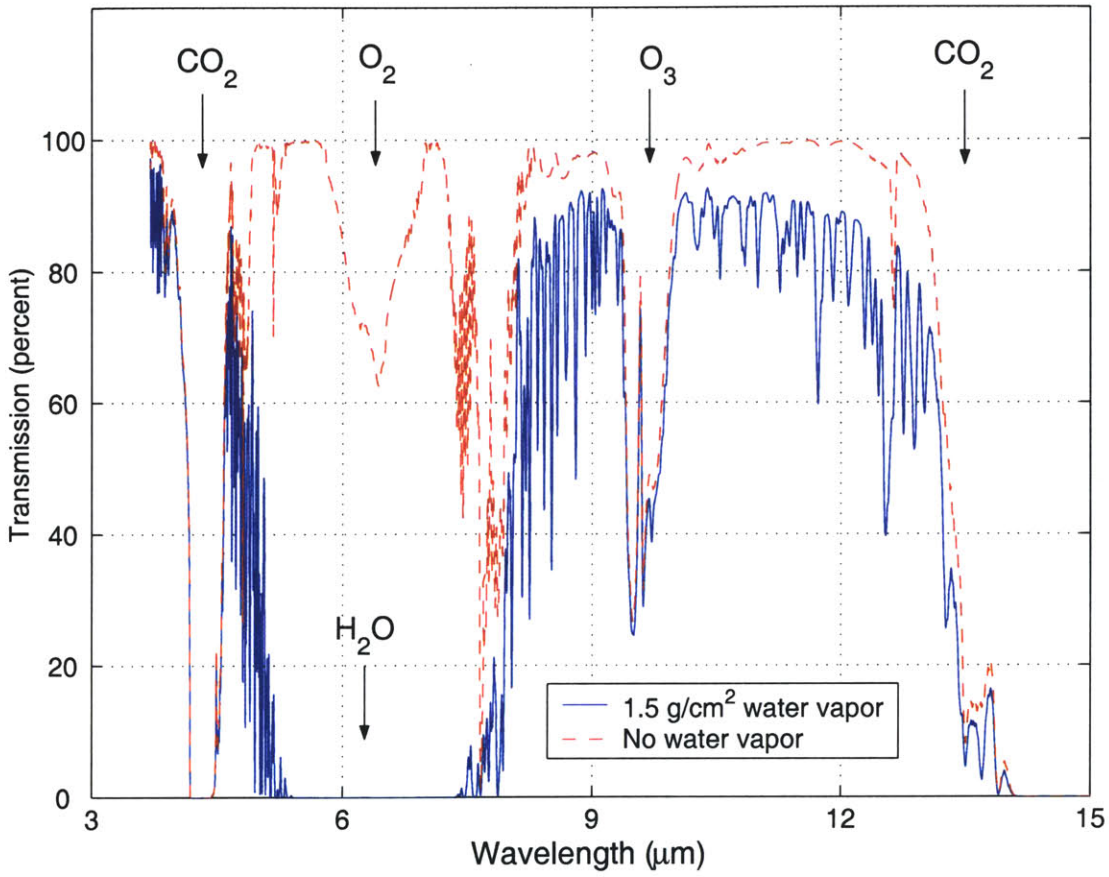

Figure 2-3: The thermal infrared absorption spectrum. Two calculations for the percent transmission (nadir view) using the 1976 Standard Atmosphere are shown, one assuming no water vapor and one assuming 1.5 g/cm^2.

2.4 Scattering of electromagnetic waves by atmospheric particles

In addition to the molecular absorption mechanisms discussed earlier, electromagnetic waves are also scattered and absorbed by much larger particles often found in the atmosphere, such as cloud water droplets, raindrops, or even dust. The scattering of electromagnetic waves upon interaction with atmospheric particles provides a tool that can be used to help retrieve many microphysical parameters related to clouds and precipitation. The fundamental concepts derived here are used in Chapter 4 to qualitatively characterize several microphysical parameters of the rainbands surrounding Hurricane Bonnie (August, 1998) based on microwave observations made by NAST-M.

2.4.1 Mie scattering

A suspended particle of geometrical cross-section A will absorb a fraction of incident power and will also scatter incident power in all directions. The ratio of absorbed power P_a (W) to incident power density S (W/m^2) is known as the absorption cross-section

$$C_a = \frac{P_a}{S}, \text{ m}^2 \tag{2.33}$$

and the ratio of C_a to the physical cross-section A is known as the efficiency factor Q_a. For a spherical particle of radius r, $A = \pi r^2$ and therefore

$$Q_a = \frac{C_a}{\pi r^2}. \tag{2.34}$$

Analogous quantities for scattering, the scattering cross-section C_s and the scattering efficiency Q_s are defined as

$$C_s = \frac{P_s}{S}, \text{ m}^2 \tag{2.35}$$

$$Q_s = \frac{C_s}{\pi r^2}. \tag{2.36}$$

The total power lost due to absorption and scattering (known as the extinction) is $P_a + P_s$ and the resulting extinction cross-section C_e and efficiency Q_e are

$$C_e = C_a + C_s, \tag{2.37a}$$

$$Q_e = Q_a + Q_s. \tag{2.37b}$$

The solution for the scattering and absorption of electromagnetic waves in free space by a dielectric sphere of radius r was formulated by Mie in terms of the "size parameter"

$$\chi = \frac{2\pi r}{\lambda} \tag{2.38}$$

and

$$n = \sqrt{\epsilon_c}, \tag{2.39}$$

where λ is the wavelength of the incident wave, n is the complex refractive index of the particle and ϵ_c is the corresponding complex dielectric constant. Mie's expressions for the scattering and extinction efficiencies of the sphere are given by

$$Q_s(n,\chi) = \frac{2}{\chi^2} \sum_{m=1}^{\infty} (2m+1)(|a_m|^2 + |b_m|^2) \tag{2.40a}$$

$$Q_e(n,\chi) = \frac{2}{\chi^2} \sum_{m=1}^{\infty} (2m+1)\mathrm{Re}\{a_m + b_m\} \tag{2.40b}$$

where a_m and b_m are known as the Mie coefficients

$$a_m = -\frac{j_m(n\chi)[\chi j_m(\chi)]' - j_m(\chi)[n\chi j_m(n\chi)]'}{j_m(n\chi)[\chi h_m(\chi)]' - h_m(\chi)[n\chi j_m(n\chi)]'} \tag{2.41a}$$

$$b_m = -\frac{j_m(\chi)[n\chi j_m(n\chi)]' - n^2 j_m(n\chi)[\chi j_m(\chi)]'}{h_m(\chi)[n\chi j_m(n\chi)]' - n^2 j_m(n\chi)[\chi h_m(\chi)]'} \tag{2.41b}$$

where $j_m(\cdot)$ and $h_m(\cdot)$ are the spherical Bessel and Hankel functions of the first kind, and the $(\cdot)'$ operator denotes the complex conjugation.

2.4.2 The Rayleigh approximation

The Mie expressions for Q_s and Q_e can be approximated with negligible error if the particle size is much smaller than the wavelength of the incident wave ($|n\chi| \ll 1$). The Rayleigh approximation is obtained by retaining only the most significant terms in the series expansion:

$$Q_s = \frac{8}{3}\chi^4 |K|^2 \tag{2.42}$$

$$Q_e = 4\chi \mathrm{Im}\{-K\} + \frac{8}{3}\chi^4 |K|^2 \tag{2.43}$$

and

$$Q_a = 4\chi \mathrm{Im}\{-K\}, \tag{2.44}$$

where K is a complex quantity defined in terms of the complex index of refraction n

$$K = \frac{n^2 - 1}{n^2 + 2} = \frac{\epsilon_c - 1}{\epsilon_c + 2}. \tag{2.45}$$

Note that in the Rayleigh limit the scattering efficiency scales as the fourth power of frequency, whereas the absorption efficiency scales linearly with frequency, for a fixed particle size and a frequency-independent index of refraction. For water, the index of refraction is frequency-dependent, and the absorption efficiency scales as frequency squared (for frequencies below 90 GHz or so) when this dependence is included.

2.4.3 Comparison of scattering and absorption by hydrometeors

Fig. 2-4 shows scattering and absorption contributions of water spheres, both in the liquid and ice phases. Deirmendjian's recursive procedure [14] was used to calculate the Mie coefficients; 80 terms were used to approximate the series. For liquid droplets, absorption is dominant in the Rayleigh region and scattering is dominant in the Mie region. For ice, scattering is dominant for all but the lowest microwave frequencies. The frequency dependence of scattering and absorption can be used to retrieve information about the particle size distributions of clouds, and the related quantity, rainrate. The distribution and type of hydrometeors found in typical clouds vary widely, and monodispersive models are inadequate. More complicated modeling is beyond the scope of this thesis; an excellent discussion can be found in [15].

2.5 Radiative transfer in a non-scattering planar-stratified atmosphere

A sensor high above the Earth's surface receives emission from the Earth and its atmosphere, along with any reflected components of solar and cosmic background radiation. Measurements of this emission allow the retrieval of many atmospheric parameters, including the temperature and water vapor profile, the amount of cloud liquid water, rainrates, and sea surface temperatures.

2.5.1 Equilibrium radiation: Planck and Kirchhoff's laws

The solution to the simple differential equation presented by Lambert's law (Eq. 2.29) is referred to as Beer's law

$$R_\nu(s'') = R_\nu(s')e^{-\int_{s'}^{s''} \kappa_\nu(s)\,ds}. \tag{2.46}$$

In addition to absorption of radiation by the gas contained within path s' to s'', Kirchhoff's law states that if in thermal equilibrium, the gas also emits radiation in an amount

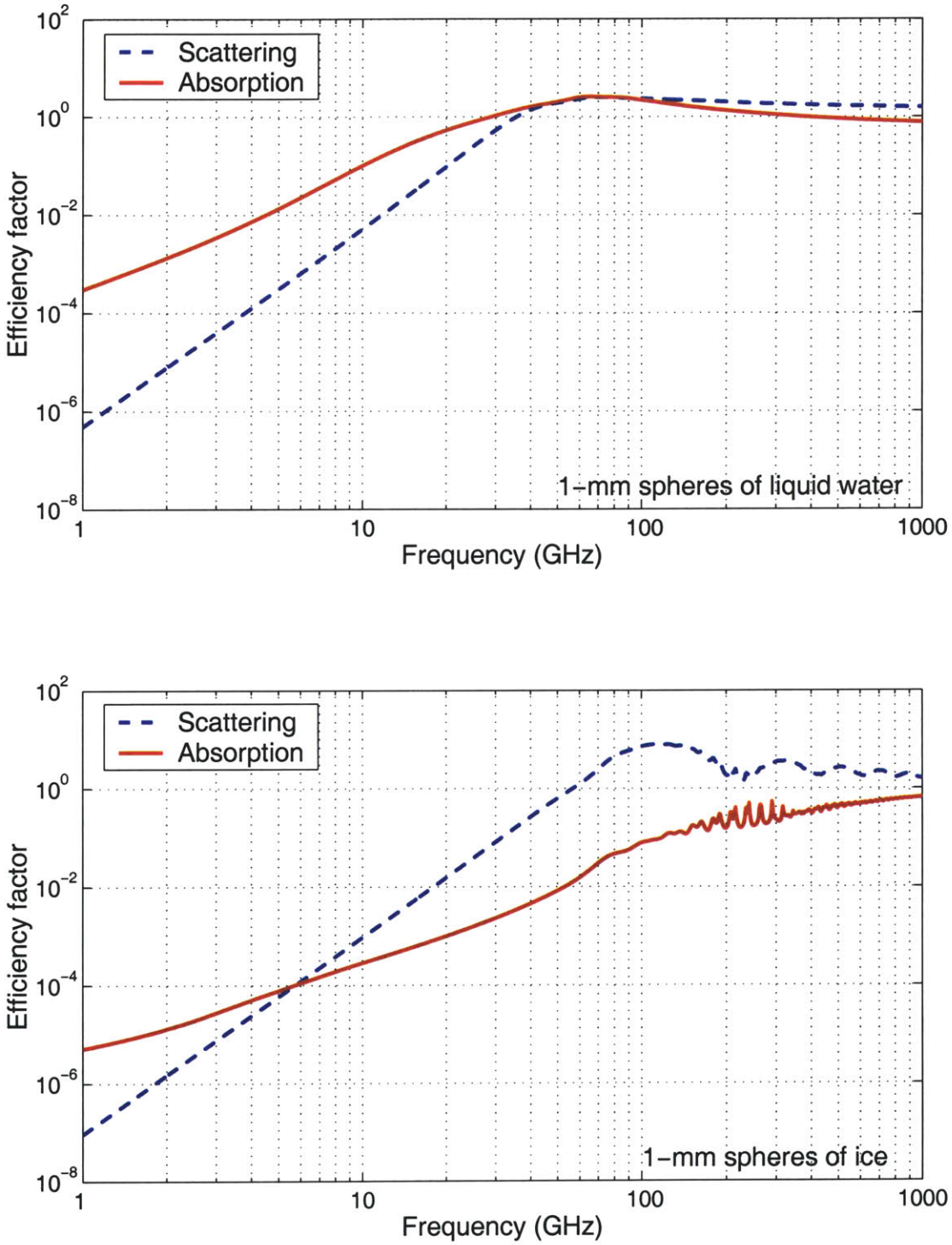

Figure 2-4: Scattering and absorption efficiency for water spheres with 1-mm radius. Liquid water spheres (273 K) are shown in the top plot and ice spheres (266 K) are shown in the bottom plot.

proportional to the absorption coefficient κ_ν:

$$R_\nu(\text{emission}) = \kappa_\nu \mathcal{J}_\nu(T), \tag{2.47}$$

where $\mathcal{J}_\nu(T)$ is the radiation intensity produced (at each of two orthogonal polarizations) by a blackbody at temperature T and frequency ν:

$$\mathcal{J}_\nu(T) = \frac{h\nu^3}{c^2} \frac{1}{e^{h\nu/kT} - 1} \ \text{W} \cdot \text{m}^{-2} \cdot \text{ster}^{-1} \cdot \text{Hz}^{-1}. \tag{2.48}$$

Values for the following physical constants can be found in the Appendix: Planck's constant (h), Boltzmann's constant (k), and the speed of light in a vacuum (c). The Planck equation exhibits a nonlinear relationship between intensity and temperature. The degree of the nonlinearity is frequency-dependent, and is shown in Fig. 2-5. The nonlinearity is most severe at the higher frequencies (shorter wavelengths) and almost nonexistent at the microwave frequencies. The approximation of the Planck radiance by the linear Taylor series term is called the Rayleigh-Jeans (RJ) approximation, and the microwave brightness temperature is defined as the scaled intensity:

$$B_\nu = \frac{c^2}{2\nu^2 k} R_\nu. \tag{2.49}$$

Note that if a radiometer is calibrated against a blackbody and all departures from the Rayleigh-Jeans law are ignored, brightness temperature is effectively redefined as

$$B_\nu = \frac{c^2}{2\nu^2 k} R_\nu + \frac{h\upsilon}{2k} \tag{2.50}$$

and accuracy is better than 0.1 K for frequencies up to 300 GHz and terrestrial temperatures. When extremely cold temperatures are encountered (e.g., the cosmic background) corrections to the RJ approximation are needed.

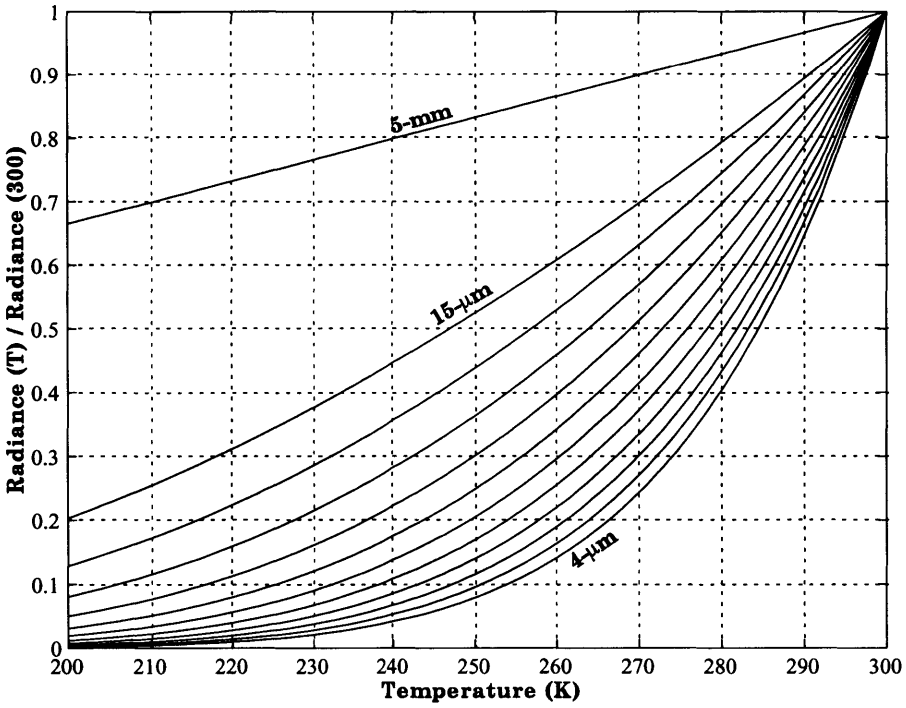

Figure 2-5: Nonlinearity of the Planck function as a function of wavelength.

2.5.2 Radiative transfer due to emission and absorption

The net change in radiation along ds due to the combination of emission and absorption is

$$dR_\nu = dR_\nu(\text{emission}) + dR_\nu(\text{absorption}). \tag{2.51}$$

Substitution of Eq. 2.29 and Eq. 2.47 into Eq. 2.51 yields the Schwartzchild equation

$$\frac{dR_\nu}{ds} = -\kappa_\nu[R_\nu - \mathcal{J}_\nu(T)], \tag{2.52}$$

which mathematically describes how radiation is transferred from one layer to another layer as a result of absorption and emission. The intensity of radiation leaving the path is therefore a function of both the absorber along the path and the temperature along the path. Passive (emission-based) sounding of constituent concentration and temperature is based upon this principle.

2.5.3 Integral form of the radiative transfer equation

Differentiation of Eq. 2.32 gives

$$d\tau_\nu(s) = -\kappa_\nu(s)\, ds, \tag{2.53}$$

where we adopt the convention that τ increases from zero downward from the top of the atmosphere to a maximum value τ^* (the opacity of the atmosphere) at the surface. Multiplying both sides of Eq. 2.52 by $e^{-\tau_\nu(s)}$ and combining terms gives

$$\frac{dR_\nu e^{-\tau_\nu(s)}}{d\tau_\nu} = -\mathcal{J}_\nu e^{-\tau_\nu(s)}, \tag{2.54}$$

which upon integration from path s' to s'' yields

$$R_\nu(s'')e^{-\tau_\nu(s'')} - R_\nu(s')e^{-\tau_\nu(s')} = \int_{\tau(s'')}^{\tau(s')} \mathcal{J}_\nu(s)e^{-\tau_\nu(s)}\, d\tau(s). \tag{2.55}$$

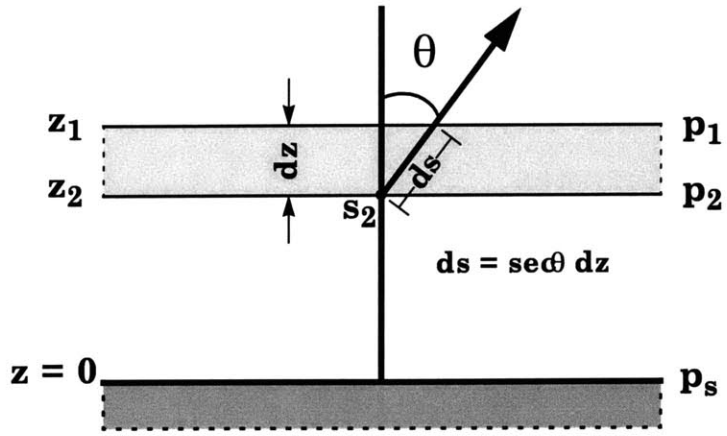

Figure 2-6: Geometry of the planar-stratified atmospheric radiative transfer equation.

Equation 2.55 can be rearranged into the integral form of the radiative transfer equation as follows:

$$R_\nu(s'') = R_\nu(s')e^{-[\tau_\nu(s')-\tau_\nu(s'')]} + \int_{s'}^{s''} \mathcal{J}_\nu(s)e^{-[\tau_\nu(s)-\tau_\nu(s'')]}\, d\tau_\nu(s). \tag{2.56}$$

The equivalent relation in terms of the absorption coefficient κ_ν is

$$R_\nu(s'') = R_\nu(s')e^{-\int_{s'}^{s''} \kappa_\nu(s)ds} + \int_{s'}^{s''} \kappa_\nu(s)\mathcal{J}_\nu(s)e^{-\int_{s}^{s''} \kappa_\nu(\sigma)\, d\sigma}\, ds. \tag{2.57}$$

The angular properties of emission have thus far been neglected, but can easily be included for the case of a horizontally-homogeneous vertically-stratified atmosphere by noting that an angular tilt of θ results in an increase in the pathlength by a factor of $\sec\theta$ (see Fig. 2-6). Optical depth is related to optical path as follows:

$$\tau(s) = \tau(z)\sec(\theta). \tag{2.58}$$

After including the angular terms, the final form of the radiative transfer equation describ-

ing the radiation intensity observed at altitude L and viewing angle θ can be formulated by including reflected atmospheric and cosmic contributions and the radiance emitted by the surface:

$$\begin{aligned} R_\nu(L) &= \int_0^L \kappa_\nu(z)\mathcal{J}_\nu[T(z)]e^{-\int_z^L \sec\theta\kappa_\nu(z')\,dz'}\sec\theta\,dz \\ &+ \rho_\nu e^{-\tau^*\sec\theta}\int_0^L \kappa_\nu(z)\mathcal{J}_\nu[T(z)]e^{-\int_0^z \sec\theta\kappa_\nu(z')\,dz'}\sec\theta\,dz \qquad (2.59) \\ &+ \rho_\nu e^{-2\tau^*\sec\theta}\mathcal{J}_\nu(T_c) \\ &+ \varepsilon_\nu e^{-\tau^*\sec\theta}\mathcal{J}_\nu(T_s) \end{aligned}$$

(2.60)

where ε_ν is the surface emissivity, ρ_ν is the surface reflectivity, T_s is the surface temperature, and T_c is the cosmic background temperature $(2.736 \pm 0.017K)$.

2.5.4 Weighting function

The first term in Eq. 2.60 can be recast in terms of the transmittance function $\mathcal{T}_\nu$(z):

$$R_\nu(L) = \int_0^L \mathcal{J}_\nu[T(z)]\left(\frac{d\mathcal{T}_\nu(z)}{dz}\right)dz. \qquad (2.61)$$

The derivative of the transmittance function with respect to altitude is often called the weighting function

$$W_\nu(z) \triangleq \frac{d\mathcal{T}_\nu(z)}{dz}. \qquad (2.62)$$

and gives the relative contribution of the radiance emanating from each altitude. Note that the Planck radiances are weighted, not the temperature profile. It is sometimes useful to define a temperature weighting function, where the temperature profile is weighted directly. One approach is to express the radiance intensity $R_\nu(L)$ in terms of a "blackbody-equivalent" brightness temperature $T_{B,\nu}(L)$ (the temperature of a blackbody that produces a radiance equivalent to $R_\nu(L)$, N.B. $T_{B,\nu}(L) \neq B_\nu$) and linearize about a nominal temperature profile $T_0(z)$ and corresponding radiance $R_{0,\nu}(L)$.

Temperature weighting function

For a particular frequency, the blackbody-equivalent radiance may be written as follows:

$$T_{B,\nu}(L) = \mathcal{J}_\nu^{-1}(W_\nu(\mathcal{J}_\nu(T_z))), \tag{2.63}$$

where $\mathcal{J}_\nu(\cdot)$ is the Planck function, $W(\cdot)_\nu$ is the integration against the weighting function, and $\mathcal{J}_\nu^{-1}(\cdot)$ is the inverse Planck function. The first-order Taylor series approximation of $\mathcal{J}_\nu^{-1}(W_\nu(\mathcal{J}_\nu(\cdot)))$ is then

$$\begin{aligned} R_\nu(L) &= R_{0,\nu}(L) + \frac{d\mathcal{J}_\nu^{-1}}{dW_\nu}\frac{dW_\nu}{d\mathcal{J}_\nu}\frac{d\mathcal{J}_\nu}{dT}[T(z) - T_0(z)] && (2.64) \\ &= W_{T,\nu}(z)[T(z) - T_0(z)] + R_{0,\nu}(L) && (2.65) \end{aligned}$$

where $W_{T,\nu}(z)$ is defined as the temperature weighting function:

$$W_{T,\nu}(z) \triangleq \left.\frac{d\mathcal{J}_\nu^{-1}}{dW_\nu}\frac{dW_\nu}{d\mathcal{J}_\nu}\frac{d\mathcal{J}_\nu}{dT}\right|_{T_0(z)}. \tag{2.66}$$

The difference between the Planck weighting function and the temperature weighting function can be significant for short-wavelength channels, as shown in Fig. 2-7. The temperature weighting functions are sharper and peak lower in the atmosphere. The RMS errors (in units of blackbody-equivalent brightness temperature) resulting from the use of the first-order approximation given by Eq. 2.65 (and assuming the weighting functions are independent of atmospheric parameters) over a representative set of atmospheric profiles $T(z)$ (with $T_0(z) = E[T(z)]$) are shown in Fig. 2-8 for the AIRS channel set. The two dominant sources of error are the nonlinearity of the Planck function (most evident in the short-wavelength channels) and the nonlinearity of the atmospheric transmittance (most evident in the water vapor channels). Temperature weighting functions are almost never used directly to evaluate the radiative transfer equation because of the large errors introduced by the exclusion of nonlinearities. Nevertheless, the temperature weighting functions provide a useful characterization of the way different layers of the atmosphere at different temperatures contribute to the radiation emitted at the top of the atmosphere.

Returning to the special case of microwave frequencies and terrestrial temperatures (i.e.,

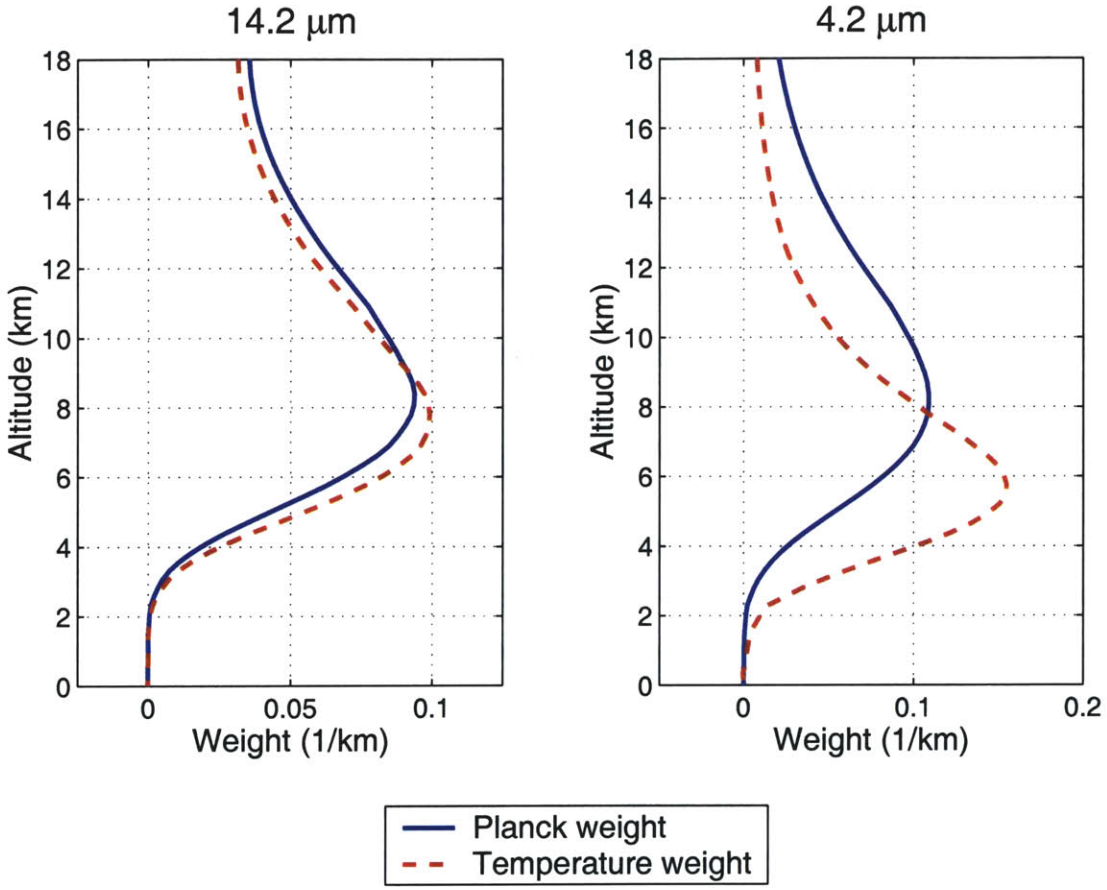

Figure 2-7: The Planck radiance weighting function and the temperature weighting function for two infrared channels.

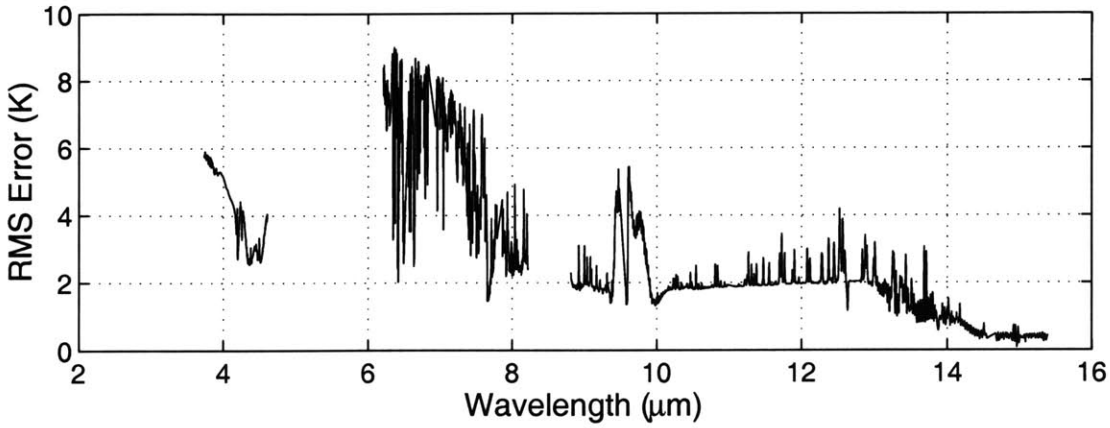

Figure 2-8: AIRS RMS radiance error due to first-order Planck approximation.

the Rayleigh-Jeans regime), Eq. 2.60 can be expressed in terms of the physical temperature profile $T(z)$ and the brightness temperature B_ν:

$$\begin{aligned} B_\nu(L) &= \int_0^L \kappa_\nu(z)T(z)e^{-\int_z^L \sec\theta\kappa_\nu(z')\,dz'} \sec\theta\,dz \\ &+ \rho_\nu e^{-\tau^*\sec\theta}\int_0^L \kappa_\nu(z)T(z)e^{-\int_0^z \sec\theta\kappa_\nu(z')\,dz'} \sec\theta\,dz \\ &+ \rho_\nu e^{-2\tau^*\sec\theta}\widetilde{T}_c \\ &+ \varepsilon_\nu e^{-\tau^*\sec\theta}T_s \end{aligned} \tag{2.67}$$

where $\widetilde{T}_c$ is the corrected cosmic background temperature

$$\widetilde{T}_c = \frac{h\nu}{2k}\frac{(e^{h\nu/kT_c}+1)}{e^{(h\nu/kT_c}-1)}. \tag{2.68}$$

Note that in the microwave case, $W_{T,\nu}(z) = W_\nu(z)$.

2.6 Passive spectrometer systems

Measurement of the Earth's radiation at a spectral resolution high enough to study details of molecular absorption bands is achieved using spectrometer sensor systems. The chapter concludes with a brief discussion of various spectrometer technologies, with a focus

on performance advantages, disadvantages, and engineering tradeoffs that must be considered when selecting an instrument to carry out a given remote sensing task. The concepts presented here are rudimentary; for more details, the reader is referred to [16] and [17].

2.6.1 Optical spectrometers

For the purpose of an elementary discussion, optical spectrometers can be grouped into three system types: prism dispersion, diffraction grating, and radiation interference. The systems differ primarily in the mechanisms used to separate incident radiation into discrete spectral components.

Prism dispersion systems

A prism spectrometer produces radiance spectra by passing the incident radiation through a prism. The prism causes the radiation to disperse (bend) by a frequency-dependent angle. The degree to which the radiation is dispersed is determined by the refractive index of the prism. The spectra are usually detected either by sweeping the dispersed radiation across a fixed detector (e.g., a photomultiplier), or sweeping the detector across the radiation field. The spectral resolution achieved by prism spectrometers is relatively coarse, and they are therefore used typically in imaging systems. The calibration of prism systems is also problematic because of the frequency dependence of the index of refraction of the prism.

Diffraction grating systems

A diffraction grating disperses radiation into spectra through angular-dependent interference patterns that result when radiation is passed through a dense array of small slits. Diffraction grating systems operate by either transmitting radiation through or reflecting radiation from a series of closely-spaced parallel lines etched on plastic film (for transmission) or a metallic surface (for reflection). Transmission gratings generally perform poorly in comparison with reflection gratings, which are used in high-performance space spectrometers. The spectral resolving power of grating spectrometers typically exceeds that of prism spectrometers by an order of magnitude, at the expense of increased instrumentation complexity. The AIRS instrument, which is simulated in Chapter 6, is a diffraction grating

spectrometer.

Interferometer systems

The interferometer spectrometer operates quite differently than the prism or grating spectrometer in that interference effects instead of dispersion effects are used to separate spectra. One of the simplest types of interferometers is the Michelson interferometer, which splits incoming radiation into two beams of unequal length by a partially-silvered plate (beam splitter) and later recombines the beams with a known path difference. The path difference can be varied uniformly by moving a mirror at a constant speed, causing the two beams to move in and out of phase at the detector. The intensity of the resulting waveform (termed the interferogram) is related to the spectral intensity of the incident radiation by the Fourier transform. The interferogram is the autocorrelation function of the optical signal. The performance of the interferometer spectrometer relative to the grating spectrometer depends on a number of factors, including the nature (e.g., mechanical and electrical) and origin (e.g., photon and thermal) of system noise. A more detailed treatment of some of these factors is given in Chapter 5 for the NAST-I interferometer spectrometer.

2.6.2 Microwave spectrometers

Microwave and optical spectrometer systems are conceptually similar. Perhaps the most pernicious source of error in microwave spectrometer systems is the instability of the receiver, and the primary difference among microwave systems is the way in which receiver sensitivity is compromised for receiver stability. Three types of microwave spectrometers[3] are now discussed.

Total power spectrometer

The simplest type of microwave spectrometer measures the power of incident radiation over a collection of bandwidths B_n, integrated over a time τ. The RMS sensitivity of the measurement at any given channel is a function of the receiver noise (T_R, expressed in

[3]The term "radiometer" is used when incident electromagnetic power is measured across a given frequency band. The term "spectrometer" is used when power across several frequency bands (or channels) is measured.

units of temperature), the incident radiation (T_A, expressed in units of temperature), the bandwidth (B_n, Hz), and the integration time[4] (τ, sec):

$$\Delta T_{rms} = \frac{T_R + T_A}{\sqrt{B_n \tau}}. \tag{2.69}$$

Equation 2.69 assumes that the receiver gain is perfectly stable. Fluctuations in receiver gain reduce the system sensitivity as follows:

$$\Delta T_{rms} = (T_R + T_A)\sqrt{\frac{1}{B_n \tau} + \left(\frac{\Delta G}{G}\right)^2} \tag{2.70}$$

where $\Delta G/G$ is the fractional receiver gain drift. It is not uncommon for the gain drift component to dominate the noise expressed in Eq. 2.70. The system noise performance of NAST-M, a total-power spectrometer, is discussed in more detail in Chapter 4. The following two microwave spectrometer systems demonstrate different approaches for minimizing the noise due to receiver gain drift.

Dicke spectrometer

The Dicke spectrometer is essentially a total-power spectrometer with two additional features: 1) a switch used to modulate the receiver input signal, and 2) a synchronous detector, placed between the detector and integrator. The modulation consists of periodically switching the receiver input between the antenna and a reference source (T_{ref}) at a rate higher than the highest significant spectral component of the gain variation. If the noise temperature of the reference source is close to the antenna temperature T_A, the system sensitivity of the Dicke spectrometer becomes

$$\Delta T_{rms} = \frac{2(T_R + T_A)}{\sqrt{B_n \tau}}. \tag{2.71}$$

[4]It is assumed for the purposes of this discussion that the detector signal is convolved with a boxcar of length τ. Other averaging kernels may be used, with tradeoffs between sensitivity and memory effects.

Table 2.3: Comparison of certain characteristics of the 4.3-μm, 15.0-μm, and 5.0-mm spectral regions [18]. Detector noise RMS: 0.15 K (IR) and 0.7 K (MW).

		4.3-μm	15.0-μm	5.0-mm
ENERGY	200 K	1.25	5000	1
(Relative Planck Radiance)	300 K	200	15000	1
TEMPERATURE SENSITIVITY	200 K	1	10	4
(Relative to Detector Noise)	300 K	20	6	1
CLOUD TRANSMISSION	Water	6%	1%	96%
	Ice	1%	1%	99.98%

Correlation spectrometer

Another possible method of stabilizing a receiver system involves the correlation of signals. Two separate receivers are used in a correlation spectrometer, and the resulting output voltages are multiplied and detected. The average value of a product of two independent noise temperatures is zero, and because only correlated noise voltages yield a DC output, receiver gain instabilities will not affect the sensitivity of the correlation spectrometer. The sensitivity of the correlation spectrometer is a factor of $\sqrt{2}$ better than the Dicke spectrometer. However, two separate receivers are needed.

2.7 Summary

The Earth's atmosphere and its interaction with electromagnetic radiation has been examined on microscopic (molecular absorption) and macroscopic (particle extinction) levels. If the atmosphere is assumed to be non-scattering, horizontally homogeneous, and vertically stratified, straightforward relations can be derived for the radiation intensity observed by a downward-viewing satellite or aircraft sensor. The frequency dependence of scattering, absorption, and the Planck radiance offers various advantages for atmospheric profile sounding in the presence of clouds (see Table 2.7). Various instrument technologies present performance advantages and disadvantages that must be considered when implementing a remote sounding system.

Chapter 3

Hyperspectral Signal Processing and Estimation Techniques

MODERN atmospheric sounders measure radiance with unprecedented resolution (spatial, spectral, and temporal) and accuracy. For example, the Advanced InfraRed Sounder (AIRS, operational in late 2002) will provide a spatial resolution of ~15 km, a spectral resolution of $\Delta\lambda/\lambda = 1/1200$ (with ~2400 channels), and a radiometric accuracy on the order of 0.1 K. Approximately 90 percent of the Earth's atmosphere is measured (in the horizontal dimension) every 12 hours or so. This wealth of data presents two major challenges from the point of view of retrieval algorithm development. The first deals with the robustness of the retrieval operator, and involves maximal use of the geophysical content of the radiance data with minimal interference from instrument and atmospheric noise (both of which will be defined in more detail later). The second deals with computational efficiency, where it is desirable to implement a robust algorithm within a given computational budget.

This chapter reviews techniques for the assessment of the information content of a measurement, and the subsequent estimation of a related geophysical parameter. Information

Mathematical notation: henceforth, lowercase characters will be used to denote scalars, uppercase characters will be used to denote vectors, and bold uppercase characters will be used to denote matrices. Bold lowercase characters will be used to denote vector-valued functions with vector inputs. The expected value operator is $E(\cdot)$ and the transpose operator is $(\cdot)^T$.

theoretic and principal component techniques are discussed, and Bayesian and non-Bayesian estimators are reviewed.

3.1 Analysis of the information content of hyperspectral data

The information content of a measurement can be defined in a number of ways. Two common scalar metrics that are used to measure the information contained in a measurement are the Shannon information content [19] and the number of degrees of freedom present in the signal, both of which are related to the eigenvalues of data covariance matrices. For the following analyses, we assume that an observation of a random radiance vector R is related to some atmospheric state vector S through a forward model $\mathbf{m}(\cdot)$ as follows

$$R = \mathbf{m}(S) + \Psi = X + \Psi, \tag{3.1}$$

where Ψ is a random error vector, and X is the "noise-less" radiance observation. For the remainder of the chapter, we assume without loss of generality that all random vectors are mean-centered.

3.1.1 Shannon information content

The Shannon definition of information content arises from information theory and depends on the entropy of the underlying probability density functions (pdf's) that characterize the measurement. The entropy of a continuous pdf $P(r)$ can be defined as

$$H(P) = -\int P(R)\log\left[P(R)\right]\, dR. \tag{3.2}$$

The base of the logarithm is usually taken to be 2, in which case the units of entropy are bits, or e, in which case the units of entropy are nats. The *information content* of a measurement in the Shannon sense can be defined as the reduction of entropy upon making a measurement of R:

$$I(X,R) = H[P(X)] - H[P(X|R)], \tag{3.3}$$

or equivalently

$$I(X,R) = H[P(R)] - H[P(R|X)]. \tag{3.4}$$

For example, the entropy of R can be calculated assuming a multivariate Gaussian distribution

$$P(R) = \frac{1}{(2\pi)^{N/2}|\mathbf{C}_{RR}|^{1/2}} \exp\left\{-\frac{1}{2}R^T\mathbf{C}_{RR}^{-1}R\right\}, \tag{3.5}$$

where $\mathbf{C}_{RR}$ is the covariance of R:

$$\begin{aligned} H[P(R)] &= \sum_{i=1}^{N} \log(2\pi e\lambda_i)^{1/2} && (3.6)\\ &= N\log(2\pi e)^{1/2} + \frac{1}{2}\log\left(\prod_{i=1}^{N}\lambda_i\right) && (3.7)\\ &= N\log(2\pi e)^{1/2} + \frac{1}{2}\log|\mathbf{C}_{RR}| && (3.8)\\ &= c_1 + \frac{1}{2}\log|\mathbf{C}_{XX} + \mathbf{C}_{\Psi\Psi}|. && (3.9) \end{aligned}$$

The information content of a measurement of R can be calculated under the assumption of Gaussianity. The covariance of R before the measurement is $\mathbf{C}_{XX} + \mathbf{C}_{\Psi\Psi}$ while the covariance of R after the measurement is $\mathbf{C}_{\Psi\Psi}$. These values are used in Eq. 3.4:

$$\begin{aligned} I(X,R) &= H[P(R)] - H[P(R|X)] \\ &= \frac{1}{2}\log|\mathbf{C}_{XX} + \mathbf{C}_{\Psi\Psi}| - \frac{1}{2}\log|\mathbf{C}_{\Psi\Psi}| && (3.10)\\ &= \frac{1}{2}\log|\mathbf{C}_{\Psi\Psi}^{-1}(\mathbf{C}_{XX} + \mathbf{C}_{\Psi\Psi})| && (3.11)\\ &= \frac{1}{2}\log|\mathbf{C}_{\Psi\Psi}^{-1/2}\mathbf{C}_{XX}\mathbf{C}_{\Psi\Psi}^{-1/2} + \mathbf{I}| && (3.12)\\ &= \frac{1}{2}\log|\widetilde{\mathbf{C}}_{XX} + \mathbf{I}|, && (3.13) \end{aligned}$$

where

$$\widetilde{\mathbf{C}}_{XX} \triangleq \mathbf{C}_{\Psi\Psi}^{-1/2}\mathbf{C}_{XX}\mathbf{C}_{\Psi\Psi}^{-1/2} \tag{3.14}$$

is the *whitened* covariance matrix describing X. Note that Eq. 3.13 can be easily calculated from the eigenvalues of $\widetilde{\mathbf{C}}_{XX}$

$$I(X,R) = \frac{1}{2}\sum_i \log(1+\lambda_i). \tag{3.15}$$

The information content of simulated hyperspectral measurements for several channel configurations is shown in Fig. 3-1. The figure shows that the information content increases as the logarithm of the number of channels, which agrees with results reported in [20].

3.1.2 Degrees of freedom

Another measure of information contained within a measurement is the number of Degrees of Freedom (DOF), where a degree of freedom can be loosely defined as an independent component of R that contains some information about X and the uncertainty of this information is smaller than the measurement error of the component. For example, if R is prewhitened and projected onto the eigenvectors of $\widetilde{\mathbf{C}}_{XX}+\mathbf{I}$, the eigenvalues of $\widetilde{\mathbf{C}}_{XX}$ give the signal-to-noise ratio (SNR) of each uncorrelated component. It is intuitive that the number of degrees of freedom should be the number of components with SNR $\gtrsim 1$. Furthermore, it is convenient to make a distinction between a degree of freedom due to signal (DOF_s) and a degree of freedom due to noise (DOF_n). The previous description assumed implicitly that the degree of freedom was due to signal. If there are N elements of R, we require

$$\mathrm{DOF}_s + \mathrm{DOF}_n = N. \tag{3.16}$$

Rodgers [21] suggests the following definitions of DOF_s and DOF_n:

$$\begin{aligned} \mathrm{DOF}_s &\triangleq \mathrm{tr}(\widetilde{\mathbf{C}}_{XX}[\widetilde{\mathbf{C}}_{XX}+\mathbf{I}]^{-1}) \\ &= \sum_i \frac{\lambda_i}{(1+\lambda_i)} \end{aligned} \tag{3.17}$$

$$\begin{aligned} \mathrm{DOF}_n &\triangleq \mathrm{tr}([\widetilde{\mathbf{C}}_{XX}+\mathbf{I}]^{-1}) \\ &= \sum_i \frac{1}{(1+\lambda_i)}. \end{aligned} \tag{3.18}$$

These definitions do not necessarily yield integer values; for example, a component with SNR $= 1$ would contribute $\frac{1}{2}$ to DOF_s and $\frac{1}{2}$ to DOF_n. Equation 3.16 is satisfied, however. As a final comment, it is interesting to note the similarity between Eq. 3.4 and Eq. 3.17, both of which depend only on the eigenvalues of $\widetilde{\mathbf{C}}_{XX}$. The number of degrees of freedom for simulated hyperspectral measurements assuming several channel configurations is shown in Fig. 3-1.

3.2 Principal components analysis (PCA)

It is often useful to decompose a random vector (of atmospheric radiance intensity observations at N frequencies, for example)

$$R \triangleq \begin{bmatrix} R_{\nu_1} \\ R_{\nu_2} \\ \vdots \\ R_{\nu_N} \end{bmatrix} \tag{3.19}$$

into a vector $\mathcal{I}_r$ of r statistically independent components (where $1 \leq r \leq N$)

$$\mathcal{I}_r \triangleq \begin{bmatrix} \mathcal{I}_1 \\ \mathcal{I}_2 \\ \vdots \\ \mathcal{I}_r \end{bmatrix} \triangleq \begin{bmatrix} f_1(R) \\ f_2(R) \\ \vdots \\ f_r(R) \end{bmatrix} \triangleq \mathbf{f}_r(R), \tag{3.20}$$

where $\mathbf{f} : \mathbb{R}^N \rightarrow \mathbb{R}^r$ is a continuous (usually nonlinear) function. The radiance vector R may be reconstructed from the independent components $\mathcal{I}_r$ (possibly with some distortion) as follows:

$$\tilde{R}_r \triangleq \begin{bmatrix} g_1(\mathcal{I}_r) \\ g_2(\mathcal{I}_r) \\ \vdots \\ g_N(\mathcal{I}_r) \end{bmatrix} \triangleq \mathbf{g}_r(\mathcal{I}_r), \tag{3.21}$$

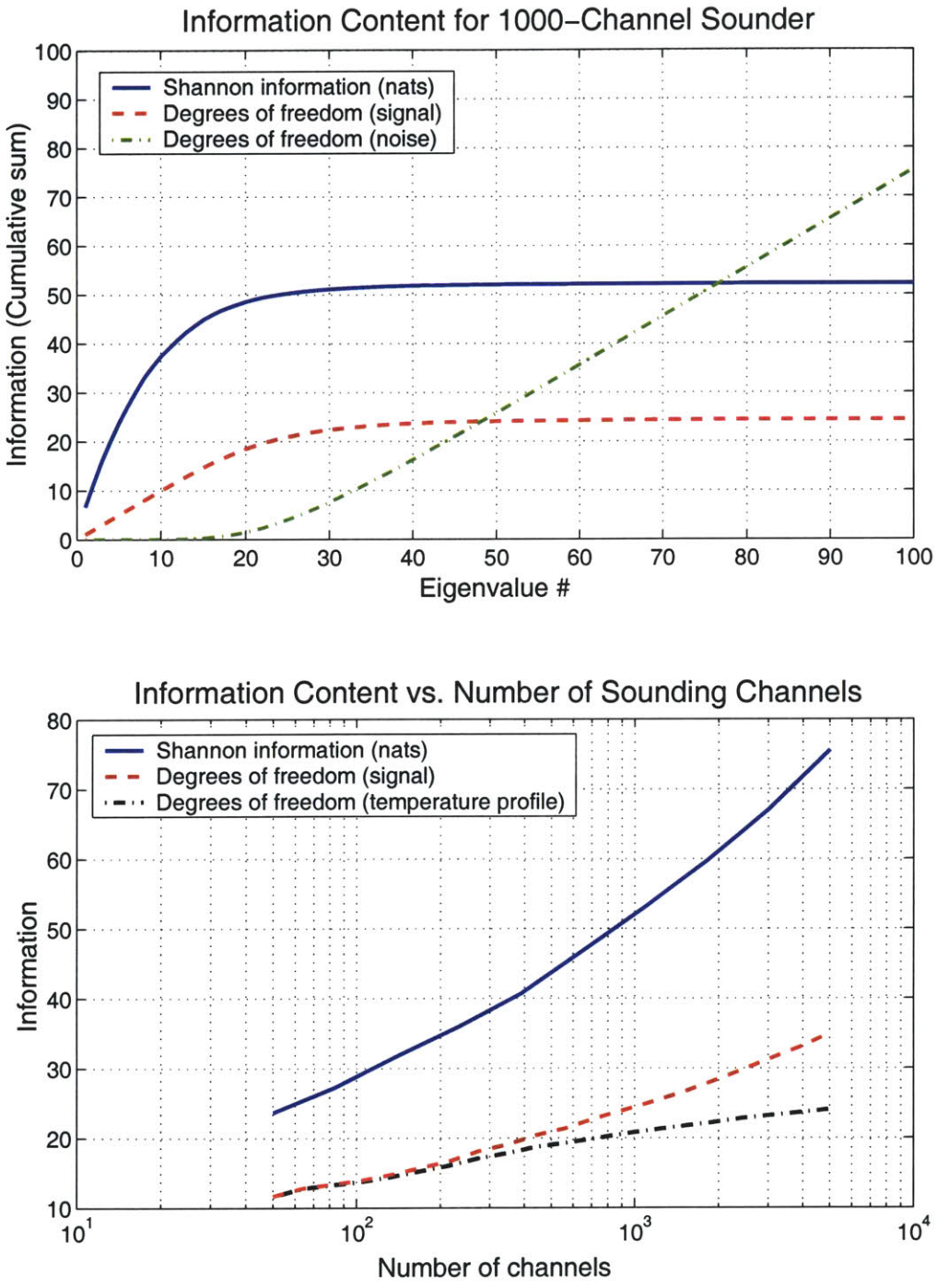

Figure 3-1: Information content analysis of a hypothetical hyperspectral IR (4 μm – 15 μm) sounder in clear air. The top graph shows the information content of a 1000-channel sounder using the Shannon and DOF metrics as a function of eigenvalue number. The bottom graph shows the relationship between information content and the number of sounding channels distributed from 4 μm – 15 μm. Also shown is the number of degrees of freedom in the signal due to the temperature profile (see Section 3.2.4).

where $\mathbf{g} : \mathbb{R}^r \rightarrow \mathbb{R}^N$ is a continuous (usually nonlinear) function. The vector-valued functions $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ are usually chosen to minimize some scalar-valued cost function

$$C(R - \tilde{R}_r) \tag{3.22}$$

over $1 \leq r \leq N$. Note that $\tilde{R}_r = R$ for $r = N$, and possibly for $r < N$ if the elements of R are statistically dependent. The functions $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ and the statistical moments of $\mathcal{I}_r$ provide a measure of the statistical structure of R.

If the cost function to be minimized is the expected value of the sum of the squares of the error of $R - \tilde{R}_r$, i.e.,

$$C(\cdot) = E\left[(R - \tilde{R}_r)^T (R - \tilde{R}_r)\right] \tag{3.23}$$

then the elements of $\mathcal{I}_r$ are called *principal components* of R.

3.2.1 Nonlinear PCA

Generally, $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ are nonlinear and cannot be found analytically. Several methods have been proposed for finding $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ using feedforward neural networks given an ensemble of observations of R [22, 23]. An autoassociative feedforward neural network was used in [24] to find $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ for remote sounding data.

3.2.2 Linear PCA

We now consider a special case where $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ and constrained to be linear functions:

$$\mathbf{f}_r(R) = \mathbf{F}R \tag{3.24a}$$

$$\mathbf{g}_r(\mathcal{I}_r) = \mathbf{G}\mathcal{I}_r, \tag{3.24b}$$

where $\mathbf{F}$ is an $r \times N$ matrix and $\mathbf{G}$ is an $N \times r$ matrix ($r \leq N$). Equation 3.21 becomes

$$\tilde{R}_r = \mathbf{GF}R \tag{3.25}$$

and the minimization to be carried out in terms of the cost function given by Eq. 3.23 is

$$\{\mathbf{F}, \mathbf{G}\} = \underset{\mathbf{F},\mathbf{G}}{\arg\min}\ E\left[(R - \mathbf{GF}R)^T(R - \mathbf{GF}R)\right]. \tag{3.26}$$

We begin by assuming that $\mathbf{G}$ is orthonormal. If this is not the case, a QR decomposition can be performed on $\mathbf{G}$ (its columns are linearly independent) and the non-orthonormal part can be included in $\mathbf{F}$. The joint minimization posed in Eq. 3.26 is separable, and $\mathbf{F}$ can be determined for a fixed choice of $\mathbf{G}$, and vice versa. For a fixed $\mathbf{G}$, R can be decomposed into two orthogonal components:

$$\begin{aligned} R &= (\mathbf{I} - \mathbf{GG}^T)R + \mathbf{GG}^T R \\ &= R_\perp + R_\parallel \end{aligned} \tag{3.27}$$

where $\mathbf{I}$ is the identity matrix. Replacing R in Eq 3.26 with 3.28 yields

$$\begin{aligned} &E\left[\left(R_\perp + R_\parallel - \mathbf{GF}R\right)^T\left(R_\perp + R_\parallel - \mathbf{GF}R\right)\right] = \\ &E\left[R_\perp^T R_\perp\right] + E\left[\left(R_\parallel - \mathbf{GF}R\right)^T\left(R_\parallel - \mathbf{GF}R\right)\right], \end{aligned} \tag{3.28}$$

where we use the fact that $R_\perp$ is orthogonal to $R_\parallel - \mathbf{GF}R$:

$$\begin{aligned} E\left[R^T(\mathbf{I} - \mathbf{GG}^T)(\mathbf{GG}^T R - \mathbf{GF}R)\right] &= \\ E\left[R^T\mathbf{GG}^T R - R^T\mathbf{GG}^T R - R^T\mathbf{GF}R + R^T\mathbf{GF}R\right] &= 0. \end{aligned} \tag{3.29}$$

For a given $\mathbf{G}$, $E\left[R_\perp^T R_\perp\right]$ does not depend on $\mathbf{F}$, and Eq. 3.26 reduces to

$$\mathbf{F} = \underset{\mathbf{F}}{\arg\min}\ E\left[\left(\mathbf{GG}^T R - \mathbf{GF}R\right)^T\left(\mathbf{GG}^T R - \mathbf{GF}R\right)\right]. \tag{3.30}$$

Upon inspection of Eq. 3.30 it is obvious that the choice of $\mathbf{F}$ that minimizes the cost function is

$$\mathbf{F} = \mathbf{G}^T. \tag{3.31}$$

Substituting $\mathbf{F} = \mathbf{G}^T$ into Eq. 3.26 we obtain

$$\mathbf{G} = \underset{\mathbf{G}}{\arg\min}\ E\left[\left(R - \mathbf{G}\mathbf{G}^T R\right)^T \left(R - \mathbf{G}\mathbf{G}^T R\right)\right], \tag{3.32}$$

where the cost function can be simplified as follows:

$$\begin{aligned} E\left[\left(R - \mathbf{G}\mathbf{G}^T R\right)^T \left(R - \mathbf{G}\mathbf{G}^T R\right)\right] &= E\left[R^T R\right] - E\left[R^T \mathbf{G}\mathbf{G}^T R\right] && (3.33) \\ &= \mathrm{tr}\left(\mathbf{C}_{RR}\right) - \mathrm{tr}\left(\mathbf{G}^T \mathbf{C}_{RR} \mathbf{G}\right) && (3.34) \end{aligned}$$

where $\mathbf{C}_{RR}$ is the data covariance matrix $E[RR^T]$. Note that the first term in Eq. 3.34 does not depend on $\mathbf{G}$, and the minimization in Eq. 3.32 is equivalent to the following maximization

$$\mathbf{G} = \underset{\mathbf{G}}{\arg\max}\ \mathrm{tr}\left(\mathbf{G}^T \mathbf{C}_{RR} \mathbf{G}\right) = \underset{\mathbf{G}}{\arg\max} \sum_{i=1}^{r} G_i^T \mathbf{C}_{RR} G_i, \tag{3.35}$$

where G_i is the i^{th} column of $\mathbf{G}$. The maximization carried out in Eq. 3.35 is the well-known quadratic maximization with unit-length constraint problem, which is solved by choosing the G_i's to be the r eigenvectors of $\mathbf{C}_{RR}$ with the r largest corresponding eigenvalues

$$\mathbf{G} = [Q_1 \,|\, Q_2 \,|\, \cdots \,|\, Q_r]. \tag{3.36}$$

3.2.3 Principal components transforms

In light of the previous discussion, the *principal components transform* (PCT) is defined as the multiplication of R by the transpose of the eigenvectors of $\mathbf{C}_{RR}$ (ordered by

eigenvalue)

$$P \triangleq \mathbf{Q}^T R, \tag{3.37}$$

where the elements of vector P are the principal components of R. The plot of eigenvalue versus eigenvalue number is known as the *screeplot.*

It is often advantageous to normalize R before computing the PCT. We briefly consider three examples.

Noise-Adjusted PCT

Prewhitening the radiance measurements before application of the PCT results in uncorrelated components with maximal signal-to-noise ratio [25]. This transform is widely-used for dimensionality reduction in many hyperspectral imaging and sounding applications (see [26] and [3] for example).

Normalized PCT

An alternative to the NAPCT is the NPCT, where each element of R is normalized by its standard deviation. The noise-adjusted PCT (NAPCT) is then applied to the normalized R. This transform is often used if the noise statistics are unknown.

Blind NAPCT

If the statistics of the noise are unknown, it may be possible to estimate them, and subsequently apply the NAPCT. This approach is called *blind* processing, and attempts to extract properties of $\mathbf{F}(S)$ and Ψ (from Eq. 3.1), usually under the assumption that $\mathbf{F}(S)$ is a linear "mixing matrix" and Ψ is a Gaussian random vector with uncorrelated elements. One example of a blind NAPCT is the Blind noise-Adjusted Principal Components Transform (BAPCT) [27], which uses the Iterated Order-Noise (ION) algorithm [1] to estimate the noise statistics.

3.2.4 Projected PC transform

Cost functions other than Eq. 3.23 are sometimes used in practice. For example, suppose we wanted to find the $r \times N$ transform matrix $\mathbf{Q}_r^T$ which gives components that, when regressed against S, minimize the resulting sum-squared error for any ordered subset of components P_r given by

$$P_r \triangleq \mathbf{Q}_r^T R. \tag{3.38}$$

The estimate of S in terms of P_r is

$$\begin{aligned} \widehat{S}_r &= \mathbf{C}_{SP_r}\mathbf{C}_{P_rP_r}^{-1}P_r & (3.39)\\ &= \mathbf{C}_{SR}\mathbf{Q}_r[\mathbf{Q}_r^T\mathbf{C}_{RR}\mathbf{Q}_r]^{-1}\mathbf{Q}_r^T R & (3.40) \end{aligned}$$

and the cost function to be minimized is given by

$$C(\cdot) = E\left[(S-\widehat{S}_r)^T(S-\widehat{S}_r)\right], \tag{3.41}$$

where $\mathbf{C}_{SR}$ is the cross-covariance of S and R. It can be shown using a derivation similar to that in Section 3.2.2 that the $\mathbf{Q}_r$'s that minimize Eq. 3.41 are the r right eigenvectors with highest singular value of the reduced-rank regression matrix $\mathbf{L}_r$:

$$\mathbf{L}_r = \mathbf{V}_r\mathbf{V}_r^T\mathbf{C}_{SR}\mathbf{C}_{RR}^{-1}, \tag{3.42}$$

where

$$\mathbf{V}_r = [V_1 \,|\, V_2 \,|\, \cdots \,|\, V_r]. \tag{3.43}$$

are the r most-significant eigenvectors of $\mathbf{C}_{SR}\mathbf{C}_{RR}^{-1}\mathbf{C}_{SR}^T$. This transform is sometimes called the Projected PCT (PPCT) because the measurements R are projected into an r-dimensional subspace of S spanned by $\mathbf{V}_r$. Note, however, that the elements of P_r are not uncorrelated, which is a basic requirement of any PCT.

Figure 3-2 compares the performance of the PC, NAPC, and PPC transforms for three specific decompositions/reconstructions of a simulated 1000-channel radiance vector:

1. Noisy radiance:

$$\widehat{R} = \mathbf{Q}_r \mathbf{Q}_r^T R, \tag{3.44}$$

2. Signal portion of noisy radiance:

$$\widehat{X} = \mathbf{Q}_r \mathbf{Q}_r^T R, \tag{3.45}$$

3. Temperature profile retrieval:

$$\widehat{T} = \mathbf{C}_{TR} \mathbf{Q}_r \left[\mathbf{Q}_r^T \mathbf{C}_{RR} \mathbf{Q}_r \right]^{-1} \mathbf{Q}_r^T R. \tag{3.46}$$

3.3 Estimation of geophysical parameters

In this section, we address the problem of estimating the state vector S given an observation of R (the retrieval or inverse problem), where we use $\widehat{S}(R)$ to denote the estimate of S given an observation of R. There are a number of ways to proceed. The model-based (or physical) approaches use knowledge of $\mathbf{F}(\cdot)$ to find an S which is consistent with an observation of R. This is usually an ill-posed problem with no mathematically unique solution. The statistical description of R and S, if available, can be used to choose the "best" solution from all the possible ones consistent with the model.

3.3.1 The Bayesian approach

The Bayesian approach to estimation involves the incorporation of *a priori* knowledge about the state vector S with knowledge gained by measuring R. Mathematically, this knowledge is formulated in terms of five related probability density functions (pdf's):

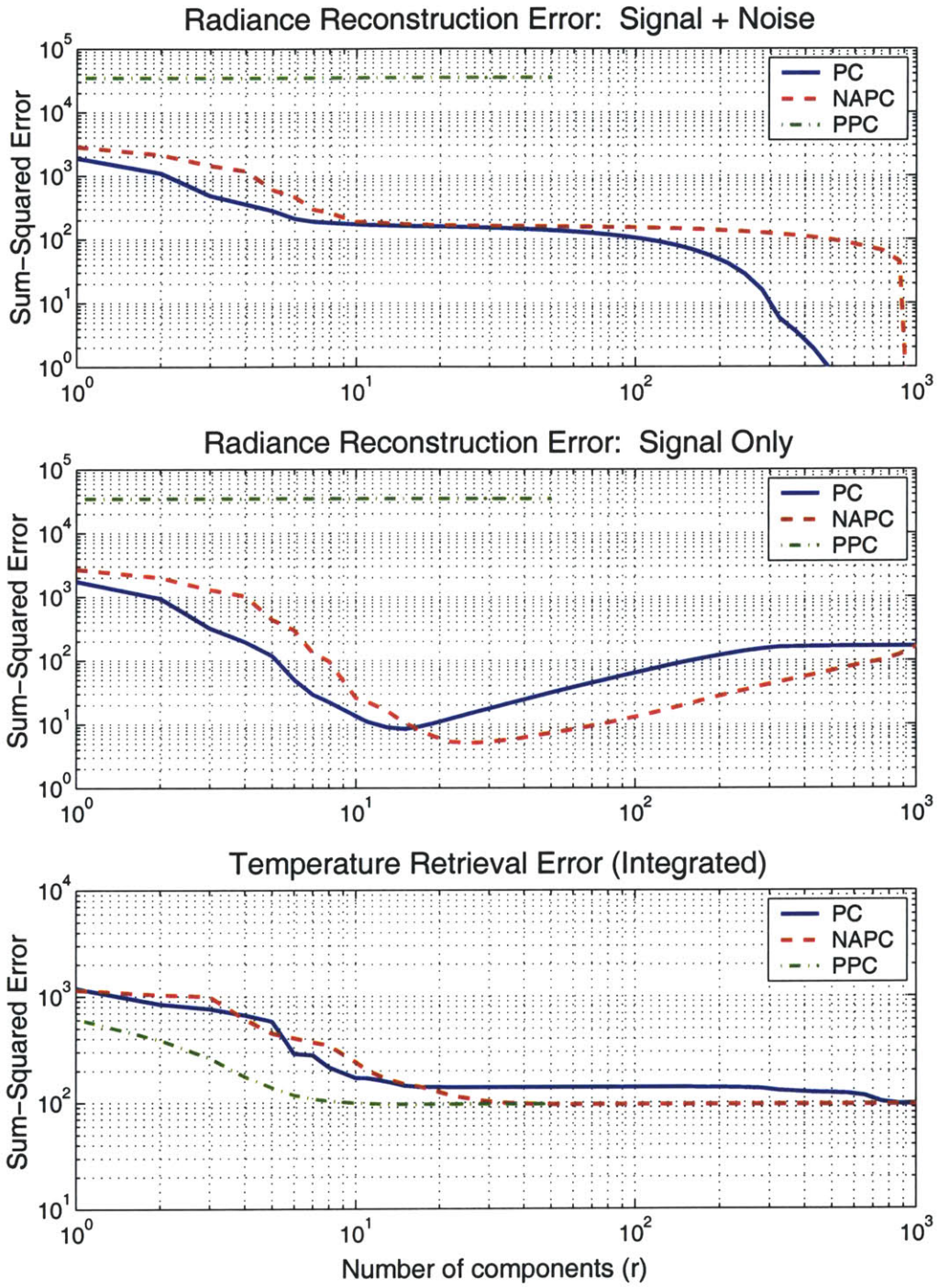

Figure 3-2: Performance comparisons of the PC, NAPC, and PPC transforms for a hypothetical 1000-channel infrared (4 μm – 15 μm) sounder. The first plot shows the distortion introduced by representing a noisy radiance vector with r components. The second plot shows the distortion of the signal portion of the radiance. The third plot shows the integrated sum-squared error of the temperature profile estimated using r components.

$P(S)$	The prior (i.e., before the measurement) pdf of state S
$P(R)$	The prior pdf of the measurement R
$P(R,S)$	The joint prior pdf of R and S
$P(R\|S)$	The conditional pdf of R given state S
$P(S\|R)$	The conditional pdf of S after measurement R. This is the quantity of interest for the solution of the estimation problem.

Bayes' theorem relates the conditional probabilities as follows:

$$P(S|R) = \frac{P(R|S)P(S)}{P(R)}. \tag{3.47}$$

Therefore, the Bayesian framework allows probabilities to be assigned to possible choices of $\widehat{S}(R)$ given knowledge of the joint and conditional probabilities of R and S. A reasonable choice for $\widehat{S}(R)$ is the value of $P(S|R)$ which has the highest probability (known as the Maximum A Posteriori (MAP) estimator).

Bayes' least-squares estimator

An alternative to the MAP estimator is the estimator that minimizes some suitable cost criterion:

$$\widehat{S}(\cdot) = \underset{\mathbf{f}(\cdot)}{\arg\min}\ C(S, \mathbf{f}(R)). \tag{3.48}$$

The mean-squared error (MSE) cost criterion is commonly chosen for this purpose and results in the following estimator, sometimes called the Bayes Least-Squares (BLS) estimator:

$$\widehat{S}(R) = E[S|R]. \tag{3.49}$$

Linear least-squares estimator

The BLS estimator has two practical disadvantages: it is often a nonlinear function of R, and it requires a complete statistical representation of the relationship between R

and S, which is rarely available. If we constrain the estimator in Eq. 3.48 to be linear, the resulting estimator depends only on a second-order characterization of the statistical relationship between R and S. This estimator is the linear least-squares estimator (LLSE):

$$\widehat{S}(R) = \mathbf{C}_{SR}\mathbf{C}_{RR}^{-1}R \tag{3.50}$$

with error covariance

$$\mathbf{C}_{\epsilon\epsilon} = \mathbf{C}_{SS} - \mathbf{C}_{SR}\mathbf{C}_{RR}^{-1}\mathbf{C}_{SR}^{T}. \tag{3.51}$$

The LLS estimator equals the BLS estimator when R and S are jointly Gaussian.

3.3.2 Bayesian alternatives

The previous estimators are based entirely on the statistical relationship between R and S. Other methods use knowledge of the forward model $\mathbf{F}(\cdot)$ together with whatever limited statistical characterization of R and S is available. One example is the minimum-information retrieval, which only depends on a "nominal" state S_0.

Minimum-information retrieval

The minimum-information retrieval picks the $\widehat{S}$ which is "closest" in the least-squares sense to S_0 and satisfies

$$\left(R - \mathbf{F}(\widehat{S})\right)^{T}\left(R - \mathbf{F}(\widehat{S})\right) = M\sigma^2, \tag{3.52}$$

where $M\sigma^2$ is a scalar quantity related to the measurement error. If we linearize about R_0, S_0 and replace $\mathbf{F}(\cdot)$ with a matrix, $\mathbf{K}$, then the minimum-information solution in the presence of noise with covariance $\mathbf{C}_{\Psi\Psi}$ is

$$\widehat{S}(R) = \mathbf{K}^{T}(\mathbf{K}\mathbf{K}^{T} + \beta\mathbf{C}_{\Psi\Psi})^{-1}R, \tag{3.53}$$

where β is some constant (in units of $1/\mathrm{T}^2$). No statistical relationships between S and R are used in the minimum-information retrieval. As a final note, if the covariance matrix of S is known and a linear model $R = \mathbf{K}S + \Psi$ is assumed, Eq. 3.53 can be modified to include the *a priori* information about S as follows:

$$\widehat{S}(R) = \mathbf{C}_{SS}\mathbf{K}^T(\mathbf{K}\mathbf{C}_{SS}\mathbf{K}^T + \mathbf{C}_{\Psi\Psi})^{-1}R. \tag{3.54}$$

If a second-order statistical characterization of R and S is available, Eq. 3.54 can be updated with this information in the form of Eq. 3.50

Figure 3-3 shows the performance of three operators used to retrieve the temperature profile from simulated AMSU radiances in clear-air: minimum-information (Eq. 3.53), linear model with known $\mathbf{C}_{SS}$ (Eq. 3.54), and the LLS estimator (Eq. 3.50). The graph demonstrates the significant impact of *a priori* statistics on the retrieval performance.

Maximum-resolution retrieval

The preceding retrieval techniques have all minimized some form of the mean-square error cost function. Another quantity that could be minimized is the *resolution* of the retrieval. Assuming a forward model and a linear retrieval ($\mathbf{D}$), the estimate of the state vector can be written as

$$\widehat{S} = \mathbf{D}R = \mathbf{D}\mathbf{K}(S + \Psi) \triangleq \mathbf{A}S + \mathbf{D}\Psi, \tag{3.55}$$

where the following error contributions are noted: $\mathbf{A} = \mathbf{D}\mathbf{K}$ is a measure of the way the observing system smoothes the profile and $\mathbf{D}\Psi$ is the "retrieval" error.

It is possible to develop a retrieval that optimizes resolution (i.e., minimizes the effects of smoothing). Backus and Gilbert [28] introduced the following "spread" function that can be minimized subject to a unit-area constraint:

$$s(z) = \sum_{ij} D_i(z)Q_{ij}(z)D_j(z), \tag{3.56}$$

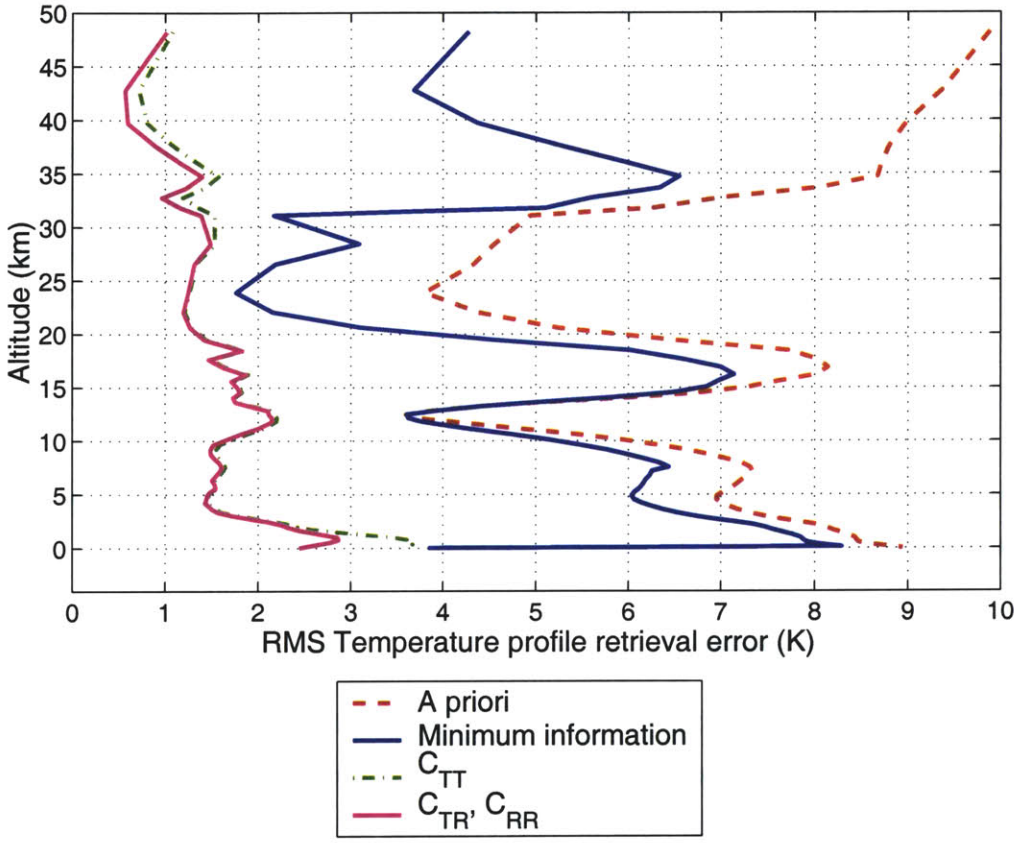

Figure 3-3: Comparison of temperature retrieval techniques. The minimum information retrieval (with $\beta = 1\mathrm{K}^{-2}$) only uses information contained in the weighting function matrix. The optimal linear model retrieval uses the weighting function matrix and $\mathbf{C}_{TT}$ only. The direct multiple regression retrieval uses statistical characterizations of $\mathbf{C}_{RR}$ and $\mathbf{C}_{TT}$.

where $Q_{ij}(z)$ depends only on the weighting functions:

$$Q_{ij}(z) = 12 \int (z - z')^2 W_i(z') W_j(z')\, dz'. \tag{3.57}$$

Given $Q_{ij}(z)$, the solution is

$$\mathbf{g}(z) = \frac{\mathbf{Q}^{-1}(z)B}{B^T \mathbf{Q}^{-1}(z)B}, \tag{3.58}$$

where the vector-valued function $\mathbf{g}(z)$ has elements $G_j(z)$, the matrix-valued function $\mathbf{Q}(z)$ has elements $Q_{ij}(z)$, and the vector B contains the areas of the weighting functions. Because no constraint has been imposed on the noise, it is likely to be large, especially if the weighting functions have significant overlap. It is possible to minimize a weighted sum of spread and noise variance, which results in the following solution

$$\mathbf{g}(z) = \frac{(\mathbf{Q} + \mu \mathbf{C}_{\Psi\Psi})^{-1}(z)B}{B^T (\mathbf{Q} + \mu \mathbf{C}_{\Psi\Psi})^{-1}(z)B}, \tag{3.59}$$

where μ is a tradeoff parameter and $\mathbf{C}_{\Psi\Psi}$ is the noise covariance matrix. The tradeoff between resolution and noise is endemic to every retrieval problem.

3.3.3 Neural networks

The artificial neural network, or simply, "neural net", has gained widespread popularity as a universal function approximator [29, 30]. The neural net is able to find input–output relationships directly from the data without requiring underlying assumptions about the distribution of the data. Furthermore, a neural net with only a single hidden layer having non-linear activation functions is capable of approximating any real-valued continuous scalar function [31]. This section introduces the basic features of neural networks and the considerations involved in designing and training a network.

Artificial Neurons

A neural network is an interconnection of simple computational elements. The computational elements, or nodes, in neural networks perform simple calculations that are typically

non-linear and analog. The output of each node is computed in a two step process. A weighted sum of the node inputs is first computed:

$$y_1 = \mathbf{W}^T\mathbf{x} \tag{3.60}$$

where $\mathbf{W} = [w_0\ w_1 \cdots w_N]^T$ is a vector of weights and $\mathbf{x} = [1\ x_1 \cdots x_N]^T$ is the vector of N input variables augmented with a one in the first entry of the vector to provide a bias. This value is then passed through a non-linear function known as the activation function of the node:

$$y = f(y_1) = f(\mathbf{W}^T\mathbf{x}) \tag{3.61}$$

The activation function is generally chosen to be monotonically increasing and differentiable to facilitate the training process; sigmoids of the form $y = \tanh(\cdot)$ are often chosen for this purpose..

Neural network training by backpropagation

The process of determining the weights of a network is called training, and generally involves minimization of the mean-squared error between the target output and the network output. Backpropagation is a common training method [32], and operates in two phases. First, the input is presented and propagated through the network to determine the output value for each node. Second, a backward pass is taken through the network to propagate the error signal back to each node and update the weights. A gradient descent algorithm is used to calculate the perturbations in the weight vectors, therefore the activation function must be differentiable. Training is continued on the network until the mean-squared error on a validation set of data (a set of data different from that used to train the network) reaches a minimum.

3.4 Summary

Information content analysis techniques including the Shannon information and degrees of freedom provide a useful characterization of the geophysical signal present in hyperspectral data. Principal components transforms can be used to represent radiance measurements in a statistically-compact form. Bayesian estimation techniques, including LLSE, offer a convenient tool for geophysical parameter estimation. Neural networks are a nonlinear alternative to LLSE estimators.

Chapter 4

NAST-M: Instrument Description and Calibration

THE National Polar-orbiting Operational Environmental Satellite System (NPOESS) Aircraft Sounder Testbed, or NAST, has recently been developed and deployed on the NASA ER-2 high-altitude aircraft. The testbed consists of two collocated cross-track scanning instruments: a Fourier-transform interferometer spectrometer (NAST-I) [33] with spectral coverage of 3.6-16.1 μm, and a passive microwave spectrometer (NAST-M) [34, 35, 36] with 17 channels near the oxygen absorption lines at 50-57 GHz and 118.75 GHz. The testbed provides the first co-registered imagery from high-resolution microwave and infrared sounders, and will provide new data that will help: 1) validate meteorological satellite environmental data record (EDR) feasibility, 2) define future satellite instrument specifications, and 3) demonstrate operational issues in ground validation, data calibration and retrievals of meteorological parameters.

In this chapter, the NAST-M instrument and calibration procedure are described. Data from NAST-M are also compared with data from a coincident NOAA-15 AMSU overpass which occurred on March 26, 1999 (00:41:30 UTC, 47.1390°N, -86.9670°W) during WINTEX (WINTer EXperiment, Wisconsin, March/April 1999).

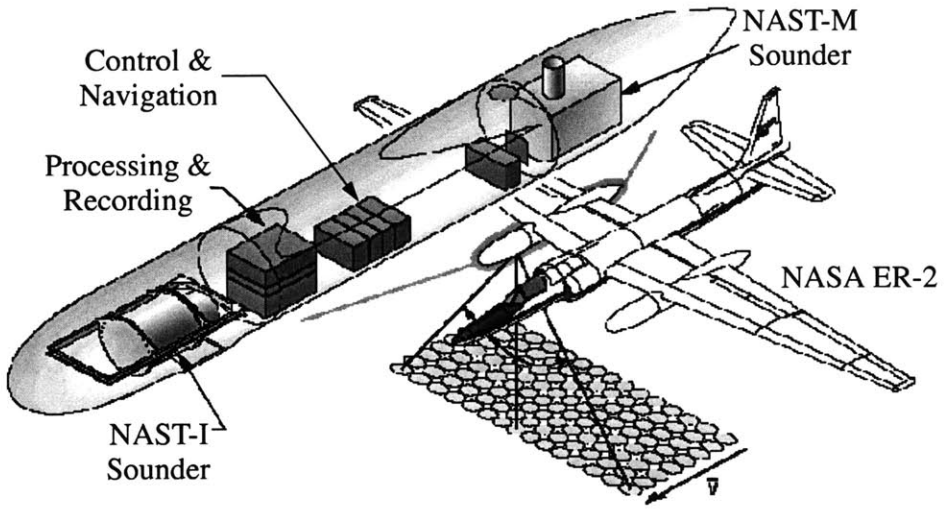

Figure 4-1: NAST ER-2 configuration.

4.1 Instrument Overview

The NAST-M instrument consists of two independent total-power radiometer systems that share a cross-track scanning reflector. The first radiometer (henceforth referred to as the "54-GHz radiometer") is a single-sideband system with eight channels from 50.3 GHz to 56.02 GHz, and the second radiometer (henceforth referred to as the "118-GHz radiometer") is a double-sideband system with nine channels from 118 ± 0.120 GHz to 118 ± 3.5 GHz. Both radiometers measure a single linear polarization; the electric field is oriented along-track at nadir and rotates with the scan angle. The package typically flies unpressurized at a nominal ER-2 cruising altitude of approximately 20 km at a speed of ~200 m/sec. The NAST configuration, as flown in the NASA ER-2, is shown in Figure 4-1. The NAST-M instrument, complete with all supporting flight hardware, weighs approximately 225 lbs., occupies a volume of approximately 15 ft^3, and consumes approximately 1.5 kW @ 120 VAC (1.2 kW of which is for heaters).

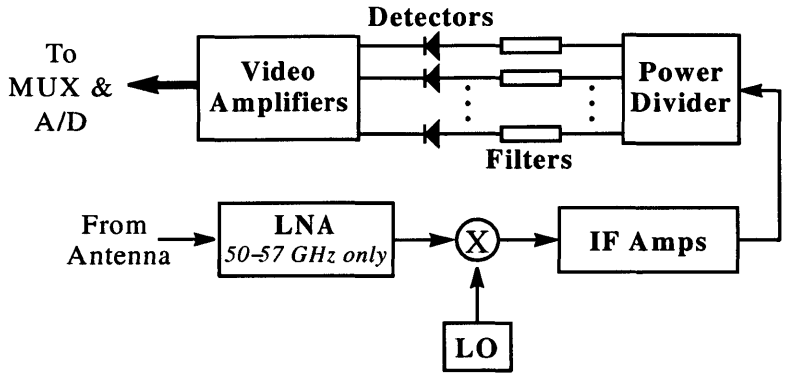

Figure 4-2: Radiometer block diagram.

4.1.1 Radiometer systems

A block diagram of the 54-GHz radiometer system is shown in Figure 4-2. The 118-GHz system is conceptually identical, except for the absence of the low-noise RF amplifier. Both systems utilize superheterodyne receivers; the local oscillator (LO) frequencies are 46 GHz and 118.75 GHz. Both LO's are temperature-controlled using thermo-electric devices to prevent frequency drift, and all amplifiers are temperature-controlled to prevent gain drift. The gain as a function of input power of the IF amplifiers, video amplifiers, square-law detectors, and A/D converters was measured [37] and found to be linear to within instrument thermal noise. The system noise temperatures of both systems are shown in Figure 4-3 The calibration and noise performance of the three channels closest to the 118.75-GHz line is degraded by LO power which is reflected off the calibration targets and re-enters the antenna feed, stimulating instabilities involving the preamplifiers. These instabilities are partially remedied by means of a small capacitor placed in series with the preamplifier input. The IF band of each radiometer is divided into channels using multi-section Chebyshev filters: four-section cavity filters are used in the 54-GHz system, and five-section lumped-element filters are used in the 118.75-GHz system. Channel specifications for the 54-GHz and 118-GHz radiometers are given in Tables 4.1 and 4.2. Additional similar radiometer systems operating near the 183-GHz water vapor line and the 425-GHz oxygen line are presently being added to the NAST-M package.

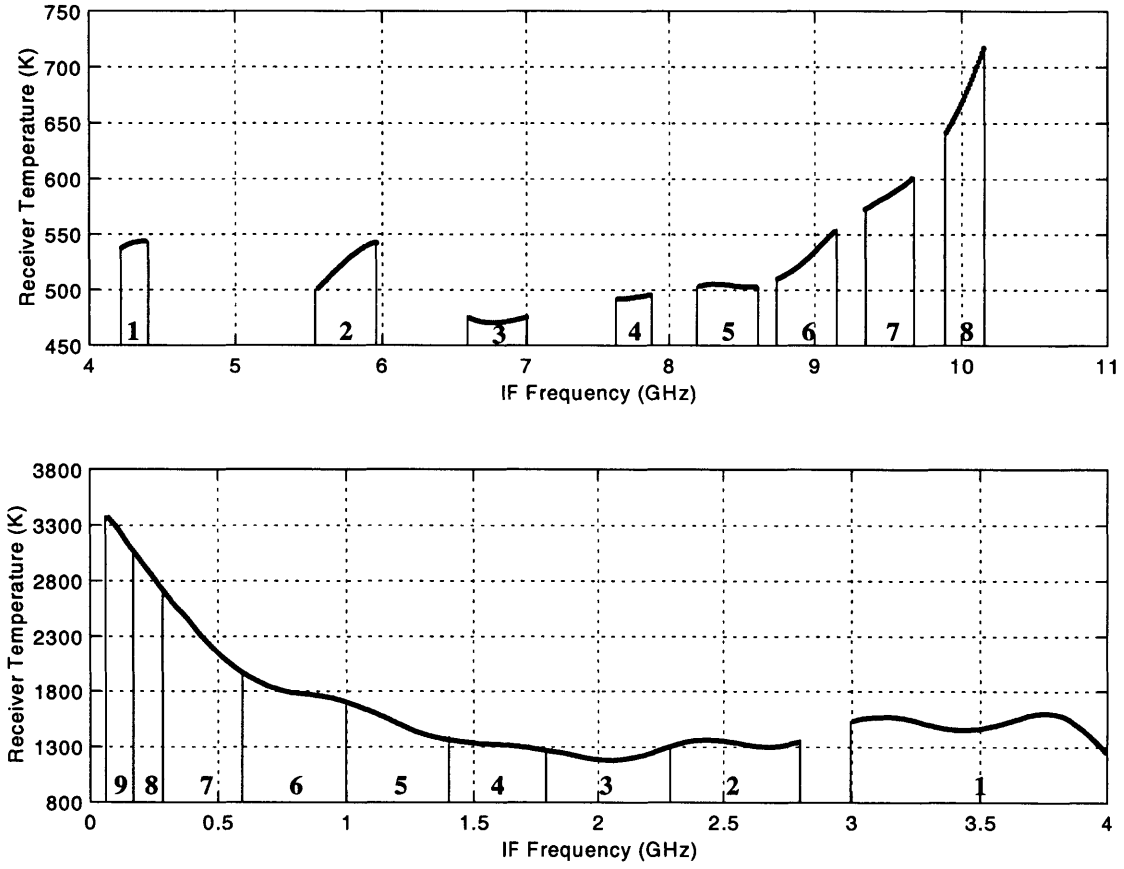

Figure 4-3: NAST-M receiver temperatures: 50-57 GHz (top), 118.75-GHz (bottom).

Table 4.1: Channel specifications for the 54-GHz radiometer. The nominal local oscillator frequency is 46 GHz. The brightness temperature offset from AMSU (at nadir) was calculated from a coincident overpass on March 26, 1999 (see Section 4.3, Fig. 4-8).

No.	Frequency (GHz)	Bandwidth (MHz)	Sensitivity (RMS K)	Offset from AMSU (K)
1	50.30	180	0.21	0.1
2	51.76	400	0.13	0.3
3	52.80	400	0.12	0.4
4	53.75	240	0.16	0.5
5	54.40	400	0.13	-0.5
6	54.94	400	0.15	-0.4
7	55.50	330	0.18	-0.2
8	56.02	270	0.18	-0.1

Table 4.2: Channel specifications for the 118.75-GHz radiometer. The nominal local oscillator frequency is 118.75 GHz The brightness temperature offset from AMSU (at nadir) was calculated from a coincident overpass on March 26, 1999 (see Section 4.3, Fig. 4-9).

No.	Frequency Offset (MHz)	Bandwidth (MHz)	Sensitivity (RMS K)	Offset from AMSU (K)
1	± 3500	1000	0.19	0.7
2	± 2550	500	0.23	0.9
3	± 2050	500	0.21	0.7
4	± 1600	400	0.25	0.6
5	± 1200	400	0.28	0.7
6	± 800	400	0.34	0.9
7	± 450	300	0.45	>2
8	± 235	130	0.90	>5
9	± 120	100	1.17	>10

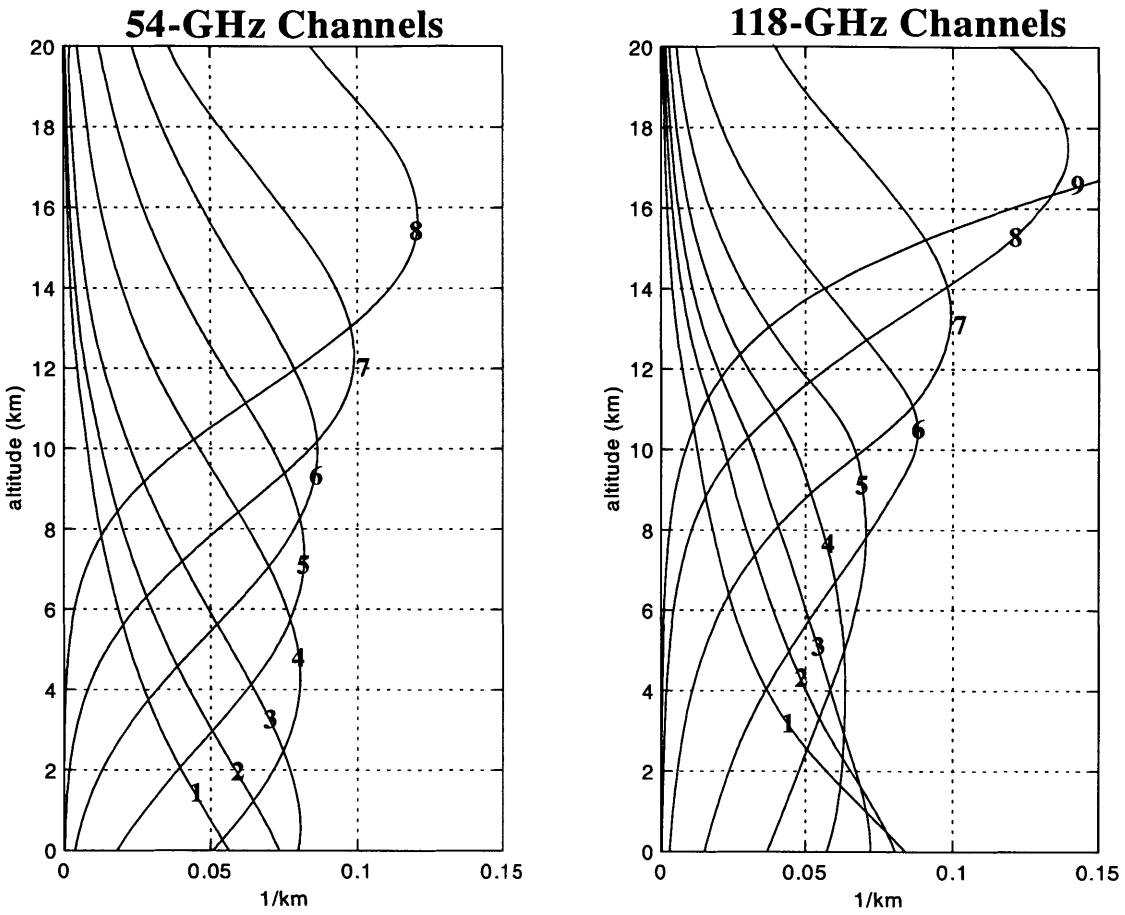

Figure 4-4: NAST-M clear-air temperature weighting functions (downward-looking). The US 1976 Standard Atmosphere over a black surface was assumed for the calculations.

Values for channel sensitivities include errors due to noisy calibration measurements. Brightness temperature offsets based on a March 26, 1999 AMSU overpass are also shown (see Section 4.3). Temperature weighting functions for the channels of both radiometers are shown in Figure 4-4.

4.1.2 Field of view

The NAST-M scanning sub-assembly is shown in Figure 4-5. The 3-dB (full-width at half-max) beamwidth for both antenna beams is 7.5° (2.6-km nadir footprint diameter at an altitude of 20 km). The downward-looking port of NAST-M allows an unobstructed view

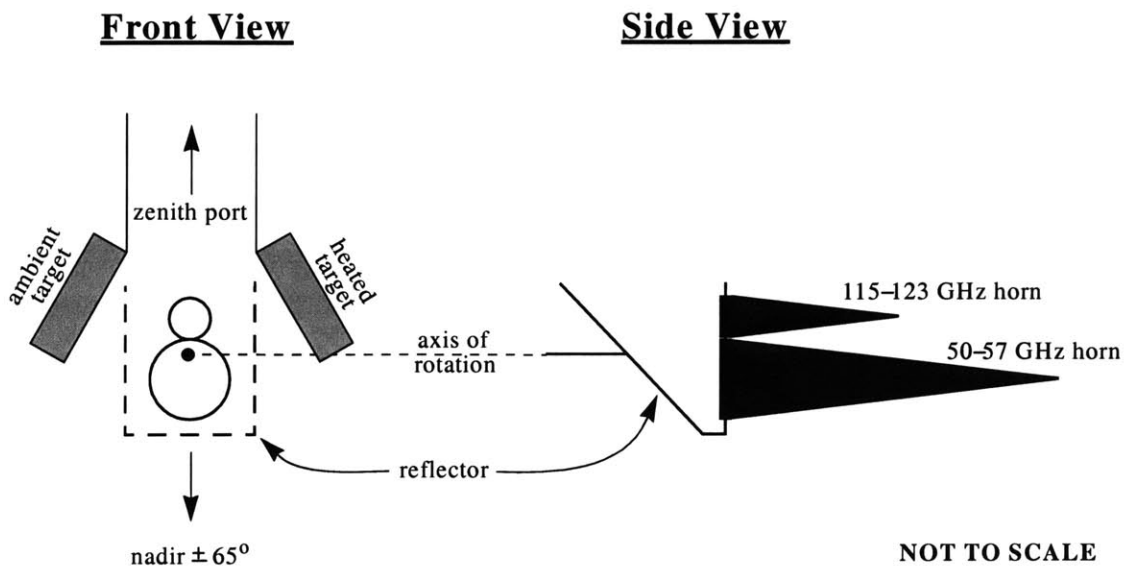

Figure 4-5: NAST-M scanning assembly.

for nadir ±65°, which yields a cross-track swath width of approximately 100 km from an altitude of 20 km. The reflector is stepped through a full rotation approximately every 5.5 seconds. A single scan consists of 19 spots across nadir ±65°, and three calibration spots: a heated internal blackbody, an ambient internal blackbody, and a zenith view through a port in the top of the instrument of the cosmic background. The nominal integration time for all spots (including calibration spots) is 100 msec.

4.1.3 Internal calibration targets[1]

Two blackbody calibration targets (20cm × 20cm × 4cm) were fabricated from aluminum and iron-loaded epoxy. Both loads have surfaces covered with tessellated pyramids machined from Emerson-Cuming CR-112 Eccosorb that are 12-mm tall with square bases with 8-mm sides. The aluminum backing extends up into the cores of the pyramids to minimize temperature gradients between pyramid bases and tips. Rectangular channels with square edges 4-mm wide and 3-mm deep were cut into the aluminum, thereby increasing

[1]**See Appendix A for a more detailed description of the construction and characterization of the internal calibration targets.**

the volume of absorbing material at the base of the pyramids and increasing return loss with negligible change in thermal conductivity. Thin-film platinum resistive temperature device (RTD) sensors were placed on the surface of the Eccosorb, embedded in the Eccosorb, and epoxied to the back of the aluminum. The temperature sensors were calibrated[2] to an accuracy of ± 0.05 K. The loads are insulated on the front with a 1-cm thick layer of Styrofoam and on the back and sides with extruded polystyrene. Time-domain reflectometry measurements at 75-110 GHz yield average return losses exceeding 30 dB (emissivity of 0.999). The thermodynamic temperature of the heated target is maintained at an average temperature of 334 ± 0.1 K and the thermodynamic temperature of the ambient target is typically 245 ± 5 K at altitude. While the thermodynamic temperature of the cosmic background is 2.736 ± 0.02 K, the measured brightness temperature can range from 2.9 ± 0.05 K for the most-transparent channel to 150 ± 5 K for the least transparent channel, depending on aircraft altitude.

4.1.4 Control and data handling

Instrument control and data collection tasks are coordinated by a microcomputer with an AMD 5x86 processor incorporating a flash-RAM hard drive and PC104 A/D cards. The Real-Time Linux [38] operating system was used. Built-in TCP/IP support allows real-time instrument control (via satellite uplink on the ER-2) and post-flight data download via Ethernet.

4.1.5 Digital video system

A wide-angle (111°), high-resolution (640x480 pixels per frame) video camera (Panasonic GP-KS162 with GP-LM3TA lens) is flown with the microwave package to provide continuous imagery of clouds and surface conditions. The video output is digitized (24-bit RGB) by a frame grabber board (one frame every five seconds), compressed (to ~56 kbps), and stored on a flash-RAM hard drive (~200 MB per 8-hour flight).

[2]See Appendix B for more details.

4.2 Instrument calibration

The radiometer output voltage C for each channel is converted into units of brightness temperature by the application of the linear calibration equation:

$$B(C) = gC + b = \begin{bmatrix} g & b \end{bmatrix} \cdot \begin{bmatrix} C \\ 1 \end{bmatrix} \equiv \mathbf{x}^T\mathbf{c}. \tag{4.1}$$

The gain (g) and baseline (b), represented by the vector $\mathbf{x}$, are derived by fitting a line to the three calibration points (C_Z, T_Z), (C_A, T_A), (C_H, T_H), where the subscripts Z, A, and H indicate the zenith, ambient, and heated calibration sources, respectively. The parameters of the linear fit are determined by weighted least-squares:

$$\mathbf{x} = (\mathbf{A}^T\mathbf{W}\mathbf{A})^{-1}\mathbf{A}^T\mathbf{W}\mathbf{b} \tag{4.2}$$

$$\mathbf{A}^T = \begin{bmatrix} C_Z & C_A & C_H \\ 1 & 1 & 1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} T_Z & T_A & T_H \end{bmatrix} \qquad \mathbf{W} = \begin{bmatrix} \sigma_Z & 0 & 0 \\ 0 & \sigma_A & 0 \\ 0 & 0 & \sigma_H \end{bmatrix}^{-2}.$$

The $\mathbf{W}$ matrix is the inverse of the error covariance of the calibration data, and includes contributions due to instrument noise (see Tables 4.1 and 4.2), unknown thermal gradients in the internal targets, and, for the 54-GHz system, antenna beam spillover. The latter two contributions are discussed in more detail in Sections 4.2.2 and 4.2.3. The calibration counts, C_H and C_A, are typically filtered over several scans to reduce sensor noise using methods such as described in [39].

4.2.1 Radiometer power spectrum

The radiometer power spectrum was estimated using 1000 consecutive observations of the heated target during a March 25, 1999 flight. Periodograms were computed and averaged using the Welch-Bartlett windowing technique [40]. The results for a typical 54-GHz channel are shown in Fig. 4-6, and the results for a typical 118-GHz channel are shown in Fig. 4-7. Also shown are least-squares fits of V_0, α, and f_c for the following noise model suggested

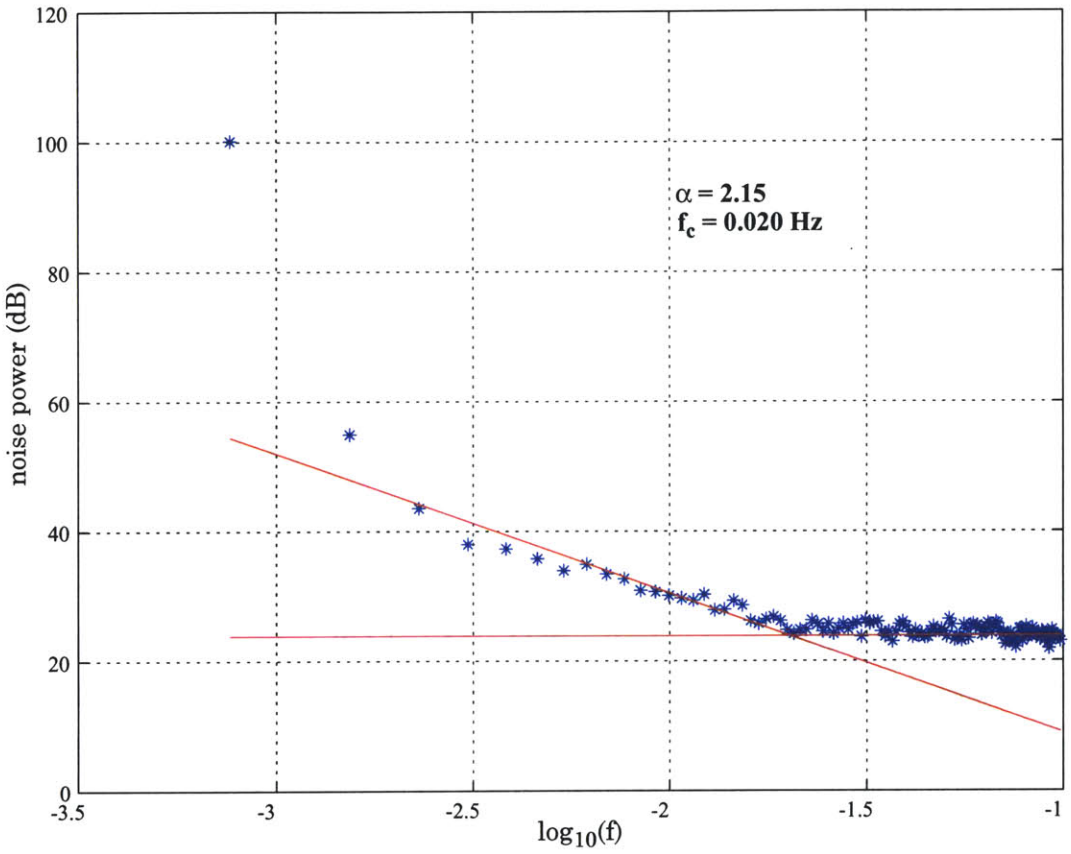

Figure 4-6: NAST-M noise power spectrum near 54 GHz. Also shown are least-squares fits of V_0, α, and f_c for the noise model given in Eq. 4.3.

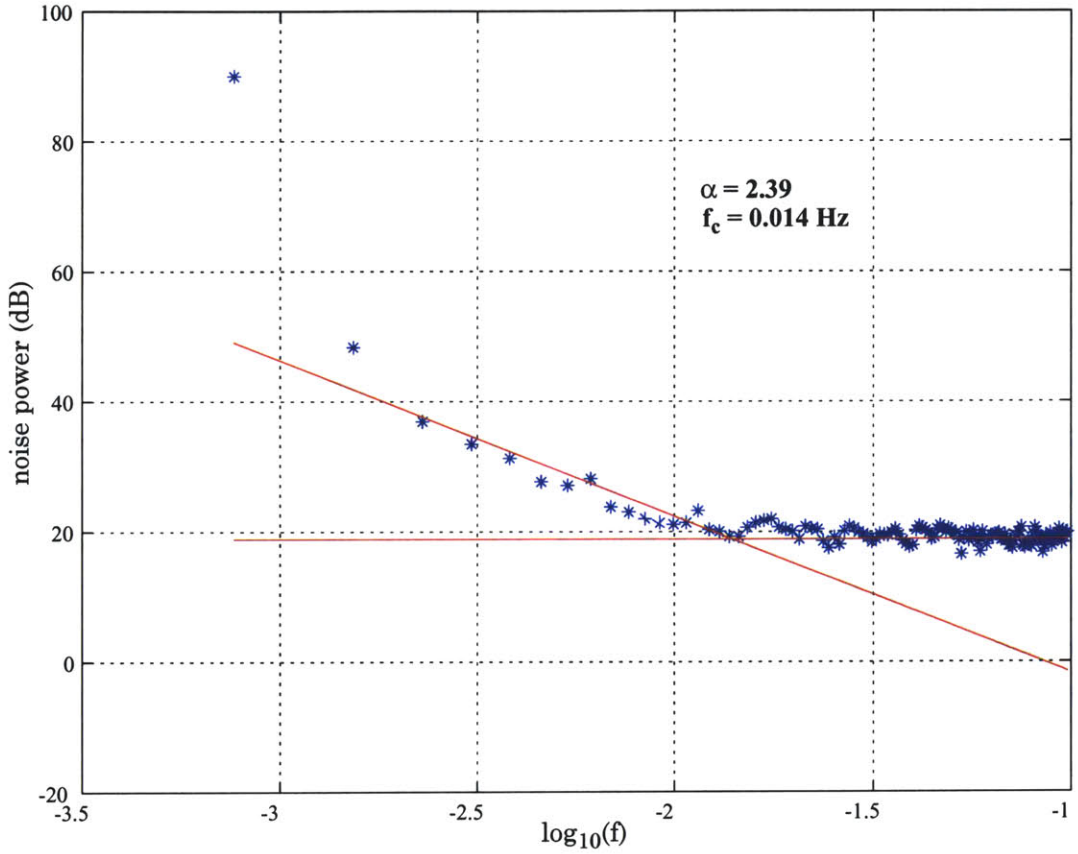

Figure 4-7: NAST-M noise power spectrum near 118.75 GHz. Also shown are least-squares fits of V_0, α, and f_c for the noise model given in Eq. 4.3.

by [41]:

$$P(f) = V_0 \left[1 + (f/f_c)^{-\alpha}\right], \tag{4.3}$$

which describes a mixture of white and $1/f$ noise. V_0 is the white noise spectral power density and f_c is the frequency at which the white and $1/f$ components are equal. The data in Figures 4-6 and 4-7 suggest that calibration data may be averaged over a period of approximately 50 seconds (10 scans). Empirical analyses [37] yielded an optimal averaging range of approximately seven scans.

4.2.2 Correction for thermal gradients in the heated target

Temperature sensors embedded throughout the heated and ambient targets allow detection of thermal gradients, both across and through the load. These must be measured at high altitudes because the low air pressure and temperature are difficult to replicate on the ground, as is the impact of this gradient on the load radiance at each frequency. Measurements of the ambient target temperature have indicated a worst-case gradient of ± 0.1 K; therefore no correction is required. Temperature differences through the heated target as large as 3 K have been observed, and correction is necessary. It is assumed that the brightness temperature B^H observed while viewing the heated target is a weighted average of the temperatures $\mathbf{T_H}$ measured by the seven embedded sensors:

$$B^H = \mathbf{w}^T \mathbf{T_H}, \text{where} \sum_{i=1}^{7} w(i) = 1. \tag{4.4}$$

Radiometric measurements at 118 ± 3.5 GHz of the ambient target and the cosmic background (known to within ± 0.1 K) are used as calibration points for determining the brightness temperature of the heated target. The solution to Eq. 4.4 is chosen such that $\sum w^2$ is minimized:

$$\mathbf{w} = \begin{bmatrix} & & \mathbf{T_H^T} & & & & \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^{\dagger} \cdot \begin{bmatrix} B^H \\ 1 \end{bmatrix} \tag{4.5}$$

Table 4.3: RTD weights calculated on March 29, 1999.

RTD 1	RTD 2	RTD 3	RTD 4	RTD 5	RTD 6	RTD 7
0.1568	0.1435	0.1402	0.1330	0.1412	0.1449	0.1402

where $(\bullet)^{\dagger}$ denotes the pseudoinverse. These weights are typically calculated for each 54- and 118-GHz channel using only the 118 ± 3.5 GHz channel for calibration. Because the penetration depth into the load is frequency-dependent, the weights are also frequency-dependent, but this dependence has been found to be negligible (on the order of ± 0.1 K) for both radiometer systems. Note that by using the zenith and ambient calibration sources to correct the heated load at 118 ± 3.5 GHz, we have effectively replaced the three-point calibration with a two-point calibration for this channel. However, all other channels use a weighted average of all three calibration sources. Weights calculated from data observed over Lake Michigan on March 29, 1999 are given in Table 4.3.

4.2.3 Characterization and correction of antenna beam spillover

An antenna beam spillover problem affecting views of the internal calibration targets for the 54-GHz radiometer results in a correctable worst-case absolute calibration bias of $\sim$3 K in the transparent channels. The "corrupted" temperature of the ambient/heated load can be modeled as a linear combination of the spillover through the zenith port, the spillover through the nadir port, and the "true" load temperature (i.e., the brightness temperature [T_A or T_H] that would be observed if there were no spillover) as follows:

$$T'_A = \eta_Z^A T_Z + \eta_N^A T_N + (1 - \eta_Z^A - \eta_N^A) T_A \tag{4.6}$$

$$T'_H = \eta_Z^H T_Z + \eta_N^H T_N + (1 - \eta_Z^H - \eta_N^H) T_H \tag{4.7}$$

The four η values (for each radiometer channel) can be accurately measured in the laboratory[3], T_Z and T_N can be estimated from flight data, and Eq. 4.6 and 4.7 can be substituted

[3]See Appendix C for more details.

in Eq. 4.5 to provide a correction. The η values depend on the azimuthal and lateral position of the reflector relative to the motor shaft, and consequently need to be measured again if the reflector position relative to the motor shaft changes. Unfortunately, the NAST-M scanning assembly underwent numerous repairs over the course of CAMEX-3 and WINTEX that affected the reflector position relative to the motor shaft, and the laboratory measurements of the η values were no longer applicable.

However, lab-derived measurements of the η values can be "tuned" for flight observations by comparing calibrated NAST-M data to radiosonde (or AMSU) data. Specifically, let B^U_{sim} and B^D_{sim} represent upwelling and downwelling brightness temperatures calculated from radiosonde data for a particular NAST-M channel. Let B^U_{obs} and B^D_{obs} represent NAST-M calibrated (using only ambient and heated targets) brightness temperatures (using lab-derived η's) when viewing the nadir and zenith positions, respectively. The tuned values of η are the solutions to the following constrained minimization problem:

$$\text{minimize} \quad (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^T(\boldsymbol{\eta} - \boldsymbol{\eta}_0) \quad \text{subject to:} \tag{4.8a}$$

$$\boldsymbol{\eta}^T \geq \begin{bmatrix} 0.001 & 0.001 & 0.001 & 0.001 \end{bmatrix} \tag{4.8b}$$

$$(B^U_{obs} - B^U_{sim})^2 \leq \sigma^2_U \tag{4.8c}$$

$$(B^D_{obs} - B^D_{sim})^2 \leq \sigma^2_D \tag{4.8d}$$

where $\boldsymbol{\eta}_0$ is the vector of laboratory measurements. The error terms σ^2_U and σ^2_D include contributions due to instrument noise (see Table 4.1), profile error (assumed to be less than ± 1 K for radiosonde observations and AMSU at NAST-M vertical resolution), and forward model error (assumed to be less than ± 0.3 K). The minimization is carried out numerically using a sequential quadratic programming (SQP) method [42]. Laboratory measurements of the parameters before and after a March 15, 1999 tuning appear in Table 4.4.

Table 4.4: Antenna beam coupling coefficients before and after March 15, 1999 tuning.

No.	Frequency (GHz)	Laboratory Measurements				After March 15, 1999 Tuning			
		η_Z^A	η_N^A	η_Z^H	η_N^H	η_Z^A	η_N^A	η_Z^H	η_N^H
1	50.30	0.0048	0.0016	0.0057	0.0014	0.0055	0.0017	0.0086	0.0024
2	51.76	0.0066	0.0021	0.0053	0.0014	0.0066	0.0021	0.0071	0.0020
3	52.80	0.0077	0.0024	0.0054	0.0013	0.0093	0.0023	0.0023	0.0010
4	53.75	0.0071	0.0024	0.0045	0.0014	0.0141	0.0024	0.0089	0.0026
5	54.40	0.0073	0.0023	0.0050	0.0016	0.0084	0.0024	0.0010	0.0010
6	54.94	0.0082	0.0027	0.0056	0.0018	0.0092	0.0028	0.0012	0.0010
7	55.50	0.0107	0.0031	0.0080	0.0023	0.0091	0.0029	0.0025	0.0010
8	56.02	0.0163	0.0044	0.0129	0.0034	0.0101	0.0032	0.0010	0.0010

4.3 Comparisons with AMSU

NAST-M radiances observed on March 26, 1999 over Lake Michigan were compared with radiances observed by AMSU in the following way. First, a temperature retrieval was performed using the AMSU-A channels [43]. A humidity profile was obtained using data from a coincident radiosonde, and the surface temperature was retrieved using the 11-μm and 12-μm channels from the Advanced Very-High Resolution Radiometer (AVHRR) on the NOAA-15 satellite. These data were used with a forward model [44] and a sea-surface model [45] to simulate NAST-M radiances for viewing angles $\pm 50°$ from nadir. Simulated brightness temperatures were band-averaged over the NAST-M passbands using laboratory measurements of the radiometer frequency response over 1200 frequencies. A Gaussian antenna beamshape was assumed in the radiative transfer calculations, as suggested by [46]. The errors associated with the simulated NAST-M brightness temperatures due to AMSU calibration/retrieval errors [43, 47], forward model errors [48], surface emissivity model errors [45], AVHRR calibration/retrieval errors [49], and temporal and spatial offsets are believed to be less than 1.5 K for all channels. Corrections for antenna sidelobe spillover affecting the 54-GHz system were derived from a March 15, 1999 overflight of Lake Michigan using temperature and humidity profiles retrieved from AMSU and surface temperature data from buoy station 45007 (NOAA National Data Buoy Center). The results are shown in Figure 4-8 and 4-9. Channels 7-9 of the 118-GHz system are significantly degraded

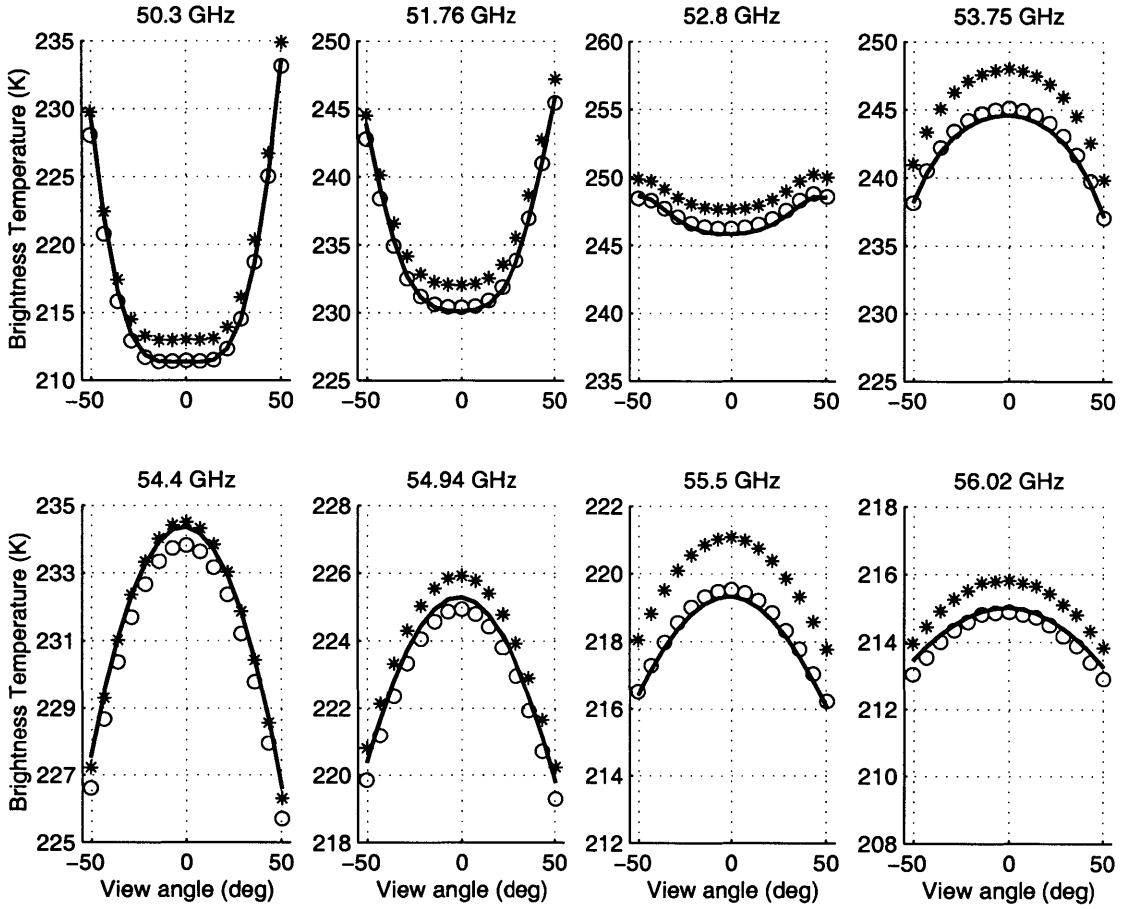

Figure 4-8: Comparison of NAST-M 54-GHz channels with AMSU (March 26, 1999 overpass, 00:41 UTC). Brightness temperatures calculated from AMSU radiances are shown with a solid line, and NAST-M radiances (corrected for antenna beam spillover) are shown with circles. NAST-M radiances before spillover corrections have been applied are indicated with asterisks.

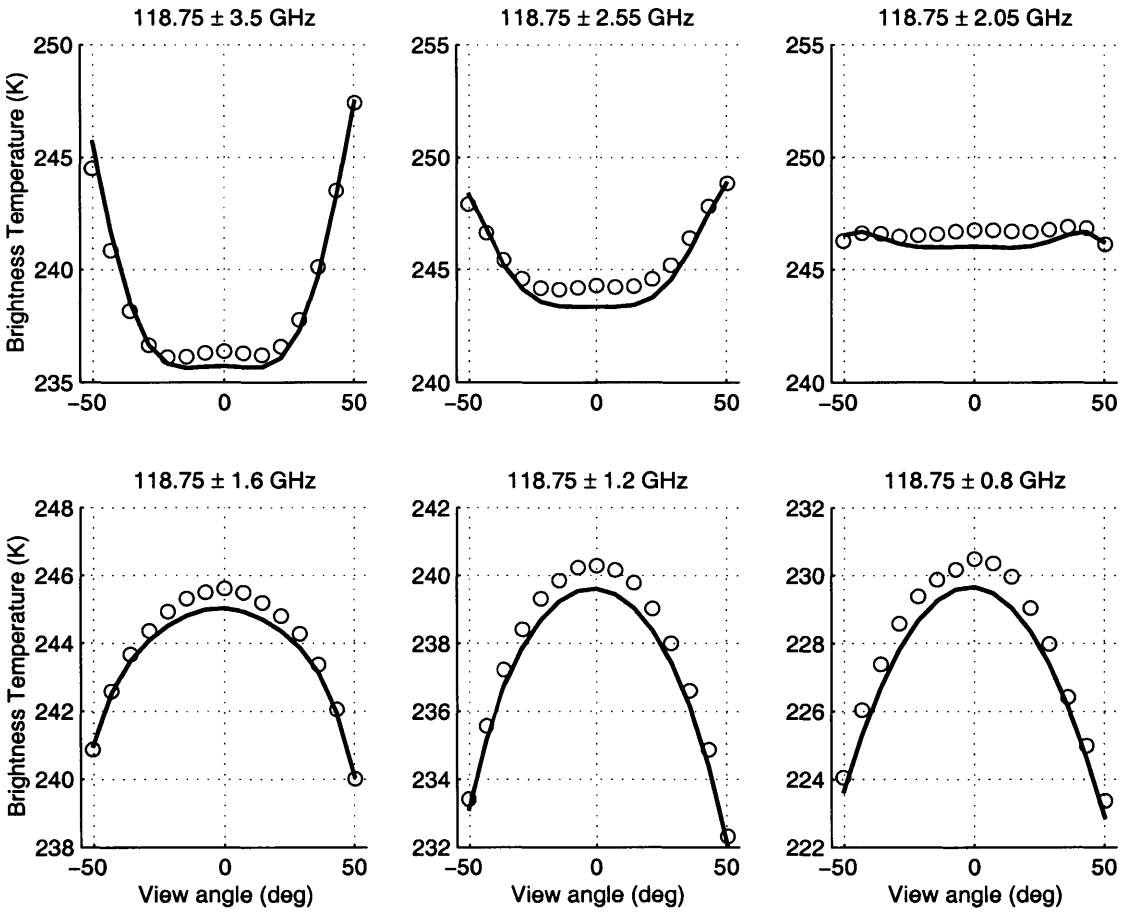

Figure 4-9: Comparison of NAST-M 118-GHz channels with AMSU (March 26, 1999 overpass, 00:41 UTC). Brightness temperatures calculated from AMSU radiances are shown with a solid line, and NAST-M radiances are shown with circles.

due to RF losses in the preamplifier circuits, and are not shown. The brightness temperatures calculated from AMSU radiances are shown with a solid line, and NAST-M radiances (averaged from 00:36 - 00:44 UTC; straight and level flight over water) are shown with circles. NAST-M brightness temperatures near 54 GHz before spillover corrections are indicated with asterisks. Agreement to within 1 K is obtained for all channels of the 54-GHz radiometer and channels 1-6 of the 118-GHz radiometer (see Tables 4-1 and 4-2 for exact brightness temperature offsets at nadir). AMSU/AVHRR/NAST-M comparisons on March 29, 1999 over Lake Michigan demonstrate similar agreement.

The excellent agreement between NAST-M and AMSU, while encouraging, is only indicative of the calibration of NAST-M relative to AMSU. More extensive studies are needed in order to validate the absolute calibration accuracy of NAST-M under all circumstances.

4.4 Discussion and Summary

NAST-M offers several noteworthy improvements over most prior aircraft and satellite-based instruments, as discussed below.

The zenith port provides: 1) an additional highly-stable calibration source for the more transparent channels, 2) a mechanism for measuring and compensating a thermal gradient in the heated target (thereby improving the calibration of the more opaque channels), and 3) a source of data for validating transmittance models.

The use of commercial, off-the-shelf (COTS) components dramatically reduced development time and cost. Thermal and mechanical environments throughout the instrument housing were engineered to ensure reliability of COTS hardware. NAST-M was designed, developed, and flown in under two years at a cost under $800,000.

NAST-M provides the first high-resolution, collocated, multiband passive microwave measurements for three-dimensional temperature sounding, cloud and precipitation studies, and transmittance modeling. The 2.6-km resolution provided by the NAST suite also facilitates interpretation of unresolved satellite observations of meteorological phenomena. The three-point calibration on NAST-M provides accurate brightness temperature measurements in two oxygen bands, permitting the first reliably accurate temperature profile

retrieval images.

NAST-I/M provides an atmospheric sounding platform with sufficient accuracy and sensitivity for validating satellite observation system concepts. Data from NAST-I and NAST-M are used in Chapters seven and eight to retrieve temperature profile images, as well as cloud and precipitation structure. The temperature profile retrieval performance in cloudy atmospheres using both NAST-I and NAST-M is significantly better than the performance obtained by using either instrument alone.

Chapter 5

NAST-I: Characterization and Reduction of White and Colored Noise

IT was demonstrated in Chapter 3 that measurement error has a significant effect on temperature profile retrieval performance. A statistically-optimal estimator requires knowledge of both the variance and correlation structure of the noise, and possibly higher-order characterizations as well. In this chapter, we present a second-order analysis of NAST-I measurement noise in both the spectral and spatial domains for two flights: Sep. 13, 1998 (CAMEX, Florida) and Mar. 29, 1999 (WINTEX, Wisconsin / Great Lakes).

5.1 Overview of NAST-I noise sources

The Michelson interferometer spectrometer was introduced in Chapter 2 and the NAST-I instrument was outlined in Chapter 4. We now briefly summarize the significant mechanisms that contribute to the NAST-I system noise [50], and discuss the mechanism identified in [51, 52] to be the dominant noise source: vibration-induced optical jitter.

5.1.1 Additive noise sources

The noise processes that are independent of the scene temperature are termed "additive". The significant sources of additive noise are:

Photon and detector noise Photon (or quantum) noise results from the statistical fluctuations in the number of photons converted to photoelectrons by a photon detector. Detector noise in semiconductor infrared detectors is produced by charge carriers generated by thermal excitation across the band gap (thermal generation-recombination noise).

Electronics noise Electronics noise is thermal in character, and arises in electronic components following the detector, primarily the input stages of the preamplifier.

Aliasing noise The spectral radiance signal is filtered before sampling: by optical components (e.g. beamsplitter and dichroics), by the detector and by the post-detection filter. Any components remaining above the Nyquist frequency are aliased.

Low-frequency "1/f" noise Semiconductor devices generally exhibit low-frequency noise with a power density of the form $c_0 f^{-a}$, where a is a constant on the order of unity. This type of noise may be correlated in the spectral domain.

Quantization noise Quantization noise is introduced by the analog-to-digital converter when the analog detector signal is represented by a finite number of bits.

Sampling errors The interferogram must be sampled at constant time offsets to faithfully restore the spectral information. If the signals triggering the sampling contain position errors, these errors will be manifested in the spectral domain.

5.1.2 Scene-dependent noise sources

The following sources of noise depend on the scene temperature measured by the interferometer, and the noise is therefore correlated in the spectral domain. The significant sources of scene-dependent noise are:

Mirror velocity errors Deviations from a constant mirror velocity (and therefore a constant sampling interval of the interferogram) affect the measured amplitude and phase of any given spectral component.

Optical jitter-induced noise Any tilt of the mirrors reduces the frequency-dependent modulation efficiency of the interferometer. The modulation caused by the jitter results in a multiplicative noise in the autocorrelation and interferogram spectral domains.

5.1.3 Mathematical description of jitter-induced noise

We now develop a mathematical relationship between the interferogram domain and the spectral domain and show how noise is transferred from one domain to the other.

The interferogram

The interferometer splits a monochromatic beam of radiation (wavenumber $= \bar{\nu} \triangleq \lambda^{-1}$) into two beams of unequal path length (path difference $= \Delta x$), which are recombined at the detector:

$$a_1 = \frac{1}{4} I_0 \cos(2\pi x \bar{\nu}), \quad a_2 = \frac{1}{4} I_0 \cos[2\pi(x + \Delta x)\bar{\nu}] \tag{5.1}$$

$$a_1 + a_2 = \frac{1}{2} I_0 [1 + \cos(2\pi \Delta x \bar{\nu})]. \tag{5.2}$$

For a polychromatic source, Eq. 5.2 is integrated over all wavenumbers:

$$s(x) = 2 \int_0^\infty S_{\bar{\nu}} [1 + \cos(2\pi \Delta x \bar{\nu})] d\bar{\nu} = k_0 + 2 \int_0^\infty S_{\bar{\nu}} \cos(2\pi \Delta x \bar{\nu}) d\bar{\nu}, \tag{5.3}$$

where $S_{\bar{\nu}}$ is the power spectral density of the source at wavenumber $\hat{\nu}$ in units of power per wavenumber. The *interferogram* is defined as the second term in Eq. 5.3:

$$m(x) = 2 \int_0^\infty S_{\bar{\nu}} \cos(2\pi \Delta x \bar{\nu}) d\bar{\nu}, \tag{5.4}$$

which can be rewritten (assuming $S_{\bar{\nu}}$ is an even function of wavenumber) as

$$m(x) \triangleq \int_{-\infty}^{\infty} S_{\bar{\nu}} e^{i2\pi\bar{\nu}x} d\bar{\nu}. \tag{5.5}$$

This is the familiar relation expressing the interferogram and the spectral intensity as Fourier transform pairs.

Effect of mirror misalignment

If the fixed mirror is tilted by a small angle (ε), the effective path difference is no longer constant for each ray in the collimated beam. The effective path difference for a given ray is a function of the position of the ray in the plane perpendicular to the beam. Letting r and ψ denote this position, the effective path difference becomes

$$x'(r, \psi) = x + 2\varepsilon r \sin\psi, \tag{5.6}$$

and the net effect of the mirror tilt is obtained by integrating over the plane perpendicular to the beam. Equation 5.3 becomes:

$$s(x) = \int_{-\infty}^{\infty} S_{\bar{\nu}}[1 + H_{\varepsilon}(\bar{\nu}, x)] d\bar{\nu}, \tag{5.7}$$

where

$$H_{\varepsilon}(\bar{\nu}, x) = \frac{1}{\pi R^2} \int_{-\pi}^{\pi} \int_{0}^{R} \cos[2\pi\bar{\nu}(x + 2\varepsilon r \sin\psi)] r \, dr \, d\psi \tag{5.8}$$

$$= \frac{2}{k} J_1(k) \cos(2\pi\bar{\nu}x), \tag{5.9}$$

where R is the radius of the collimated beam, $k \triangleq 4\pi\bar{\nu}\varepsilon R$, and $J_1(\cdot)$ is the Bessel function of the first kind. The interferogram becomes:

$$m(x) = \int_{-\infty}^{\infty} S_{\bar{\nu}} \left[\frac{2}{k} J_1(k)\right] e^{i2\pi\bar{\nu}x} d\bar{\nu}. \tag{5.10}$$

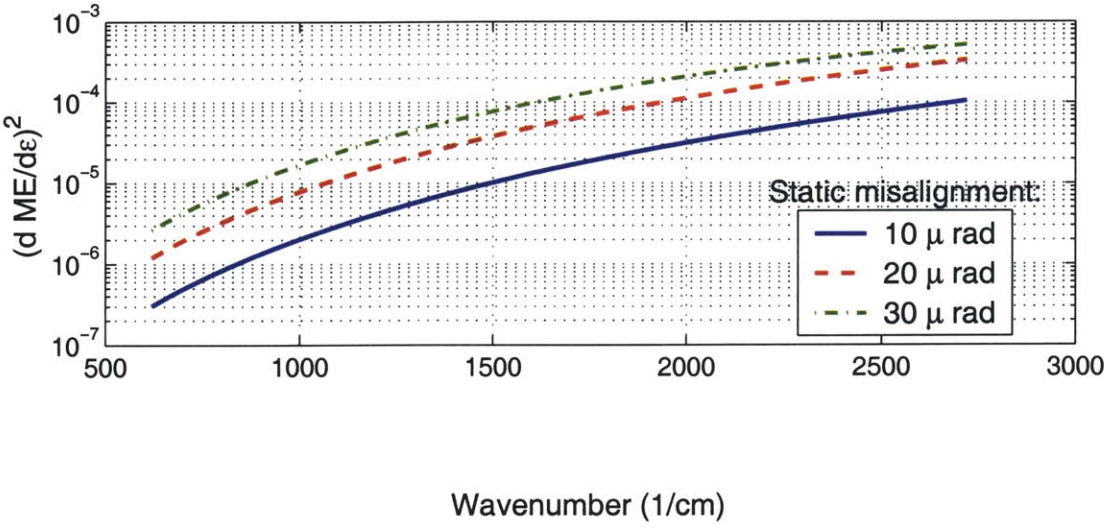

Figure 5-1: The contribution of the $\frac{d\,\mathrm{ME}(\bar{\nu})}{d\varepsilon}$ term in the Kelly model of NAST-I jitter-induced noise [51].

The term $\left[\frac{2}{k}J_1(k)\right]$ is known as the *modulation efficiency*, $\mathrm{ME}(\bar{\nu},\varepsilon)$. If the mirror tilt is constant over the entire interferogram, then $S_{\bar{\nu}}$ is attenuated by a factor of $\mathrm{ME}(\bar{\nu},\varepsilon)^{-1}$. In practice, the angular tilt is a random variable, with power spectrum P_{jitter}.

Model for jitter-induced noise

Kelly et al. [51] suggest the following model for the jitter-induced component of NAST-I noise in RMS radiance units:

$$\mathrm{NEDR}_{\mathrm{jitter}}(\bar{\nu}) = \sqrt{\left[\left(B(\bar{\nu},T_{\mathrm{scene}}) - B_{\mathrm{bkgd}}(\bar{\nu})\right)^2\left(\frac{d\,\mathrm{ME}(\bar{\nu})}{d\varepsilon}\right)^2\right] * P_{\mathrm{jitter}}}, \tag{5.11}$$

where $*$ is the convolution operator. Values of $\frac{d\,\mathrm{ME}(\bar{\nu})}{d\varepsilon}$ for static tilts of 10, 20, and 30 microradians are shown in Fig. 5-1.

Table 5.1: Flight segments used for NAST-I noise analyses.

Deployment	Date	Time (UTC)	Latitude (°N)	Longitude (°W)
CAMEX-3	Sep. 13/14, 1998	23:44:15 – 23:48:15	24.69 – 25.13	77.83 – 77.56
WINTEX	Mar. 29/30, 1999	1:05:31 – 1:09:26	43.07 – 43.53	87.39 – 87.12

5.2 Summary of flight data used for noise analyses

One representative flight was chosen from both the CAMEX-3 and WINTEX deployments. Only data within $\pm \sim 20°$ of nadir (middle five pixels) was retained. The correlation effects of the atmosphere were removed by selecting homogeneous, cloud-free regions and operating only on the difference of adjacent pixels (with the exception of the ION analysis, for which all flight data was used). High-resolution infrared data from MAMS (CAMEX-3) and MAS (WINTEX) was used to identify cloud-free regions over water. The details of each flight segment are given in Table 5.2, where the spatial and temporal bounds apply to the cloud-free regions.

5.3 Analysis of spectrally-uncorrelated noise

The ION algorithm [1] was used to estimate the spectrally-uncorrelated component of the NAST-I system noise. ION estimates of the RMS variation of the uncorrelated noise are shown in Fig. 5-2. Also shown are similar estimates based on the analysis of calibration data. The ION estimates are in excellent agreement with the calibration estimates. The principal discrepancies arise in the water vapor bands and in extremes of the shortwave band. These discrepancies could be due to a number of factors, including water vapor in the instrument canister that affected the calibration data to a greater extent than the scene data, and possibly and varying instrument background temperature.

Histograms of the ION-estimated noise sequences for two temperature-band channels with weighting function peaks near 8 km are shown in Fig. 5-3. The Gaussian pdf associated with thermal noise processes is clearly revealed. The spectral correlation of the

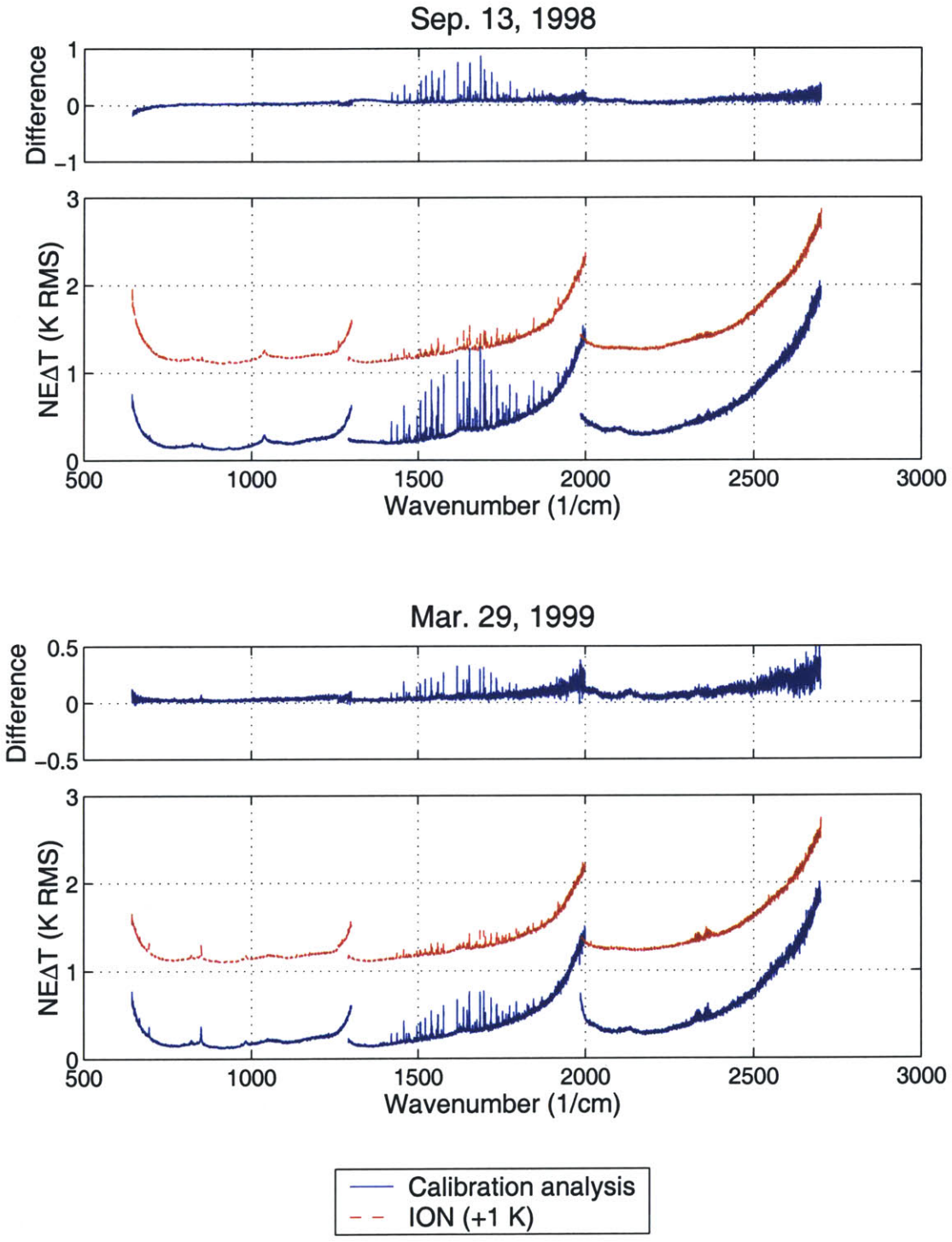

Figure 5-2: Estimates of the RMS variation of the uncorrelated component of NAST-I system noise for two flights and two methods. Estimates using the ION algorithm are offset by 1 K from estimates obtained using calibration analysis. Also shown are NEΔT differences (K RMS) between the ION estimates and the calibration estimates.

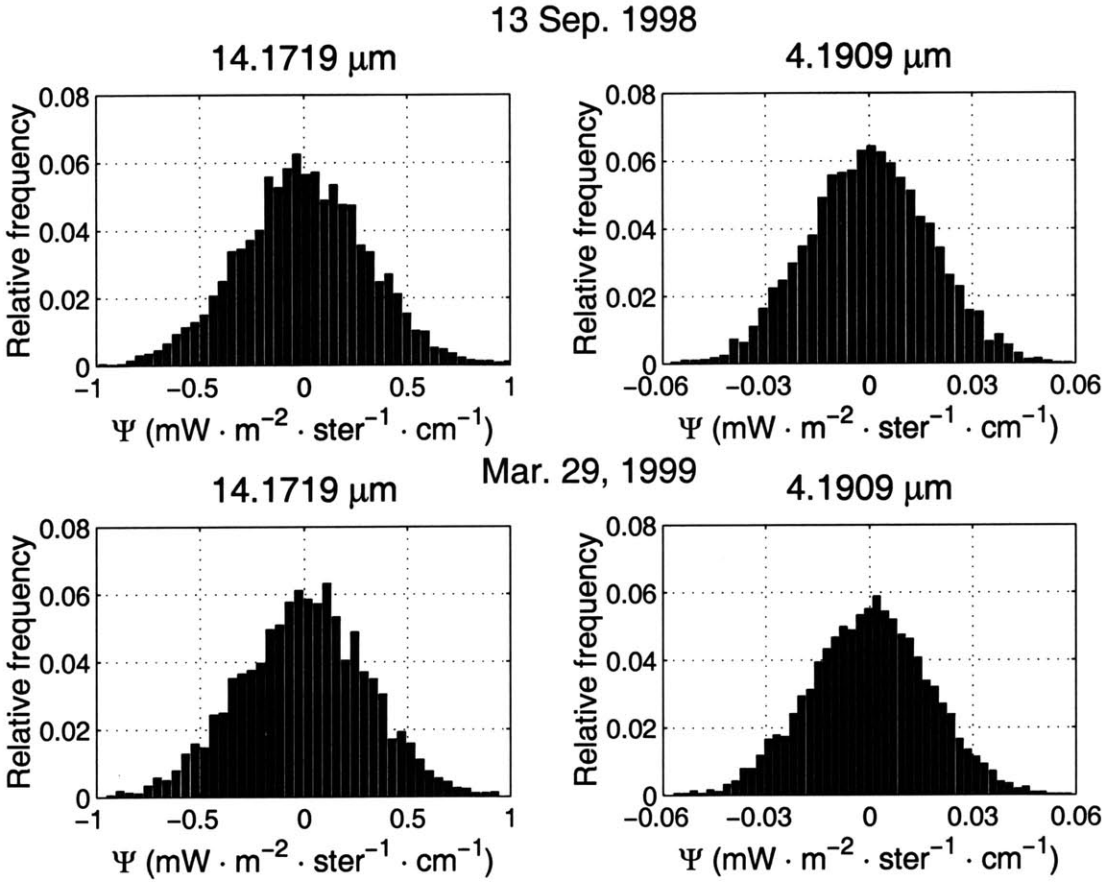

Figure 5-3: Histograms of ION-estimated noise sequences for two temperature-band channels. The weighting function of both channels peaks near 8 km.

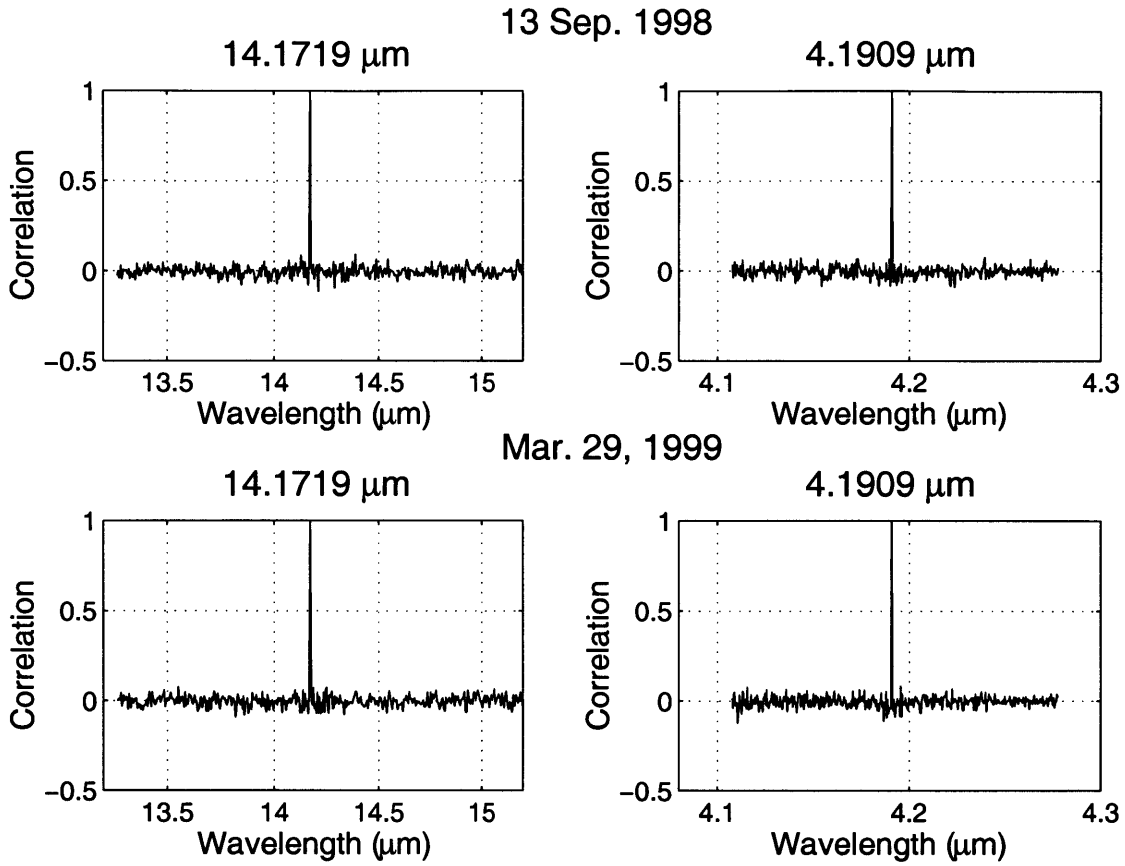

Figure 5-4: Spectral correlation of ION-estimated noise sequences for two temperature-band channels. The weighting function of both channels peaks near 8 km.

ION-estimated noise sequences for the same two temperature-band channels selected in Fig. 5-3 is shown in Fig. 5-4. The shape is characteristic of a white-noise process.

Finally, the RMS variation of the spectrally-correlated noise component was estimated in the following way. First, the ION-estimated noise sequences were subtracted from the 100 NAST-I pixels in the cloud-free region given in Table 5.2. Adjacent-pixel differences were then taken in the cross-track direction and the RMS variation of the residuals was computed. The resulting estimate of correlated RMS variation may be artificially high for two reasons. If there is a horizontal temperature gradient (or thin cloud layer undetected by MAMS) in the cross-track direction, this will appear as correlated noise. In addition, if

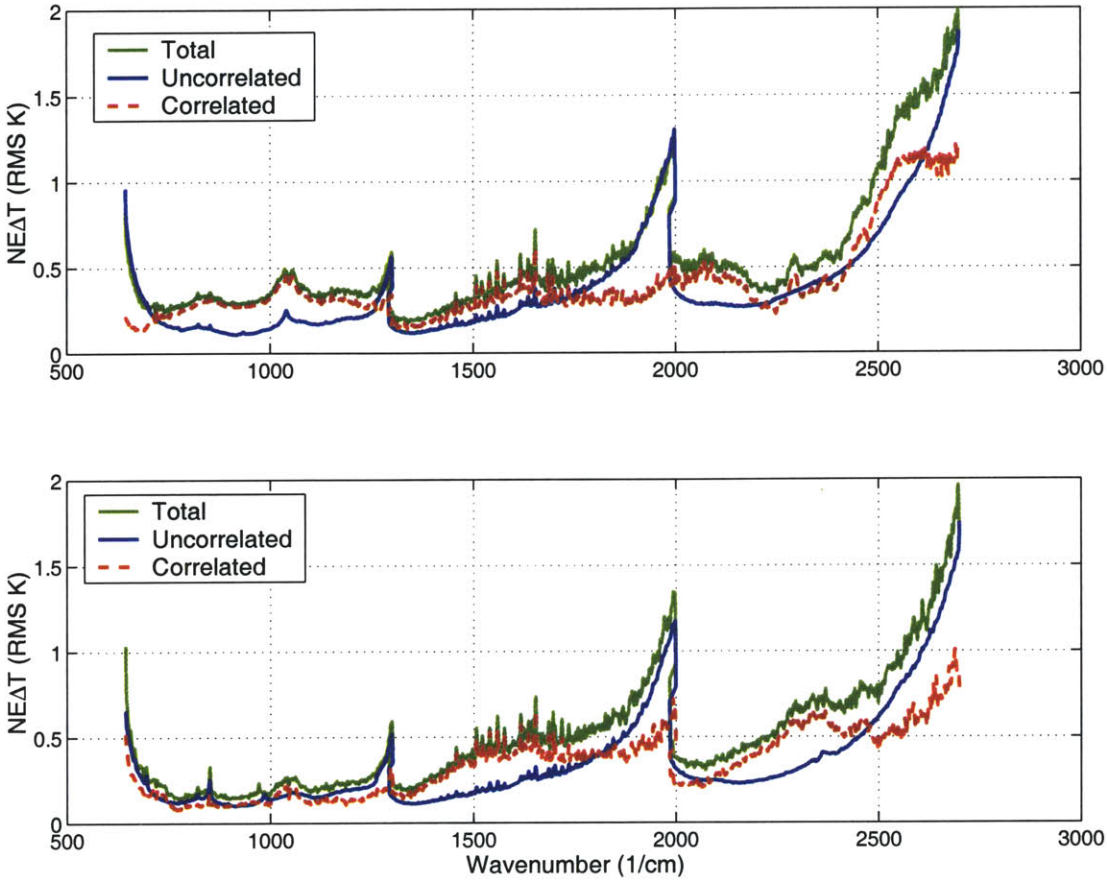

Figure 5-5: The estimated RMS variation of the spectrally correlated and uncorrelated noise components of NAST-I system noise. Also shown is the RMS variation of the total noise component.

the noise is spectrally correlated but spatially uncorrelated, the differencing operator will amplify the noise. The spatial correlation structure of the spectrally correlated noise will be examined in a future section. The estimates of the RMS variation of the correlated noise are shown in Fig. 5-5. Several features are noteworthy. First, the correlated noise component is dominant in all three spectral bands for the Sep. 13, 1998 flight. However, the correlated noise component is significantly larger in the middle portion of the shortwave band on Mar. 29, 1999 than on Sep. 13, 1998.

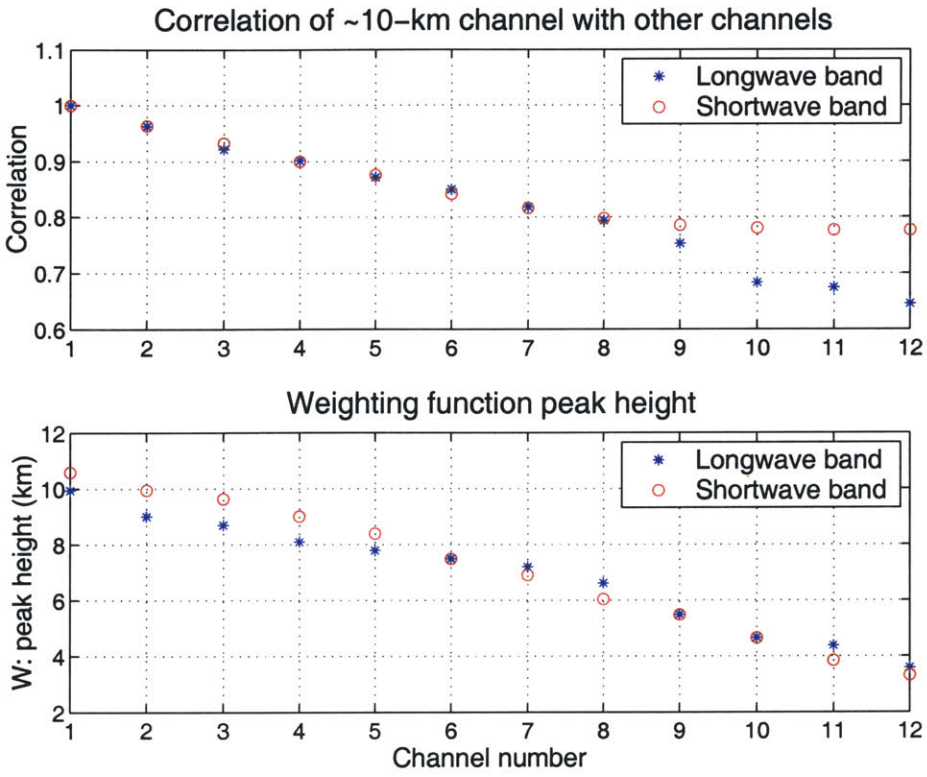

Figure 5-6: Properties of simulated NAST-I channels used in Fig. 5-7. The top graph shows the correlation between a channel with weighting function peak near 10 km and 11 other channels with weighting function peaks from approximately 2–10 km. The bottom graph show the altitude at which the weighting function for each channel is maximum.

5.4 Analysis of spectrally-correlated noise

We limit the present study of the spectral correlation structure of the NAST-I system noise to a representative set of twelve channels from each of the two temperature bands. The channels are chosen such that the peaks of the weighting functions sample the atmosphere uniformly from approximately 2 to 10 km. The weighting function peak heights for the channel sets are shown in Fig. 5-6. Note that peak heights of the temperature weighting functions for the shortwave bands (see Fig. 2-7, for example) are significantly lower than the Planck weighting function peak heights shown in Fig. 5-6. This is the reason the spectral correlation of the shortwave channels asymptotes after channel nine.

Also shown in Fig. 5-6. is the spectral correlation calculated from simulated NAST-I radiances over a set of 10,000 atmospheric profiles. This correlation structure will be used to delineate atmospheric and non-atmospheric correlation.

The spectral correlation between the channel with the highest weighting function peak and the other channels was calculated after taking cross-track adjacent pixel differences. The results are shown in Fig. 5-7. Also shown are the spectral correlation results from simulated NAST-I data. The shortwave bands for the Sep. 13, 1998 flight exhibit spectral correlation that is fairly consistent with with the simulated atmospheric correlation structure. Both bands for the Mar. 29, 1999 flight deviate slightly from the simulated atmospheric correlation structure, with the longwave band exhibiting a relatively high degree of spectral decorrelation for small spectral offsets. This decorrelation could be caused by an over/underestimation of the uncorrelated noise component for the segment of flight data. The spectral correlation structure exhibited by the longwave band on Sep. 13, 1998 is much higher than predicted by simulated data, and could be the result of correlated instrument noise.

As a final observation, the correlation structure of both bands differs significantly from the Sep 13., 1998 flight to the Mar. 29, 1999 flight. A similar observation was noted by [52], and the difference was attributed primarily to: 1) different vibration environment due to new instrument shock mounts installed after CAMEX-3, and 2) a change in system responsivity due to a reduction in the hot blackbody setpoint used after CAMEX-3.

5.5 Analysis of spatial noise correlation

The spatial correlation of the spectrally-correlated noise component was investigated by computing the correlation in the cross-track and along-track directions. Again, the lack of data precludes a full characterization of the correlation structure. We therefore compute the spatial correlation for a set of NAST-I "pseudochannels" (a linear combination of NAST-I channels).

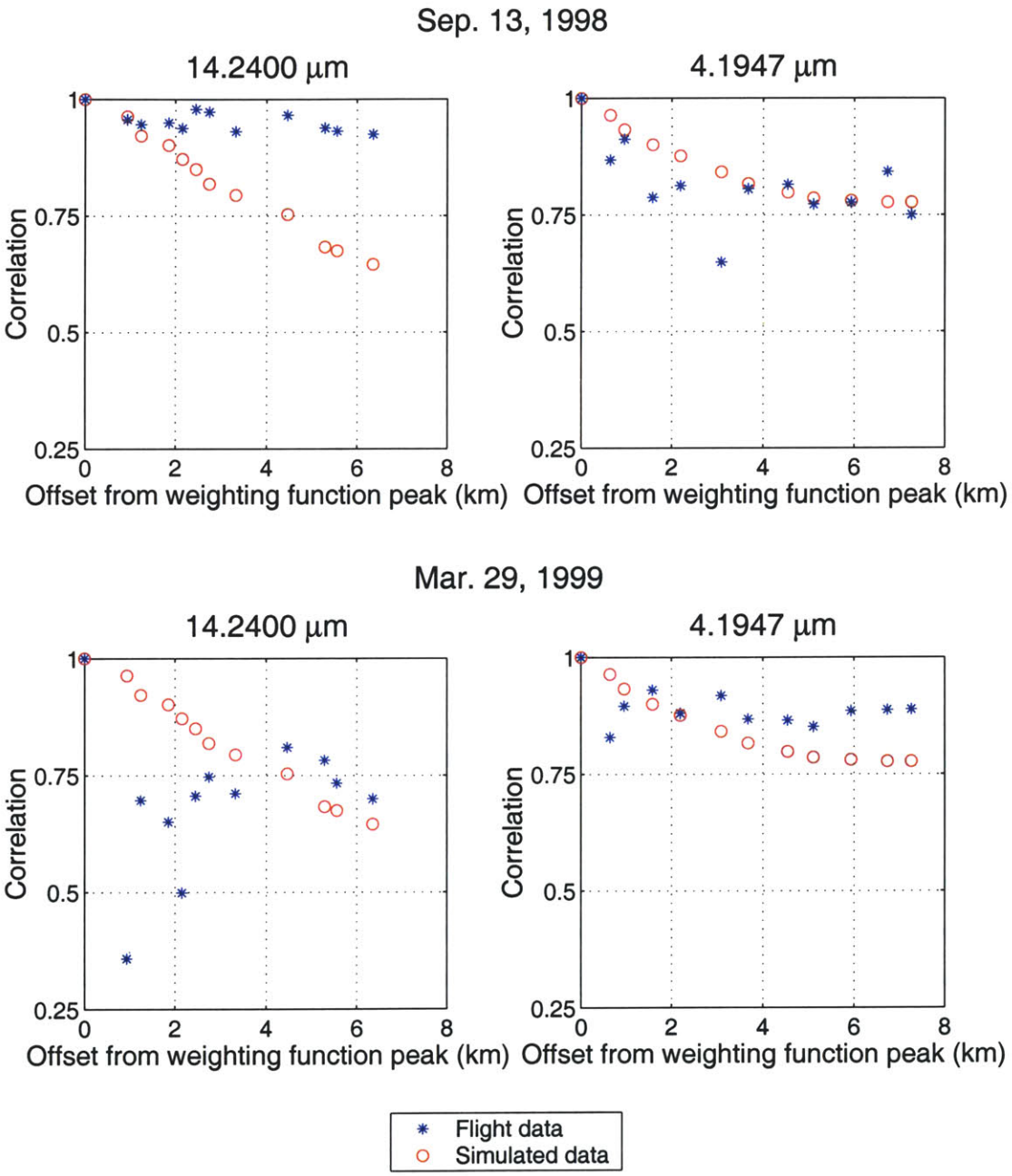

Figure 5-7: Correlation structure of NAST-I spectrally-correlated noise component. The correlation structure of simulated radiances over an ensemble of 10,000 atmospheric profiles is also shown.

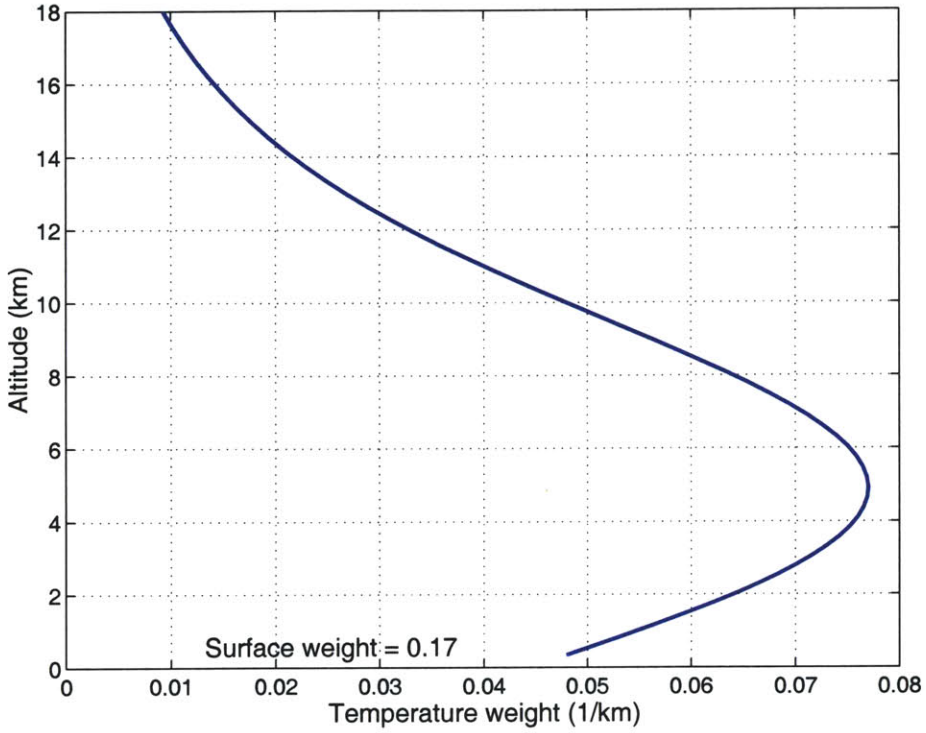

Figure 5-8: Template temperature weighting function used for the NAST-I pseudochannels.

5.5.1 Formulation of NAST-I pseudochannels

Two NAST-I "pseudochannels" were constructed (one for the longwave band and one for the shortwave band), each with the temperature weighting function shown in Fig. 5-8. This weighting function is very similar to that of the 54.4-GHz and 118.75 ± 1.6-GHz NAST-M channels. Define the weighting function matrix for a band of N NAST-I channels as:

$$\mathbf{K} = \begin{bmatrix} W_{T,\nu_1}[k] \\ W_{T,\nu_2}[k] \\ \vdots \\ W_{T,\nu_N}[k] \end{bmatrix}, k = 1, \ldots, M, \tag{5.12}$$

where $W_{T,\nu_i}[k]$ is a discretization over M atmospheric levels of $W_{T,\nu_i}(z)$ (defined in Eq. 2.66). We seek the linear combination of NAST-I channels, A, that satisfies

$$\mathbf{K}^T A = W_{\text{Template}}, \tag{5.13}$$

where the M-element row vector W_{Template} is the template weighting function shown in Fig. 5-8. Equation 5.13 is a linear system of M equations with N unknowns. N is usually much larger than M for hyperspectral sounders, and the system of equations is underdetermined. We therefore pick the solution that minimizes the noise of the pseudochannel, $A\mathbf{C}_{\Psi_T\Psi_T}A^T$, where $\mathbf{C}_{\Psi_T\Psi_T}$ is the noise covariance matrix of the band of NAST-I channels expressed in Noise-Equivalent ΔT (NEDT[1]). This constrained minimization problem can be solved by introducing a vector of M Lagrange multipliers (Φ) and setting the derivative with respect to A equal to zero:

$$\frac{d}{dA}\left[A\mathbf{C}_{\Psi_T\Psi_T}A^T + \Phi^T(\mathbf{K}^T A - W_{\text{Template}})\right] = 0, \tag{5.14}$$

which results in the following set of linear equations:

$$\begin{bmatrix} 2\mathbf{C}_{\Psi_T\Psi_T} & \mathbf{K} \\ \mathbf{K}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} A \\ \Phi \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ W_{\text{Template}} \end{bmatrix} \tag{5.15}$$

with solution

$$\begin{bmatrix} A \\ \Phi \end{bmatrix} = \begin{bmatrix} 2\mathbf{C}_{\Psi_T\Psi_T} & \mathbf{K} \\ \mathbf{K}^T & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0} \\ W_{\text{Template}} \end{bmatrix}. \tag{5.16}$$

5.5.2 Calculation of spatial noise correlation

The spatial correlation in the cross-track direction was calculated after taking adjacent-pixel differences in the along-track direction. The spatial correlation in the along-track direction was calculated after taking adjacent-pixel differences in the cross-track direction.

[1]Radiance units (NEDR) can be converted to NEDT as follows: NEDT $\triangleq \mathcal{J}_\nu^{-1}(\mathcal{J}_\nu(T_0) + \text{NEDR}) - T_0$, where T_0 is some nominal temperature. $T_0 = 290$ K in this thesis, unless stated otherwise.

The results are shown in Fig. 5-9. The relatively large correlation present in the along-track direction is an artifact of a smoothing algorithm employed by the distributor of the NAST-I data: a triangular filter with coefficients of $\frac{1}{4}$, $\frac{1}{2}$, and $\frac{1}{4}$ was convolved with the NAST-I data in the along-track direction. If the noise were spatially white before the application of the smoothing, the spatial correlation after the smoothing for a one-pixel offset would be $\frac{2}{3}$, which is consistent with the values shown in Fig. 5-9. The spatial correlation appears to be significantly smaller than the spectral correlation, which suggests that noise can be filtered spatially.

5.6 Summary

The ION-estimated RMS variation of the spectrally uncorrelated component of NAST-I system noise is in excellent agreement with calibration analysis. Uncorrelated noise can therefore be removed by subtracting the ION-estimated noise sequences. The spectrally correlated component of NAST-I system noise was dominant in all bands for Sep. 13, 1998. This component was smaller by approximately a factor of two (in RMS) in the longwave band for Mar. 29, 1999. The spectral correlation structure for the longwave band on Sep. 13, 1998 and the shortwave band on Mar. 29, 1999 are not consistent with correlation structure due to simulated atmospheric variability. The spatial correlation is smaller than the spectral correlation, and sufficiently low that spatial filtering should reduce its effects significantly. For example, Figures 5-5 and 5-9 suggest that a two-dimensional spatial filter with the following coefficients would reduce the variance of the correlated noise component component below that of the nominal uncorrelated (thermal) component for both the Sep. 13, 1998 and Mar. 29, 1999 flights:

$$h = \begin{bmatrix} \frac{1}{18} & \frac{1}{9} & \frac{1}{18} \\ \frac{1}{9} & \frac{1}{3} & \frac{1}{9} \\ \frac{1}{18} & \frac{1}{9} & \frac{1}{18} \end{bmatrix}. \tag{5.17}$$

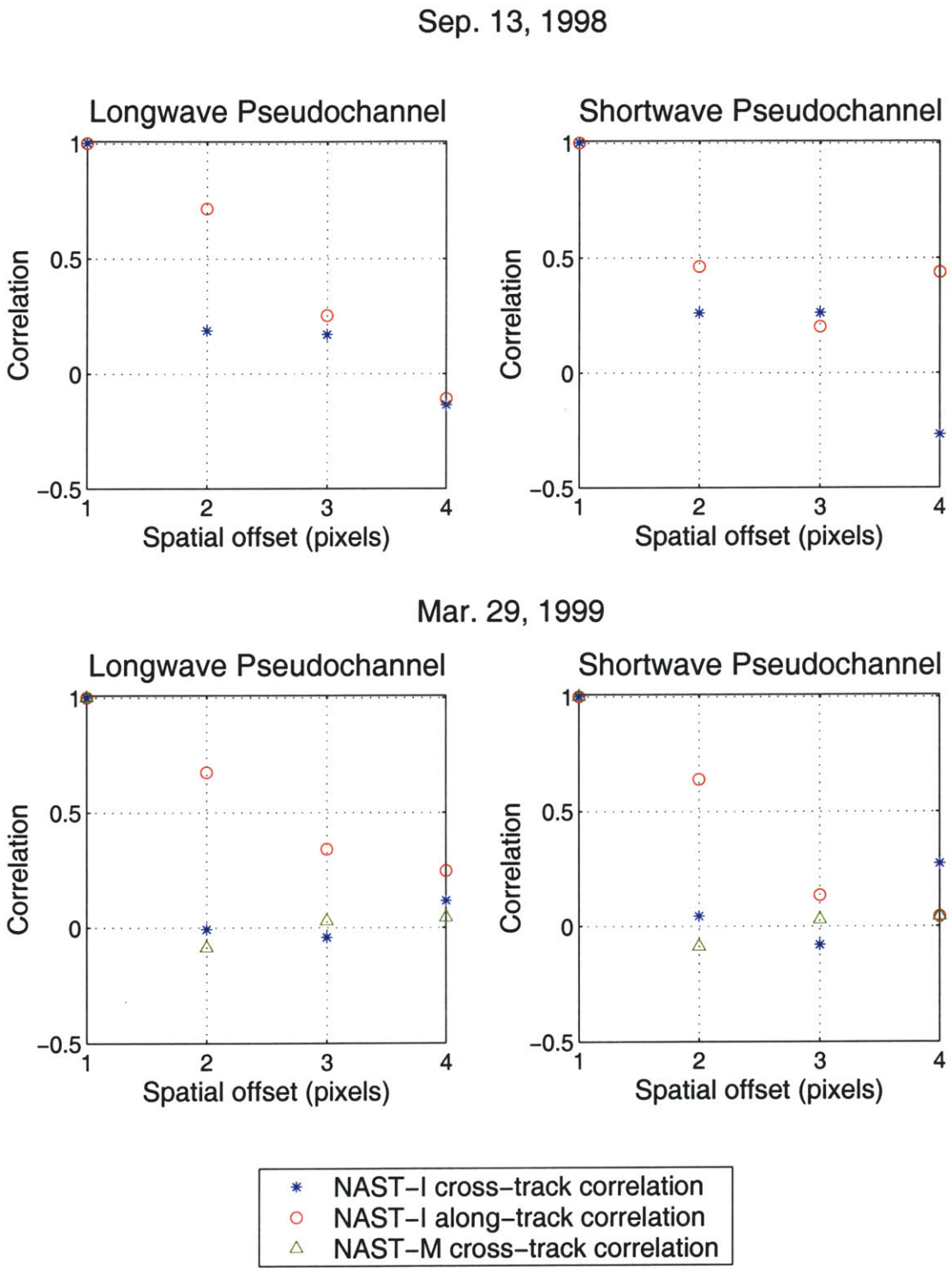

Figure 5-9: Spatial correlation of the spectrally correlated component of NAST-I system noise. Correlation was computed in both the cross-track and along-track directions. Cross-track correlation computed from spatially-coincident NAST-M data (118.75 ± 1.6 GHz, weighting function peak near 5 km) observed on Mar. 29, 1999 is shown for comparison.

Chapter 6

Retrieval of Cloud-Cleared Temperature Profiles: Simulation Studies

SIMULATED radiance observations from the Atmospheric InfraRed Sounder (AIRS) [53], Advanced Microwave Sounding Unit–A (AMSU–A) [54], and (AMSU–B[1]) [55] instruments are used in this chapter to derive and evaluate methods for retrieving the temperature profile (single-pixel) or clear-column AIRS radiance (multiple-pixel) in the presence of clouds. Principal components analyses are performed on the radiance data to determine the contributions of clouds and the temperature profile to radiance variation. The significant superiority of a neural network relative to a linear least-squares estimator (LLSE) is demonstrated for both temperature profile and clear-column radiance (cloud clearing) retrievals.

[1]The AQUA satellite launched May 4, 2002, carrying the AIRS and AMSU instruments, among others. AMSU–B was replaced with the Humidity Sounder–Brazil (HSB), which is identical to AMSU–B, except that it has no 89-GHz channel.

6.1 Introduction

The temperature profile retrieval performance of airborne and spaceborne atmospheric infrared sounders is significantly degraded by clouds. Methods for combining infrared and microwave radiance data to retrieve temperature profiles and clear-column infrared radiances in the presence of clouds have been proposed by several investigators (see [3, 56, 57], for example). Neural network estimators have been proven effective for retrieving temperature profiles from hyperspectral infrared measurements in clear air [58, 59]. In the following sections, novel methods are derived using combined infrared and microwave measurements together with neural network estimators to retrieve the temperature profile and clear-column infrared radiances in the presence of clouds.

6.2 Simulation of cloudy radiance observations

The `cld2lr` profile database constructed and used by the AIRS science team in approximately 1995 [60] was used to simulate radiances from the AIRS, AMSU–A, and AMSU–B instruments. The atmospheric profile statistics, cloud models, and surface model used in the `cld2lr` database are summarized, and the radiative transfer codes used to perform the radiance calculations are briefly described. Finally, the assumed instrument specifications are outlined.

6.2.1 Atmospheric profile data

A set of 12,000 atmospheric profiles (uniformly distributed seasonal and global coverage) was generated using a Numerical Weather Prediction (NWP) model [60]. The mean and standard deviation of the temperature, water vapor, and ozone profiles generated by the model are shown in Fig. 6-1.

6.2.2 Vertical cloud model

Clouds at two levels were introduced into every profile by relating the cloud-top altitude and fraction to a relative-humidity threshold. Histograms of the cloud fraction and cloud-top altitude for the two cloud layers are shown in Fig. 6-2. Cloud tops were assumed to

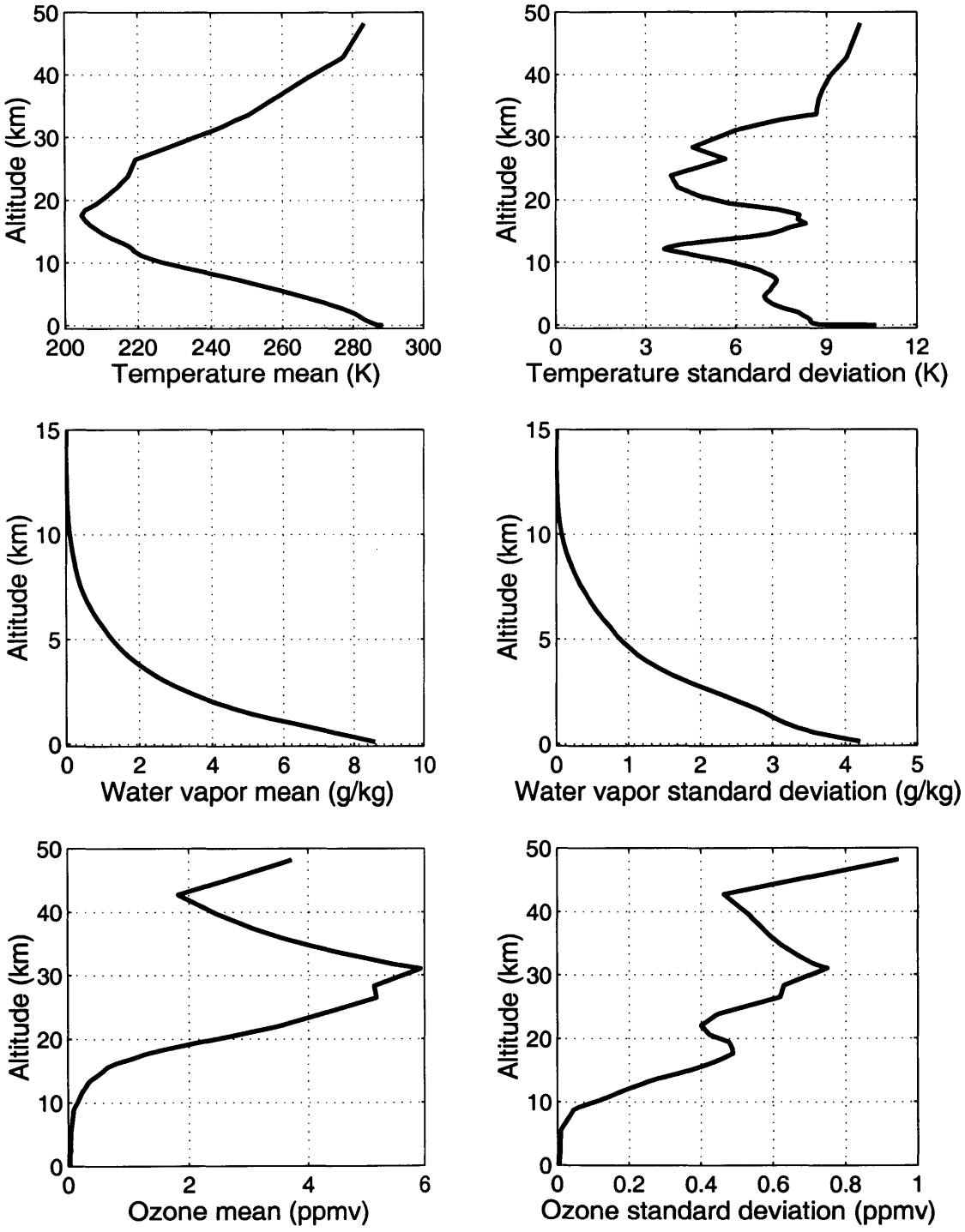

Figure 6-1: Temperature, water vapor, and ozone profile statistics for the `cld2lr` database, given in units of Kelvin, grams of water vapor per kilogram of dry air, and parts per million (volume), respectively.

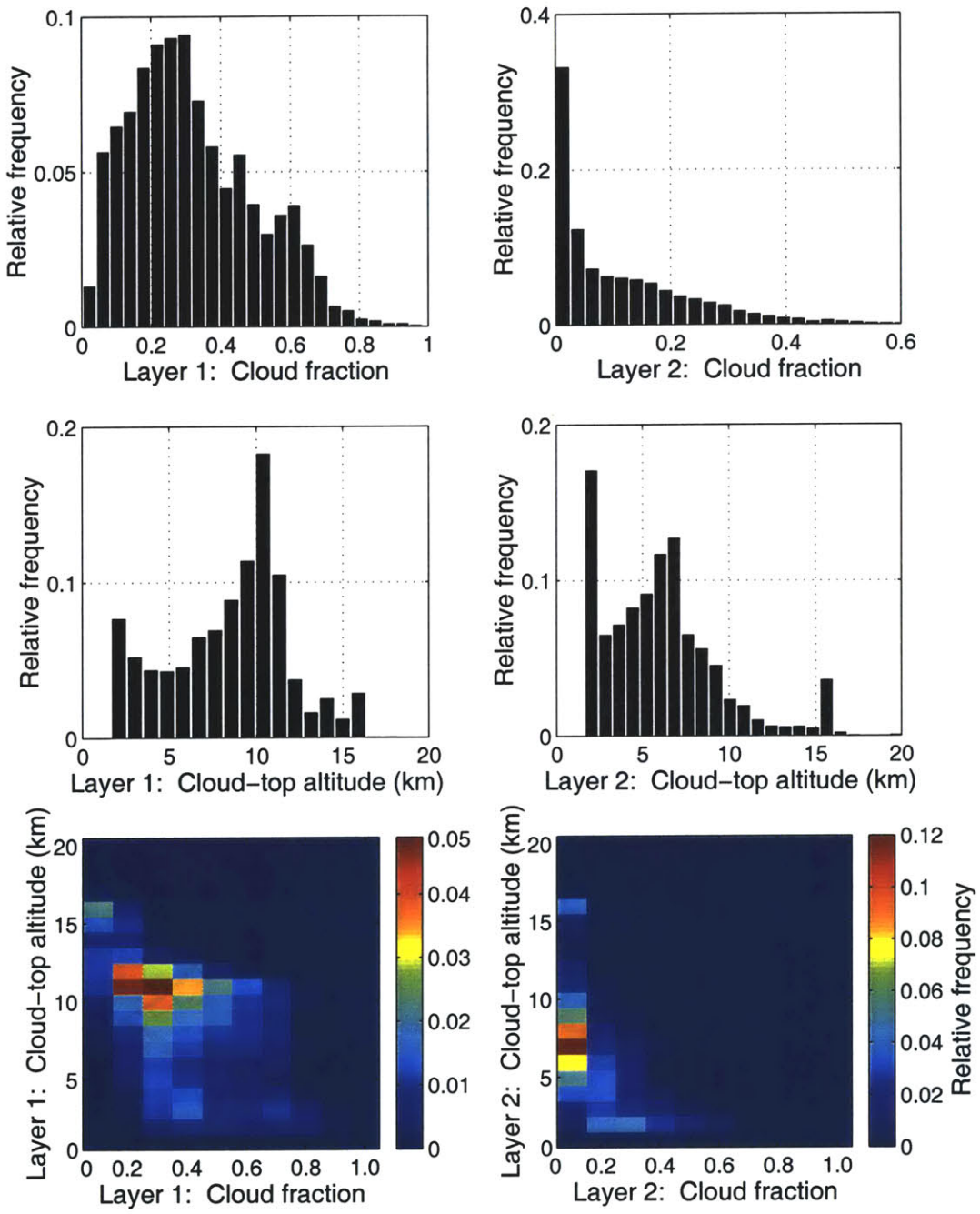

Figure 6-2: Histograms of cloud fraction and cloud-top altitude for each of two layers in the `cld2lr` cloud database.

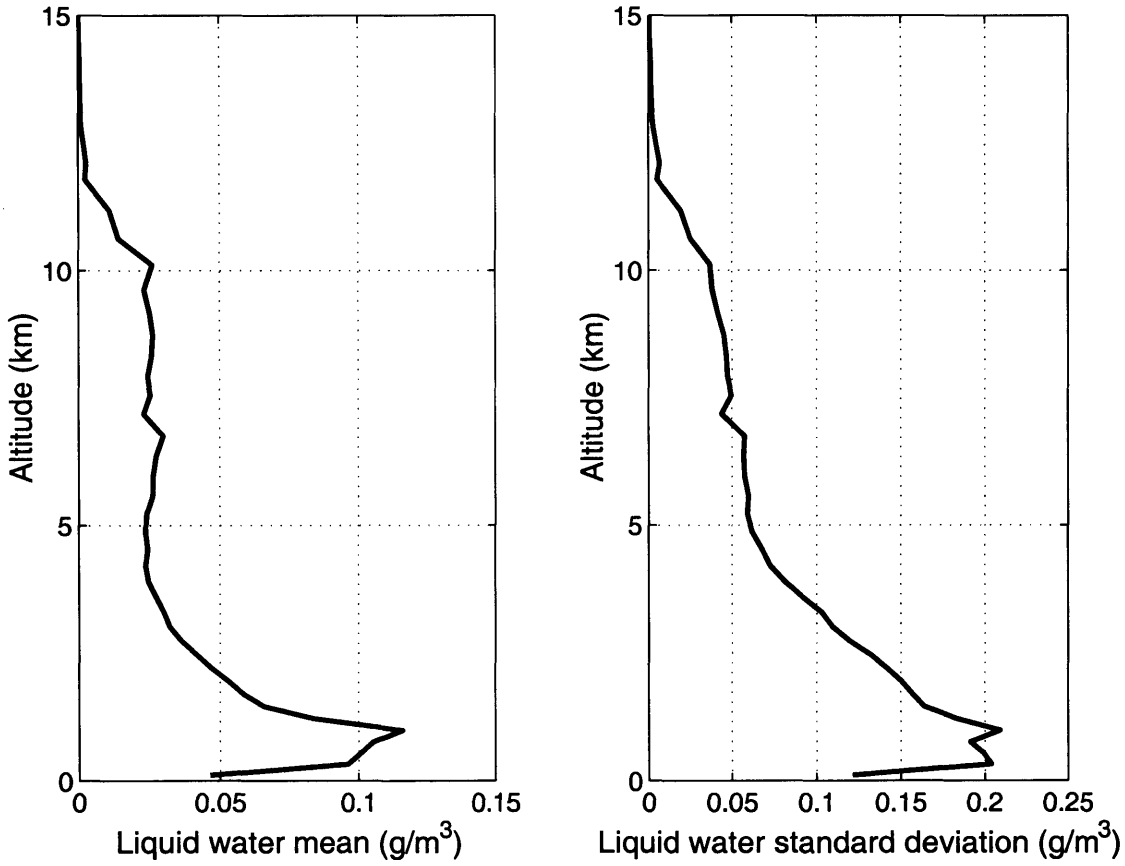

Figure 6-3: Cloud liquid water statistics for the `cld2lr` database.

be opaque to the infrared channels with a non-random emissivity of 0.9. Solar reflection was not modeled. Microwave channels were perturbed by clouds only through liquid water absorption; scattering was not modeled. Cloud liquid water statistics are shown in Fig. 6-3.

6.2.3 Horizontal cloud models

Two horizontal cloud models were used in the simulation analyses: "single-pixel" and "multiple-pixel". The single-pixel model assumes that the cloud and surface properties of each pixel are independent. For the multiple-pixel model, AIRS/AMSU–B radiance vectors were assembled into 3×3 "nontuples", where each nontuple shared the same two

cloud altitudes; the cloud altitudes were independent from cluster to cluster, and generally consistent with the associated atmosphere. The surface emissivity across each nontuple was fixed. A corresponding AMSU–A brightness temperature vector was generated for each AIRS/AMSU–B nontuple. The temperature and moisture profile over the nontuple was assumed to be homogeneous.

6.2.4 Surface model

Random surface emissivities were chosen for each profile, and were modeled as random variables with clipped-Gaussian pdf's (with the clip at $+1\sigma$). The mean and standard deviation assumed for the AIRS instrument were 0.975 and 0.025, respectively. The mean and standard deviation assumed for the AMSU instrument were 0.95 and 0.05, respectively. The surface emissivities of one instrument were assumed to be uncorrelated with any other; emissivities of each channel of a given instrument were assumed to be perfectly correlated.

6.2.5 Radiative transfer codes

The profiles in the `cld2lr` database with the assumptions outlined above were used to simulate radiance observations of the AIRS and AMSU instruments. The AIRS Fast Transmittance Codes [61, 62] were used to calculate clear-air transmittances for the infrared channels. The Rapid Transmittance Algorithm [63, 64, 65] was used to calculate transmittances for the microwave channels. The `airsbt` radiative transfer package [60] was used to calculate cloudy radiances for the single-pixel and multiple-pixel cloud models.

6.2.6 Instrument specifications

The AIRS instrument provides spectral coverage over the 3.74–4.61 μm, 6.20–8.22 μm, and 8.8–15.4 μm infrared bands, with approximately 2375 channels. The specified weighting function peak height and modeled (specified) measurement noise (NEΔT) for each channel are shown in Fig. 6-4. AIRS will provide a nominal footprint diameter of 13.5 km at nadir.

The AMSU–A instrument measures microwave radiation in 15 channels from 23.8–89.9 GHz at a nominal footprint diameter of 40.5 km at nadir. The AMSU–B instrument mea-

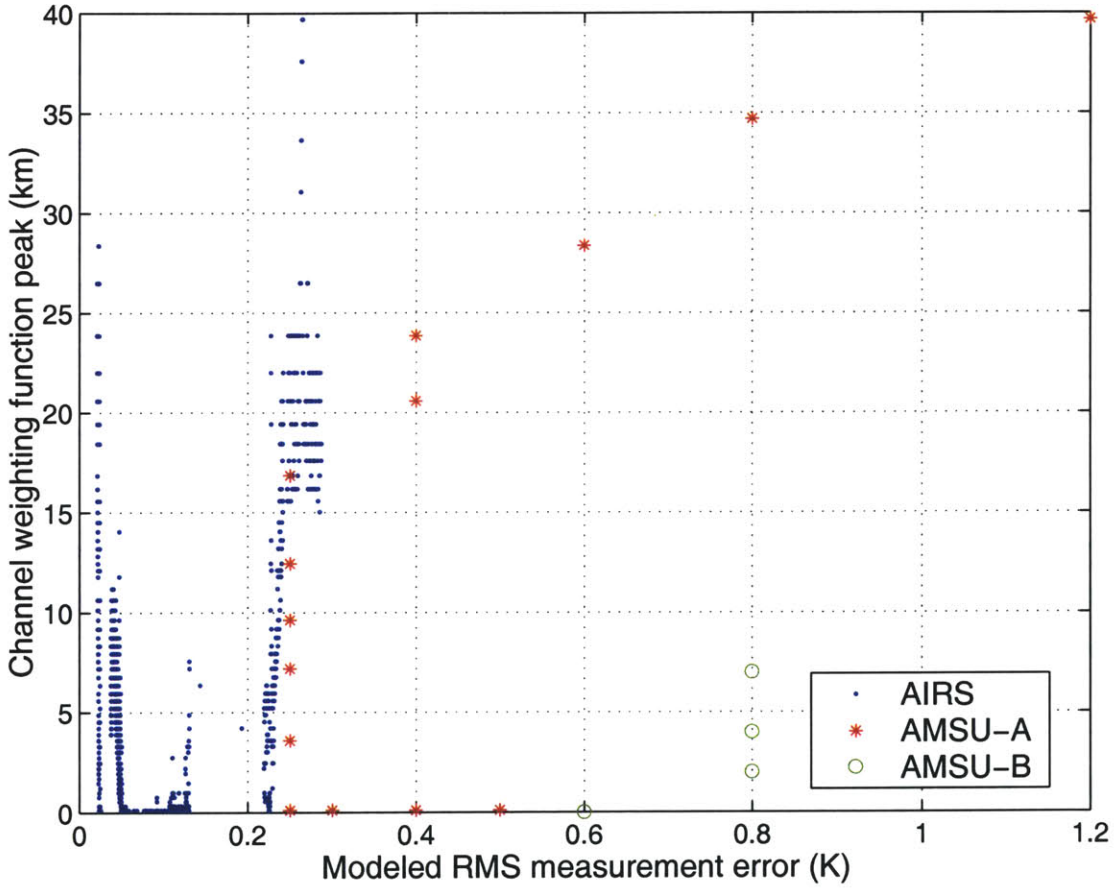

Figure 6-4: Weighting function peak heights and modeled RMS measurement error for the AIRS, AMSU–A, and AMSU–B instruments.

Table 6.1: Channel specifications for AMSU-A. The weighting function peak heights were calculated using the US standard atmosphere viewed at nadir over a nonreflective surface.

Channel No.	Center freq.(s) (MHz)	Bandwidth (MHz)	ΔT_{RMS} (K)	Weighting function peak height (km)
1	23,800 ± 72.5	2 × 125	0.30	0
2	31,400 ± 50	2 × 80	0.30	0
3	50,300 ± 50	2 × 80	0.40	0
4	52,800 ± 105	2 × 190	0.25	0
5	53,596 ± 115	2 × 168	0.25	4
6	54,400 ± 105	2 × 190	0.25	7
7	54,940 ± 105	2 × 190	0.25	10
8	55,500 ± 87.5	2 × 155	0.25	12
9	57,290.344 = f_{LO}± 87.5	2 × 155	0.25	17
10	f_{LO}± 217	2 × 77	0.40	21
11	f_{LO}± 322.2 ± 48	4 × 35	0.40	24
12	f_{LO}± 322.2 ± 22	4 × 15	0.60	28
13	f_{LO}± 322.2 ± 10	4 × 8	0.80	35
14	f_{LO}± 322.2 ± 4.5	4 × 3	1.20	40
15	89,000 ± 900	2 × 1000	0.50	0

sures microwave radiation in 5 channels from 88–191 GHz at a nominal footprint diameter of 13.5 km at nadir. The bandpass characteristics of the AMSU–A and AMSU–B instruments are summarized in Tables 6.1 and 6.2.

6.3 Principal components analysis of AIRS/AMSU

Principal components analysis (PCA) techniques were performed on AIRS/AMSU data to determine a more compact representation of the information contained in the radiance spectra. This reduces the number of inputs to the retrieval operator, and in the case of a neural network, provides a more stable network and significantly reduces training time because fewer nodes are needed. In this section we compute the noise-adjusted principal components (and associated eigenvalues) of clear and cloudy radiance data, the noise-adjusted principal components of the infrared radiance perturbations due to clouds, and the projected (temperature) principal components of clear and cloudy radiance data.

Table 6.2: Channel specifications for AMSU–B. The weighting function peak heights were calculated using the US standard atmosphere viewed at nadir over a nonreflective surface.

Channel No.	Center freq. (GHz)	Bandwidth (GHz)	ΔT_{RMS} (K)	Weighting function peak height (km)
1	89 ± 1	2×1	0.6	0
2	150 ± 0.9	2×1	0.6	0
3	183.31 ± 1	2×0.5	0.8	2
4	183.31 ± 3	2×1	0.8	4
5	183.31 ± 7	2×2	0.8	7

6.3.1 NAPC of clear and cloudy radiance data

The 2371 AIRS radiances were converted from spectral intensities to brightness temperatures using Eq. 2.48, and were concatenated with the 20 microwave brightness temperatures into a single vector R of length 2391. The noise-adjusted principal components were computed as follows:

$$P_{\mathrm{NAPC}} = \mathbf{Q}^T R, \tag{6.1}$$

where $\mathbf{Q}$ are the eigenvectors of $\widetilde{\mathbf{C}}_{XX} + \mathbf{I}$, sorted in descending order by eigenvalue. $\widetilde{\mathbf{C}}_{XX}$ is the prewhitened covariance matrix given by Eq. 3.14. The eigenvalues corresponding to the top 100 NAPC's are shown in Fig. 6-5 for simulated clear-air and cloudy data. Also shown are scatterplots of the first three NAPC's.

The eigenvalues of the 90 lowest-order terms are very similar. The principal differences occur in the three highest order terms, which are dominated by channels with weighting function peaks in the lower part of the atmosphere. The eigenvalues associated with the clear-air and cloudy NAPC's cluster into roughly five groups: 1, 2–3, 4–9, 10–11, 12–100. The first 11 NAPC's capture 99.96 percent of the total radiance variance for both the clear-air and cloudy data. The top three NAPC's of both clear-air and cloudy data appear to be jointly-Gaussian to a close approximation, with the exception of clear-air NAPC 1 versus NAPC 2.

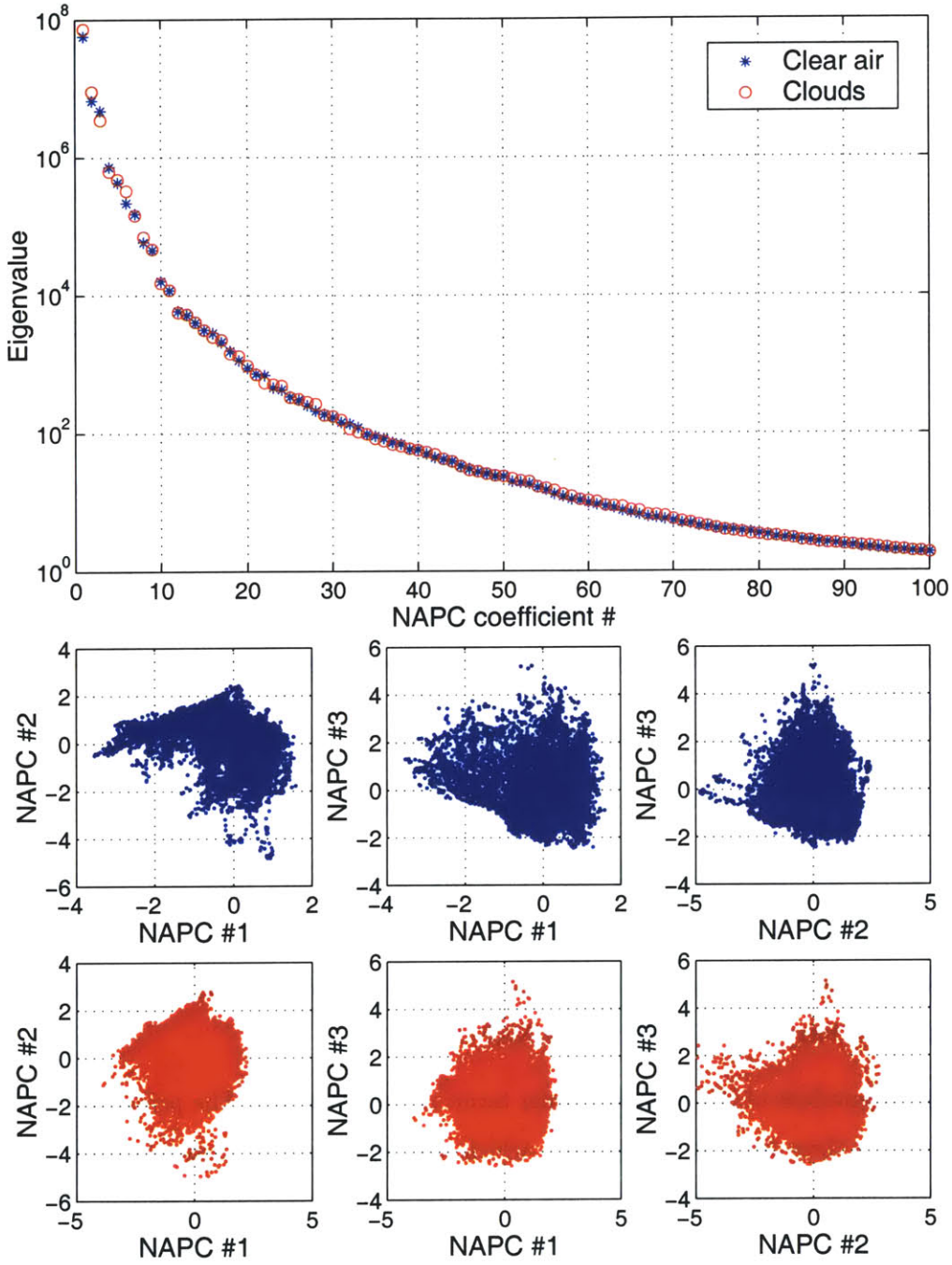

Figure 6-5: Noise-adjusted principal components transform analysis of clear and cloudy simulated AIRS/AMSU data. The top plot shows the eigenvalues of each NAPC coefficient for clear and cloudy data. The middle row presents scatterplots of the three clear-air NAPC coefficients with the largest variance (shown normalized to unit variance). The bottom row presents scatterplots of the three cloudy NAPC coefficients with the largest variance (shown normalized to unit variance).

6.3.2 NAPC of infrared cloud perturbations

Define the infrared cloud perturbation ΔR_{IR} as

$$\Delta R_{\mathrm{IR}} \triangleq R_{\mathrm{IR}}^{clr} - R_{\mathrm{IR}}^{cld}, \tag{6.2}$$

where R_{IR}^{clr} is the clear-air infrared brightness temperature and R_{IR}^{cld} is the cloudy infrared brightness temperature. The NAPC's of ΔR_{IR} were calculated using the method described above. The results are shown in Fig. 6-6.

The six highest-order NAPC's of ΔR_{IR} capture approximately 99.96 percent of the total cloud perturbation variance, which suggests that there are more degrees of freedom in the atmosphere than there are in the clouds. Furthermore, there is significant crosstalk between the cloud perturbation and the underlying atmosphere, and this crosstalk is highly nonlinear and non-Gaussian. Evidence of this can be seen in the scatterplot of NAPC #1 versus NAPC #2, shown in the lower left corner of Fig. 6-6. The temperature weighting functions of NAPC #1 and NAPC #1 are shown in Fig. 6-7. NAPC #1 consists primarily of surface channels and NAPC #2 consists primarily of channels that peak near 3–6 kilometers and channels that peak near the surface. Therefore, NAPC #1 is sensitive principally to the overall cloud amount, while NAPC #2 is also sensitive to cloud altitude.

6.3.3 PPCA of clear and cloudy radiance data

The projected principal components transform discussed in Chapter 3 was used to identify temperature information contained in the clear and cloudy radiances. Figure 6-8 shows the mean temperature profile retrieval error for the reduced-rank regression operator given in Eq. 3.42 as a function of rank (the number of PPC coefficients retained) for clear-air and cloudy radiance data. Both curves have asymptotes near 15 coefficients.

6.4 Single-pixel temperature retrievals

The `cld2lr` profile/radiance data set was used to derive and evaluate a retrieval of the average tropospheric temperature in 1-km thick slabs (vertically). Retrievals were con-

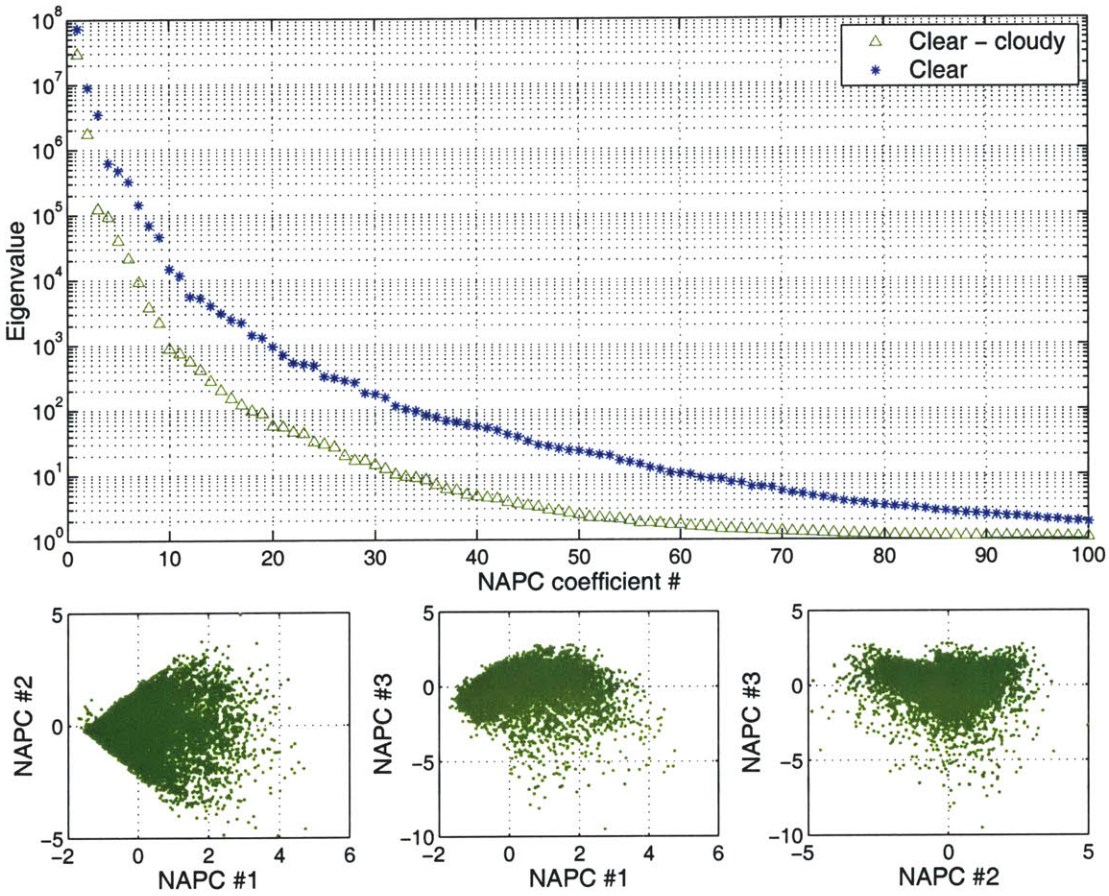

Figure 6-6: Noise-adjusted principal components transform analysis of the cloud impact (clear radiance - cloudy radiance) for simulated AIRS data. The top plot shows the eigenvalue of each NAPC coefficient of cloud impact, along with the NAPC coefficients of clear-air data (shown in Fig. 6-5). The bottom row presents scatterplots of the three cloud-impact NAPC coefficients with the largest variance (shown normalized to unit variance).

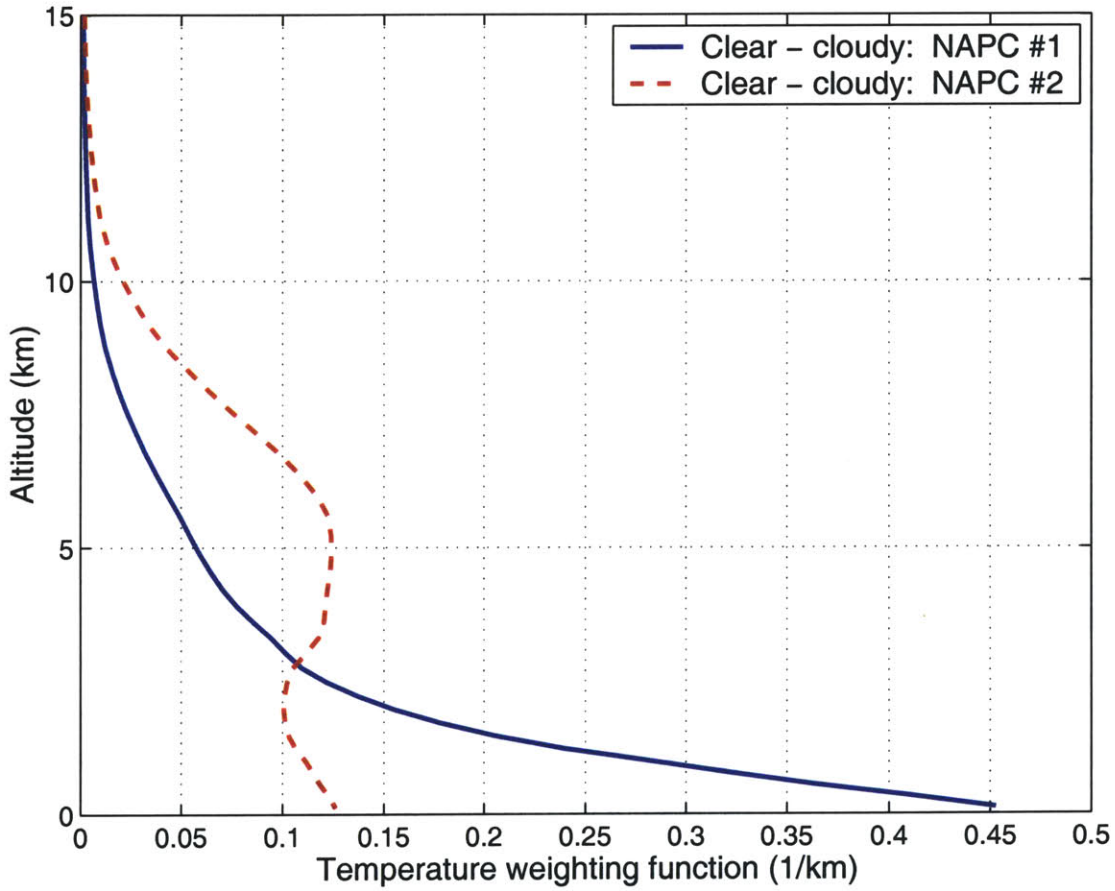

Figure 6-7: Temperature weighting functions of the first two noise-adjusted principal components of AIRS cloud perturbations (clear radiance - cloudy radiance).

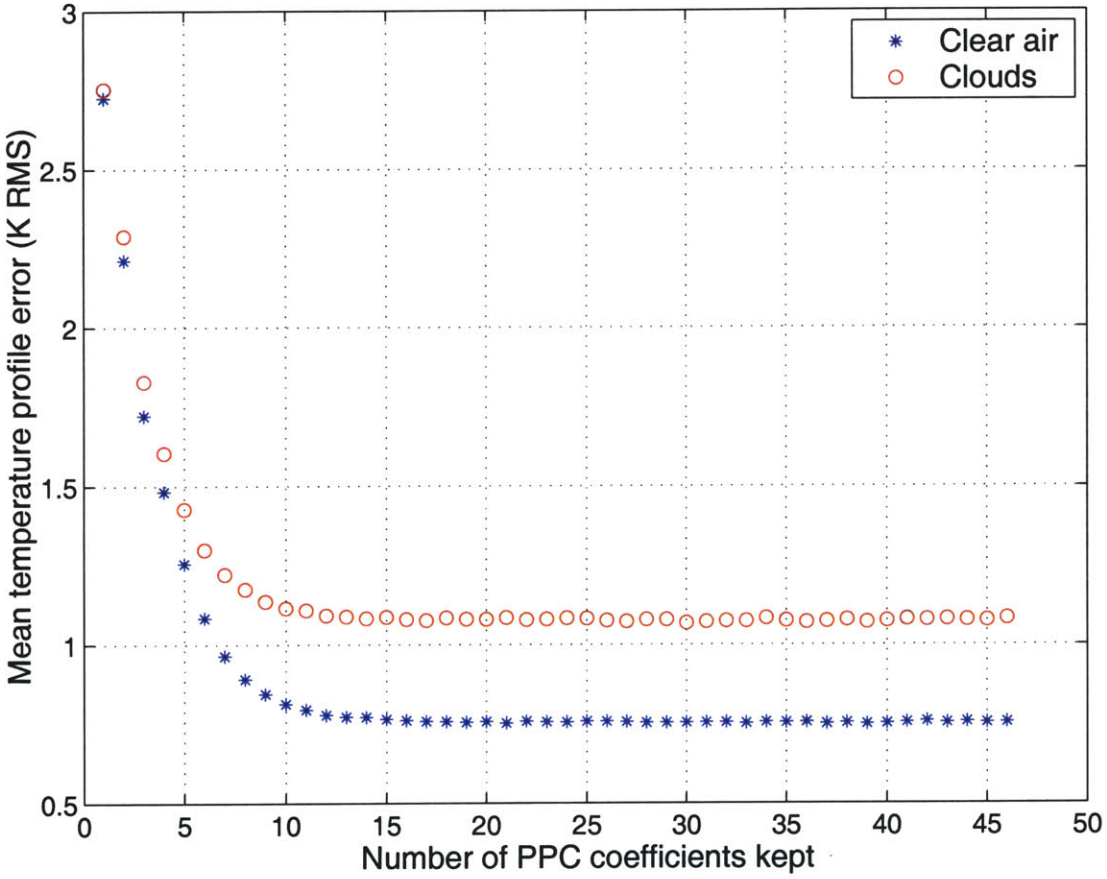

Figure 6-8: Projected principal components transform analysis of clear and cloudy simulated AIRS/AMSU data. The mean temperature profile retrieval error (K RMS) is shown as a function of the number of PPC coefficients used in a linear regression for simulated clear and cloudy data.

structed using LLSE and neural network techniques.

6.4.1 Training and validation sets

The `cld2lr` data set was divided into a "training set", which was used to calculate statistical information and train the neural network, and a separate "validation set", which was used to test the performance of the retrievals. The validation set was constructed by choosing 2,000 profiles, evenly-spaced throughout the dataset. The remaining 10,000 profiles comprised the training set.

6.4.2 LLSE retrieval

The LLSE retrieval discussed in Chapter 3 was to estimate the temperature profile in the lower 15 kilometers of the atmosphere, in approximately 0.33-km increments. These estimates were then smoothed to obtain the 1-km thick layer averages. The infrared radiance data was converted into units of brightness temperature before the sample covariance matrices were computed from the training set. All instrument noise was assumed to be uncorrelated.

6.4.3 Neural network retrieval

A neural network was trained to estimate the temperature profile in the lower 15 kilometers of the atmosphere. Five separate networks were trained, each of which estimated the temperature in a 3-km slab of atmosphere, in approximately 0.33-km increments. The inputs and topology of the five networks were identical.

Neural network inputs

The AIRS and AMSU observation vectors were consolidated into one vector, and the PPC transform was performed. The 30 highest-order coefficients were retained. These 30 coefficients were normalized to unit-variance and input to the neural networks. The instrument noise was regenerated for each epoch of training.

Network topology

The neural network was comprised of one hidden layer with 20 nodes, and one linear output layer with approximately six outputs. These values were chosen by considering the degrees of freedom in the radiances (approximately 20, see Fig. 3-2) and the number of training vectors. Empirical analyses indicated that the number of training vectors should exceed the number of degrees of freedom in the neural network (number of inputs *times* number of hidden nodes + number of output nodes) by at least a factor of ten. The following sigmoidal activation function was used for the hidden layer:

$$f(x) = \frac{2}{1 + e^{-2x}} - 1. \tag{6.3}$$

Training algorithm

The Levenberg-Marquardt training algorithm was used. Training times for a single network averaged two hours on an AMD Athlon 1-GHz PC. A typical training curve is shown in Fig. 6-9.

6.4.4 Error analysis

The RMS errors for the LLSE and neural network retrievals are shown in Fig. 6-10 for clear and cloudy atmospheres. The neural network estimator significantly outperforms the LLSE in both cases. The sensitivity of the retrieval to instrument noise is examined by repeating the retrieval with instrument noise set to zero. The difference in retrieval errors (with and without noise) is shown in the first panel of Fig. 6-11. The atmospheric contribution to the retrieval error (i.e., the noise-free retrieval error) is shown in the second panel of Fig. 6-11. Finally, the differences (net minus LLSE) in error contributions for the two methods are shown in the third panel of Fig. 6-11. It is noteworthy that the neural net is a much better filter of instrument noise than is the LLSE.

As a final test of sensitivity to instrument noise, the LLSE and neural network retrievals were repeated while varying the instrument noise between 10 percent and 1000 percent of its nominal value. The resulting retrieval errors are shown in Fig. 6-12. The neural network retrieval is significantly less sensitive to instrument noise than is the LLSE retrieval.

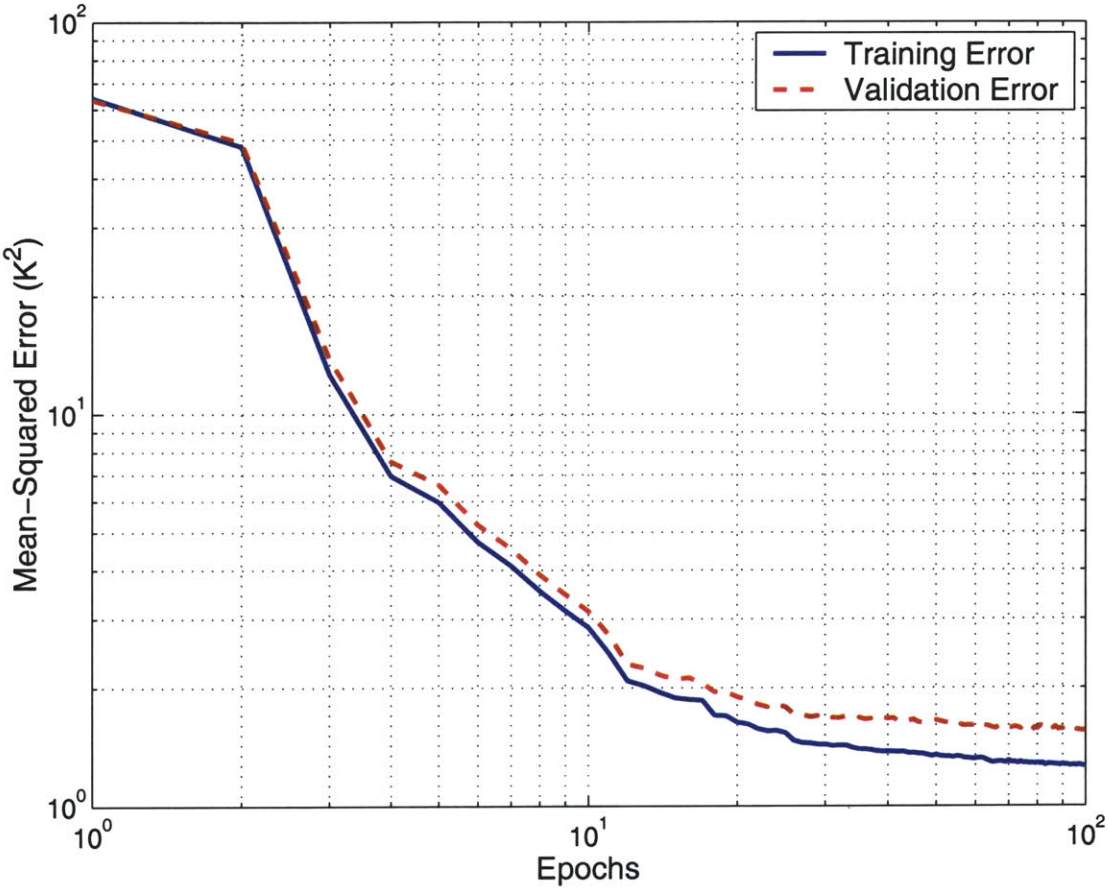

Figure 6-9: Example learning curve for the neural network used to perform single-pixel temperature profile retrievals. The Levenberg-Marquardt training algorithm was used.

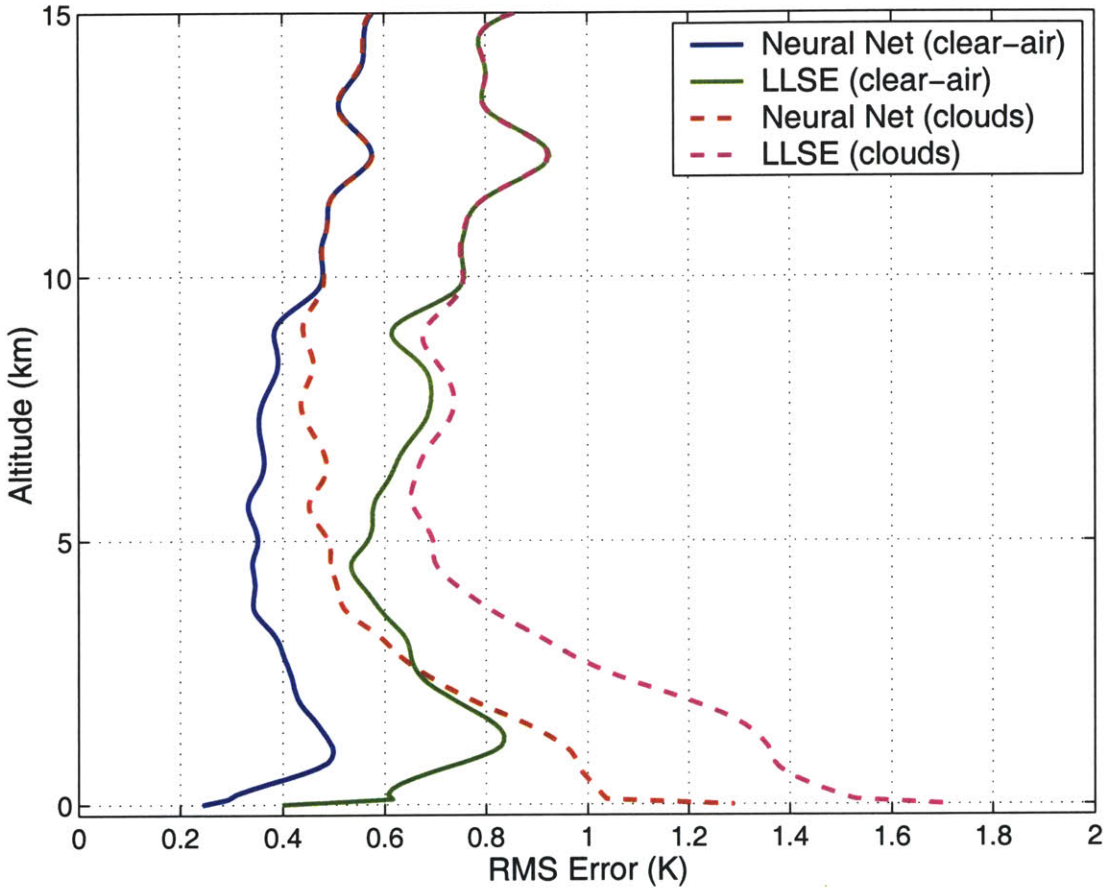

Figure 6-10: RMS temperature profile retrieval error for the neural network and LLSE estimators. Results for clear air and clouds are shown. Surface emissivities were modeled as random variables with clipped-Gaussian pdf's; mean = 0.975, standard deviation = 0.025 (AIRS); mean = 0.95, standard deviation = 0.05 (AMSU).

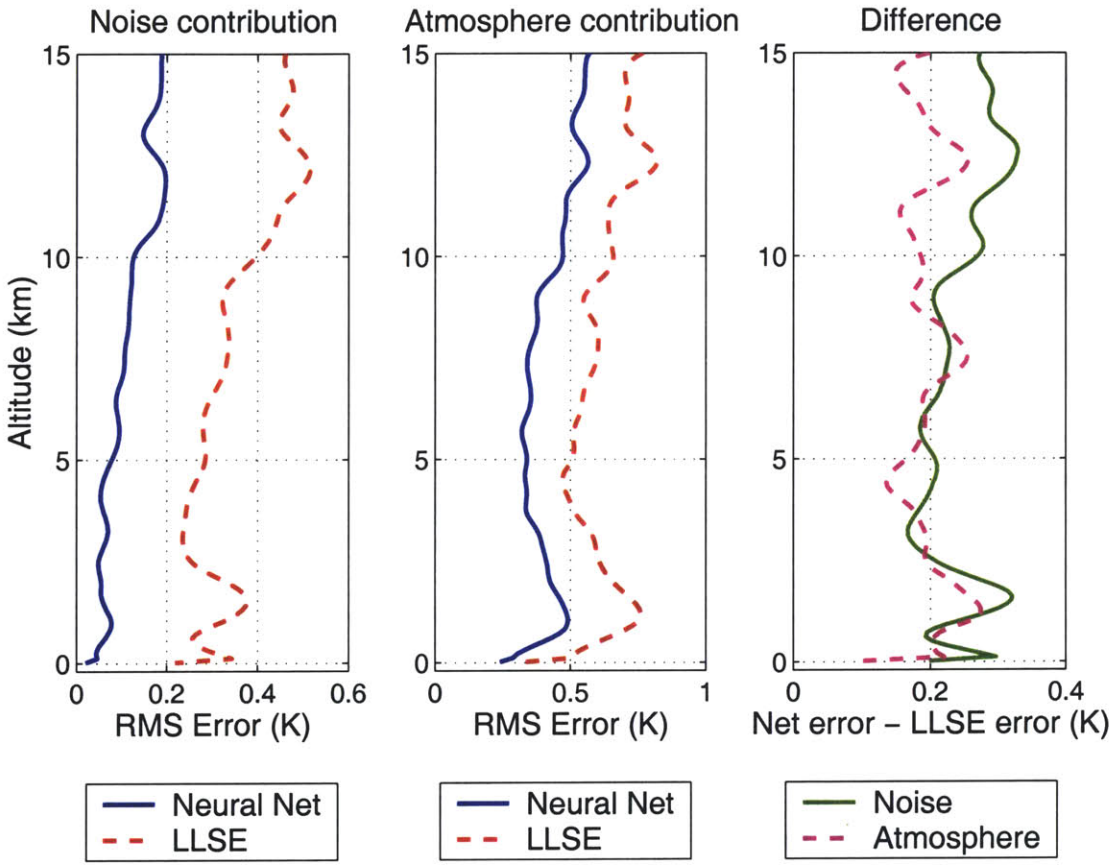

Figure 6-11: Contributions to temperature profile retrieval error due to measurement noise and atmospheric noise for the neural network and LLSE estimators.

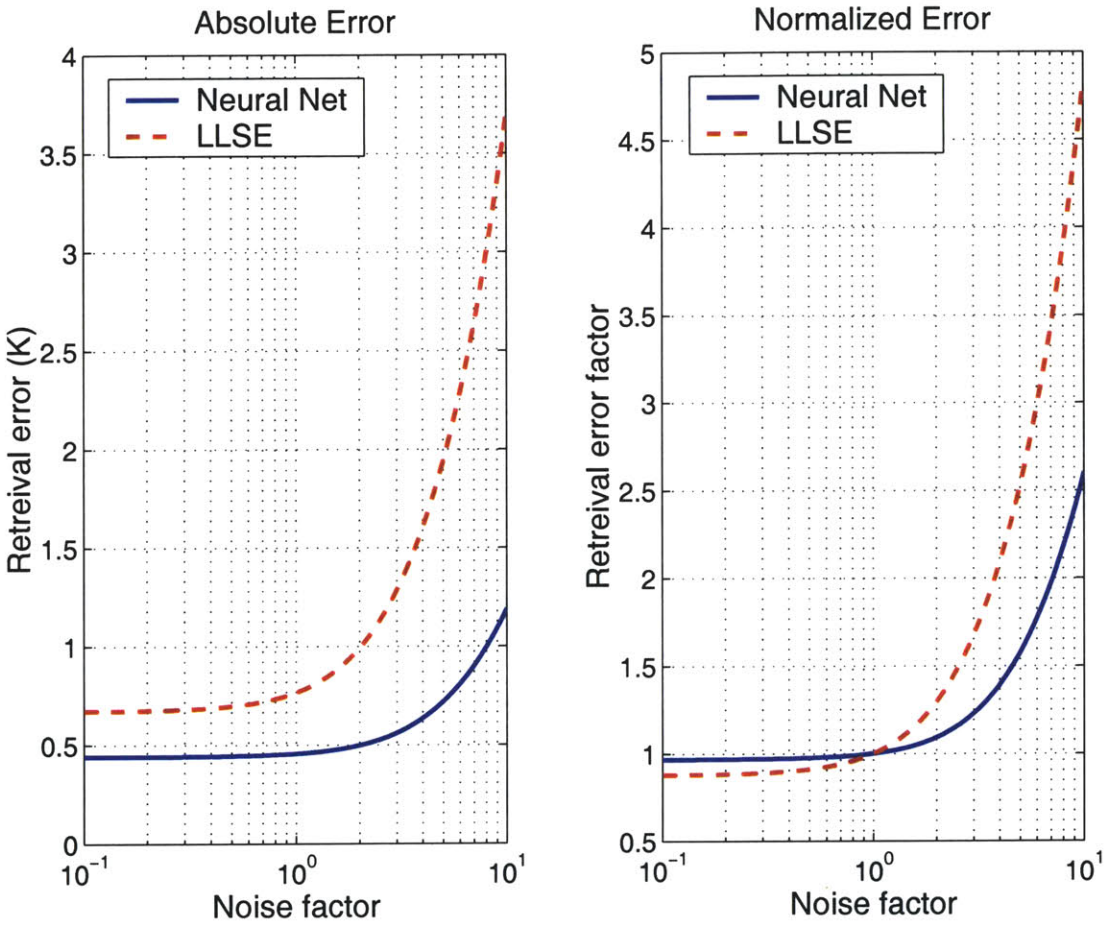

Figure 6-12: Sensitivity of temperature profile retrieval to measurement noise. The mean RMS error over 15 km is shown as a function of the noise amplification factor (nominal noise shown in Fig. 6-4).

6.5 Cloud Flagging

It is often useful to estimate a (possibly channel-dependent) measure of infrared "cloud-impact", which can subsequently be used as an input to a cloud clearing routine. We henceforth refer to this quantity as a flag. While "flags" are generally regarded as "0" or "1", we allow the single-pixel cloud flags to vary between $\pm\infty$. Consider the following example:

$$f(\nu) \equiv M_{obs}(\nu) - \hat{M}(R_{obs}(\forall\nu)) \ \ (K) \tag{6.4}$$

where $f(\nu)$ is a cloud flag for infrared radiance $R_{obs}(\nu)$, $M_{obs}(\nu)$ is an observed microwave "pseudochannel" (linear combination of microwave channels, for example) peaking near $R_{obs}(\nu)$, and $\hat{M}(R_{obs}(\forall\nu))$ is an estimate of $M_{obs}(\nu)$ using all infrared channels. One possible choice for the microwave pseudochannels is the linear combination that yields the sharpest (in the Backus-Gilbert [28] sense) temperature weighting function with the same peak as the companion infrared channel. Fig. 6-13 shows plots of "true" cloud impact (defined as $R^{clr} - R^{cld}$) versus the Backus-Gilbert flag for nine different nontuples (a 4.19-μm AIRS channel peaking near 5 km is shown). The relationship for a given nontuple is nearly linear, and a reasonable cloud-clearing methodology would be to derive a regression line and extrapolate back to an estimate of $R^{clr}(\nu)$.

6.6 Statistical Cloud Clearing

The problem of estimating an infrared clear-column radiance vector from one or more cloud-impacted radiance observations has been addressed by a number of investigators (see [66] and [67], for example). Various strategies have been suggested including spatial processing, spectral processing, and a combination of both. We examine the latter approach here using only statistical methods based on a large and simulated global data set, the `cld2lr` database discussed earlier.

In the previous example, statistics are only used to derive the cloud flags for each nontuple separately. One could construct an ensemble of nontuples and derive a statistical

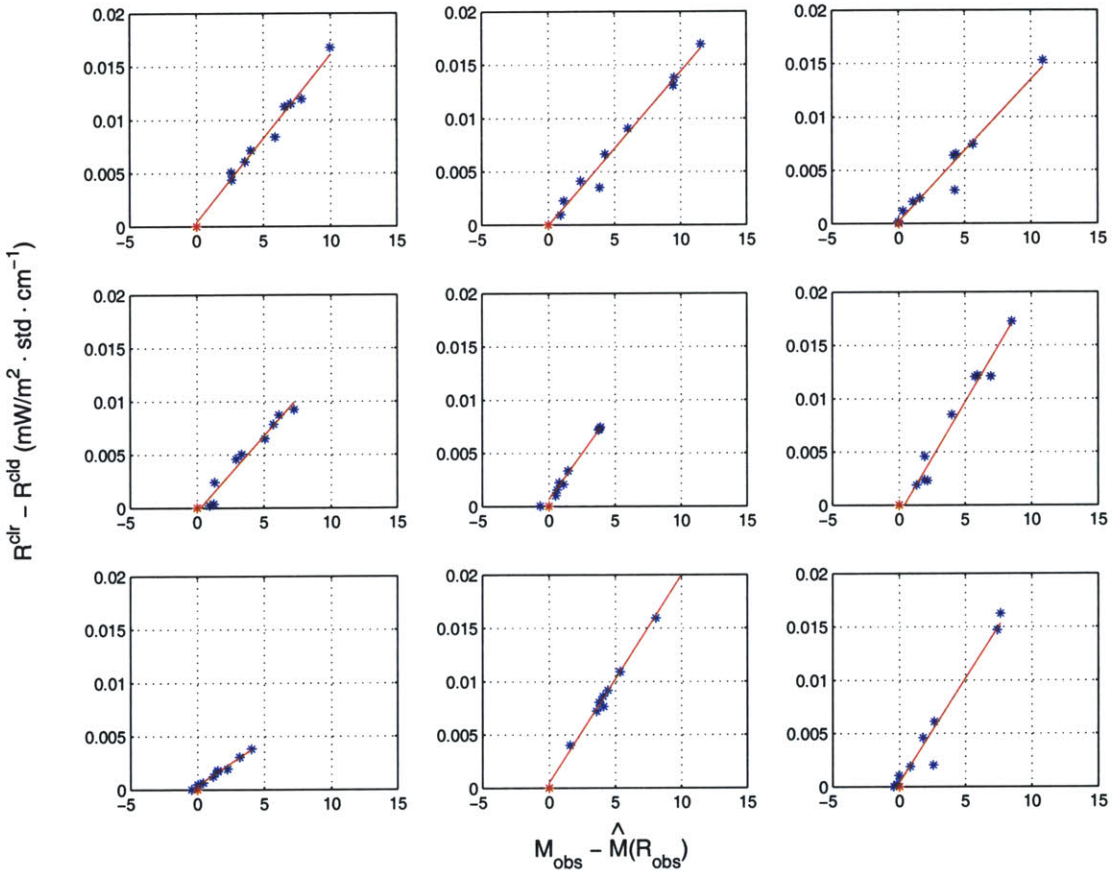

Figure 6-13: Regressions of 4.19-μm radiances R_{obs} and microwave-derived cloud flags in degrees Kelvin (see text). Nine representative pixel clusters are shown.

Table 6.3: Input channel configuration for the neural network 3×3 radiance cloud-clearing algorithm.

	Warmest pixel	Each of eight remaining pixels (difference from warmest pixel)
AIRS	20 NAPC's	4 NAPC's
AMSU–B	All 5 channels	All 5 channels
AMSU–A	8 most transparent channels $\neq$ 89 GHz	

relationship between the clear-column infrared radiance and the cloud-impacted nontuple/microwave observations. Two statistical estimators were used here: linear regression and a neural network. The estimators were constructed using a training set of approximately 5000 nontuples and a validation set of approximately 1000 nontuples.

A neural network with two hidden sigmoidal layers (30 and 20 nodes, respectively) and one linear output layer was trained to estimate 30 NAPC's of the clear-column AIRS radiance from 105 inputs, derived as follows. Several 10-μm AIRS channels were used to identify the warmest of nine pixels. 20 noise-adjusted principal components (NAPC's) [26] of the 2371-element AIRS radiance vector corresponding to this pixel were used. The five AMSU–B channels corresponding to this pixel were also used. The remaining eight pixels were expressed in terms of the difference from the warmest pixel. Four NAPC's of the infrared difference for each of eight pixels were used. All five AMSU–B differences were used. Finally, the eight most-transparent AMSU–A channels were used (not including the 89-GHz channel, which is duplicated on AMSU–B). The input channel configuration is summarized in Table 6.3. The neural network was trained to estimate 30 NAPC's of the clear-column AIRS radiance.

Representative results are shown in Fig. 6-14, Fig. 6-15, and Fig. 6-16. RMS cloud clearing errors are shown as a function of channel weighting function peak height. For Fig. 6-14 and Fig. 6-15, the channels with the lowest RMS error at each height were chosen; the resulting curves therefore establish a lower bound on RMS errors using these methods. In Fig. 6-16, the RMS cloud clearing error for all channels is shown for the neural network clearing method. The sensitivity of error performance to weighting function width is examined in Section 6.7.2.

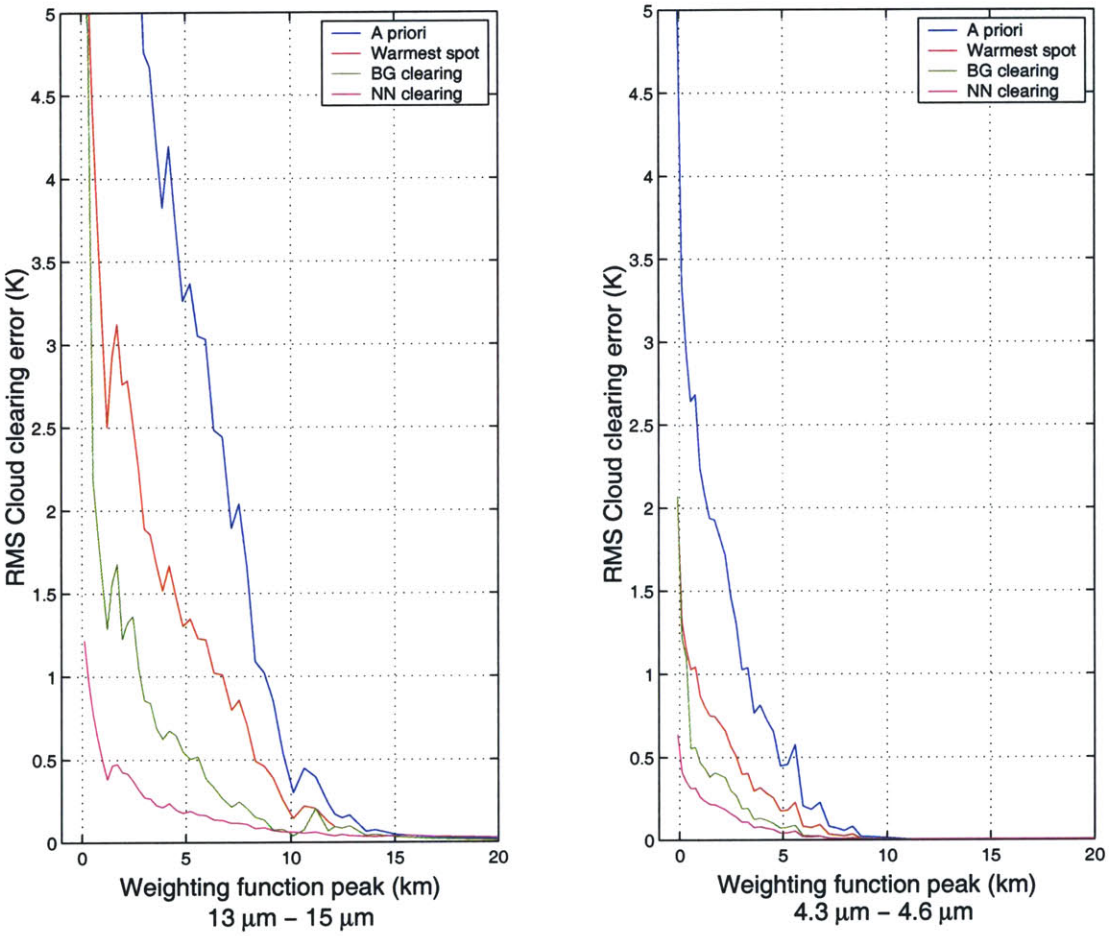

Figure 6-14: Cloud clearing performance bounds. Results from four methods are shown (from left to right): neural network, Backus-Gilbert nine-point regression, warmest spot, and *a priori*.

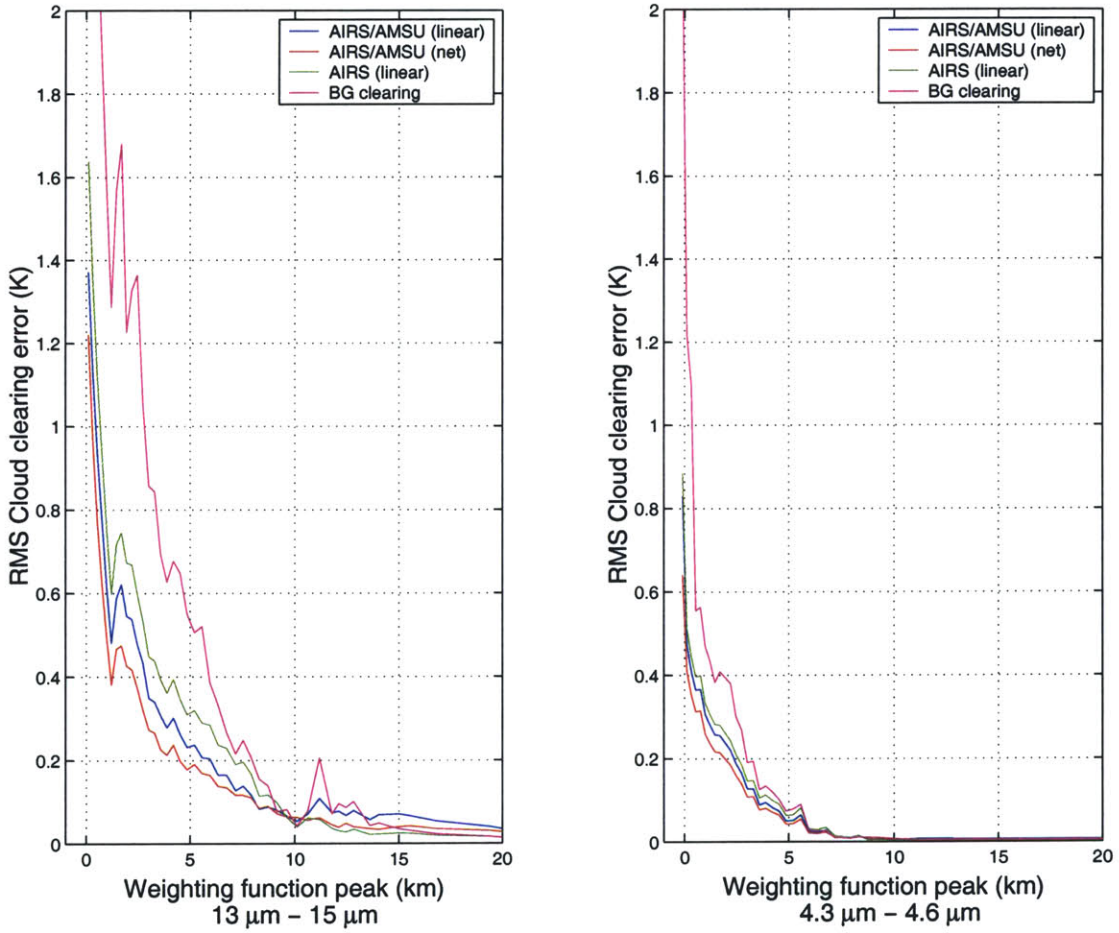

Figure 6-15: Cloud clearing performance bounds. Results from four methods are shown (from left to right): neural network, linear, linear (IR only), and Backus-Gilbert.

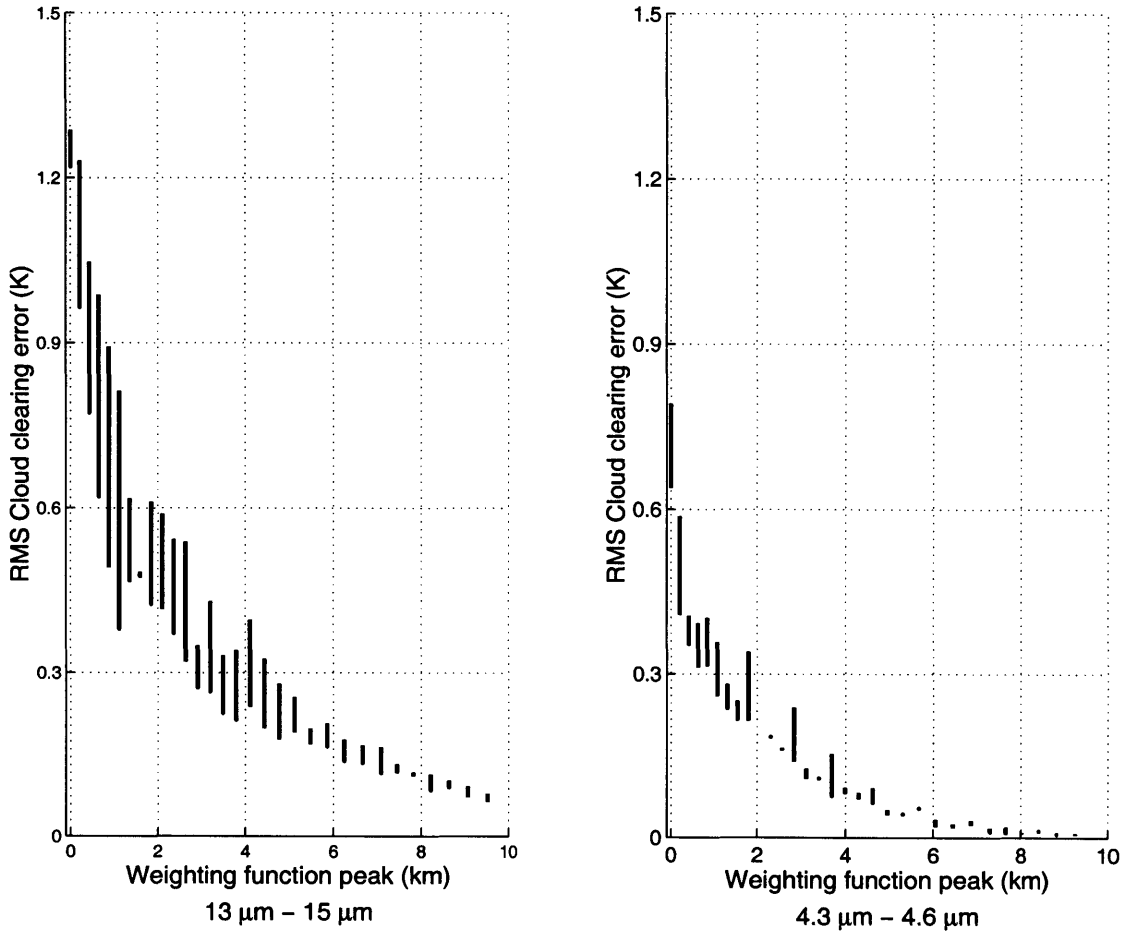

Figure 6-16: Cloud clearing performance of the neural network clearing method, where the RMS error range of each altitude is shown.

The cloud-clearing error associated with channels near 15 μm is higher than the cloud-clearing error associated with channels near 4 μm. This is because the *a* priori cloud variance is higher near 15 μm as shown in Fig. 6-14 and demonstrated for CAMEX-3 NAST-I observations (see Fig. 8-1). If the cloud-clearing error is expressed in terms of a ratio to the *a* priori, the error near 4 μm is greater than the error near 15 μm by approximately a factor of two at altitudes below eight kilometers.

Several features of Fig. 6-15 are noteworthy. The neural network improved on the performance of the linear regression by approximately 0.2 K in the most transparent channels of the shortwave band. The removal of microwave channels degraded performance by approximately 0.25 K in the most transparent channels of the longwave band. The use of statistics significantly improves performance relative to the Backus-Gilbert clearing method.

This is expected, because the Backus-Gilbert method is a subset of the linear regression method in two ways. First, the linear combination of microwave channels which give the sharpest peak is non-optimal statistically (see Section 3.3.2. Second, a better estimate of the radiance statistics is obtain by computing the sample covariances over an *e*nsemble of nontuples, rather than for each nontuple individually, as is done in the Backus-Gilbert regression method.

As a final point, it should be noted that the relationship between the cloud flags and the infrared radiance perturbation becomes more nonlinear near the surface, due to cross-talk from microwave surface emissivity. The neural network method is able to represent this nonlinearity, and outperform the linear regression method near the surface.

6.7 Sensitivity analyses

6.7.1 Surface emissivity

The impact of random surface emissivity was studied by comparing the performance of the nontuple linear statistical cloud clearing method for three combinations of surface emissivity randomness (for sets of nontuples, the surface emissivity within each nontuple is identical, as before): 1) AIRS and AMSU surface emissivity both random, with statistics given in Section 6.2.4, 2) AIRS surface emissivity fixed at 0.9 (nonrandom) and AMSU

surface emissivity random (as before), and 3) AIRS surface emissivity random (as before) and AMSU surface emissivity fixed at 0.9 (nonrandom). The best results were obtained when both the infrared and microwave surface emissivities were fixed at 0.9. The microwave surface emissivity impacted performance the most; cloud clearing errors increased by 0.75 K in the most transparent long-wave channels if the random microwave surface emissivity model was used instead of the fixed 0.9 model. Variable infrared surface emissivities had very little impact on performance.

6.7.2 Weighting function width

The relationship between weighting function width and cloud clearing error is also of interest. Figure 6-18 shows the cloud clearing errors for channels peaking near 1.25 km as a function of weighting function width (second moment, normalized by the second moment of a nominal channel peaking near 1.25 km). Channels with sharp weighting functions were more difficult to clear than channels with broad weighting functions because the high-altitude content of sharp function is less.

6.8 Summary

Simulated AIRS/AMSU radiances were used to develop and evaluate methods for retrieving the temperature profile in the presence of clouds and for flagging and clearing cloud-impacted infrared radiances. The neural network single-pixel temperature retrieval outperformed the linear statistical method by over 30 percent (K RMS) for all levels of the atmosphere. For nontuple cloud clearing, both linear and nonlinear statistical methods yielded results superior to flag-based regression methods, such as illustrated in Fig. 6-13. Combining microwave and infrared radiances at the pixel level improved performance, although microwave surface emissivity uncertainties were a key contributor to residual errors. The neural network nontuple cloud-clearing algorithm yielded the best results: residual RMS radiance errors for broad weighting functions peaking below 10 km of 0.1–1.2K and 0.1–0.7K for the 13–15 μm and 4.3–4.6 μm bands, respectively; no nontuples were rejected due to clouds. These errors approximately double for the sharpest weighting functions

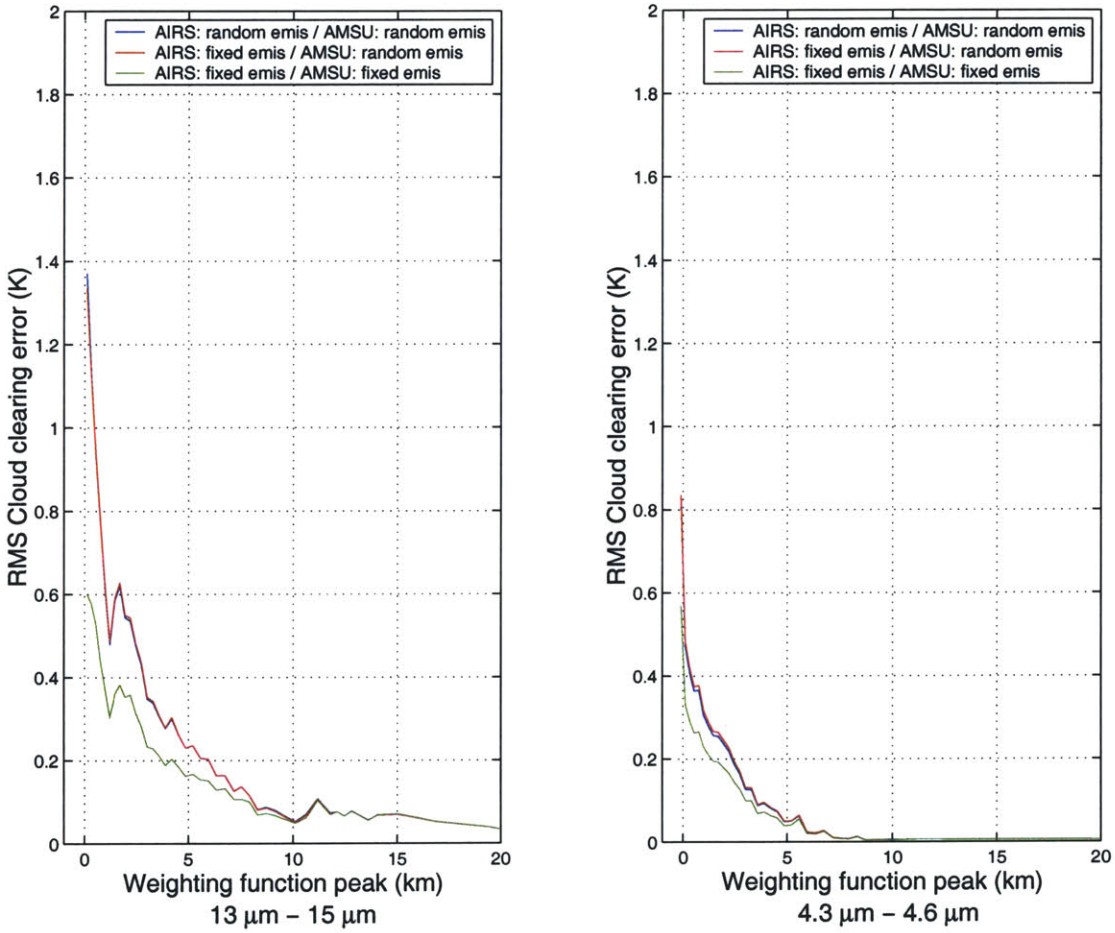

Figure 6-17: Impact of random surface emissivity on cloud clearing performance. The curve on the left (on each graph) corresponds to fixed microwave and infrared surface emissivity. The remaining two curves (on each graph) are nearly indistinguishable, and correspond to random microwave surface emissivity plus: (curve 2) fixed infrared emissivity, and (curve 3) random infrared surface emissivity.

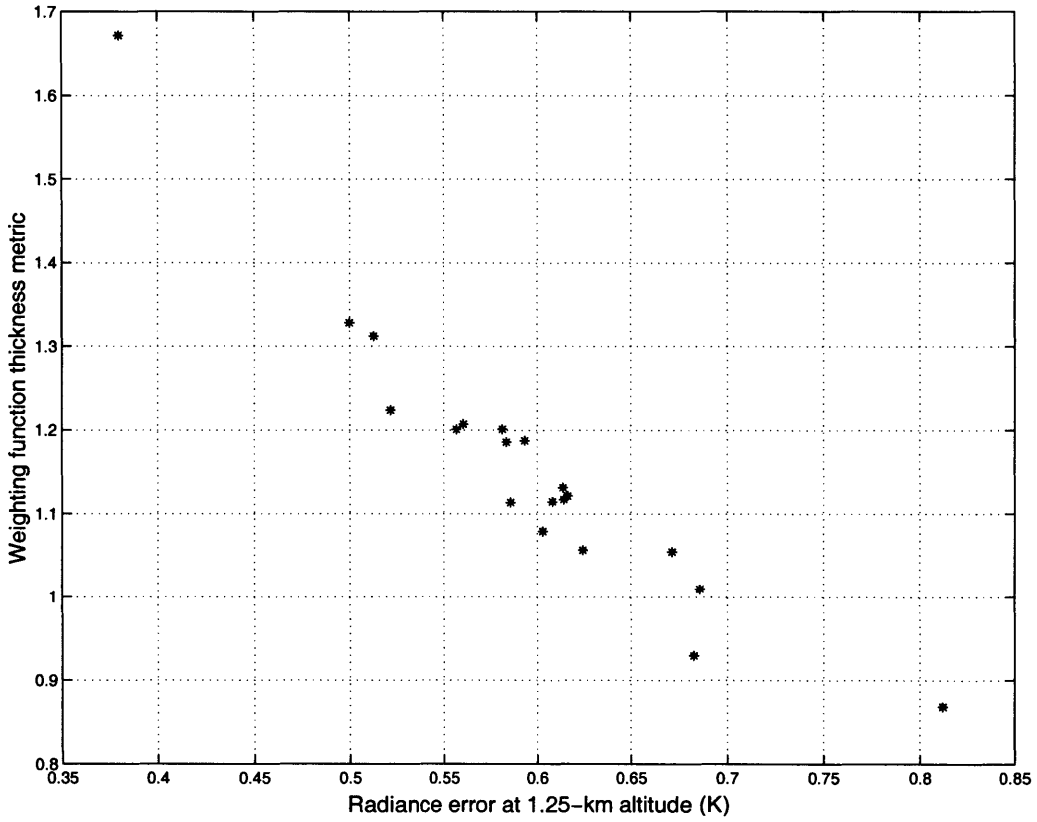

Figure 6-18: Cloud clearing error versus weighting function width.

peaking at the same altitudes.

For direct assimilation of radiances into NWP models, channels peaking at altitudes below 1–2 km may be problematic. Conversely, channels peaking above some low altitude ($\sim$ 0–3 km) should all be usable. By rejecting some soundings, we expect lower-altitude channels can be utilized (see Section 8.3).

Chapter 7

NAST Observations of Cloud and Precipitation Structure During CAMEX-3

AIRBORNE atmospheric sounding instruments with high spatial and spectral resolution provide a convenient platform for the development and validation of transmittance codes, surface models, cloud models, and present and future remote sensing systems. The study of clouds is particularly important, as they present the biggest challenge for high-fidelity sounding of the vertical temperature and moisture structure of Earth's atmosphere. In this chapter, infrared imagery collected by NAST-I and the Multispectral Atmospheric Mapping Sensor (MAMS) and microwave imagery collected by NAST-M during CAMEX-3 are used to characterize the impact of clouds and precipitation on sounding data.

7.1 Microwave observations of thermal and precipitation structure in Hurricane Bonnie

Aircraft-based imaging of temperature and precipitation using passive microwave radiometry has been studied by a number of investigators (see [68], [69], and [70] for example). Multispectral microwave sounders exploit the frequency dependence of scattering

from hydrometeors to provide information about particle sizes [71], cloud-top altitudes [69], and rain rate [72, 73]. Recent studies [70] have demonstrated the ability of high-resolution microwave imagery to clearly resolve hurricane eyewalls and warm cores within the eyes.

7.1.1 NAST-M brightness temperature images of Hurricane Bonnie

Figure 7-1 presents the brightness temperature images observed in the window channels associated with the NAST-M 54- and 118-GHz radiometers during an overflight of Hurricane Bonnie on August 26, 1998. The lower brightness temperatures observed in the center of the eye near 50 GHz result partly from the reflection of cold space from the ocean surface, whereas such surface effects are largely absent near 118 GHz because of the greater absorption there by tropospheric water vapor. The cold temperatures observed near 118 GHz (upper left of the top image) arise due to scattering of cold space off hydrometeors in a rain band on the edge of the hurricane eye. A band of moist air or cloud crosses the eye near 16:30 UTC. Other interesting features include the Rayleigh-scattering cloud in the lower left-hand corner (prominent near 118-GHz and invisible near 50-GHz) and the "eye within the eye" visible near 50 GHz, marking the touch-down of the very dry air descending in the eye.

7.1.2 Retrieval of the temperature perturbation in the eye of Hurricane Bonnie

The temperature profile of the warm core in the eye of Hurricane Bonnie (relative to clear air nearby) was retrieved using NAST-M observations in the following way. The eight temperature sounding channels near 54 GHz and the six near 118 GHz were used as inputs to a neural network with one sigmoidal hidden layer of 55 nodes to retrieve temperature profiles [37]. A training ensemble of 500 radiosondes ($\pm 45°$ latitude, all seasons) selected from the TIGR profile set [74] were used with the forward and sea-surface models described in Section 4.3 to produce simulated brightness temperatures at the NAST-M frequencies. Temperatures were retrieved along-track (nadir only) at 22 levels ranging from 0 to 16 km. The retrievals were filtered (in the along-track direction only) with a triangular filter of length five, and were then bilinearly interpolated by a factor of two in the vertical and

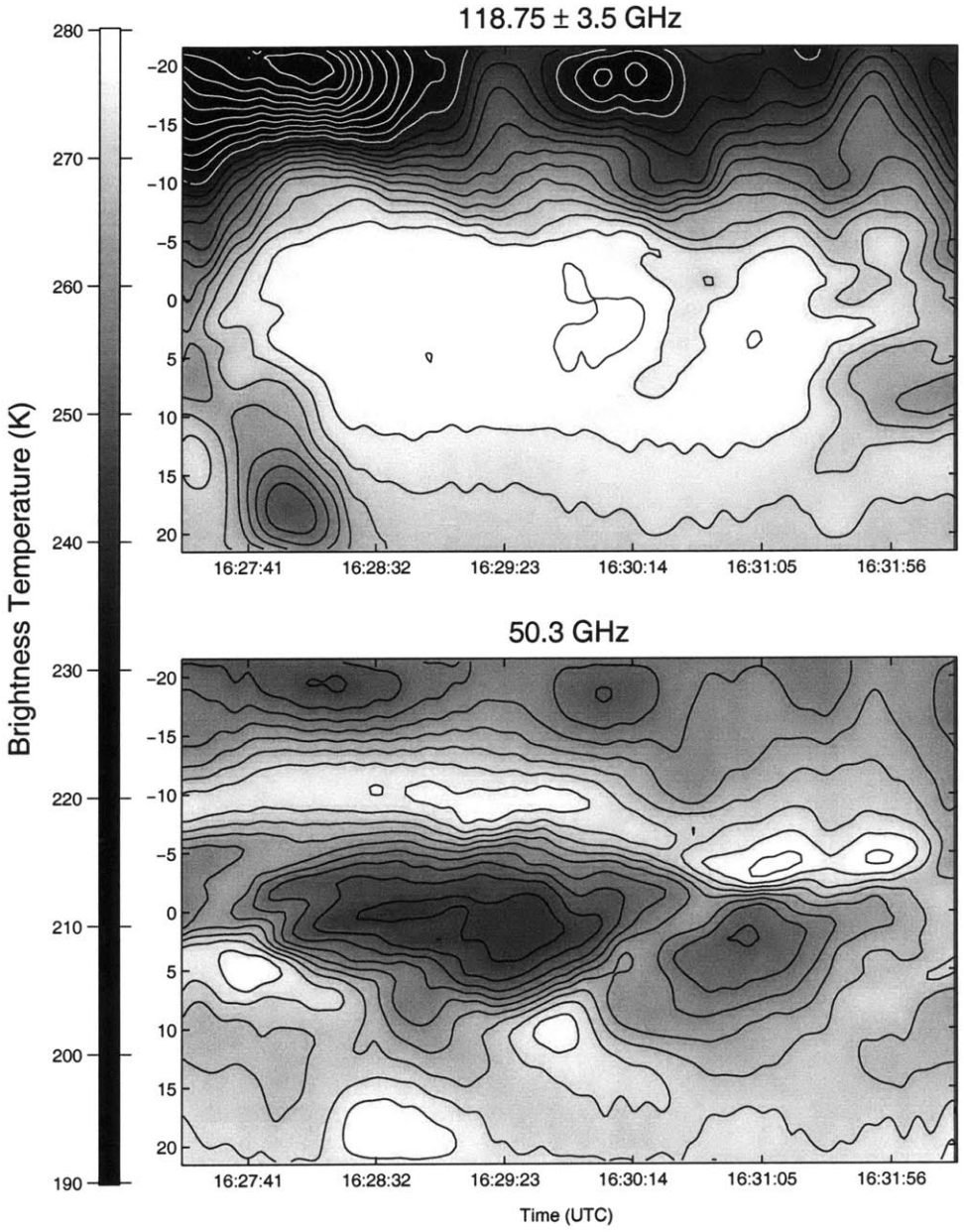

Figure 7-1: Brightness temperature images over the eye of Hurricane Bonnie, August 26, 1998, at window frequencies near 50.3 GHz and 118.75 ± 3.5 GHz. The swath width is approximately 40 km at an altitude of 10 km. Contour lines are drawn in black for every 4-K change from 240 K to 280 K (warmest contour at 280 K), and in white for every 5-K change from 190 K to 235 K (warmest contour at 280 K). Values on the ordinate indicate distance from the flight track in kilometers.

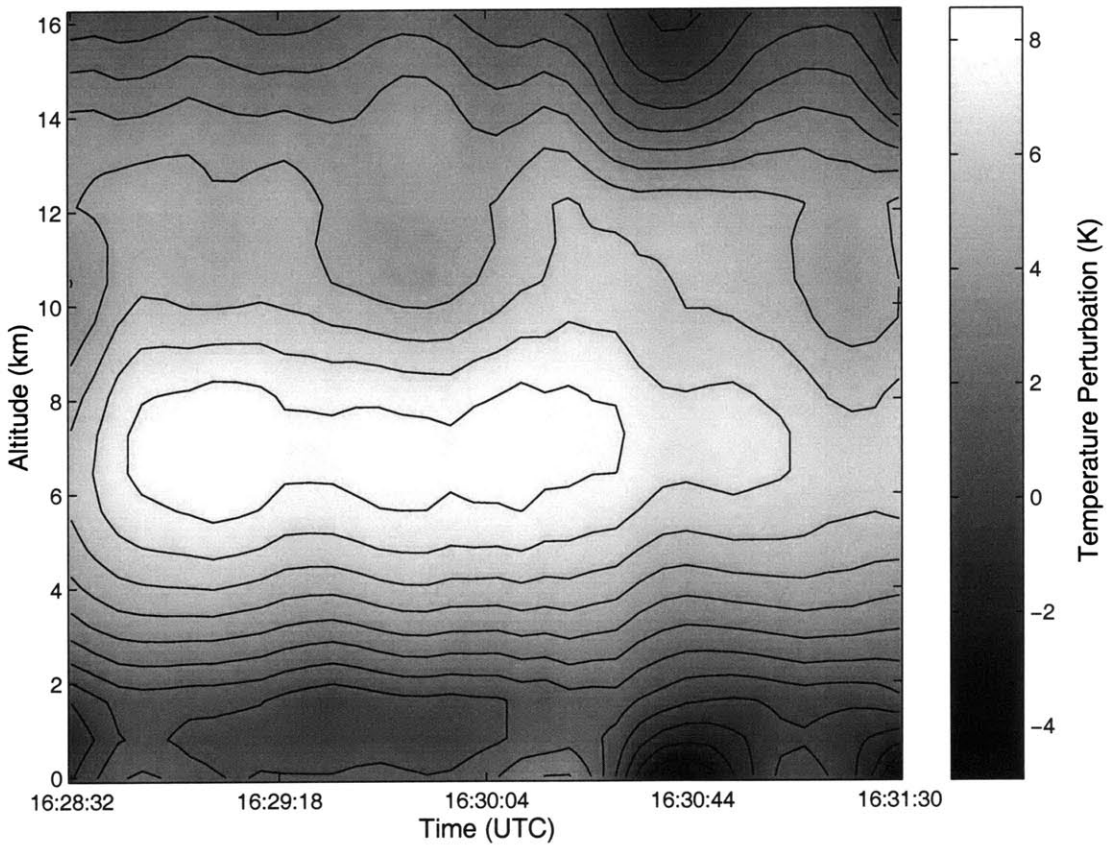

Figure 7-2: Warm core temperature profile retrieval for the eye of Hurricane Bonnie, August 26, 1998, relative to clear air 180 km to the east. The horizontal extent of the retrieval is approximately 35 km. Contour lines are drawn for every 1-K change in retrieved temperature, with the warmest contour at 7.7 K.

along-track directions. Figure 7-2 illustrates the difference between the temperature profile retrieved by NAST-M near nadir as it crossed the eye of Hurricane Bonnie on August 26, 1998 near 16:30 UTC, and the temperature profile observed a few minutes earlier in relatively clear air 180 km to the east. The warm core aloft appears to peak near 8 K at altitudes between 5 and 10 km. The magnitude and form of hurricane warm cores are well-known indicators of wind speed and structure of hurricanes, and have been used to monitor hurricane strength [72, 75, 76, 77].

7.1.3 Analysis of cloud particle size distributions using NAST-M

The combination of 54- and 118-GHz spectral data reveals information about hydrometeor size distributions and cell-top altitudes, and about the correlated parameters, vertical wind velocity and precipitation rate. Precipitation increases directly with the vertical velocity of saturated air into cold condensing regions, provided we neglect re-evaporation at lower altitudes. Figure 7-3(a) illustrates how such spectral data reveals precipitation structure. This figure shows a narrow cold band near 50.3 GHz that corresponds to higher precipitation rates in the eyewall of Hurricane Bonnie on August 23, 1998, while Figure 7-3(b) also highlights smaller hydrometeors that delineate the broader (cold) rain bands visible near 118.75 ± 3.5 GHz from 19:52 - 19:53 and 19:56 UTC.

From such data it is difficult to retrieve separately precipitation parameters such as rain rate, drop size, cell-top altitude, and cloud density because they are highly correlated. For example, high vertical winds will generally increase the rate of condensation, support larger drops aloft, push to higher altitudes, and result in greater cloud densities. Nonetheless, certain trends in the distinctive effects of these various parameters are evident in radiative transfer calculations based on simple cloud models. Figure 7-4 suggests how spectral differences between the 54- and 118-GHz oxygen absorption bands can reveal information about hydrometeor sizes and cell-top altitudes. The distribution of lower brightness temperatures extending downwards and to the left results from greater numbers of larger hydrometeors aloft at higher altitudes, and therefore generally greater precipitation rates. Observation offsets orthogonal to this reference direction "z" provide additional information. For example, the cloud model presented by Gasiewski and Staelin [80] exhibited iso-altitude contour lines in the $53.65/118.75 \pm 1.45$ GHz plane that were generally oriented between the z and horizontal (53.65-GHz) axes.

A similar set of theoretical contours [79, 78] is overlaid on Figure 7-4 based on a tropical standard atmosphere with 1 g/m^3 water (ice, below 0° C) in spherical drops having exponentially distributed diameters from the surface up to a variable cloud-top altitude. The contours are labeled with median mass-weighted drop diameters (half the mass is in larger drops) and cell-top altitudes. The contour lines associated with constant drop size are generally oriented between the z and vertical (118.75 ± 3.5 GHz) axes, and closer to z,

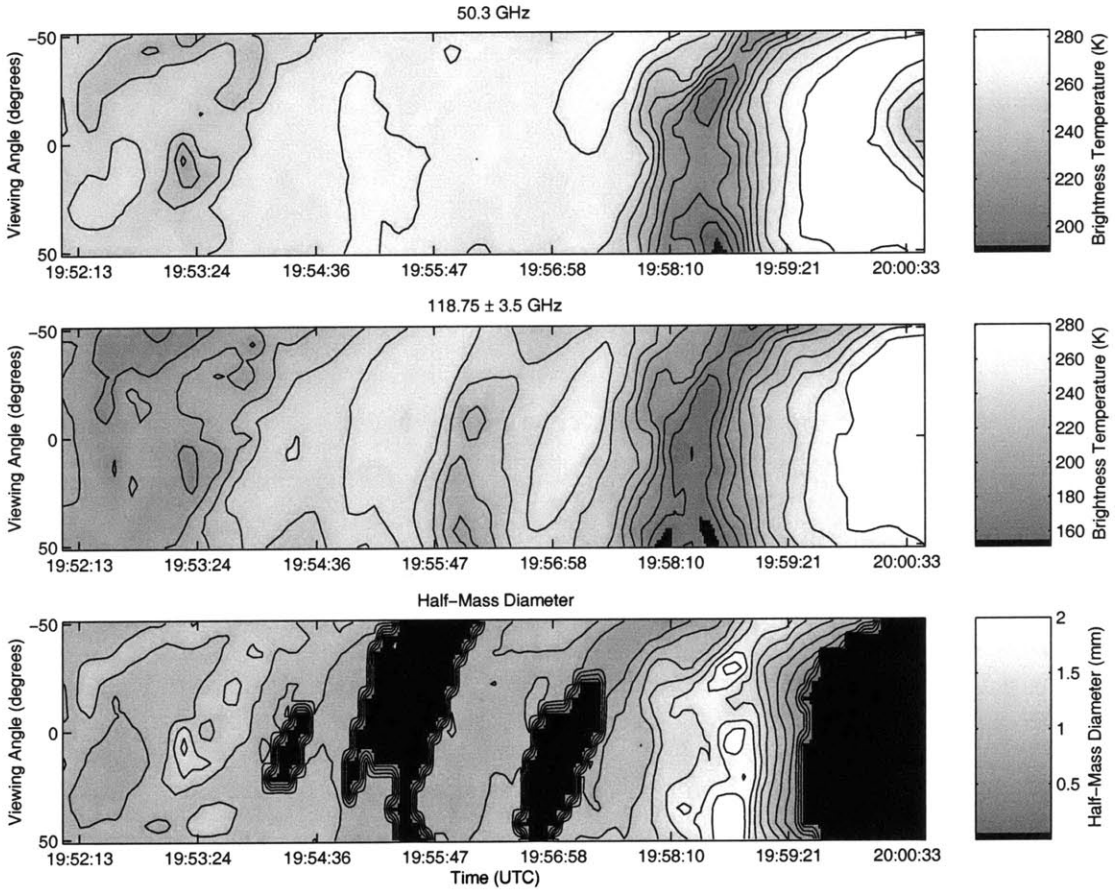

Figure 7-3: NAST-M observations of one eyewall and two rainbands of Hurricane Bonnie, August 23, 1998. (a) Radiance image near 50.3 GHz (contour lines every 10 K, warmest contour at 273.5 K), (b) radiance image near GHz (contour lines every 15 K, warmest contour at 267.5 K), (c) retrieved particle size index [78] (contour lines every 0.2 mm, largest contour at 1.8 mm); larger values correspond to larger particles (see Figure 7-4).

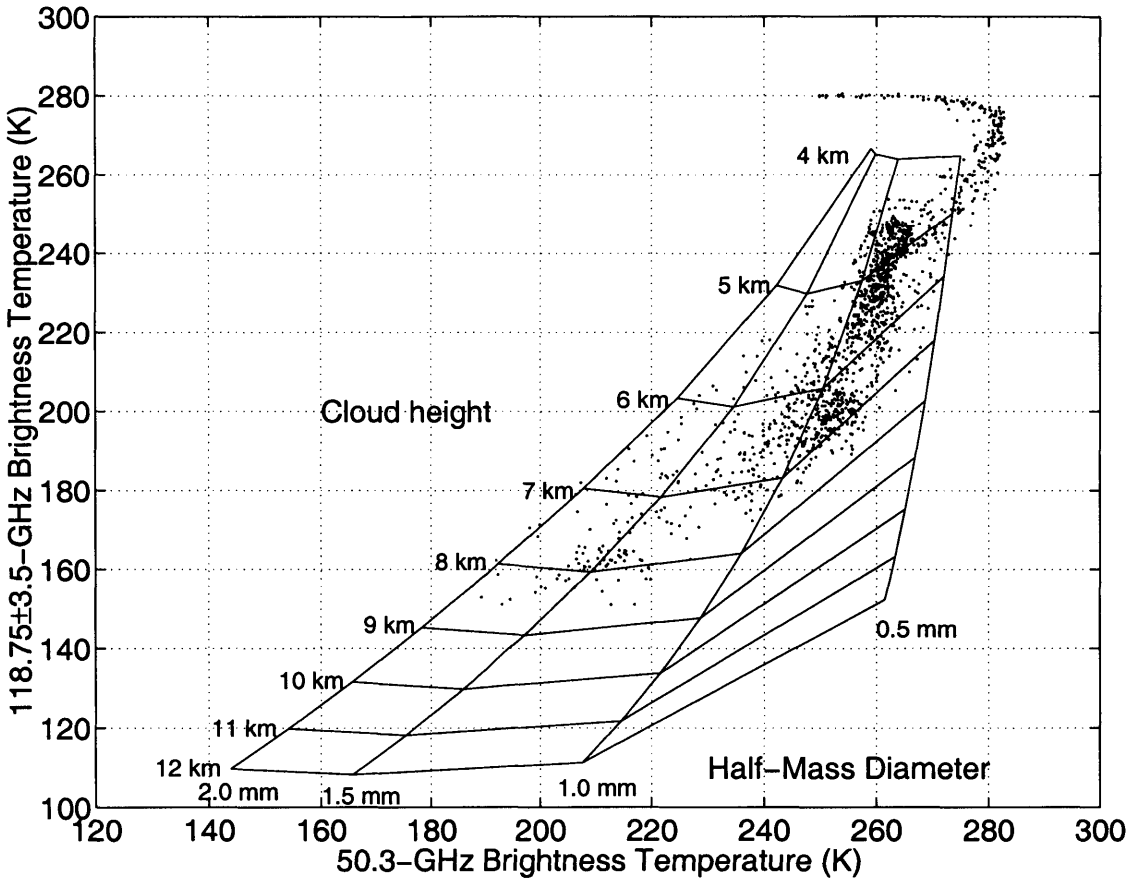

Figure 7-4: Relative radiances observed over Hurricane Bonnie, August 23, 1998 near 50.3 GHz and 118.75 ± 3.5 GHz illustrating the nominal effects of particle size and the definition of the size metric illustrated in Figure 7-3(c). A set of theoretical contours [79, 78] is also shown, based on a tropical standard atmosphere with 1 g/m^3 water (ice, below 0° C) in spherical drops having exponentially distributed diameters from the surface up to a variable cloud top altitude.

with larger drops corresponding to colder 50.3-GHz brightness temperatures. These same contours applied to the data of Figure 7-3(a-b) yield the inferred-drop-size image shown in Figure 7-3(c), which suggests that the largest particles are found toward the inner edge of the hurricane eyewall, consistent with stronger convection there. No diameters are presented for inferred cell-top altitudes below 4.9 km because this model and data are less reliable there. Particle diameter is only one of the links between spectral observations and precipitation rates; others include cell-top altitude, absolute albedo, and the adjacent temperature and humidity profiles. Further treatment of these retrieval issues requires additional modeling and data, and are being pursued separately [73].

7.2 The Multispectral Atmospheric Mapping Sensor (MAMS)

The Multispectral Atmospheric Mapping Sensor (MAMS) [81] is a multispectral scanner that measures reflected radiation from the Earth's surface and clouds in eight visible/near-infrared bands, and thermal emission from the Earth's surface, clouds, and atmospheric constituents (primarily water vapor) in four infrared bands. The 5.0 mRa aperture of MAMS produces an instantaneous field-of-view (IFOV) resolution of 100 m at nadir from the nominal ER-2 altitude of 20 km. The width of the entire cross-track field of regard is 37 km. The MAMS channel characteristics are given in Table 7.1.

7.3 The CAMEX-3 radiance/cloud-truth database

7.3.1 Data description

Channel 12 (12.4-μm) of MAMS was used to derive an estimate of the cloud fraction (percent) and mean cloud perturbation (K) in each NAST-I/M pixel. Channel 11 (11.2-μm) was used when channel 12 was unavailable. Data from four daytime flights over the Atlantic ocean were used: two flights (August 23 and 26, 1998) near Hurricane Bonnie, one flight (August 30, 1998) near Hurricane Danielle and one flight (September 13, 1998) near Andros Island, Bahamas. A catalog of cloudy CAMEX-3 data used in the thesis is shown in Table 7.2.

Table 7.1: Channel specifications for the Multispectral Atmospheric Mapping Sensor (MAMS).

Channel	Spectral range (μm)	NEΔT (K)
1	0.42–0.45	
2	0.45–0.52	
3	0.52–0.60	
4	0.57–0.67	
5	0.60–0.73	
6	0.65–0.83	
7	0.72–0.99	
8	0.83–1.05	
9	6.20–6.90	0.3
10	10.32–12.02	0.1
11	10.32–12.02	0.1
12	12.20–12.56	0.4

Table 7.2: Catalog of cloudy flight data available from CAMEX-3 for NAST–I, NAST–M, and MAMS. Only scan angles between $\pm$ 15°are included.

Date	Available Pixels			
	NAST–I & MAMS	NAST–M & MAMS	NAST–I & NAST–M	All Three
23 Aug 98	1383	650	7655	650
26 Aug 98	389	389	8640	389
30 Aug 98	612	612	3650	612
13 Sep 98	704	704	3970	704
21 Sep 98	2507	0	0	0
22 Sep 98	1650	0	0	0
Totals	7245	2355	23915	2355

7.3.2 MAMS-derived estimates of cloud parameters

The cloud fraction and mean cloud perturbation estimates were derived as follows. The mean and standard deviation of all MAMS pixels falling within a NAST-I/M pixel are calculated. The NAST-I/M pixel is classified as "clear" if both the following conditions are satisfied:

1. The mean MAMS pixel temperature exceeds a flight-dependent threshold T_C. This threshold is determined from an empirical examination of the MAMS data, and is typically several degrees below the warmest pixel temperature.

2. The standard deviation of the MAMS pixels is below 0.25 K.

For those NAST-I/M pixels classified as "cloudy", the cloud fraction (percent) is defined as

$$\mathrm{CF} = \frac{\text{number of MAMS pixels } \leq (T_C - 1.5 \text{ K})}{\text{total number of MAMS pixels}} \times 100 \tag{7.1}$$

and the mean cloud perturbation ΔT is the difference in brightness temperature between the mean of all the MAMS pixels in the NAST-I/M pixel and the nearest clear-air temperature (determined by MAMS).

This cloud classification algorithm was used to construct a NAST-I radiance database of 7245 pixels, where only scan angles between ± 15 degrees were included. NAST-M data were available for 2355 of these 7245 pixels. For each cloudy NAST-I/M pixel, a companion "clear" pixel was established by choosing the closest clear pixel as defined by MAMS. Representative results of the cloud classification algorithm are shown in Fig. 7-5, and results for each flight segment used in the database can be found in Appendix D. The resulting histograms of cloud fractions and mean MAMS brightness temperature perturbations are shown in Fig. 7-6.

7.3.3 Error due to slowly-varying cloud amount

One problem that was encountered was the failure of MAMS to distinguish between slowly-varying (spatially) cloud cover and clear air. One example of this is illustrated in Fig. 7-7 for a flight segment from August 23, 1998. A bias of almost eight degrees results

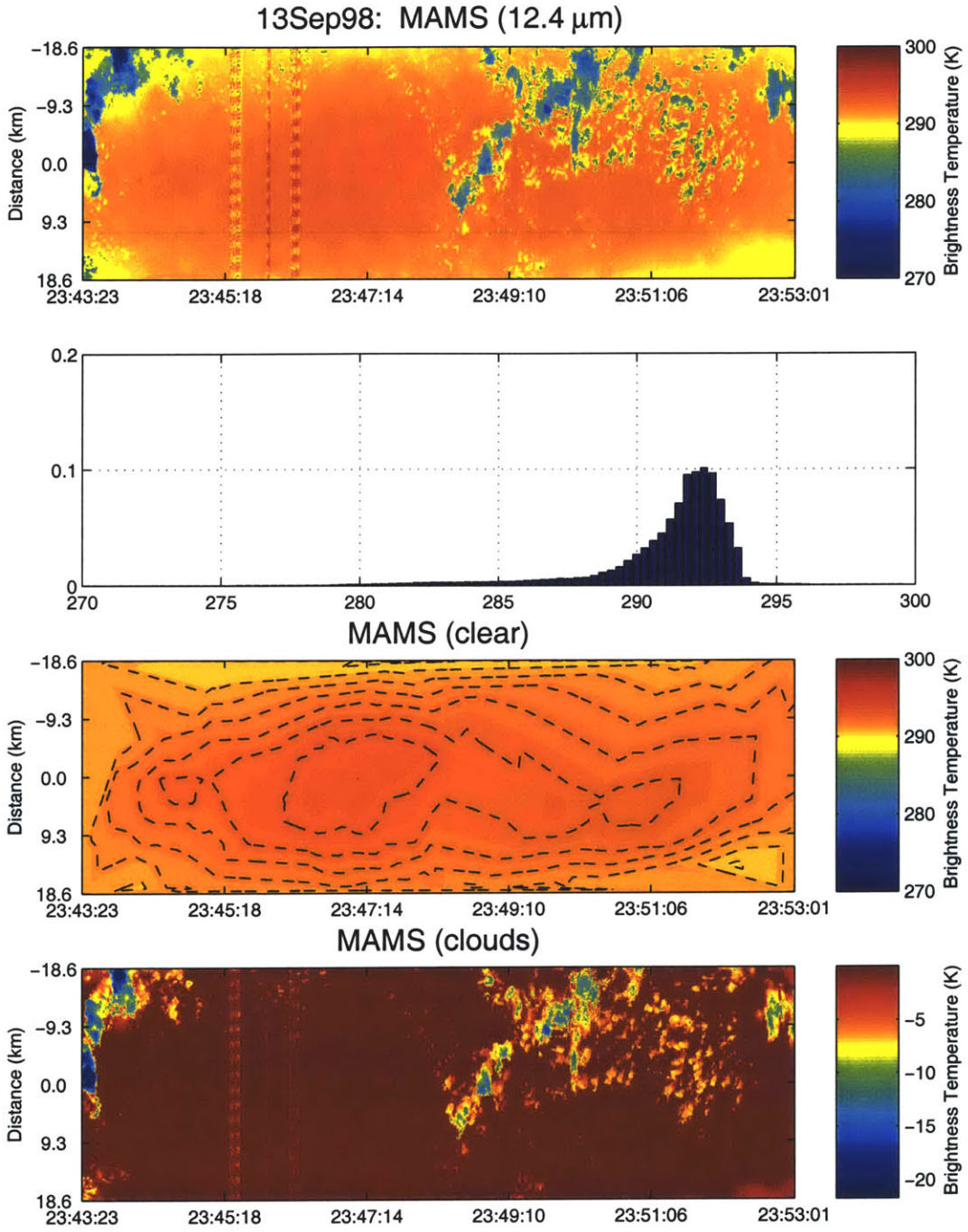

Figure 7-5: Representative MAMS cloud imagery from CAMEX-3. The top image shows MAMS brightness temperatures over a 40×120 km swath. A histogram of pixel brightness temperatures for the 12.4-μm image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

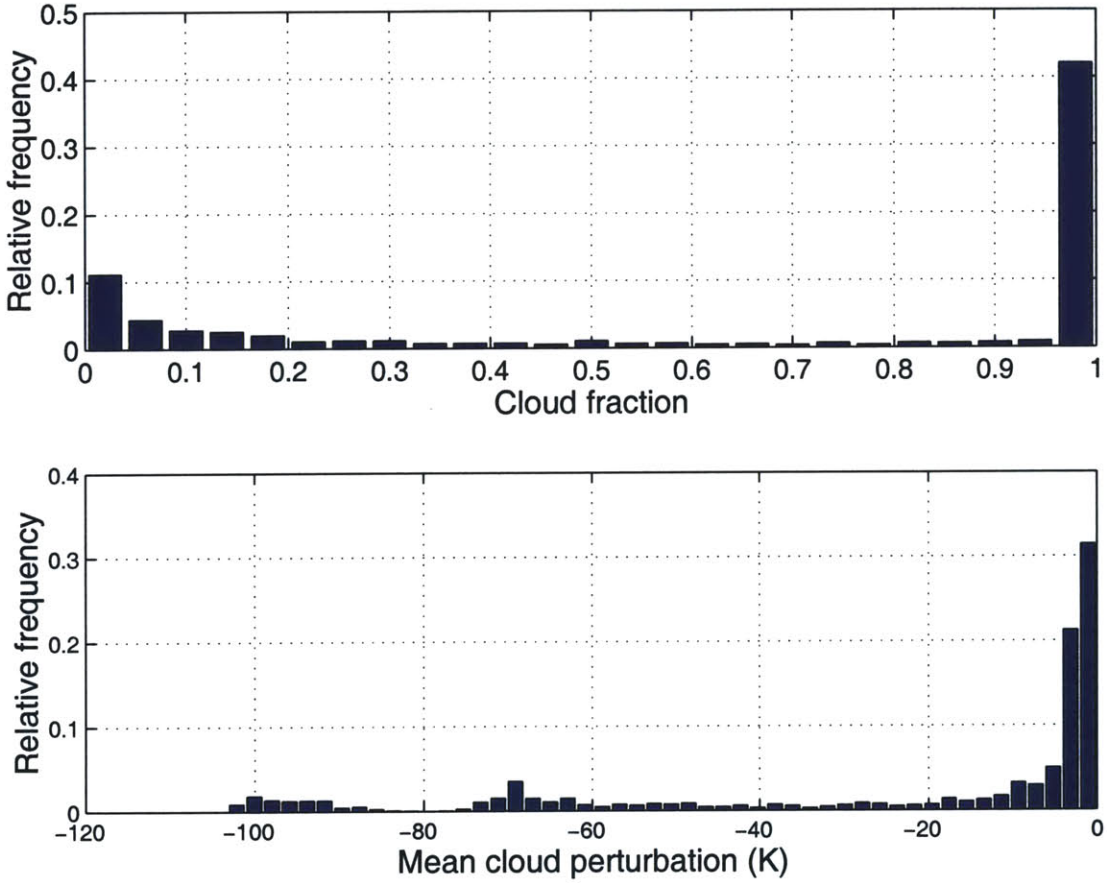

Figure 7-6: Histograms of cloud fractions and mean MAMS brightness temperature perturbations for the CAMEX-3 radiance/cloud-truth database.

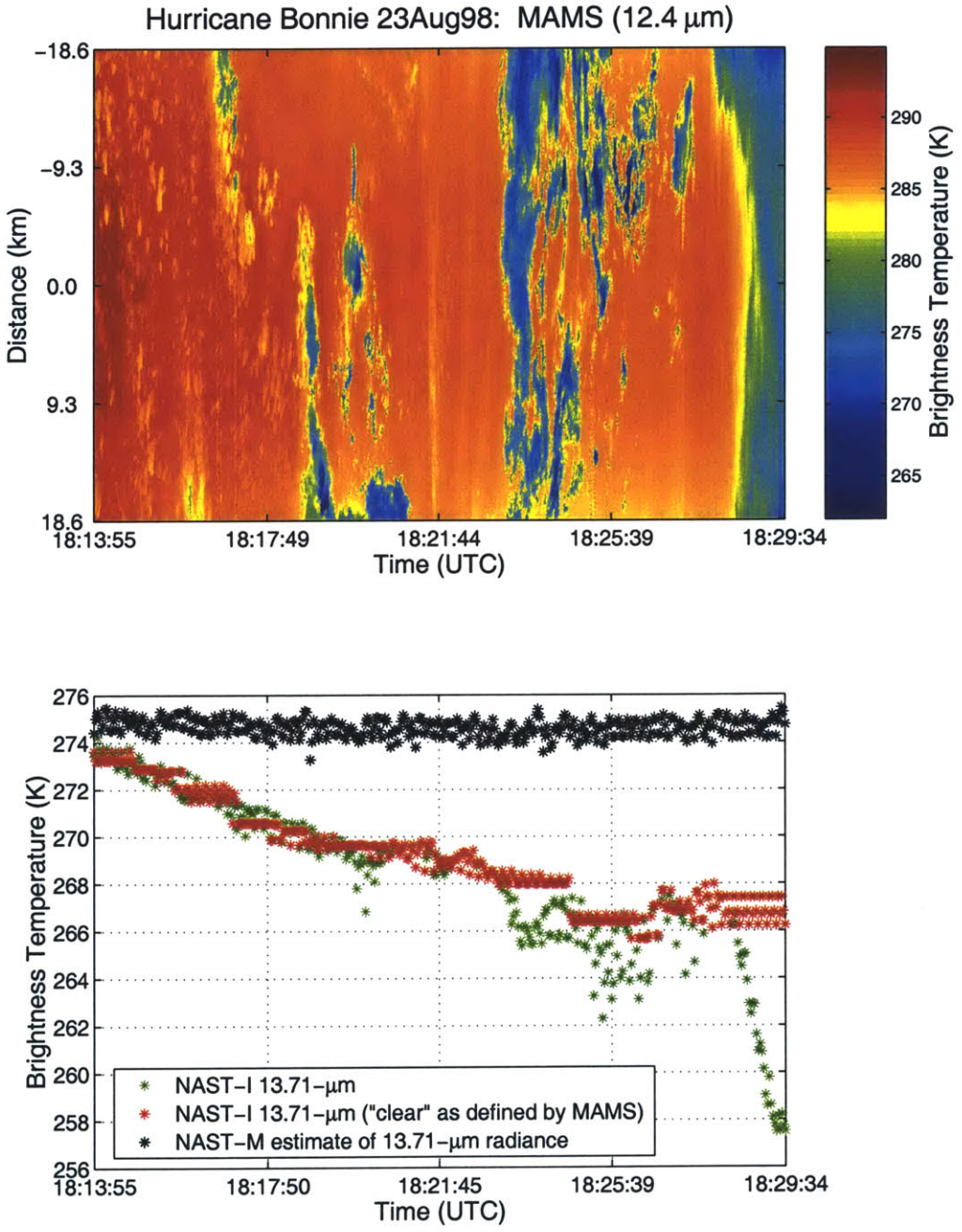

Figure 7-7: Example of clear-air misclassification by MAMS. The top plot shows a brightness temperature image for the 12.4-μm MAMS channel. The bottom plot shows the microwave estimate of the NAST–I 13.71 μm channel (weighting function peak near 4 km) exhibiting no drift, while the MAMS-derived estimate of the clear-air brightness temperature at 13.71 μm drifts cold as the cloud deck is encountered. Also shown is the brightness temperature of the cloudy 13.71 NAST–I μm channel, which decreases sharply in brightness temperature near 18:29 UTC.

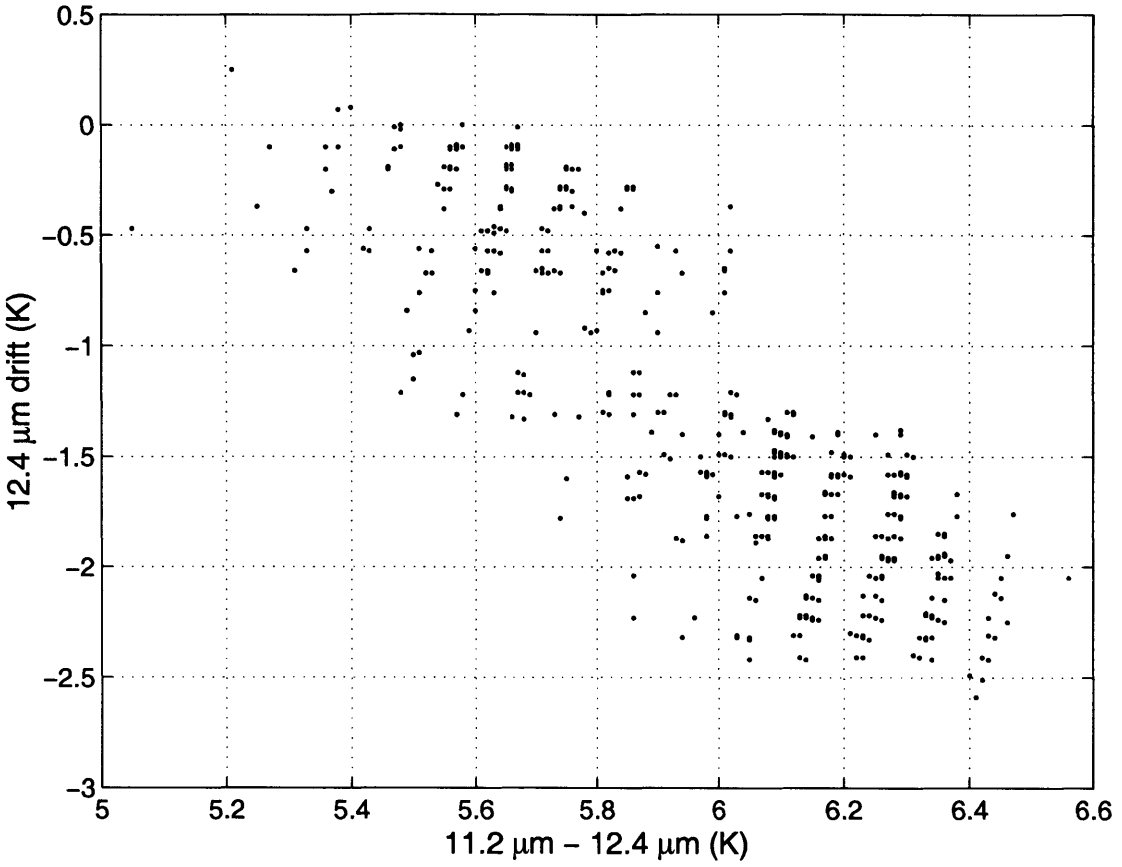

Figure 7-8: Scatterplot of the brightness temperature drift in the 12.4-μm MAMS channel as a function of the brightness temperature difference between the 11.2-μm and 12.4-μm MAMS channels.

as a gradually-intensifying cloud layer is misclassified as clear air. The 12.4-μm channel is mildly sensitive to water vapor, and it is possible that absorption due to water vapor is responsible for the drift. A split-window approach using 11.2-μm data to correct for the error due to water vapor absorption was explored. Figure 7-8 shows the drift in the 12.4-μm channel versus the difference between the split-window channels for a portion of the data shown in Fig. 7-7. The data suggest that a correction could be made to reduce the drift in the 12.4-μm channel if the correction proved to be predictable globally. No attempt was made to implement a correction because the drifting phenomenon occurred infrequently in the CAMEX-3 data selected for the analyses.

7.4 Principal components analysis of NAST-I cloud impact

The principal components analysis techniques applied to AIRS/AMSU data in Section 6.3 were applied to NAST-I CAMEX-3 data for three cases: clear-air, cloudy, and clear-cloudy ("cloud perturbation"), where MAMS was used as described earlier to determine whether each NAST-I pixel was cloudy or clear. In each case, the covariance matrix was prewhitened using ION estimates of NAST-I system noise (see Section 5.3). The NAST-I radiances were not converted into brightness temperatures, because some channels give negative readings for extremely cold scenes, including clouds, due to calibration problems.

The total radiance perturbation (expressed as NEΔT) due to clouds is shown in Fig. 7-9. Note that the RMS NAST-I radiance variability due to clouds is more than twice the RMS atmospheric temperature profile variability in the `cld2lr` database (shown in Fig. 6-1).

The eigenvalues corresponding to the top 100 NAPC's for clear and cloud-contaminated pixels are shown in Fig. 7-10. Also shown are scatterplots of the first three NAPC's. In both clear and cloudy cases, two coefficients capture most of the radiance variance. The radiance variation due to clouds is much greater than the radiation variation due to the atmosphere. This is expected, because only four flights were used to compile the pixel database and the atmospheric variation over the four flights was small (i.e., not "global"). The scatterplots for clear-air NAPC's are less Gaussian and show more clustering than the cloudy NAPC's. The clear-air NAPC's appear to fall within the space of simulation NAPC's. A detailed comparison of the CAMEX-3 cloud-impact NAPC's and the simulated cloud-impact NAPC's is given in Chapter 8.

The NAPC's of NAST-I radiance perturbations due to clouds (clear radiance minus cloudy radiance) are shown in Fig. 7-11. The characteristics of the cloud perturbation NAPC's are very similar to the cloudy NAPC's, which again suggests the clouds are the dominant source of radiance variation. Furthermore, the scatterplots exhibit Gaussian characteristics, with the exception of NAPC #2 versus NAPC #3, which is neatly divided into two clusters.

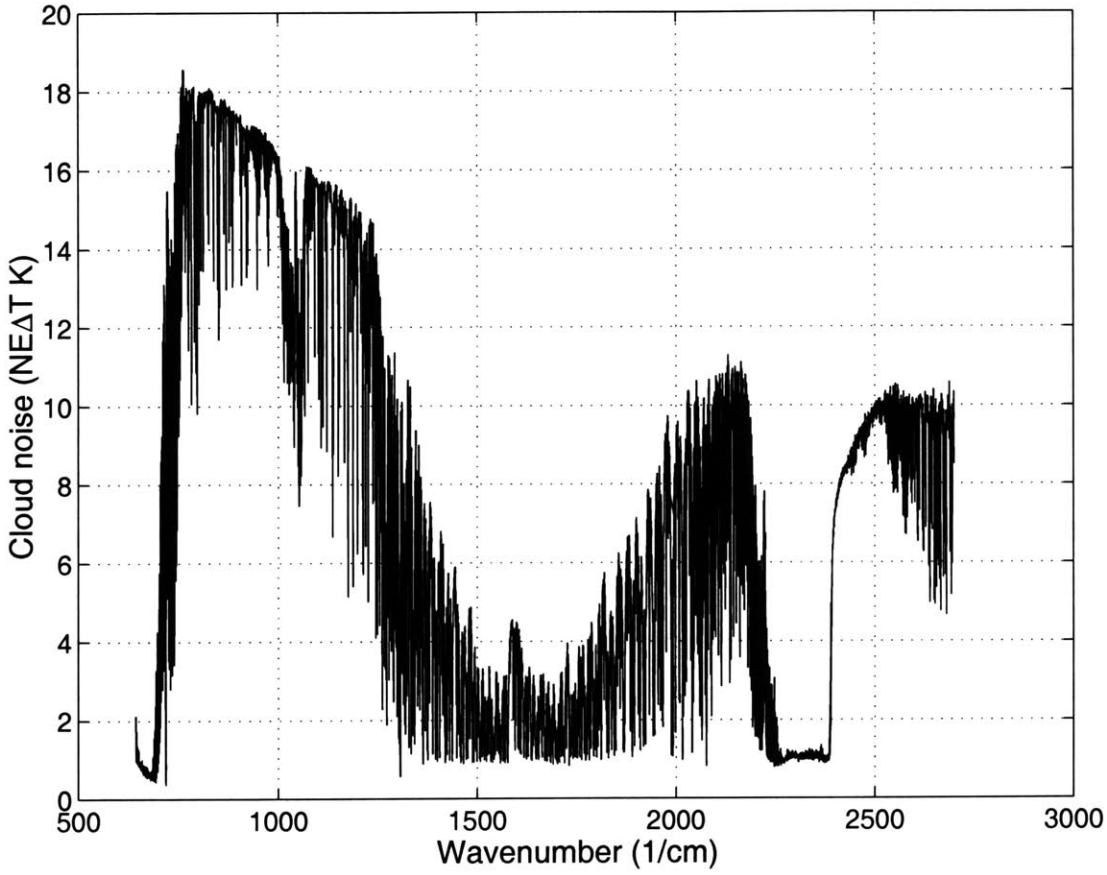

Figure 7-9: RMS NAST-I radiance variation due to clouds.

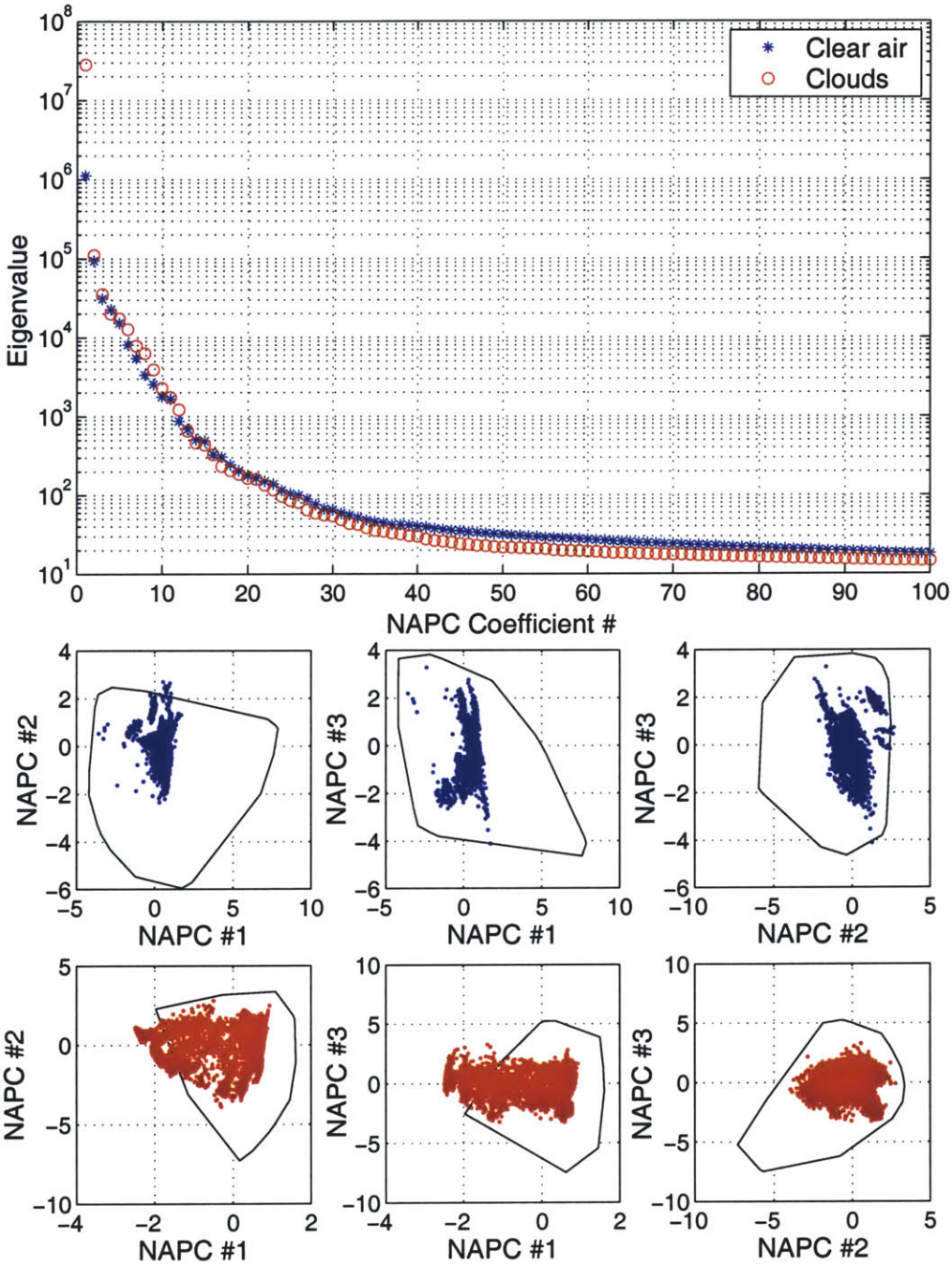

Figure 7-10: Noise-adjusted principal components transform analysis of clear and cloudy NAST-I data collected during CAMEX-3. The top plot shows the eigenvalues of each NAPC coefficient for clear and cloudy data. The middle row presents scatterplots of the three clear-air NAPC coefficients with the largest variance (shown normalized to unit variance). Also shown are the convex hulls of the corresponding simulation NAPC scatterplots (normalized in the same way as the CAMEX-3 scatterplots). The bottom row presents scatterplots of the three cloudy NAPC coefficients with the largest variance (shown normalized to unit variance).

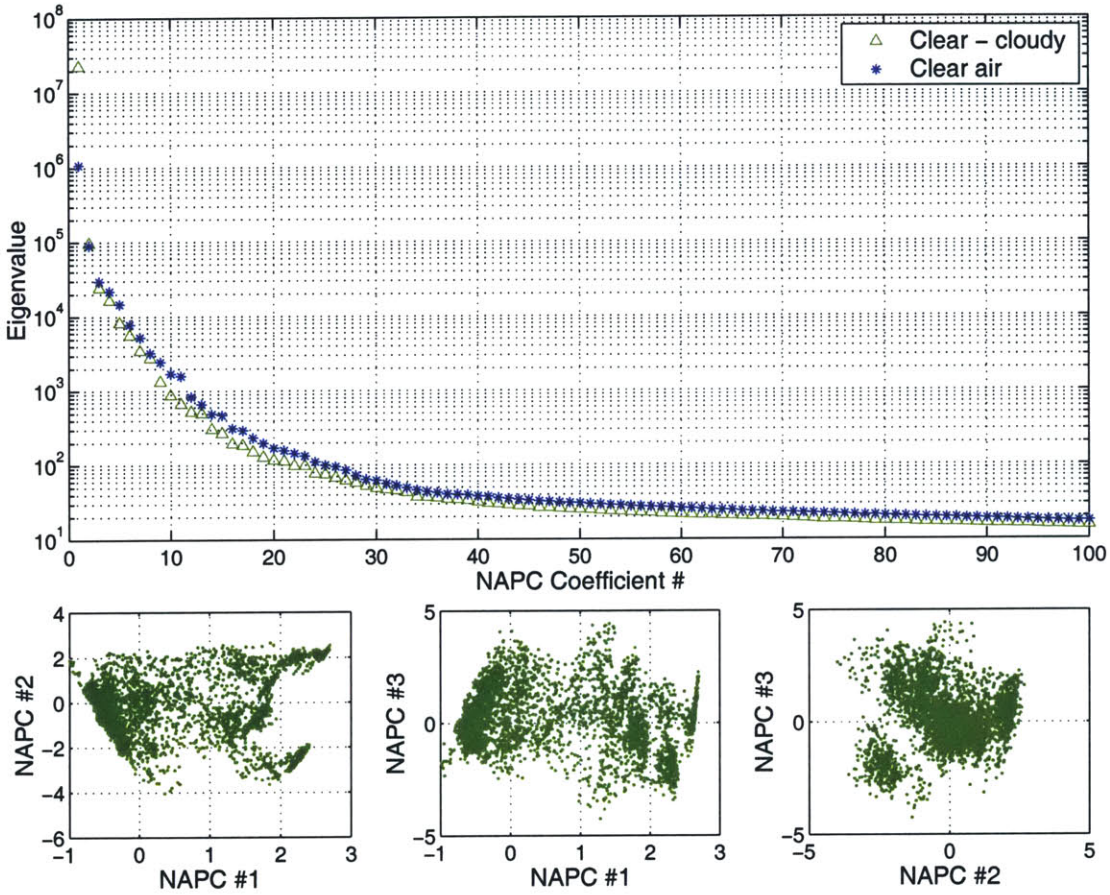

Figure 7-11: Noise-adjusted principal components transform analysis of the cloud impact (clear radiance - cloudy radiance) for NAST-I data collected during CAMEX-3. The top plot shows the eigenvalue of each NAPC coefficient of cloud impact, along with the NAPC coefficients of clear-air data (shown in Fig. 7-10). The bottom row presents scatterplots of the three cloud-impact NAPC coefficients with the largest variance (shown normalized to unit variance).

7.5 Summary

The utility of microwave dual-absorption-band measurements for characterization of convective precipitation was illustrated by observations of rainbands in Hurricane Bonnie. Dual-band microwave data collected by NAST–M during CAMEX-3 was used to provide:

1. The first 54/118-GHz particle size and cell-top altitude retrievals of convective precipitation. The ER-2 platform provided high spatial resolution of this storm's structure, including its warm core and the larger hydrometeors seen near the eye walls.

2. The first 54/118-GHz high-resolution temperature retrievals in the eye of a hurricane. The warm core and convection morphology of Hurricane Bonnie (August, 1998) was clearly revealed both by aircraft-based microwave brightness temperature imagery and temperature retrievals within the eye. Clouds above most of the eye precluded the use of infrared data for temperature sounding.

NAST-M, NAST-I, and MAMS were used to assemble a radiance database of 2355 pixels from four daytime flights during CAMEX-3. For the first time, principal components analysis of hyperspectral infrared observations of clear-air and cloudy data were performed, which quantitatively demonstrated both the amount and spectral structure of the radiance variability due to cloud perturbations. The NAST-I cloud perturbation observations are compared to simulation results in Chapter 8.

Chapter 8

Analysis of cloud impact on atmospheric sounding performance using NAST

DATA collected by airborne infrared (NAST–I, MAMS) and microwave (NAST–M) sensors during CAMEX-3 (Florida, Summer, 1998) are used to quantify the cloud impact on atmospheric sounding performance in several ways. The degrees of freedom introduced by clouds were estimated in Chapter 7 by applying principal component transforms to cloud-perturbed (clear minus cloudy) NAST–I spectra. These results are compared to those obtained using NAST–I radiance simulations with two-level clouds. The radiance cloud-clearing performance for NAST-type clouds is evaluated and compared with results for simulated two-level clouds. Finally, a neural network is trained to identify soundings associated with large cloud-clearing errors. When these soundings are rejected, the errors are modestly reduced.

8.1 Comparison of cloud information content

A thorough statistical comparison of the information content of simulated and observed hyperspectral sounding data is impossible in this study given the limited atmospheric variability observed over only four flights in three weeks during CAMEX-3. However, the

observed cloud characteristics are sufficiently random that statistical comparisons of cloud effects (including crosstalk from solar and surface reflection) are justified. We begin by defining a cloud impact vector:

$$\Delta R = R^{\mathrm{clr}} - R^{\mathrm{cld}} \tag{8.1}$$

where R^{cld} is the observed (or simulated) infrared radiance vector upwelling from the top of a cloudy atmosphere, and R^{clr} is the corresponding radiance that would have been observed if the cloud effects were removed. Henceforth, a radiance vector observed by NAST–I will be denoted R_{obs} and a simulated radiance vector will be denoted R_{sim}. An observed (simulated) vector of NAST–M brightness temperatures will be denoted B_{obs} (B_{sim}).

8.1.1 Simulation of NAST radiances

A set of 12,000 NWP-generated atmospheric profiles (uniformly distributed seasonal and global coverage) was used to generate a database of simulated NAST–I/M radiances. Random surface emissivities were chosen for each profile; they were uniformly distributed from 0.95 to 1.0 for NAST–I and from 0.6 to 0.7 for NAST–M (consistent with an ocean surface). The surface emissivities of one instrument were assumed to be uncorrelated with the other; emissivities of each channel of a given instrument were assumed to be perfectly correlated. Clouds were introduced at two levels into the profile set by relating the cloud top pressure and fraction to a relative humidity threshold as discussed in Section 6.2.2. The resulting cloud fractions were approximately uniformly distributed between 0 and 100 percent. Cloud tops were assumed to be opaque to the infrared channels with a non-random emissivity of 0.9. Microwave channels were perturbed by clouds only through liquid water absorption; scattering was not modeled, therefore the following results apply only to non-precipitating pixels. Clear-air NAST–I radiances were computed using the UMBC NAST–I Fast Transmittance Codes [62], and NAST–M brightness temperature were computed using a line-by-line model [44].

8.1.2 Principal components analysis of cloud impact

The total radiance perturbation (expressed as NEΔT) due to clouds is shown in Fig. 8-1 for the CAMEX-3 NAST–I observations and the simulated NAST–I observations. Also shown is the cloud noise remaining after the two most-significant cloud-impact NAPC's have been removed. This residual higher-order noise presumably would be more difficult to clear. The relatively large errors in the shortwave channels result from reflected solar radiation off the cloud tops, and are not well-represented by the two NAPC's. This could be explained in part by the nonlinearity of the Planck function, and a nonlinear principal components transform would achieve greater energy compaction. Note that the RMS NAST–I radiance variability due to clouds during the four CAMEX-3 flights is approximately twice that observed over 12,000 simulated profiles. The ION algorithm was used to filter the instrument noise from the cloud noise in the CAMEX-3 observations to facilitate comparisons with the simulation-derived cloud noise, which is free from instrument noise.

Another interesting feature of Fig. 8-1 is the fact that the cloud noise variance near 15 μm is significantly larger than the cloud noise variance near 4 μm. This was also evident in the AIRS simulations of cloud noise (see Fig. 6-13). One cause of this may be the larger water vapor opacities associated with the 4-μm channels that reduces cloud/background radiance differences.

The eigenvalues of $E(\overline{\Delta R}_{sim}\overline{\Delta R}_{sim}^T)$ and $E(\overline{\Delta R}_{obs}\overline{\Delta R}_{obs}^T)$ are shown in Fig. 8-2, where ΔR was normalized for unity noise on each channel. We use an overbar to denote that the mean has been removed from a random vector. 5000 NAST–I channels (evenly distributed among the 8632) were used in the calculations. About six times more cloud variance is evident in the CAMEX-3 hurricane-season observations, and NAST exhibits more degrees of freedom. However, both NAST observed and simulated data have fewer than ~20 degrees of freedom that appear to be above any noise threshold.

The eigenvector distributions associated with the eigenvalues shown in Fig. 8-2 are shown in Fig. 8-3 using nine scatterplots, where the scatterplot in position (i, j) corresponds to the i^{th} eigenvector of the simulated ΔR data plotted against the j^{th} eigenvector of the CAMEX-3 observed ΔR data. The first two pairs of eigenvectors (graphs [1,1] and [2,2]) are similar, but the third eigenvectors pair (graph [3,3]) differ substantially. These

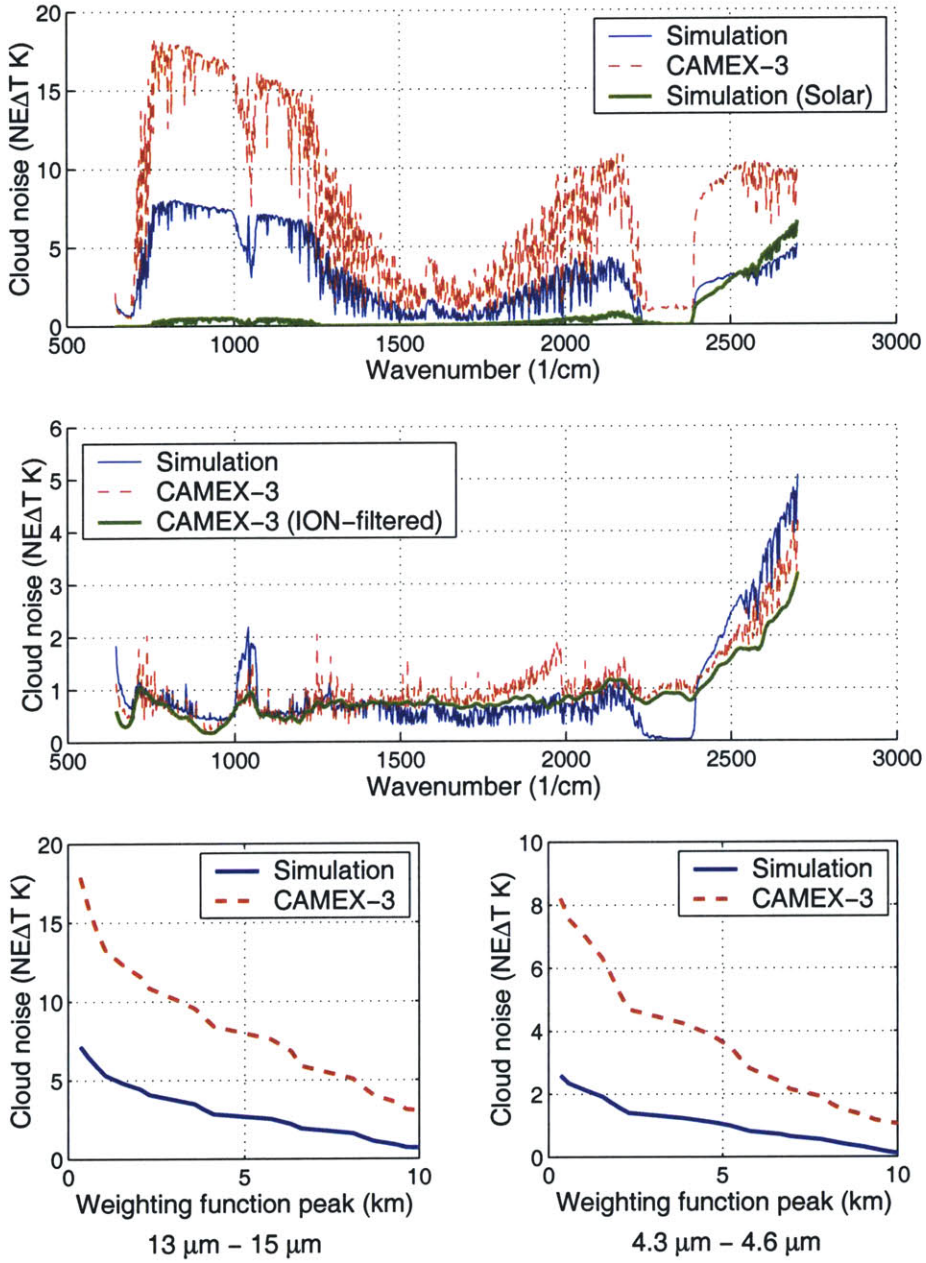

Figure 8-1: Top plot: comparison of RMS NAST–I cloud noise observed during CAMEX-3 and simulated using `cld2lr` database. The thick line (small amplitude below 2000 cm^{-1}) shows the RMS variation due to solar reflection. The middle plot shows the cloud error after the two most-significant NAPC's have been removed. The thick line (slightly below the CAMEX-3 line on the bottom plot) shows the cloud noise after instrument noise has been filtered with the ION algorithm. The bottom two plots present the cloud noise shown in the top graph as a function of weighting function peak altitude to facilitate comparisons with the AIRS simulation results shown in Fig. 6-13.

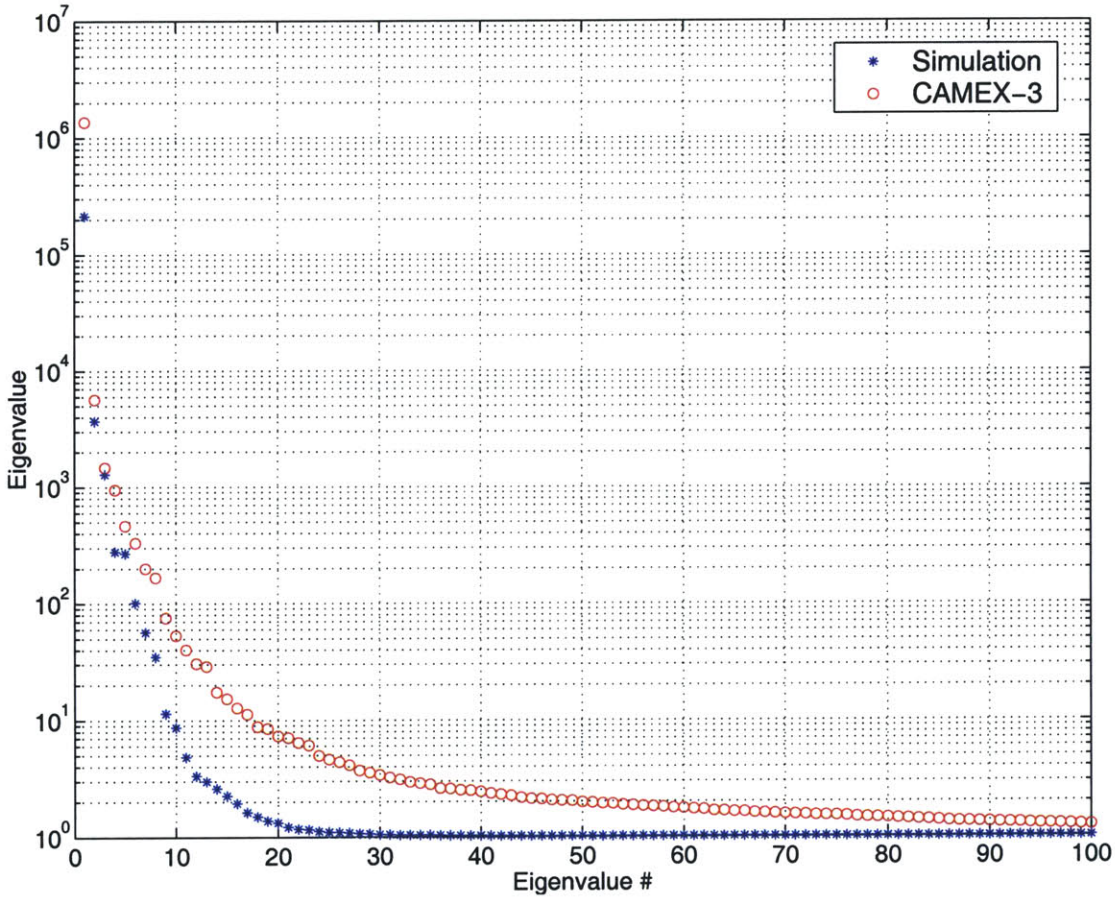

Figure 8-2: Observed (circles) versus simulated (asterisks) cloud-impact Noise-Adjusted Principal Components (NAPC's) for 5000 NAST–I channels. Solar, surface, and cloud effects contribute to both.

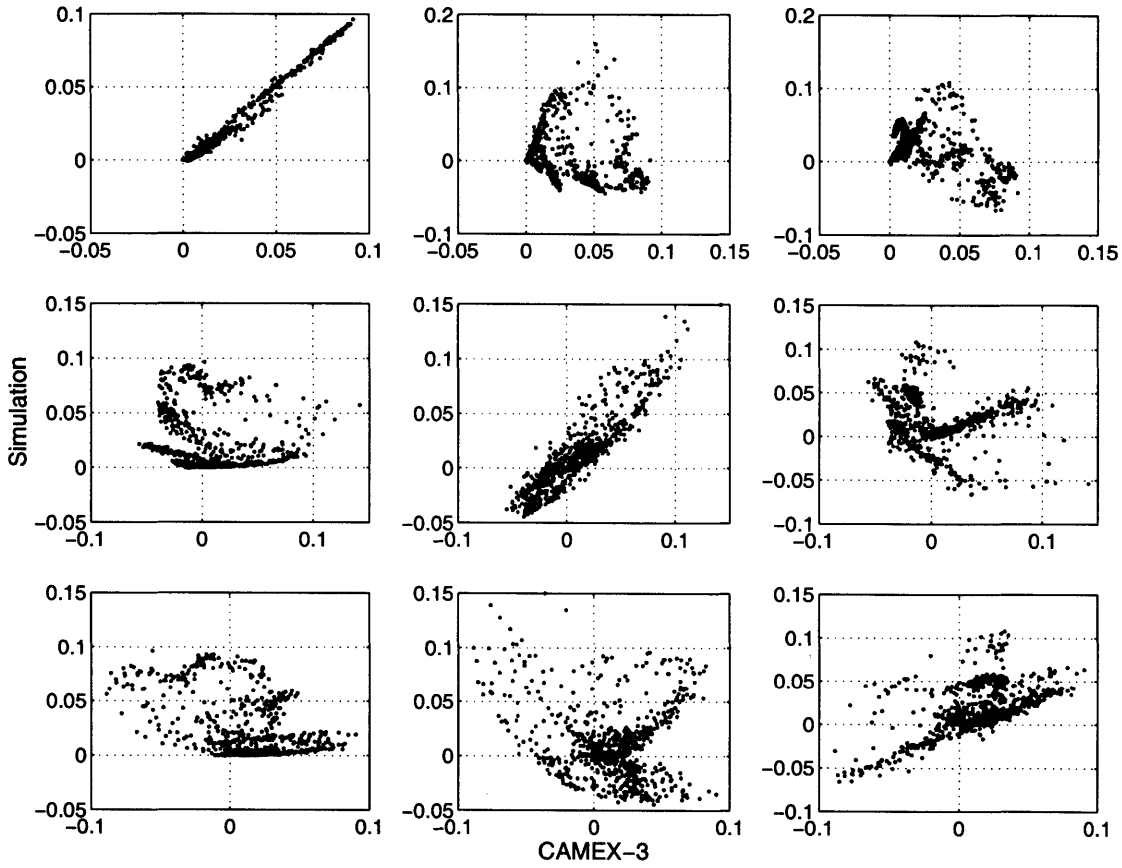

Figure 8-3: Observed versus simulated NAST–I cloud-impact eigenvectors associated with the eigenvalues shown in Fig. 8-2; solar, surface, and cloud effects contribute to both. In each subplot, 1000 elements of the CAMEX-derived and simulation-derived eigenvectors are shown.

results suggest that the cloud models used in the simulations capture large-scale features of cloud observations, but may fail to capture fine spectral structure. Scatterplots of NAPC's exhibit similar characteristics, and indicate that the CAMEX-3 cloud perturbations are statistically less global than the simulated cloud perturbations. As a final comment, it should be noted that transmittance errors could contribute to the fine-scale discrepancies. This hypothesis could be examined by comparing simulated and observed clear-air principal components. Unfortunately, the lack of a statistically-diverse set of clear-air pixels from CAMEX-3 precludes that analysis here.

8.2 Comparison of cloud clearing performance

Numerous investigators (see [66, 67], for example, as well as Section 6.6 of this thesis) have suggested methods for producing clear-column radiances given observations of cloudy radiances. We follow the development in [82] here, with no claims of optimality. To summarize, CAMEX-3 observations of cloud impact (ΔR_{obs}) are combined with simulated clear-air radiances (R_{sim}^{clr}) to form a "synthetic" cloudy radiance vector ($\widetilde{R}^{cld}$):

$$\widetilde{R}^{cld} = R_{sim}^{clr} - \Delta R_{obs}. \tag{8.2}$$

Both microwave and infrared radiances are included in $\widetilde{R}^{cld}$. The simulated clear-column radiance R_{sim}^{clr} is estimated from $\widetilde{R}^{cld}$ using a linear least-squares estimator. The error covariance of the estimate is given by Eq. 3.51:

$$\mathbf{C}_{\epsilon\epsilon} = E(\overline{R}_{sim}^{clr}\left[\overline{R}_{sim}^{clr}\right]^T) - E(\overline{R}_{sim}^{clr}\left[\overline{\widetilde{R}}^{cld}\right]^T)\left\{E(\left[\overline{\widetilde{R}}^{cld}\right]\left[\overline{\widetilde{R}}^{cld}\right]^T)\right\}^{-1} E(\overline{\widetilde{R}}^{cld}\left[\overline{R}_{sim}^{clr}\right]^T), \tag{8.3}$$

and it can easily be shown that $\mathbf{C}_{\epsilon\epsilon}$ depends only on $E(\overline{R}_{sim}^{clr}[\overline{R}_{sim}^{clr}]^T)$, $E(\overline{R}_{sim}^{clr}[\overline{\Delta R}_{obs}]^T)$, and $E(\overline{\Delta R}_{obs}[\overline{\Delta R}_{obs}]^T)$.

The error covariance $\mathbf{C}_{\epsilon\epsilon}$ was computed using two assumptions regarding the cross-covariance $E(\overline{R}_{sim}^{clr}[\overline{\Delta R}_{obs}]^T)$: 1) the cross-covariance is zero, or 2) the cross-covariance is the same as that calculated from simulated radiances. Fig. 8-4 shows the cloud clearing

results for three cases: 1) simulated radiances, 2) CAMEX-3 radiances (with zero correlation between ΔR_{obs} and R_{sim}^{clr}), and 3) CAMEX-3 radiances (using simulation-derived correlation between ΔR_{obs} and R_{sim}^{clr}). 2355 NAST–I/NAST–M/MAMS pixels were used in the analysis.

The CAMEX-3 results are worse than the simulation results by roughly a factor of two, but it must be noted that the NAST–I cloud impact observed during CAMEX-3 was approximately a factor of two larger than that exhibited by the simulated profiles. Similar analyses on a more statistically diverse set of NAST–I observations are needed for definitive conclusions. Nevertheless, the results shown here are consistent with the performance predicted by simulation studies.

8.3 Rejection of cloudy soundings

A neural network with a single hidden layer (10 sigmoidal nodes with transfer function given by Eq. 6.3) was trained to estimate the magnitude of the error made by the regression operator discussed previously. Soundings corresponding to high error estimates (relative to a pre-defined threshold) by the neural network were discarded.

8.3.1 Training and validation datasets

Clear-air radiances (microwave and infrared) simulated using the `cld2lr` profile database were combined with CAMEX-3 $\widetilde{R}^{cld}$ observations as described by Eq. 8.2. Training and validation sets were constructed as follows. Of the 2355 CAMEX-3 pixels for which NAST-I, NAST-M, and MAMS data are available, 1962 pixels were randomly assigned to the training set and 393 pixels were randomly assigned to the validation set. The `cld2lr` profile database was randomly segmented into 10,000 training vectors and 2,000 validation vectors. The 1962 CAMEX-3 training pixels were randomly matched to the 10,000 `cld2lr` training profiles, and the 393 CAMEX-3 validation pixels were randomly matched to the 2,000 `cld2lr` validation profiles. Each CAMEX-3 pixel was therefore matched to approximately five different `cld2lr` profiles. No correlation was assumed between ΔR_{obs} and R_{sim}^{clr}.

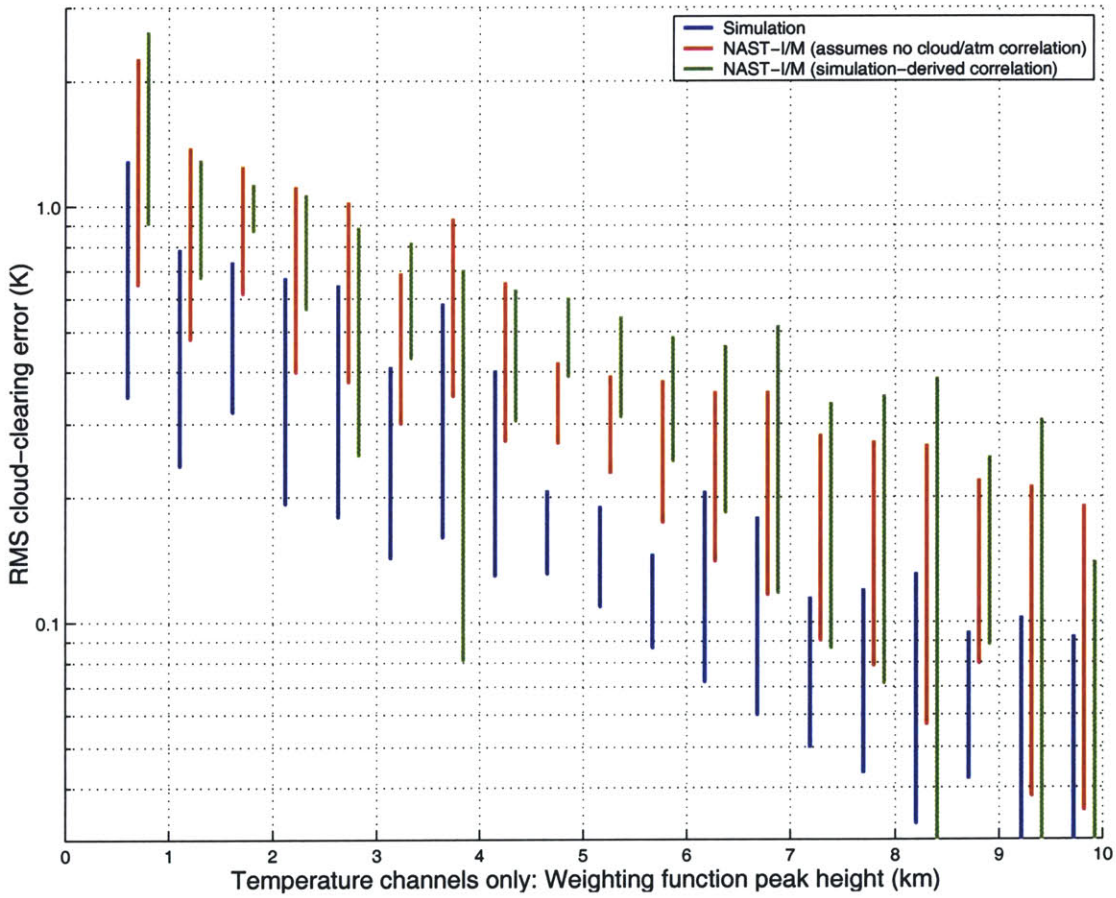

Figure 8-4: Comparison of simulated and CAMEX-3 observed cloud clearing performance. Results are shown in groups of three (from left to right): Simulation, NAST–I/M assuming no cloud and atmospheric correlation (offset to the right by 0.15 km), and NAST–I/M assuming model-based cloud and atmospheric correlation (offset to the right by 0.15 km).

8.3.2 Neural network input and output format

The input vectors were comprised of the 14 most-significant NAPC's of the infrared component of $\widetilde{R}^{cld}$ and 16 NAST-M channels (channel 9 of the 118-GHz system was not used). The network was trained to estimate one output: the mean of the trace of $\mathbf{C}_{\epsilon\epsilon}$ (given by Eq. 8.3).

8.3.3 Training algorithm

The Levenberg-Marquardt training algorithm was used. Training was typically stopped after approximately 50 epochs, corresponding to less than one hour on an AMD Athlon 1-GHz PC.

8.3.4 Performance results

The cloud-clearing performance improved as shown in Fig. 8-5 for the simulation case presented in Fig. 8-4. The "ideal" (dashed) curves in Fig. 8-5 represent the cloud-clearing error improvement if the soundings associated with the largest mean-squared cloud-clearing error are eliminated from the validation set. The neural network sometimes misidentifies soundings as excessively cloudy, and the resulting cloud-clearing errors (solid curves) are slightly worse than ideal. For example, the RMS error for the channels peaking below 1km dropped by ~ 0.15K after the neural network excluded 10% of the soundings. Similar analysis using statistics of aircraft observations showed that RMS single-pixel radiance cloud-clearing errors dropped below 1K for all channels peaking above 2km if 10% of the soundings were excluded by the neural network.

8.4 Discussion

The chapter concludes with a brief synopsis of other cloud-clearing experiments that were either attempted and produced inconclusive or specious results, or could not be attempted given the limited statistical diversity of the NAST cloud observations. Finally, implications of the results presented in this chapter for next-generation sounders such as the AIRS and (Cross-track Infrared Sounder) CrIS instruments are given.

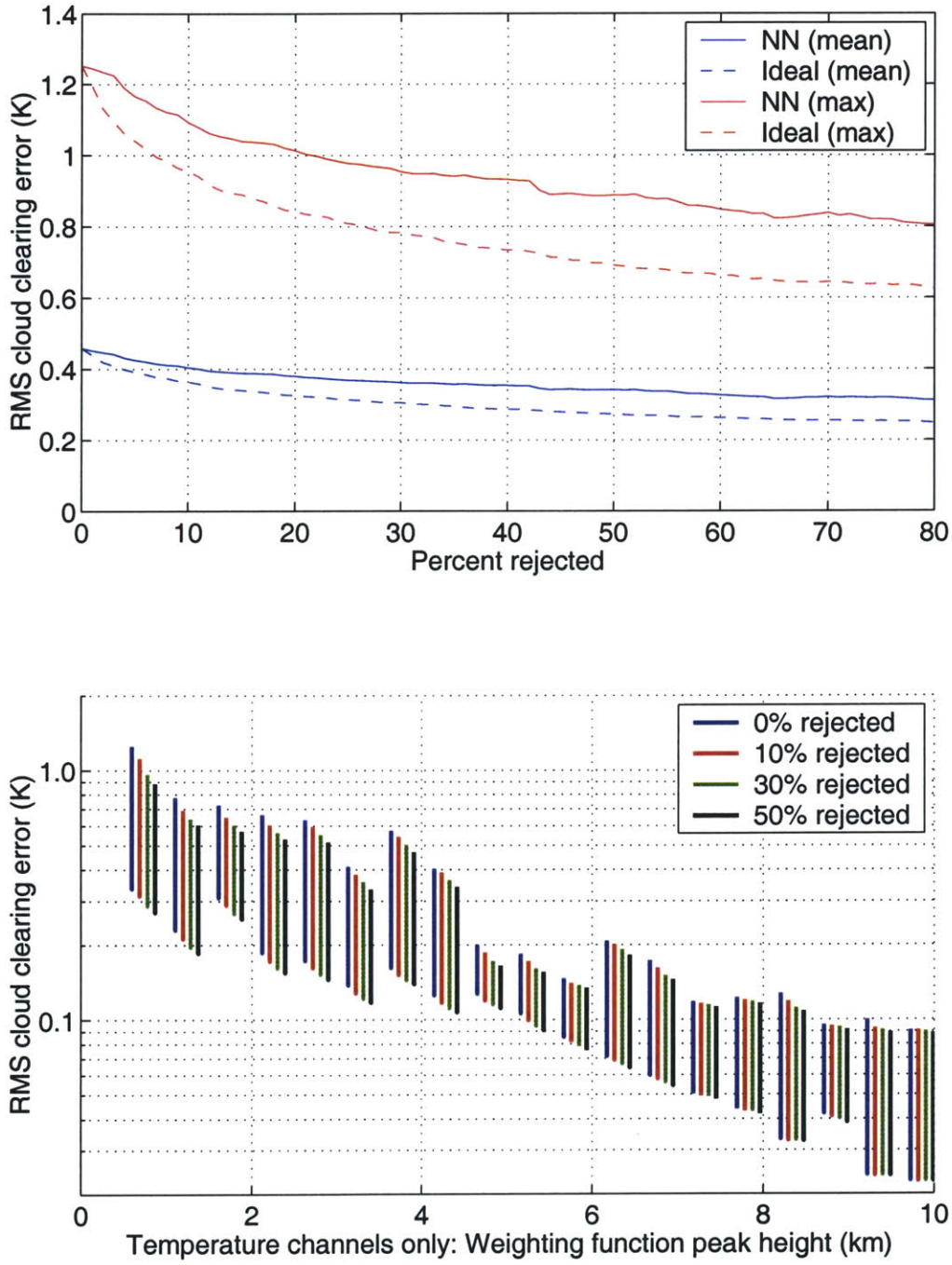

Figure 8-5: Simulated radiance performance improvement by rejection of soundings. Only simulated 0-10 km temperature channel errors are shown. Curves labeled "ideal" represent the reduction in error that would result if the neural network estimates of MSE are replaced with the true values of MSE. Results are shown on the top top graph for the worst-case (most transparent) temperature channel and the mean across all temperature channels. Results are shown on the bottom graph for four rejection criteria (from left to right): no rejections, 10% rejected, 30% rejected, 50% rejected.

8.4.1 Other cloud-clearing experiments

Cloud clearing of NAST data using statistics derived only from observations

The 2355 CAMEX-3 pixels were divided into a training and validation set. The training set was used to derive a linear least-squares estimator, which was applied to the validation set. The resulting cloud-clearing errors were very low (below 0.5 K RMS, for all channels). While encouraging, this level of performance is artificially good due to the limited degree of atmospheric variability present in the CAMEX-3 observations. Additional observations are needed to ensure that a "global" set of atmospheres comprises the training set. Such a training set could be assembled quite easily from satellite data (e.g., AIRS, which includes high-resolution visible/near-infrared channels that could be used to provide cloud truth), and the performance of statistical estimators could be reliably evaluated.

Nonlinear and multi-pixel cloud clearing

Simulated cloud-clearing studies in Chapter 6 demonstrated that nonlinear methods operating on multiple pixels achieved the lowest RMS cloud-clearing errors for all AIRS channels. Only linear, single-pixel methods were applied to NAST CAMEX-3 observations, due to the lack of a statistically-diverse dataset, and the failure of the cloud models to adequately represent the fine-scale spectral and spatial structure of CAMEX-observed clouds. Again, a comprehensive set of satellite observations would facilitate an examination of the more-advanced nonlinear and multi-pixel methods.

"Hole hunting" approaches

A final area worthy of study is the use of high-resolution infrared instruments to identify cloud-free pixels (the so-called "hole-hunting" approach). One potential liability of this method was demonstrated with MAMS data in Chapter 7–the inability to detect and correct slow drifts due to low-level clouds. However, it was also demonstrated that MAMS data was in most cases effective in identifying cloud-free areas. Further analyses with more comprehensive datasets are again warranted.

8.5 Summary

Principal components analysis of infrared cloud perturbations suggest large-scale spectral similarities between simulated and observed radiance data. Fine-scale spectral structure appears to differ substantially between simulated and observed cloud impact. Single-pixel cloud clearing performance for both simulated and observed radiance data is better than 1K RMS for all temperature channels peaking above 3km. Neural network rejection of 10 percent of soundings achieves better than 1K RMS performance for all temperature channels peaking above 2km. A dataset more statistically diverse than CAMEX-3 is needed to evaluate fully the statistical single- and multi-pixel cloud-clearing methods developed in Chapter 6. Nonetheless, results obtained for CAMEX-3 are consistent with simulation results, and suggest that results could be improved if a global CAMEX-3 training set were available.

Chapter 9

Conclusions

9.1 Summary of the thesis

THE objective of this thesis was to improve combined microwave/infrared atmospheric temperature retrievals, and the error characterization thereof, in the presence of clouds. This was accomplished in two areas: 1) The development of improved instrumentation (NAST-M) and collection of data, and 2) the development and evaluation of signal processing and estimation techniques, including novel combinations of principal components transforms, neural network estimators, and iterated order-noise estimates.

NAST-M provides the first high-resolution, collocated, multiband (54, 118 GHz) passive microwave measurements for three-dimensional temperature sounding, cloud and precipitation studies, and oxygen transmittance modeling. The 2.6-km resolution provided by the NAST suite also facilitates interpretation of unresolved satellite observations of meteorological phenomena. The three-point calibration on NAST-M provides accurate brightness temperature measurements in two oxygen bands, permitting the first reliably accurate temperature profile retrieval images. The utility of dual-absorption-band measurements for characterization of the drop size distribution and cell-top height for convective precipitation was illustrated by observations of rainbands in Hurricane Bonnie. The ER-2 platform provided high spatial resolution of this storm's structure, including its warm core. The warm core and convection morphology of Hurricane Bonnie (August, 1998) was clearly revealed both by aircraft-based microwave brightness temperature imagery and temperature

retrievals within the eye.

The ION algorithm was shown to be a very useful tool for the blind analysis of hyperspectral noise processes. The ION-estimated RMS variation of the spectrally uncorrelated component of NAST-I system noise is in excellent agreement with calibration analysis. ION was used to identify and characterize the spectrally and spatially correlated components of NAST-I system noise.

Principal components analyses on clear, cloudy, and clear-minus-cloudy simulated AIRS radiances revealed that cloud effects are the dominant source of radiance variation, but dominate only two to three degrees of freedom. The principal components of cloud impact are non-Gaussian, and respond to different cloud properties (e.g., cloud fraction and cloud height).

Neural network estimators were trained on projected principal components of simulated clear and cloudy global AIRS/AMSU data. The neural network single-pixel temperature retrieval outperformed the linear statistical method by over 30 percent (K RMS) for all levels of the atmosphere, and exhibited significantly less sensitivity to measurement noise. Cloud-cleared radiance errors less than 1 K were achieved for all atmospheric levels above 1 kilometer.

Both linear and nonlinear statistical methods yielded results superior to flag-based regression methods. The neural network nontuple cloud-clearing algorithm yielded the best results: residual RMS radiance errors for weighting functions peaking below 10 km of 0.1–1.2K and 0.1–0.7K for the 13–15 μm and 4.3–4.6 μm bands, respectively; no nontuples were rejected due to clouds. Combining microwave and infrared radiances at the pixel level improved performance by approximately 0.2 K in the most-transparent channels, although microwave surface emissivity uncertainties were a key contributor to residual errors.

NAST-M, NAST-I, and MAMS were used to assemble a radiance database of $\sim$2500 pixels from four daytime flights during CAMEX-3. Principal components analyses were performed on NAST-I observations, and the results were compared to those obtained with simulated data. Single-pixel cloud clearing performance for both simulated and observed radiance data is better than 1K RMS for all temperature channels peaking above 3km, based on all NAST cloud-impacted data plus global simulated clear-air radiances. Neural

network rejection of 10 percent of soundings achieves better than 1K RMS performance for all temperature channels peaking above 2km.

9.2 Main contributions

The main thesis contributions are summarized as follows.

9.2.1 NAST-M

The NAST-M sounder is a passive microwave spectrometer with 17 channels near the oxygen absorption lines at 50-57 GHz and 118.75 GHz. It is a cross-track scanning instrument with a 2.6-km diameter footprint at nadir, and cross-track field of regard of approximately 100 kilometers. It has collected over 300 hours of flight data and flown on two aircraft: the NASA ER-2 and the Scaled Composites Proteus.

9.2.2 Techniques for the blind characterization of hyperspectral noise processes

ION analysis on NAST-I data from CAMEX-3 and WINTEX demonstrated that the spatial correlation of noise is smaller than its spectral correlation, and sufficiently low that spatial filtering should reduce its effects significantly.

9.2.3 Novel techniques for the estimation of atmospheric temperature profiles and clear-column infrared radiances

Novel combinations of projected principal components transforms and neural networks were developed for infrared and infrared/microwave data that were computationally efficient relative to iterative nonlinear model-based methods. In addition, the single-pixel and nontuple neural network estimators yielded approximately 30 percent lower RMS retrieval errors than linear statistical methods and provided greater immunity to measurement noise. Statistical nontuple cloud-clearing methods were shown to be superior (in some case by more than 1 K in the most-transparent AIRS channels) to generalized adjacent-pixel (Backus-Gilbert clearing) methods.

9.2.4 Application of principal components analysis to simulated and observed NAST data to characterize cloud impact and its effect on cloud-clearing performance

Principal components analysis of NAST-I CAMEX-3 radiance perturbations due to clouds suggest large-scale spectral similarities between simulated and observed radiance data. Fine-scale spectral structure appears to differ between simulated and observed cloud impact, but this could be a statistical artifact.

9.3 Implications for next-generation sounders

The results obtained in the thesis have significant implications in terms of the development of operational algorithms for next-generation sounders such as AIRS and CrIS (Cross-track Infrared Sounder). The most important of these are summarized below.

9.3.1 Instrumentation issues

- Inclusion of microwave channels improved cloud clearing and temperature profile retrieval performance by approximately 0.2 K in the infrared window channels.

- Based on preliminary MAMS hole-hunting experiments, further experiment and study of clear-air misclassification is warranted.

- If next-generation interferometer sounders exhibit vibration-induced noise as NAST–I does, spatial smoothing might be needed. Experiments with NAST–I suggest such noise can be reduced to levels below nominal instrument thermal noise with modest 3×3 spatial averaging.

- Absolute calibration using aircraft instruments can be problematic because the calibration parameters of NAST–M and other microwave instruments at altitude are different from calibration parameters at the surface.

9.3.2 Algorithm issues

- The first statistical, multi-pixel cloud-clearing algorithm was developed and evaluated, and shown to achieve performance superior to generalized adjacent-pixel approaches for simulated global AIRS/AMSU radiances.

- A new altitude-based method for the operational benchmarking of cloud-clearing performance was introduced. The performance at a given altitude depends on the channel weighting function width.

- Neural networks were superior to LLSE methods for cloud clearing simulated global AIRS/AMSU radiances.

- Analyses of a limited set (not global) of NAST observations suggest statistical cloud-clearing methods could be effective, but further study with a more comprehensive set of measurements is needed.

- Analyses of both simulated and observed infrared cloud impacts indicate that the number of degrees of freedom due to clouds is small, but the total contribution to radiance variance is large.

- The rejection of bad (excessively cloudy) pixels helps reduce cloud-clearing error variances, but not dramatically.

9.3.3 New products

- Based on NAST–M results presented in the thesis, next-generation microwave sounders such as ATMS (Advanced Technology Microwave Sounder, 30-km spatial resolution) should be able to provide improved retrievals of cloud particle size and cell-top height on a global scale.

- NAST–M also demonstrated the potential of high-resolution microwave sounders to penetrate the eyes of hurricane and measure warm anomalies and deduce hurricane energy dynamics.

9.4 Suggestions for further research

9.4.1 NAST-M upgrades

The addition of radiometers with spectral coverage near 23, 31, 183, and 425 GHz would allow the estimation of atmospheric water vapor profiles and improve the estimation of temperature profiles, cloud microphysical properties, and precipitation rates. Additionally, the performance of the high-frequency bands could better model current and future NPOESS instruments, and could be evaluated as a potential component of a geostationary instrument, where small antenna size is of paramount importance.

9.4.2 NAST-I noise characterization

The analyses performed in Chapter 5 could be applied to a more extensive dataset, including calibration data. Detailed comparisons with model-based results are also warranted.

9.4.3 Simulation analyses

Improved cloud models (frequency-dependent, two-formation, for example) and surface models (frequency-dependent or varying every 15-km, for example) could be evaluated and compared with NAST observations.

9.4.4 NAST observations: cloud clearing and temperature profile retrievals

Cloud clearing and temperature profile retrieval analyses using a globally-representative (geographically and seasonally) set of NAST observations over a variety of surfaces (land, snow, etc.) and cloud conditions would provide a better understanding of how performance is affected by atmospheric conditions. A more thorough comparison of simulated and observed radiance statistics could also be carried out.

Appendix A

Construction and Characterization of NAST-M Blackbody Targets

by: M. J. Schwartz and W. J. Blackwell

TWO blackbody calibration targets (20cm × 20cm × 4cm) were fabricated from aluminum and iron-loaded epoxy. Both loads have surfaces covered with tessellated pyramids machined from Emerson-Cuming CR-112 Eccosorb that are 12-mm tall with square bases with 8-mm sides. The aluminum backing extends up into the cores of the pyramids to minimize temperature gradients between pyramid bases and tips. Rectangular channels with square edges 4-mm wide and 3-mm deep were cut into the aluminum, thereby increasing the volume of absorbing material at the base of the pyramids and increasing return loss with negligible change in thermal conductivity. Thin-film platinum resistive temperature device (RTD) sensors were placed on the surface of the Eccosorb, embedded in the Eccosorb, and epoxied to the back of the aluminum. The loads are insulated on the front with a 1-cm thick layer of Styrofoam and on the back and sides with extruded polystyrene. Figure A-1 shows a detail of the load design. One-inch thick aluminum was machined to give a bed of pointed pegs. The pyramidal tips were machined with a carbide tool with a 3:1 angle on its flutes. A thin wall of unmachined aluminum around the perimeter forms the basin into which the castable resin (CR) was poured and permits placement of heaters to minimize lateral temperature gradients in the hot load. Eccosorb CR-112 was poured

over the aluminum structure and then the pyramidal faces were machined into the Eccosorb with the tool raised 3 mm above the level at which the pyramidal tips were machined on the aluminum cores. The resulting layer of Eccosorb is just less than 1 mm thick on the tips of the pyramids and ranges from 3 mm to 6 mm thick in the valleys between the pyramids. Machining was done by the MIT Central Machine Shop[1].

A.1 Eccosorb Casting

The Eccosorb was cast following the manufacturer's recommendations, as follows. Parts X and Y were heated to $150°F$, thoroughly mixed individually, using a paddle on an electric drill. The resin (Part X) must be stirred prior to measuring, as it becomes stratified in storage. The two parts were mixed in a coffee can, placed in a vacuum chamber[2] and bubbles were removed by vacuum evacuation (~8 minutes.) Care must be taken to lower the pressure slowly or the mixture will foam out of the container. The mixture has a pot-life of 1 hour, and significant stratification is observed in cross-sections of samples cured at room temperature. After the mixture stopped foaming, it was immediately poured into the pre-heated aluminum load and bubbles were removed a second time by vacuum evacuation. The load was placed in an oven at $200°F$ for 1 hour and then the temperature was reduced to $165°F$. Shrinkage of the epoxy during curing resulted in some cracking between the aluminum sides and the absorber in the corners of the loads. This shrinkage may be reduced with slower (cooler) curing. The increased stratification of the absorber may be an advantage if stronger absorption is desired deeper in the load.

Further experiments should be made in the curing process. The temperature of the oven might be reduced as soon as the viscosity of the epoxy resin has increased sufficiently to inhibit settling of the suspended absorber particles. Over a period of several days, the material will cure even at room temperature, and low temperature curing is expected to minimize shrinkage, or at least to allow relaxation of the material into the mold as it shrinks.

[1] Andrew Gallant (617-258-0789), agallant@MIT.EDU

[2] The vacuum chamber was borrowed from the Physics Junior Lab, Jay Kirsch, 617-253-3421, jkirsch@MIT.EDU

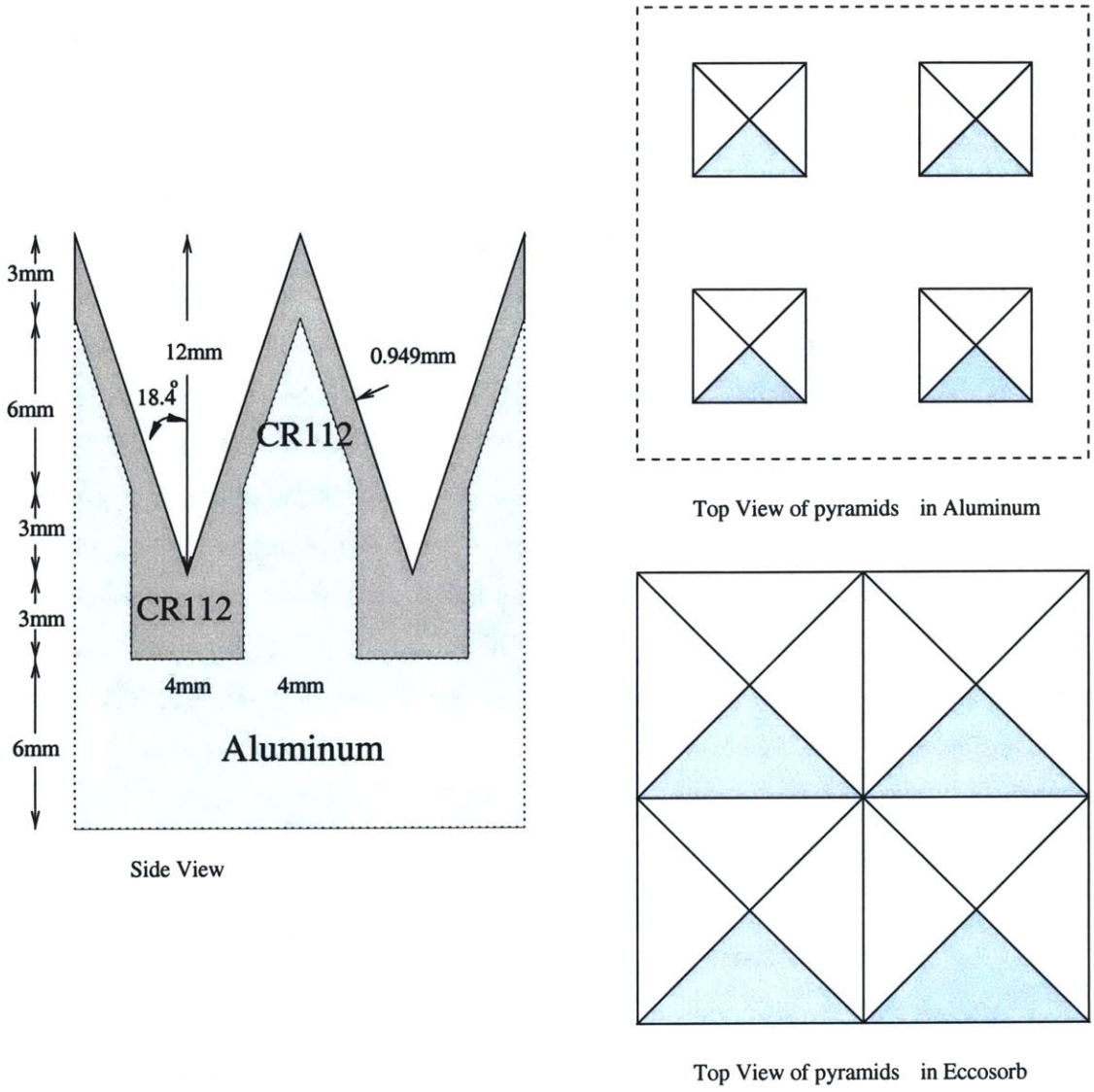

Figure A-1: Calibration load detail.

A.2 RTD Placement

The temperature sensors used in calibration load construction are the same sensors used throughout the NAST-M package, only lacking the plastic packaging and pre-wiring. The sensor elements are thin-film, platinum RTDs (Minco 5245PD) and their small size (1.6 mm $\times$ 2.2 mm $\times$ 1.1 mm) allows them to be embedded within the loads after construction without significantly altering their microwave or thermal properties. RTDs were placed in the tips of the aluminum spikes by drilling holes (approximately 1 mm in diameter) through to the back of the aluminum, filing a slightly larger socket in the tip, and epoxying the sensor into place with thermally-conductive epoxy. Four-wire connections to the sensors were made with extremely fine-gauge, low-thermal-conductivity wire. It was difficult to work with this wire, and its use was probably not warranted. RTDs were also placed against the sides of the spikes in the trenches between the pyramids. The hot load has additional RTDs on the back and sides of the aluminum block and on the surface of the pyramids in a corner of the load. Wires for the RTD on the load surface run over the surface of the load for a short distance. Multiple RTDs allow control and independent monitoring of the hot load temperature and reveal gradients as large as 3 K from the front to back of the hot load during flights. The positions of the sensors are shown in Figures A-2 and A-3. RTD A was disconnected before CAMEX-3 because of a wiring problem, and is no longer used.

A.3 Load Insulation

The loads are insulated by a 1-cm layer of Styrofoam insulation. Extruded polystyrene cut from bats intended for house construction were used on the sides and back of the load. It is less crumbly than expanded polystyrene (AKA Dylite, AKA coffee-cup material.) Expanded polystyrene, also from home insulation bats, was used for the window. Extruded polystyrene is lossy (typically 1 dB/ft near 100 GHz) since the demise of CFCs as a foaming gas[83]. The window surface is covered with a thin film (HR500/2S).

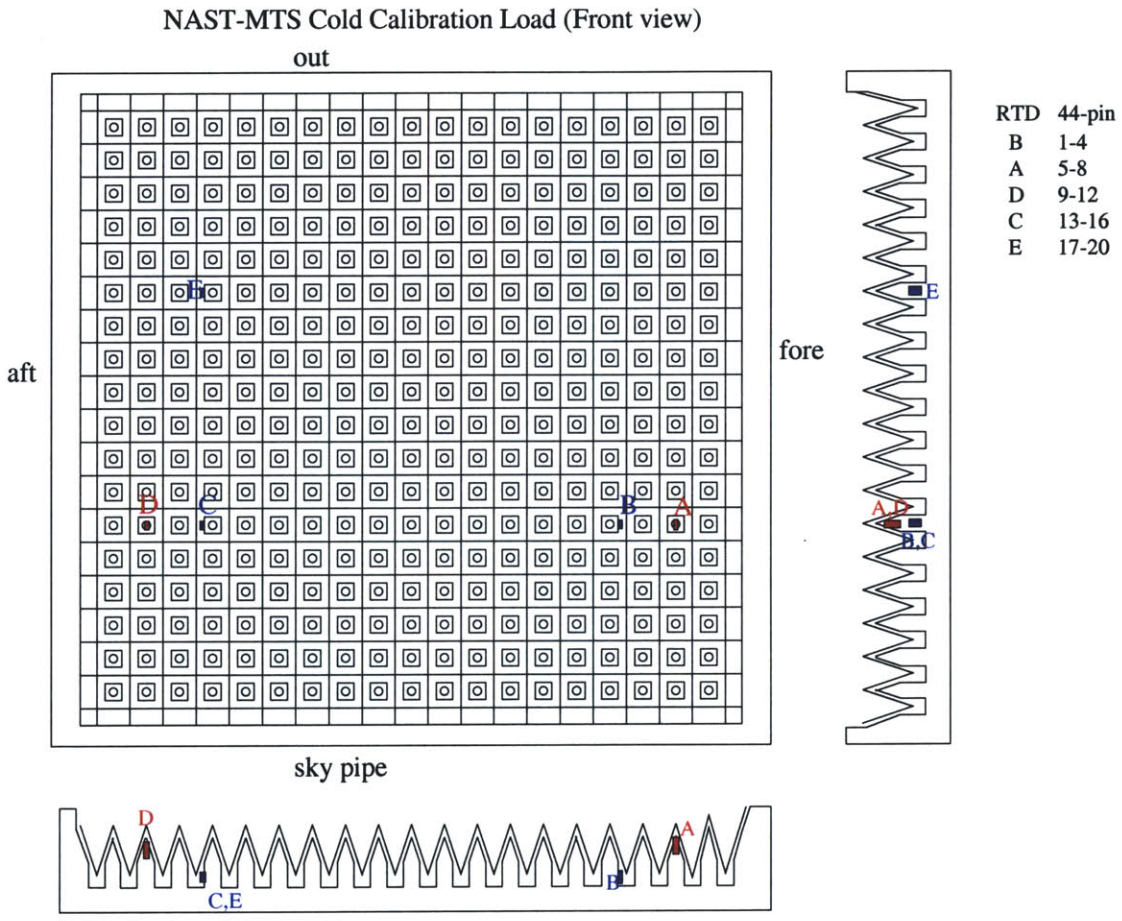

Figure A-2: Cold load RTD placement.

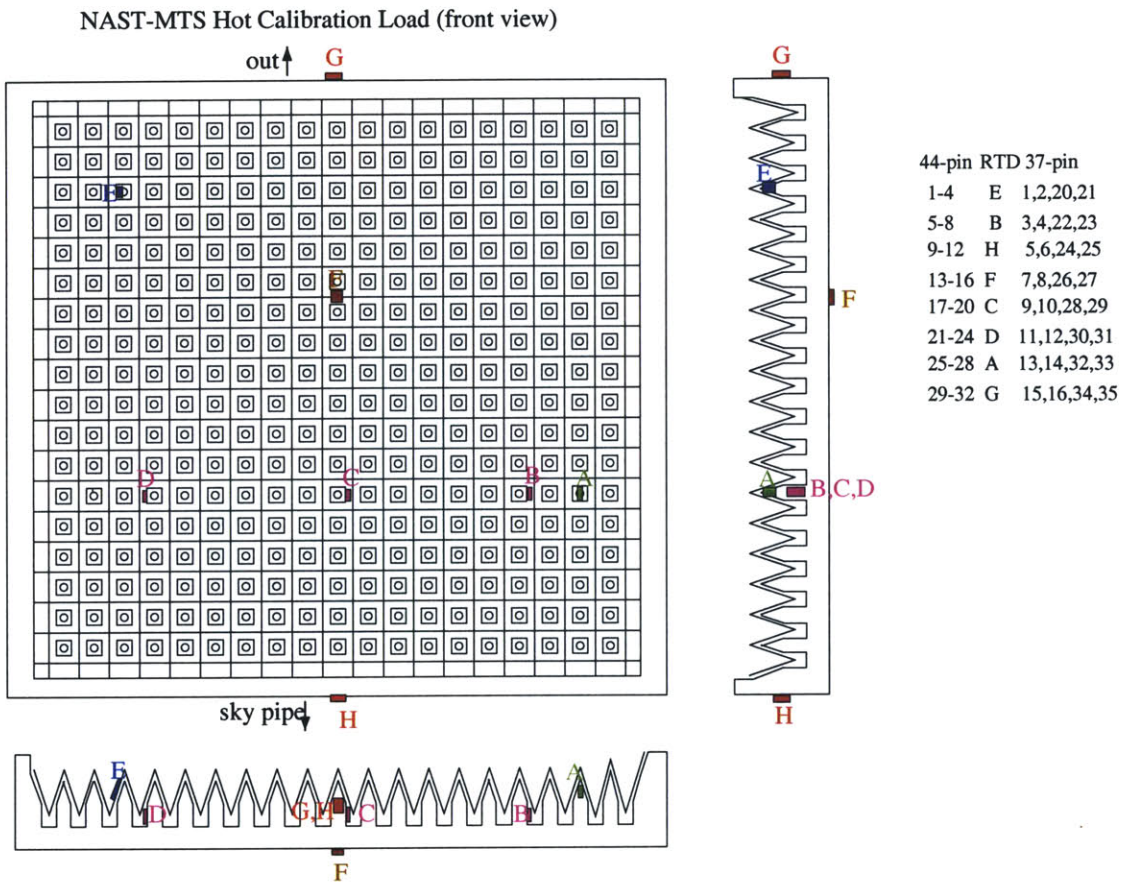

Figure A-3: Hot load RTD placement.

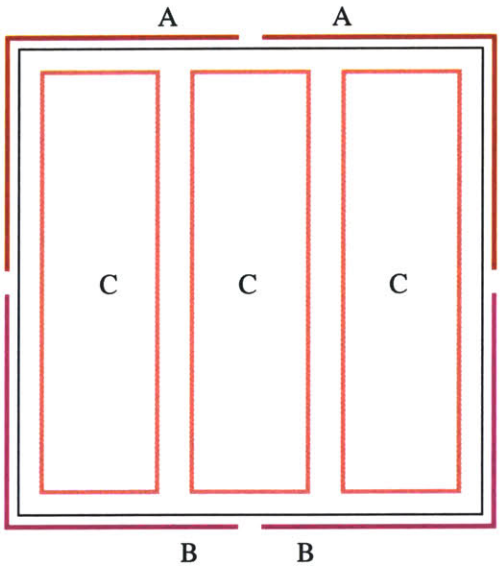

Figure A-4: Hot load heater placement.

A.4 Hot Load Heater Placement

Minco Thermofoil heaters were placed on the back and sides of the hot load's aluminum backing. The heaters are controlled in three banks, labeled 'A', 'B' and 'C' in Figure A-4. Heaters labeled A and B are 1 in by 6 in and those labeled C are 2 in by 6 in. All heaters provide 5 W/in^2 when operated at 115 V. Relays allow each bank to be run either in parallel or in series, so the heaters on the sides (A and B) can provide either 120 W or 30 W and the heaters on the back can provide either 180 W or 20 W. The hardware exists for independent computer control of the banks.

A.5 Load Emissivity Measurement

Measurements were made of the load surface reflectivity in the lab at MIT by viewing the load with the radiometer while changing the scene behind the radiometer from 293 K to 77 K. The observation was repeated with the load replaced by a polished aluminum surface. This measurement was made at approximately 45° incidence angle, while the load is viewed at near normal incidence in flight. The load reflectivity was estimated to be less than

1.5%, but the measurement was considered to be somewhat crude. Further measurements were made with the help of Ed Tong at the Harvard Smithsonian Center for Astrophysics using an HP8510C high-frequency network analyzer and an HP85105A mm-wave controller. Figure A-5 shows the time-domain reflections from the hot load. The approximately -40 dB reflection at 0.6 ns is a reflection from the feed horn used in the test setup. Reflections at markers 2 and 3 are from the front and back of the Styrofoam insulation respectively. The difference in round-trip times of 0.1 ns corresponds to the insulation thickness of 1.5 cm. The most significant reflection (marker 1) is from a surface 2.44 cm deeper than the back of the insulation, which is the level of the aluminum plane at the bottom of the moats between pyramids. Sweeps through two accessible bands of the test setup (75-110 GHz and 185-205 GHz) are shown in Figures A-6 and A-6. These figures show total reflections to be more than 25-dB below the baseline established with an aluminum plate. Values for the extremal frequencies in the higher-frequency band are not considered to be reliable. The time-domain plot, Figure A-5 indicates that load performance is, in fact, considerably better than expected. The largest reflection is actually due to mismatch in the horn of the test-setup. The largest return from the load is 44-dB below the baseline. It is reasonable to believe that this behavior holds for the NAST-M 118-GHz band. Frequencies comparable to the 50-57-GHz NAST-M band were not accessible to the test setup at the CFA, however it is reasonable to believe that returns are far less than the -20-dB level and likely better than -30-dB. The largest Bragg peak from the periodic load surface would be straight backwards, and no large changes in backscatter intensity were seen for small changes in the viewing angle, so there appears to be no measurable Bragg scattering from the periodic pyramidal structure. The combination of a -30-dB load (99.9% emissivity) with a metallic shroud on the scan mirror redirecting reflections back to the load makes error due to load reflectivity inconsequential compared to sources of error such as temperature gradients in the hot load.

Observation from 20-km altitude of cold-space up the sky-pipe provides a further test of load emissivity and instrument linearity, however there are significant temperature gradients from the front to the back of the hot load resulting in uncertainties in the correct effective microwave temperatures of the loads. Cold-sky observations are consistent with theory to within the uncertainty of the load temperature.

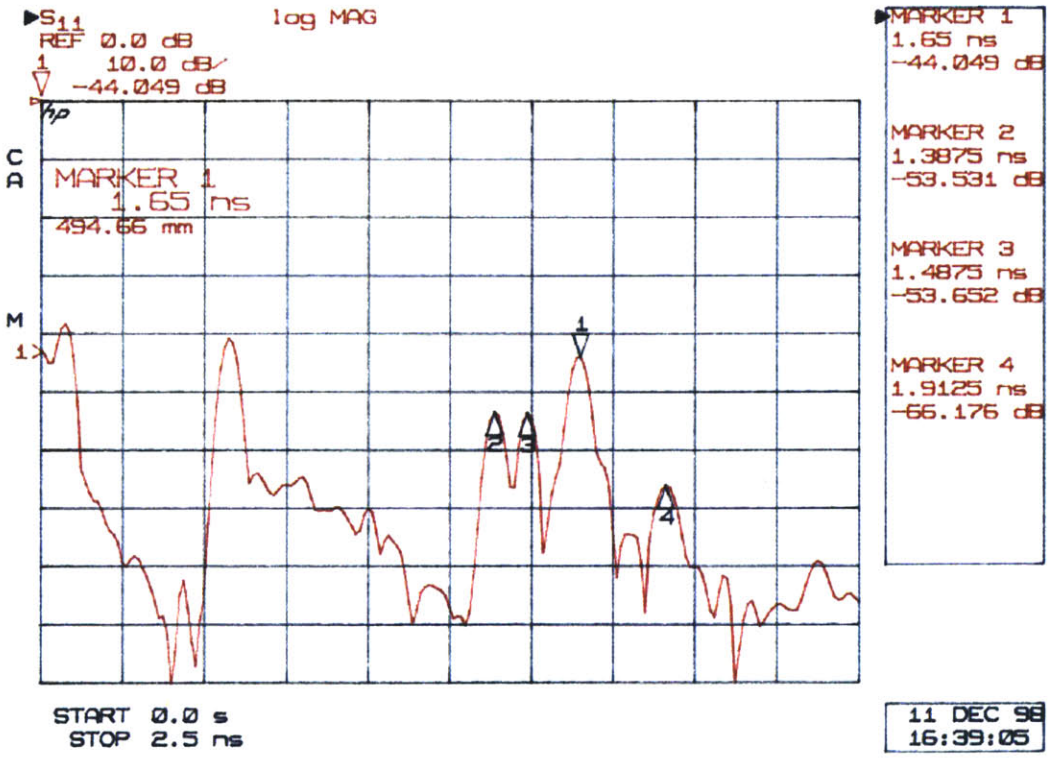

Figure A-5: Time domain reflection from the NAST-M hot load on CFA network analyzer for 75-110-GHz band.

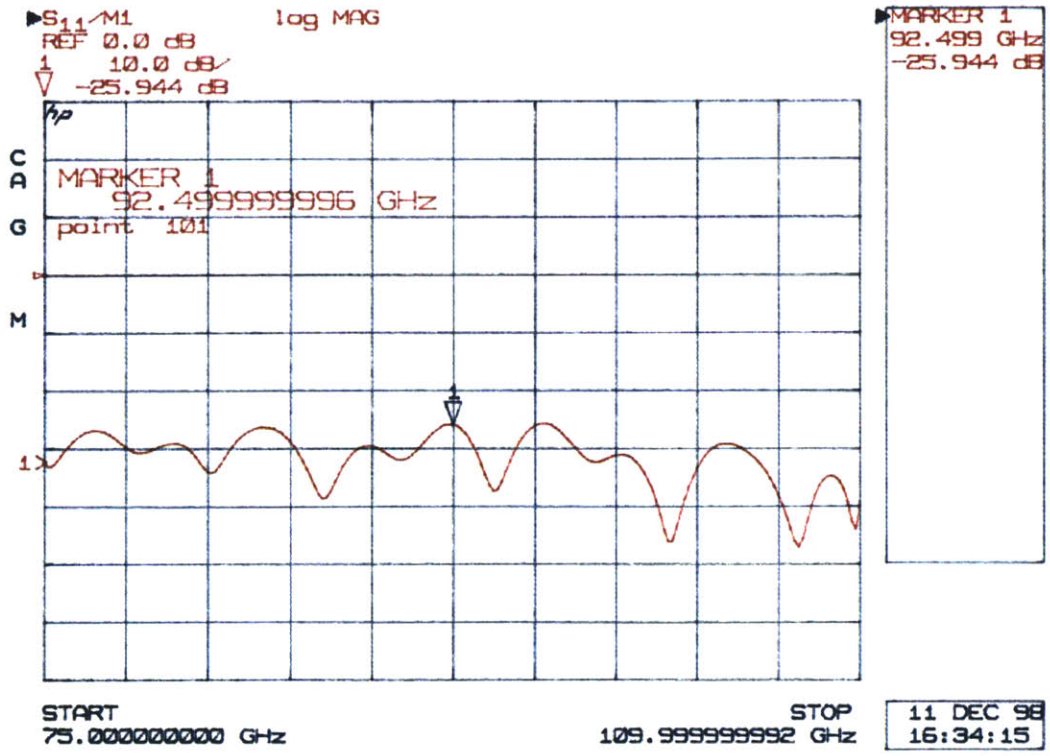

Figure A-6: NAST-M hot load on CFA network analyzer for 75-110-GHz band.

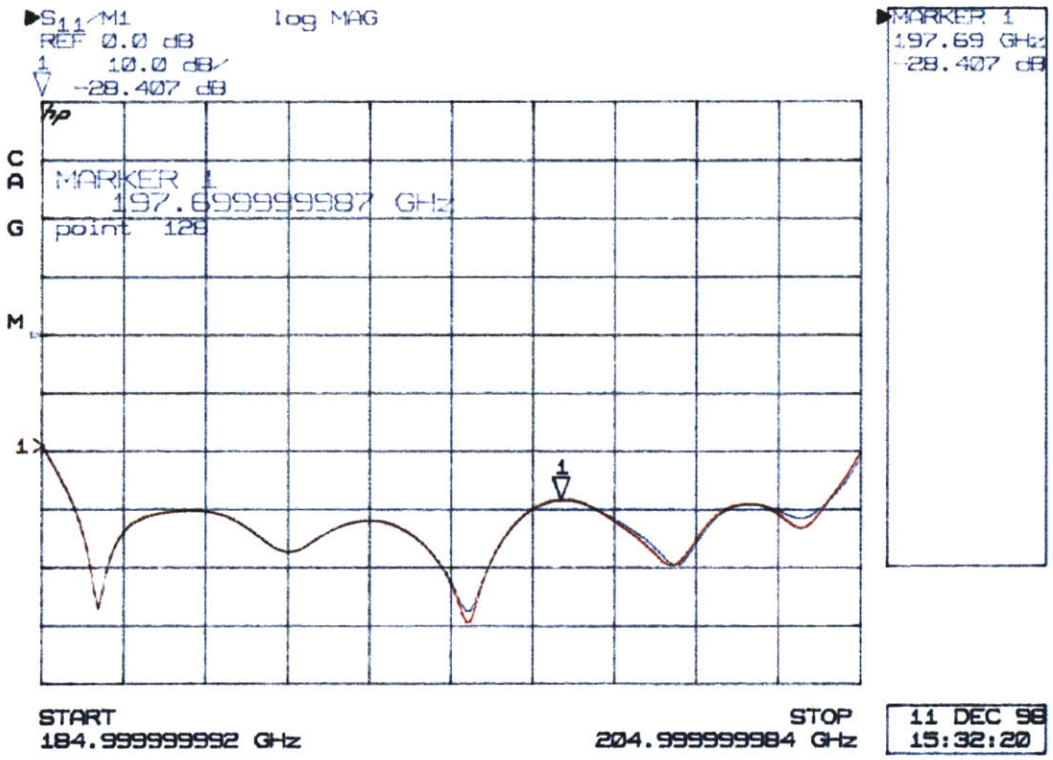

Figure A-7: NAST-M hot load on CFA network analyzer for 185-205-GHz band.

Appendix B

Calibration of the NAST-M Temperature Measurement System

THE calibration of the system used for monitoring the thermodynamic tempertures of the internal blackbody targets is considered in this chapter. Platinum Resistive Devices (PRD's)[1] are used throughout the instrument to monitor the physical temperatures of various components. The resistance of these devices changes as a function of temperature. The RTD's used in NAST-M have a nominal resistance of 100 ohms at a temperature of 273.15 K (0° C) and an average temperature coefficient of resistance (ohms/°C) of 0.385. Four-wire sensors are used to minimize error due to lead resistance. The resistances of the RTD's are measured using a custom measurement board, consisting of a precision, programmable current loop and precision multiplexing/data conversion circuitry.

B.1 Calibration of the RTD Measurement Board

The RTD measurement board uses a 500–μA precision current source to excite a number of sensors in series. The voltage of a particular sensor is read by a multiplexed A/D converter. A three-point calibration that relates resistance to voltage was determined empirically using three precision loads (75.00, 100.00, and 123.64 ohms, ±0.01%, Precision

[1]PRD's are a subclass of sensors sometimes called "Resistive Temperature Devices" or RTD's

Resistive Products, 319-394-9131).

- **Step One.** Uncorrected resistance is calculated by assuming ideal RTD and ideal current source as follows:

$$R_o = V/100.0/0.0005 \tag{B.1}$$

  where V is the voltage across the RTD measured by the A/D converter.

- **Step Two.** Resistance estimates are corrected as follows:

$$R = aR_o^2 + bR_o + c \tag{B.2}$$

  where a, b, and c are calibration coefficients listed in Table B.1.

B.2 Calibration of RTD's

The relation between resistance and temperature for the 100-ohm platinum RTD is slightly nonlinear. A sixth-degree polynomial is used to approximate the nonlinear function:

$$T(R) = c_6R^6 + c_5R^5 + c_4R^4 + c_3R^3 + c_2R^2 + c_1R^1 + c_0. \tag{B.3}$$

The polynomial coefficients are given in Table B.2.

B.2.1 Calibration of blackbody target RTD's

After applying the nonlinear correction (Eq. B.3), the calibration accuracy of a given RTD is approximately ± 0.5 K due to manufacturing inconsistencies. This level of accuracy is acceptable for all measurements of temperature throughout NAST-M, with the exception of the blackbody targets. To improve the calibration accuracy, a Lakeshore silicon-diode temperature sensor (DT-471) was factory calibrated to an accuracy exceeding 50 mK. All RTD's used in the blackbody targets were calibrated to the DT-471 standard by bringing all sensors to thermal equilibrium in an insulated heating/cooling chamber. The RTD's

#	a	b	c
0	0.00013869229702	1.05576801631774	-0.18566149495149
1	0.00013927449982	1.05601524836986	-0.21071077543916
2	0.00014101188313	1.05583499326767	-0.21265659171809
3	0.00016028346776	1.05219005877087	-0.05570389095445
4	0.00016359095485	1.05165643333656	-0.01735960966884
5	0.00015449511948	1.05366333238956	-0.14613707227112
6	0.00012924860764	1.05777736501955	-0.32908159419907
7	0.00012123259481	1.06015684751307	-0.46752392030633
8	0.00014737301848	1.05296646036783	0.06926327292225
9	0.00013661152544	1.05478226766433	0.01041032855573
10	0.00015526736239	1.05131521323410	0.15657835900886
11	0.00027867551630	1.02281074165312	0.98765315257951
12	0.00016965491704	1.04808488190457	0.27515788488792
13	0.00027329381037	1.02340523687909	0.83660973808433
14	0.00015832269411	1.05045324866483	0.13125171044363
15	0.00018046449911	1.04680581738250	0.29443331096929
16	0.00019307408658	1.04463145740507	0.36997423747051
17	0.00016305632600	1.05011961063873	0.12773389760980
18	0.00017014051452	1.04896097290410	0.17623048835974
19	0.00014204448793	1.05444988738145	-0.06501945085812
20	0.00018157776638	1.04510265591639	0.26204372425943
21	0.00016431337341	1.05021599143216	0.12158037553736
22	0.00016170307567	1.05069217328780	0.09240611245069
23	0.00012879666351	1.05328255971528	0.23304204528815
24	0.00012649508141	1.05346450033812	0.23388196317961
25	0.00013180089015	1.05275604272476	0.25598169443189
26	0.00013853491999	1.05130575862168	0.32099509243437
27	0.00012036610526	1.05473580636087	0.18189976755318

Table B.1: Calibration coefficients for the 28 sensors used in the WINTEX deployment: March 1999

c_6	c_5	c_4	c_3
7.529465633863e-15	-6.212690630360e-12	1.984031043875e-09	-2.924240004671e-07

c_2	c_1	c_0
1.725939573222e-03	2.240811515378e+00	-2.421646935839e+02

Table B.2: Polynomial coefficients for 100-ohm platinum RTD.

Operating point	RTD 1 (E)	RTD 2 (B)	RTD 3 (H)
335 K	1.20	0.45	0.70

RTD 4 (F)	RTD 5 (C)	RTD 6 (D)	RTD 7 (G)
1.10	.37	-0.63	.77

Table B.3: Calibration offsets (K) for hot load RTD's. The operating point at which the calibration was performed is given in the leftmost column. The RTD designations used in Appendix A is denoted in parentheses.

Operating point	RTD 1 (B)	RTD 2 (A)	RTD 3 (D)	RTD 4 (C)	RTD 5 (E)
235 K	.48	.64	.44	.42	.46
297 K	.325	.28	.20	.15	.15

Table B.4: Calibration offsets (K) for ambient load RTD's. The operating points at which the calibrations were performed is given in the leftmost column. The RTD designations used in Appendix A is denoted in parentheses.

used in the heated target were calibrated at 335 K, and the RTD's used in the ambient target were calibrated at 297 K and 235 K. The biases (i.e., values that must be added to uncalibrated RTD readings) for the RTD's in the heated target are given in Table B.3, and the biases for the RTD's in the ambient target are given in Table B.4.

Appendix C

Laboratory Measurements of NAST-M Antenna Beam Spillover

AN antenna beam spillover problem affecting views of the internal calibration targets for the 54-GHz radiometer results in a correctable worst-case absolute calibration bias of ~3 K in the transparent channels. The "corrupted" temperature of the ambient/heated load can be modeled as a linear combination of the spillover through the zenith port, the spillover through the nadir port, and the "true" load temperature (i.e., the brightness temperature [T_A or T_H] that would be observed if there were no spillover) as follows:

$$T'_A = \eta_Z^A T_Z + \eta_N^A T_N + (1 - \eta_Z^A - \eta_N^A) T_A \tag{C.1}$$

$$T'_H = \eta_Z^H T_Z + \eta_N^H T_N + (1 - \eta_Z^H - \eta_N^H) T_H \tag{C.2}$$

The antenna beam coupling coefficients (η values) for each radiometer channel can be accurately measured in the laboratory by switching calibration loads of known temperature in and out of the zenith and nadir ports of NAST-M. A sketch of the laboratory setup is shown in Figure C-1.

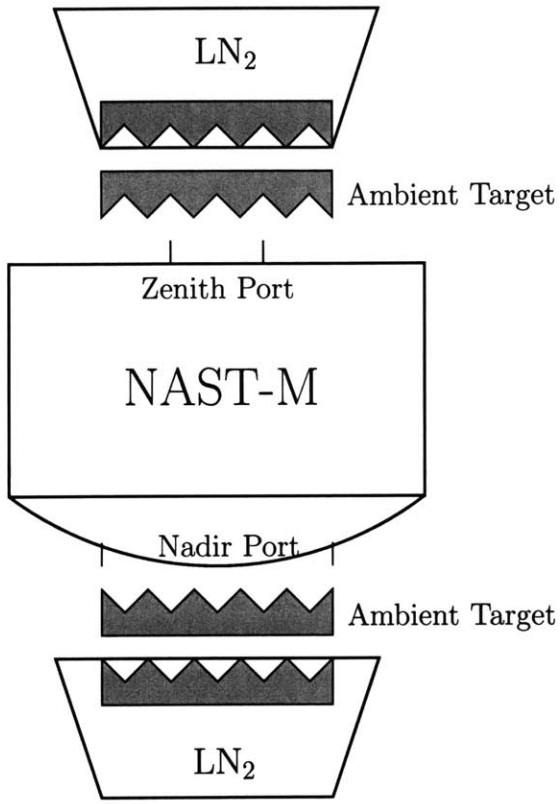

Figure C-1: Labatory setup for measurement of antenna beam spillover.

C.1 Experimental procedure

By switching the ambient targets in and out of the zenith and nadir ports, four operating modes are possible: $\frac{295K}{295K}$ (ambient target at zenith and nadir ports), $\frac{295K}{77K}$ (ambient target at zenith port and liquid nitrogen at nadir port), $\frac{77K}{295K}$ (liquid nitrogen at zenith port and ambient target at nadir port), and $\frac{77K}{77K}$ (liquid nitrogen at zenith and nadir ports). Fifty spots are observed for each of the operating modes: [Z 24H N 24A], where the following notation is used: Z = zenith port, H = heated load, N = nadir port, and A = ambient load. The heated and ambient targets are observed at 24 angles ranging from 40.2° to 67.8° in steps of 1.2°. A "superscan" contains 200 spots defined as:

$$\left[\text{[Z 24H N 24A]}_{\frac{295K}{77K}} \quad \text{[Z 24H N 24A]}_{\frac{295K}{295K}} \quad \text{[Z 24H N 24A]}_{\frac{77K}{295K}} \quad \text{[Z 24H N 24A]}_{\frac{77K}{77K}} \right] \tag{C.3}$$

Six superscans were collected (and averaged) for two cases: 1) The reflector was moved to the far end of the shaft, and 2) The reflector was positioned as close as possible to the chain sprocket. The total travel between these two points is approximately 3/8". The two nadir views of liquid nitrogen and views of the intenal ambient target near 54° observed during the $\frac{295K}{295K}$ cycle were used as calibration points. The following sections describe the mathematical methods used to calculate the antenna beam coupling coefficients.

C.2 Regression solutions for the beam coupling coefficients

Equations C.1 and C.2 can be applied to each of the four test cases, giving eight equations and four unknowns (i.e., the four beam coupling coefficients). The resulting overdetermined system of linear equations can then be solved using regression. The case of the heated target is slightly more complicated, because T_H is not well-known. T_H is therefore considered unknown, and a linear system of equations (with four equations and three unknowns) can be solved using regression, as before.

C.2.1 Case I: Ambient internal target

Equation C.1 can be rewritten as follows:

$$(T'_A - T_A) = (T_Z - T_A)\eta_Z^A + (T_N - T_A)\eta_N^A. \tag{C.4}$$

Equation C.4 can be expressed in matrix form by including the terms for all four test cases as follows:

$$\mathcal{T}_A = \begin{bmatrix} T'_A(\frac{295K}{77K}) - T_A \\ T'_A(\frac{295K}{295K}) - T_A \\ T'_A(\frac{77K}{295K}) - T_A \\ T'_A(\frac{77K}{77K}) - T_A \end{bmatrix} \tag{C.5}$$

$$\mathbf{A}_A = \begin{bmatrix} T_Z(77K) - T_A & T_N(295K) - T_A \\ T_Z(295K) - T_A & T_N(295K) - T_A \\ T_Z(295K) - T_A & T_N(77K) - T_A \\ T_Z(77K) - T_A & T_N(77K) - T_A \end{bmatrix} \tag{C.6}$$

$$\boldsymbol{\eta}_A = \begin{bmatrix} \eta_Z^A \\ \eta_N^A \end{bmatrix} \tag{C.7}$$

where $T_N(77K)$ is defined to be 79.5 K (warming of ~2.5 K assumed, due to the reflection of radiation off the surface of the liquid nitrogen [84, 85]) and $T_Z(77K)$, $T_Z(295K)$, and $T_N(295K)$ are determined from calibrated radiometric data. The matrix equation to be solved is therefore

$$\mathcal{T}_A = \mathbf{A}_A \boldsymbol{\eta}_A \tag{C.8}$$

with solution

$$\boldsymbol{\eta}_A = (\mathbf{A}_A^T \mathbf{A}_A)^{-1} \mathbf{A}_A^T \mathcal{T}_\mathcal{A}. \tag{C.9}$$

C.2.2 Case II: Heated internal target

Equation C.2 can be expressed in matrix form by including the terms for all four test cases as follows:

$$\mathcal{T}_\mathcal{H} = \begin{bmatrix} T'_H(\frac{295K}{77K}) \\ T'_H(\frac{295K}{295K}) \\ T'_H(\frac{77K}{295K}) \\ T'_H(\frac{77K}{77K}) \end{bmatrix} \tag{C.10}$$

$$\mathbf{A}_H = \begin{bmatrix} T_Z(77K) & T_N(295K) & 1 \\ T_Z(295K) & T_N(295K) & 1 \\ T_Z(295K) & T_N(77K) & 1 \\ T_Z(77K) & T_N(77K) & 1 \end{bmatrix} \tag{C.11}$$

$$\boldsymbol{\eta}_H = \begin{bmatrix} \eta_Z^H \\ \eta_N^H \\ (1 - \eta_Z^H - \eta_N^H) T_H \end{bmatrix} \tag{C.12}$$

where $T_N(77K)$, $T_Z(77K)$, $T_Z(295K)$, and $T_N(295K)$ are defined as before. The matrix equation to be solved is therefore

$$\mathcal{T}_H = \mathbf{A}_H \boldsymbol{\eta}_H \tag{C.13}$$

with solution

$$\boldsymbol{\eta} = (\mathbf{A}_H^T \mathbf{A}_H)^{-1} \mathbf{A}_H^T \mathcal{T}_\mathcal{H}. \tag{C.14}$$

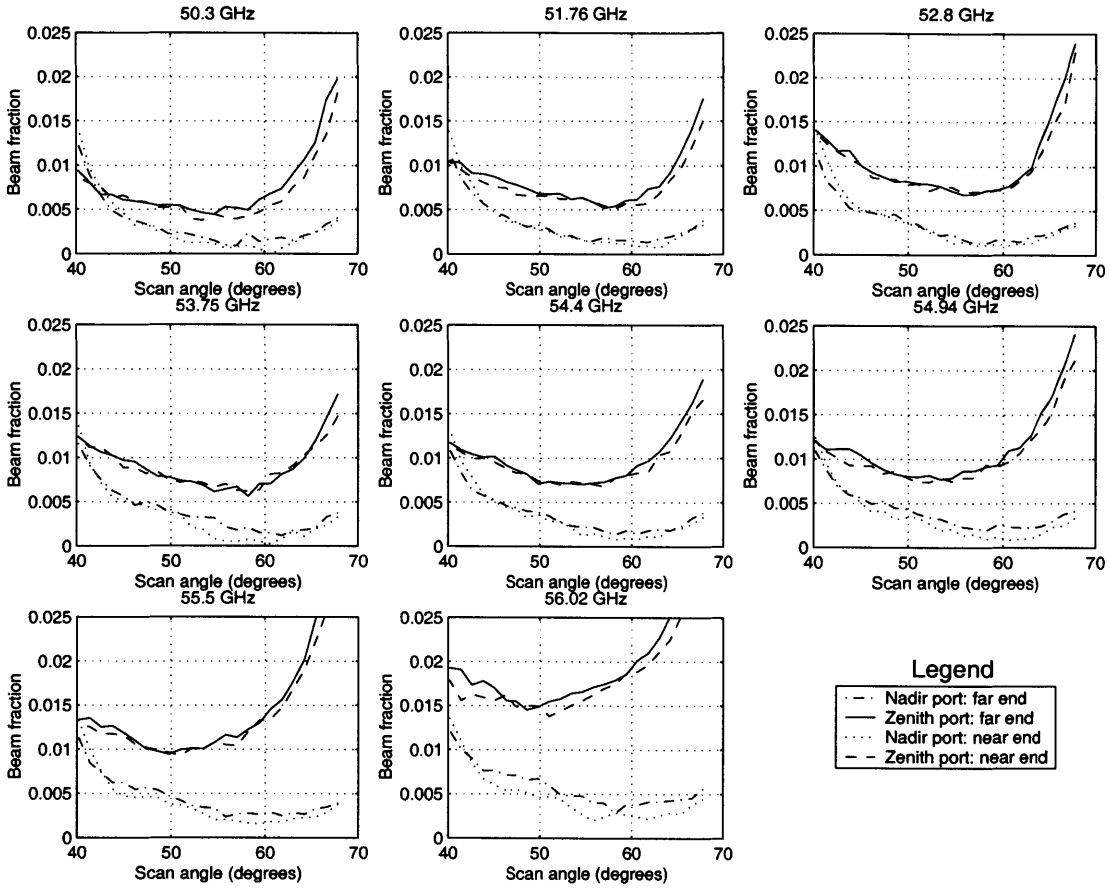

Figure C-2: Antenna beam coupling coefficients for ambient load near 54 GHz.

The antenna beam coupling coefficients are shown in Figures C-2 through C-5. The values used in the thesis are an average of the values obtained at the two extreme lateral reflector positions, and are given in Table C.1.

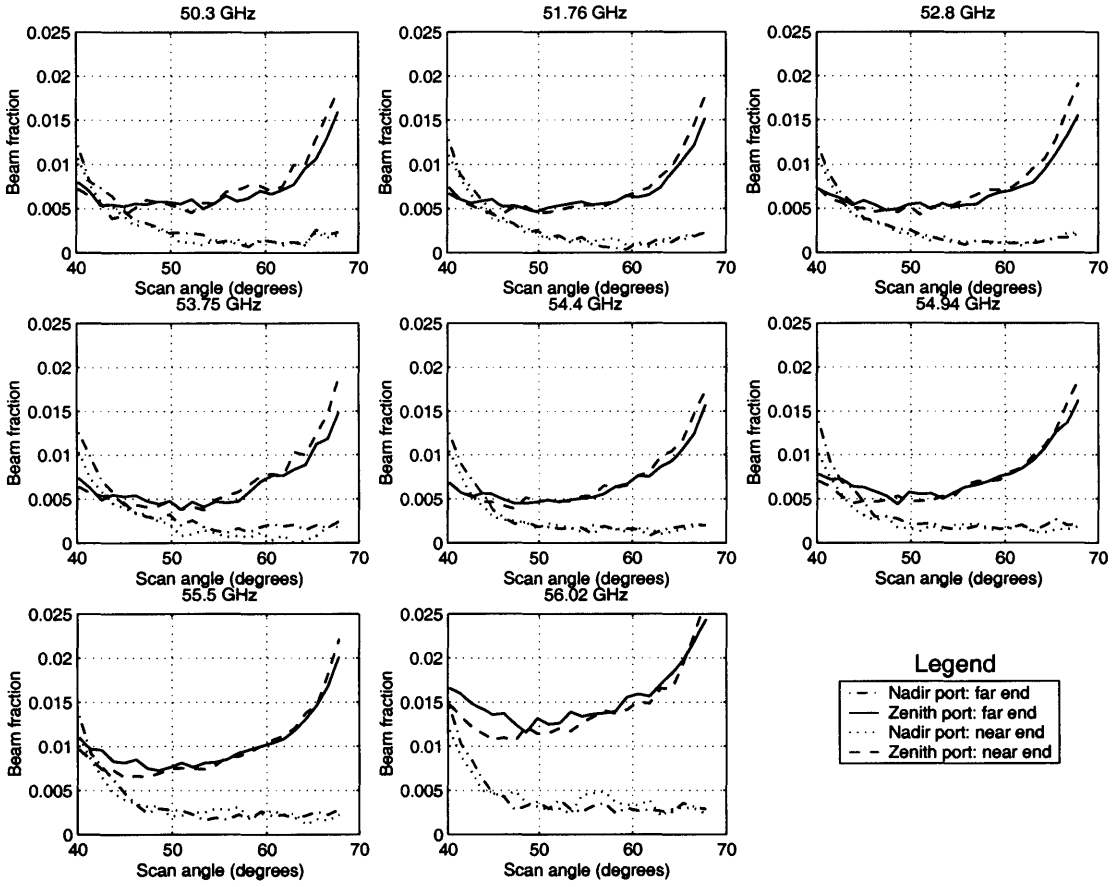

Figure C-3: Antenna beam coupling coefficients for heated load near 54 GHz.

Table C.1: Laboratory measurements of antenna beam coupling coefficients near 54 GHz.

No.	Frequency (GHz)	η_Z^A	η_N^A	η_Z^H	η_N^H
1	50.30	0.0048	0.0016	0.0057	0.0014
2	51.76	0.0066	0.0021	0.0053	0.0014
3	52.80	0.0077	0.0024	0.0054	0.0013
4	53.75	0.0071	0.0024	0.0045	0.0014
5	54.40	0.0073	0.0023	0.0050	0.0016
6	54.94	0.0082	0.0027	0.0056	0.0018
7	55.50	0.0107	0.0031	0.0080	0.0023
8	56.02	0.0163	0.0044	0.0129	0.0034

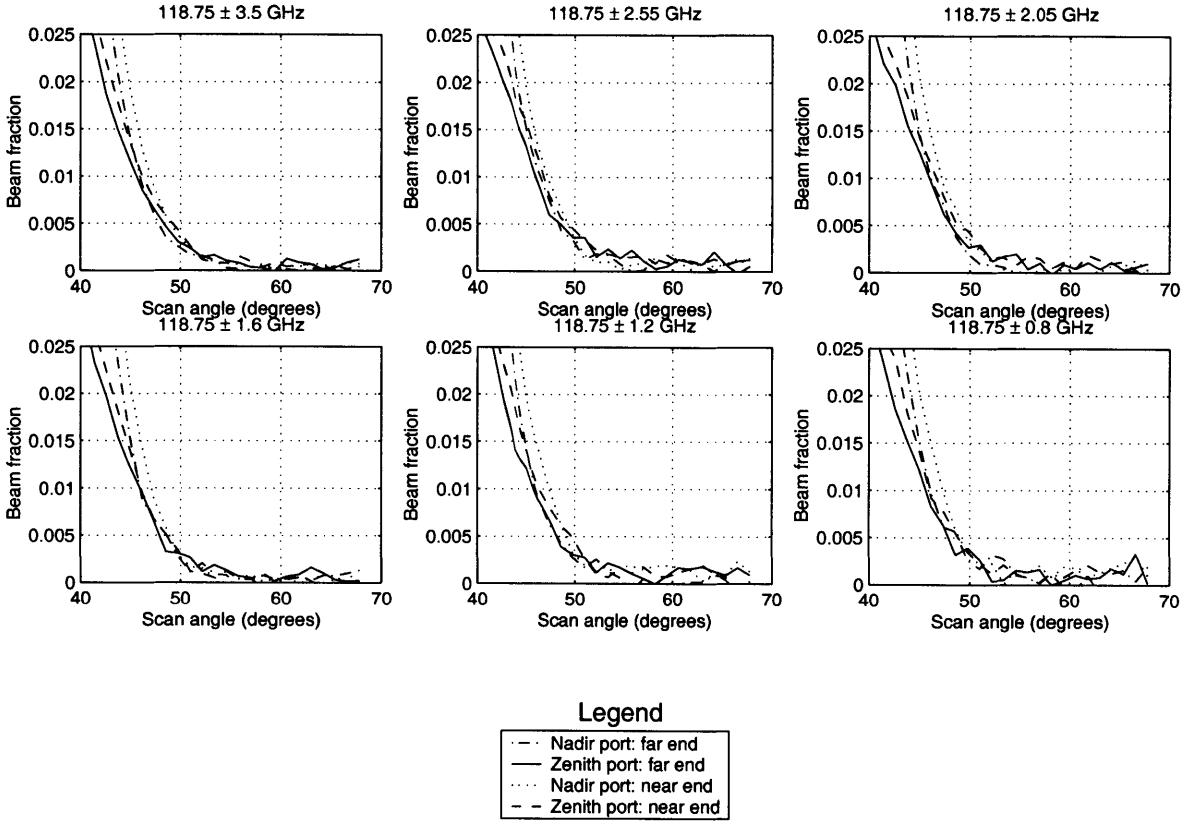

Figure C-4: Antenna beam coupling coefficients for ambient load near 118.75 GHz.

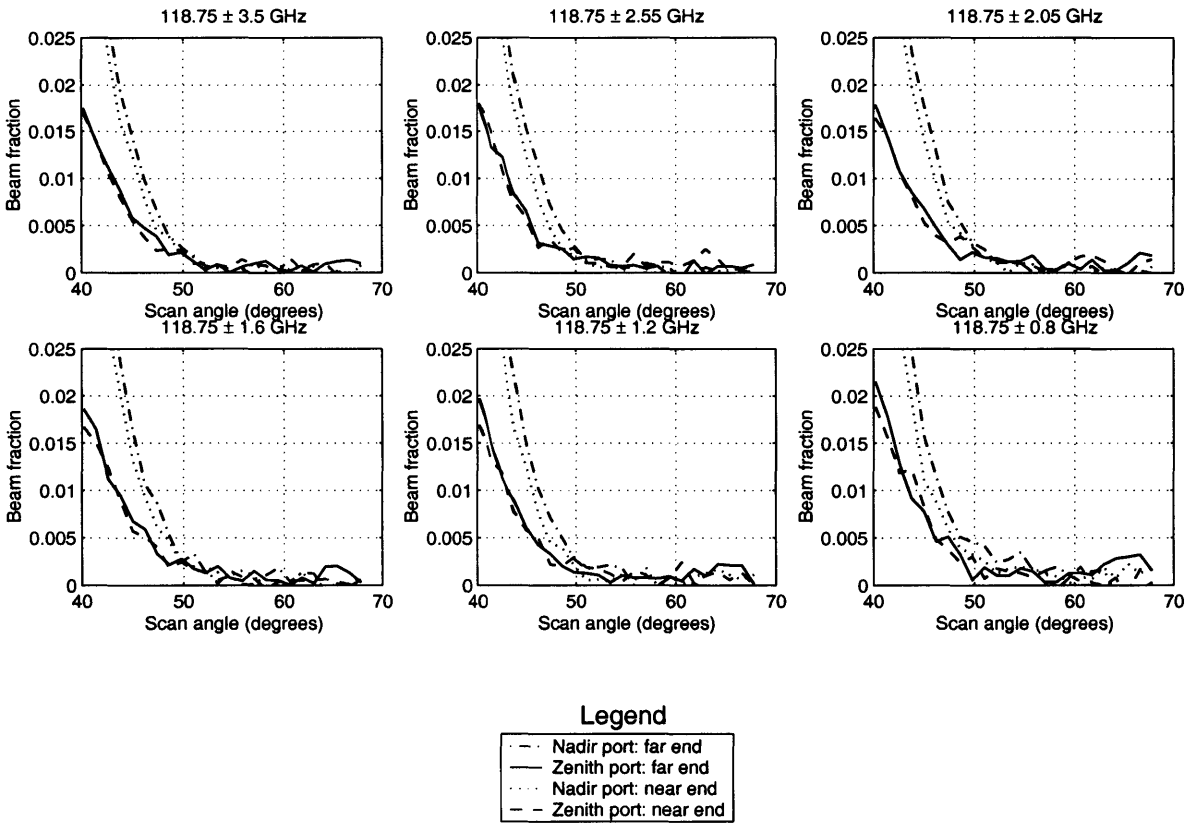

Figure C-5: Antenna beam coupling coefficients for heated load at 118.75 GHz.

Appendix D

MAMS-Derived Cloud Imagery from CAMEX-3

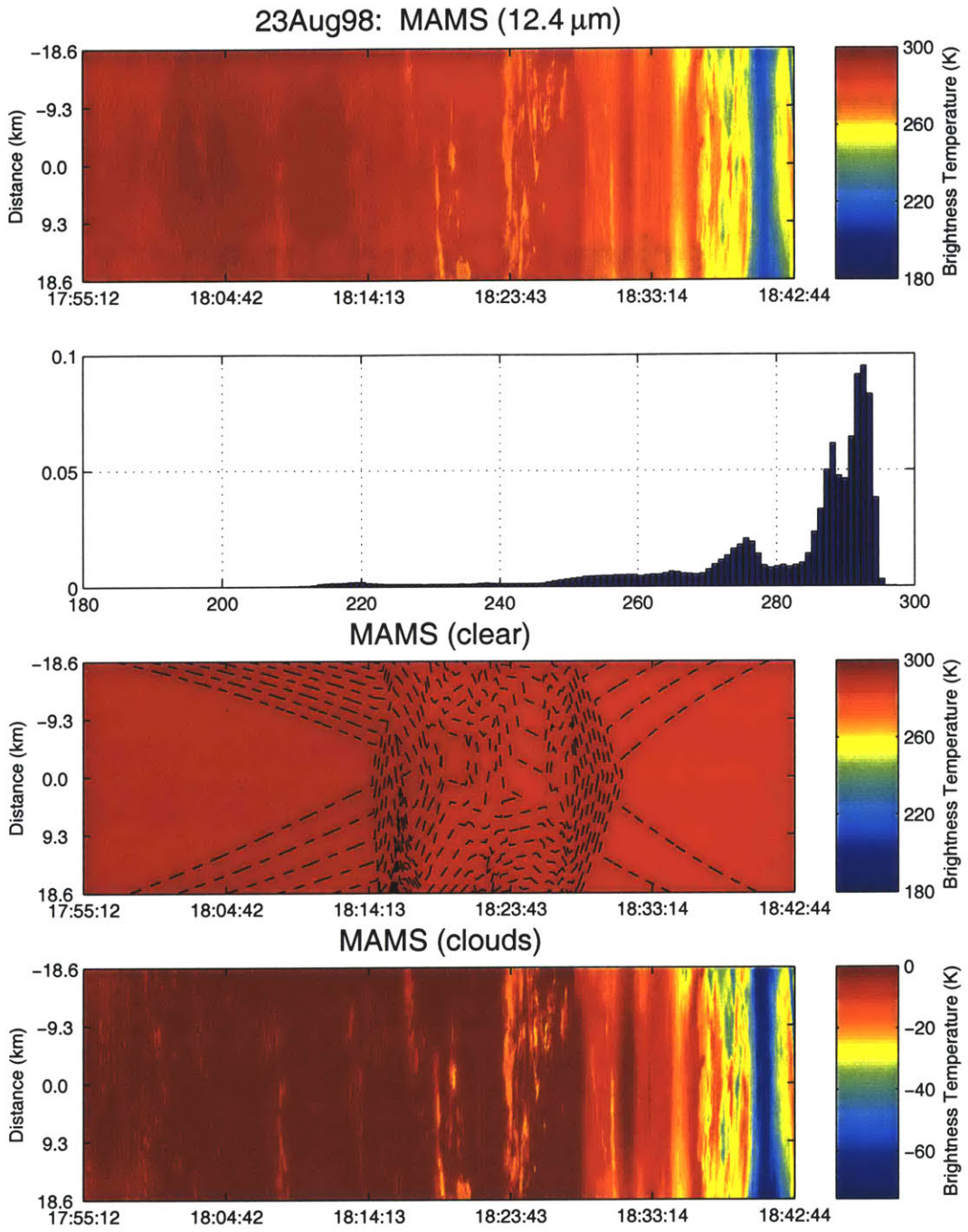

Figure D-1: MAMS imagery from August 23, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

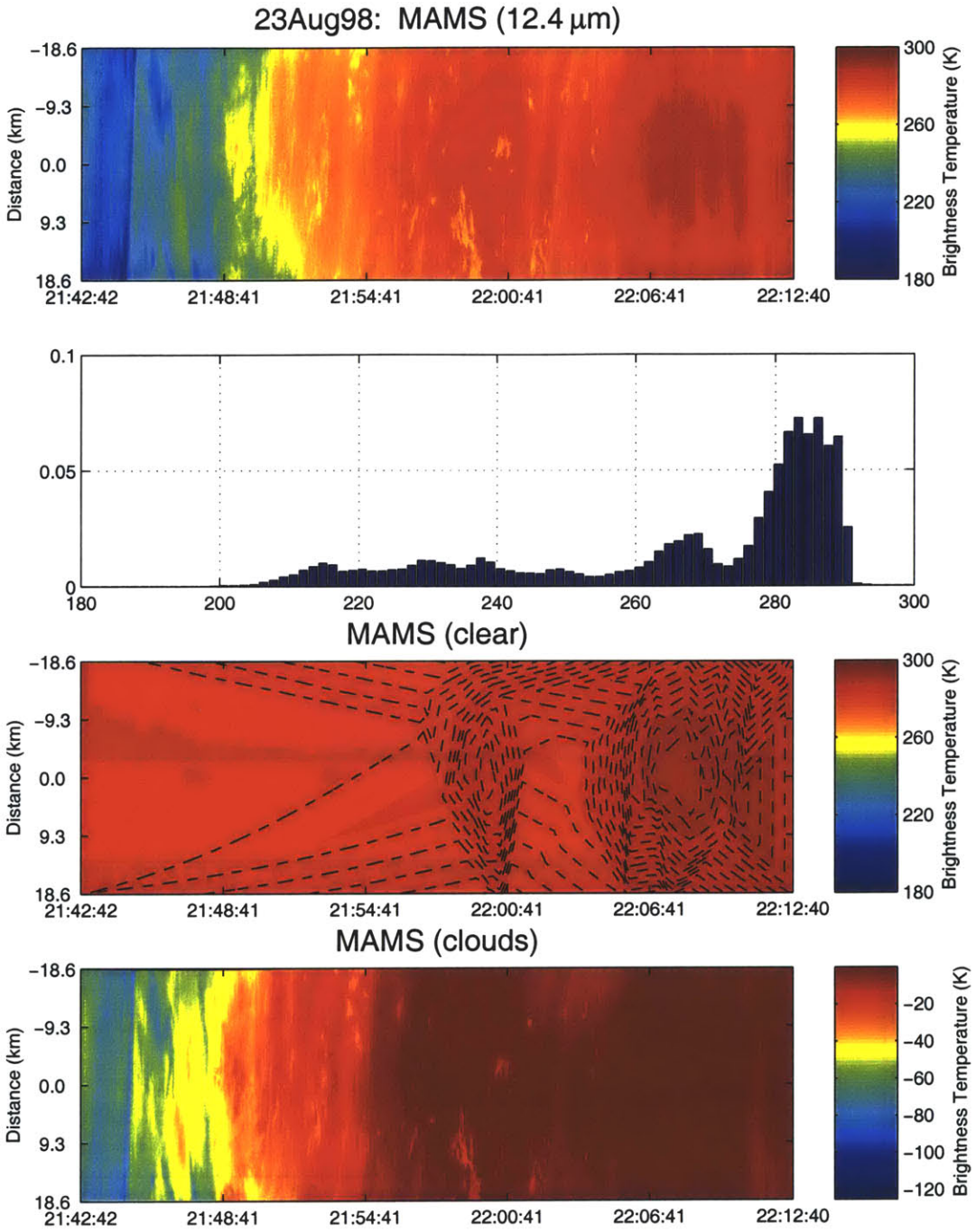

Figure D-2: MAMS imagery from August 23, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

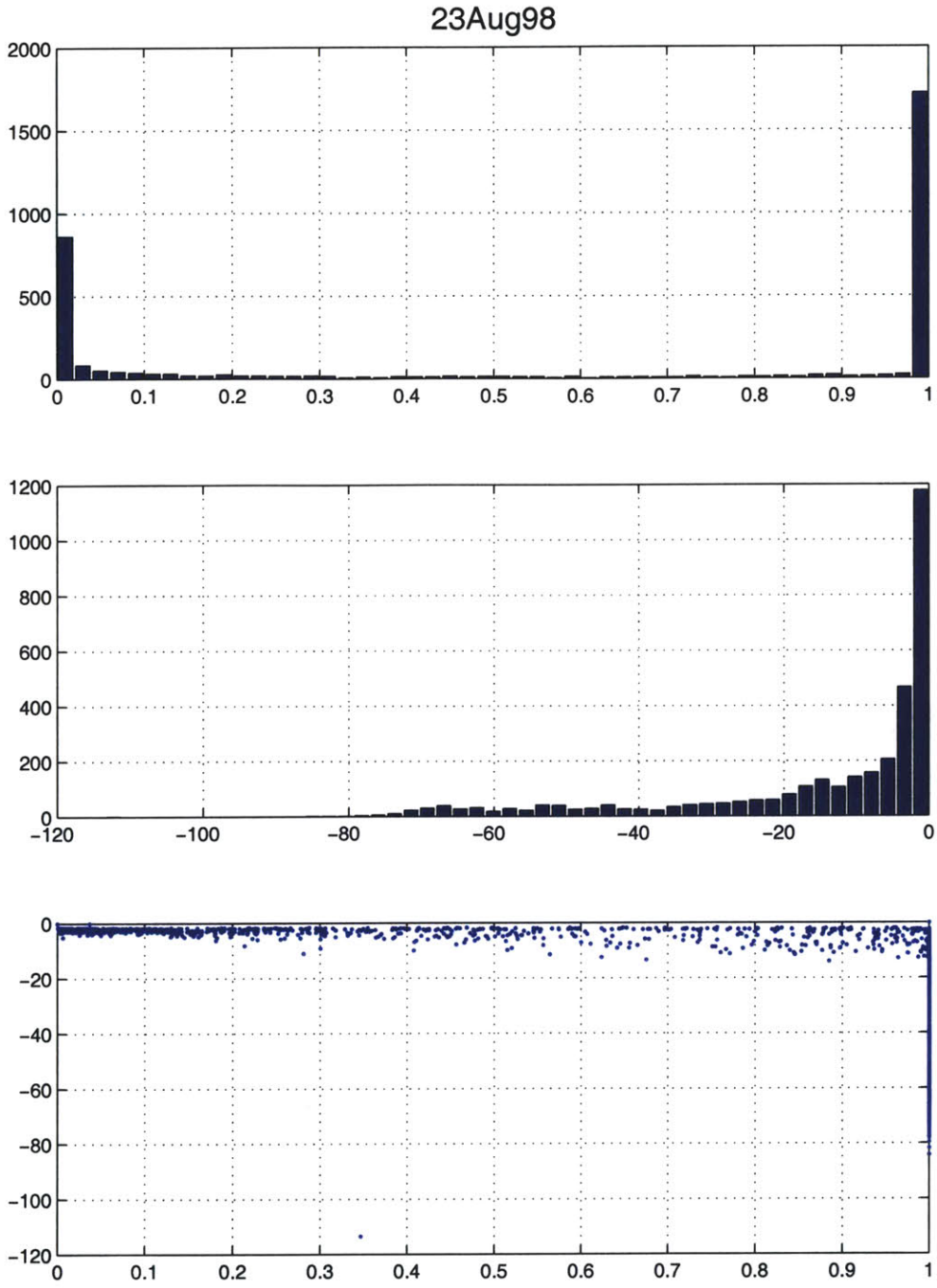

Figure D-3: MAMS-derived cloud statistics from August 23, 1998. The top panel shows a histogram of cloud fraction. The second panel shows a histogram of mean cloud perturbation. A scatterplot of cloud fraction versus mean cloud perturbation is given in the last panel.

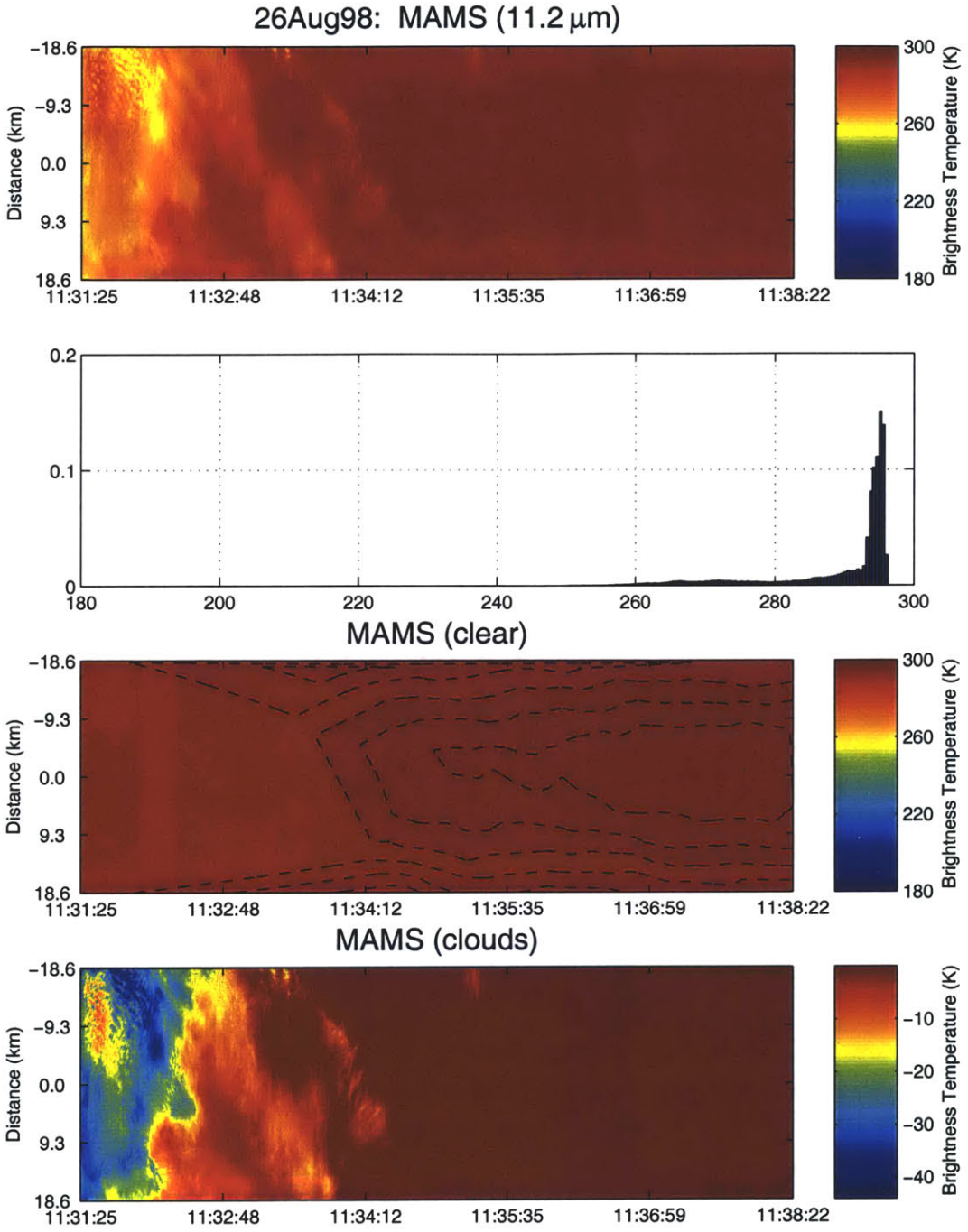

Figure D-4: MAMS imagery from August 26, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

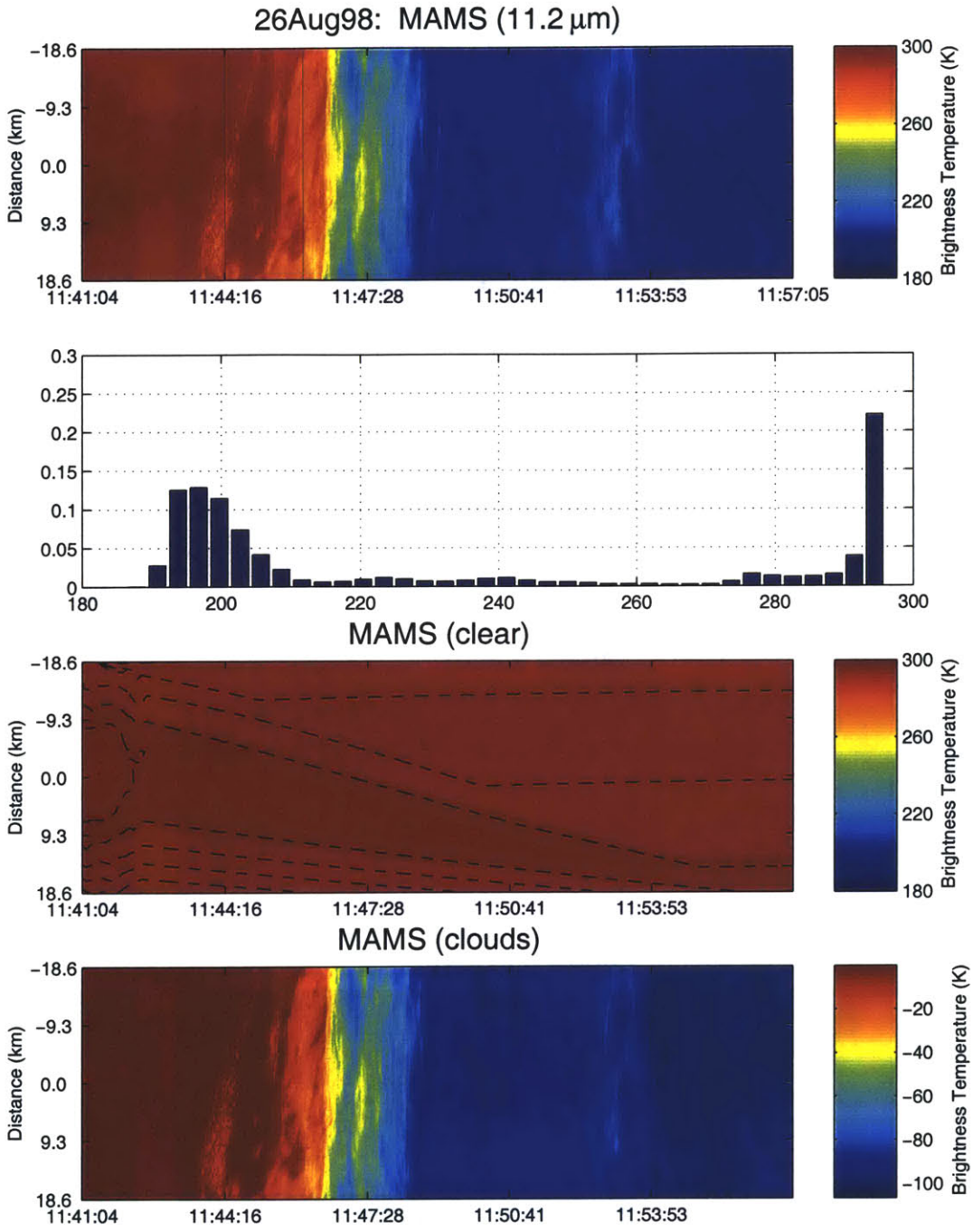

Figure D-5: MAMS imagery from August 26, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

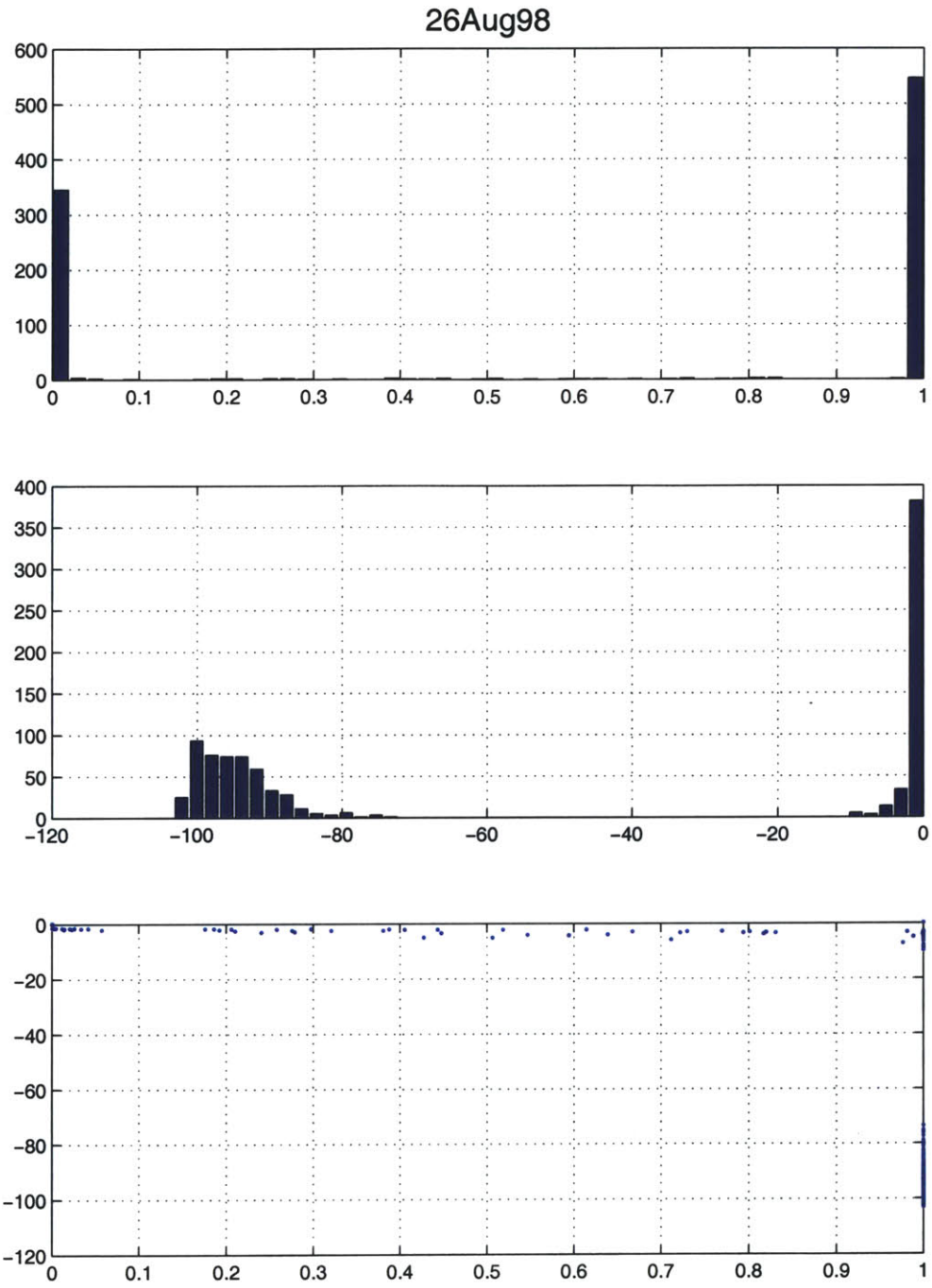

Figure D-6: MAMS-derived cloud statistics from August 26, 1998. The top panel shows a histogram of cloud fraction. The second panel shows a histogram of mean cloud perturbation. A scatterplot of cloud fraction versus mean cloud perturbation is given in the last panel.

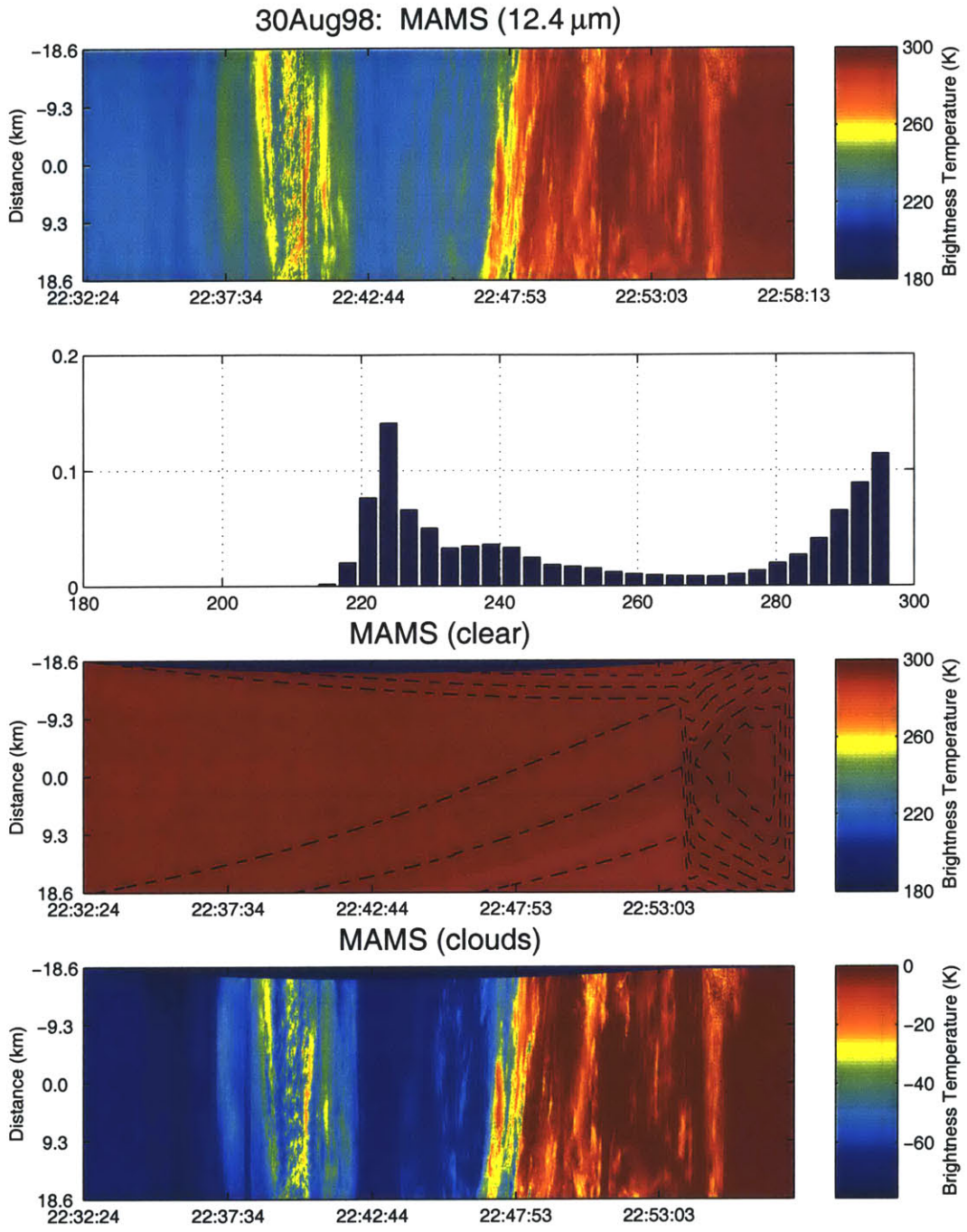

Figure D-7: MAMS imagery from August 30, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

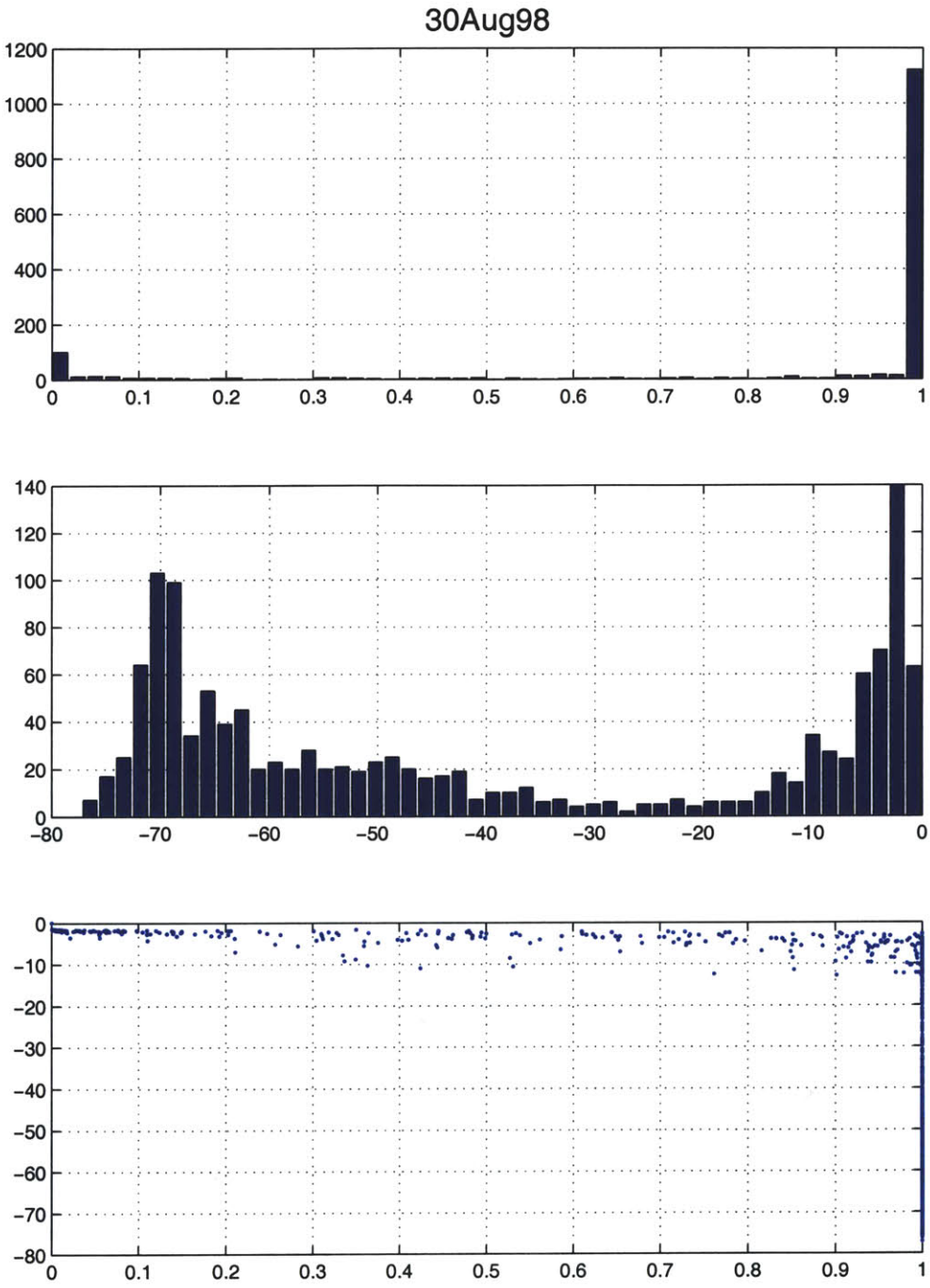

Figure D-8: MAMS-derived cloud statistics from August 30, 1998. The top panel shows a histogram of cloud fraction. The second panel shows a histogram of mean cloud perturbation. A scatterplot of cloud fraction versus mean cloud perturbation is given in the last panel.

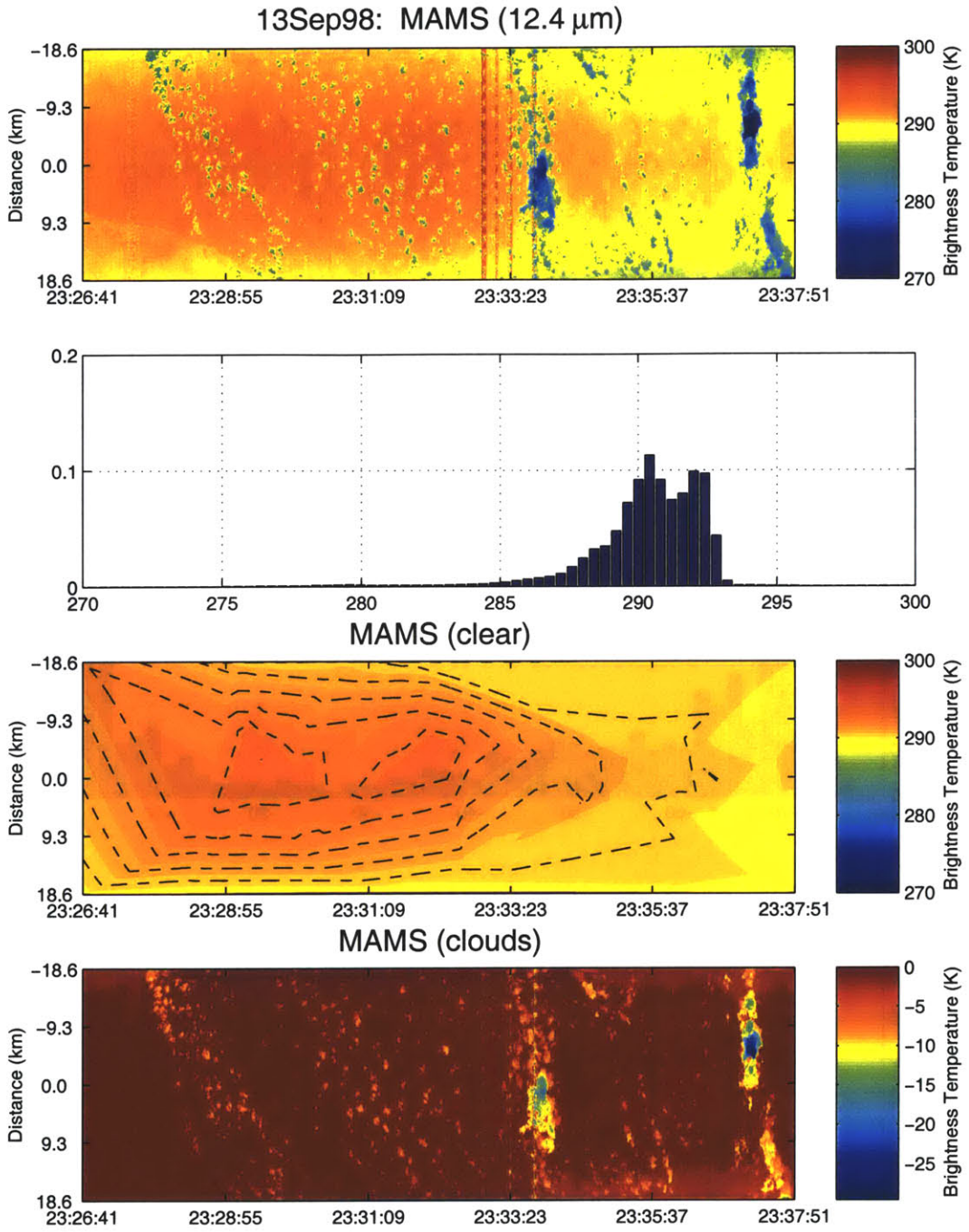

Figure D-9: MAMS imagery from September 13, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

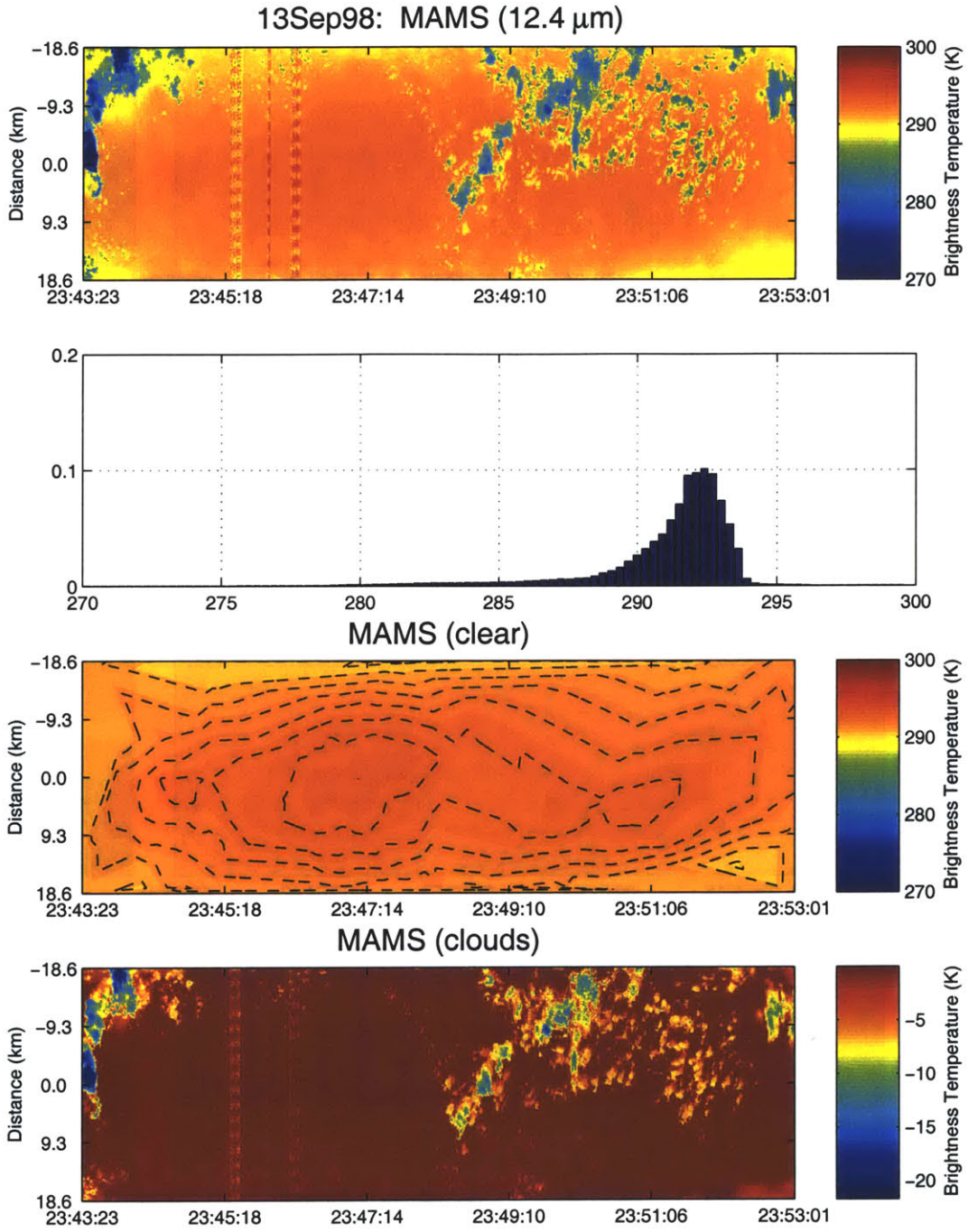

Figure D-10: MAMS imagery from September 13, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

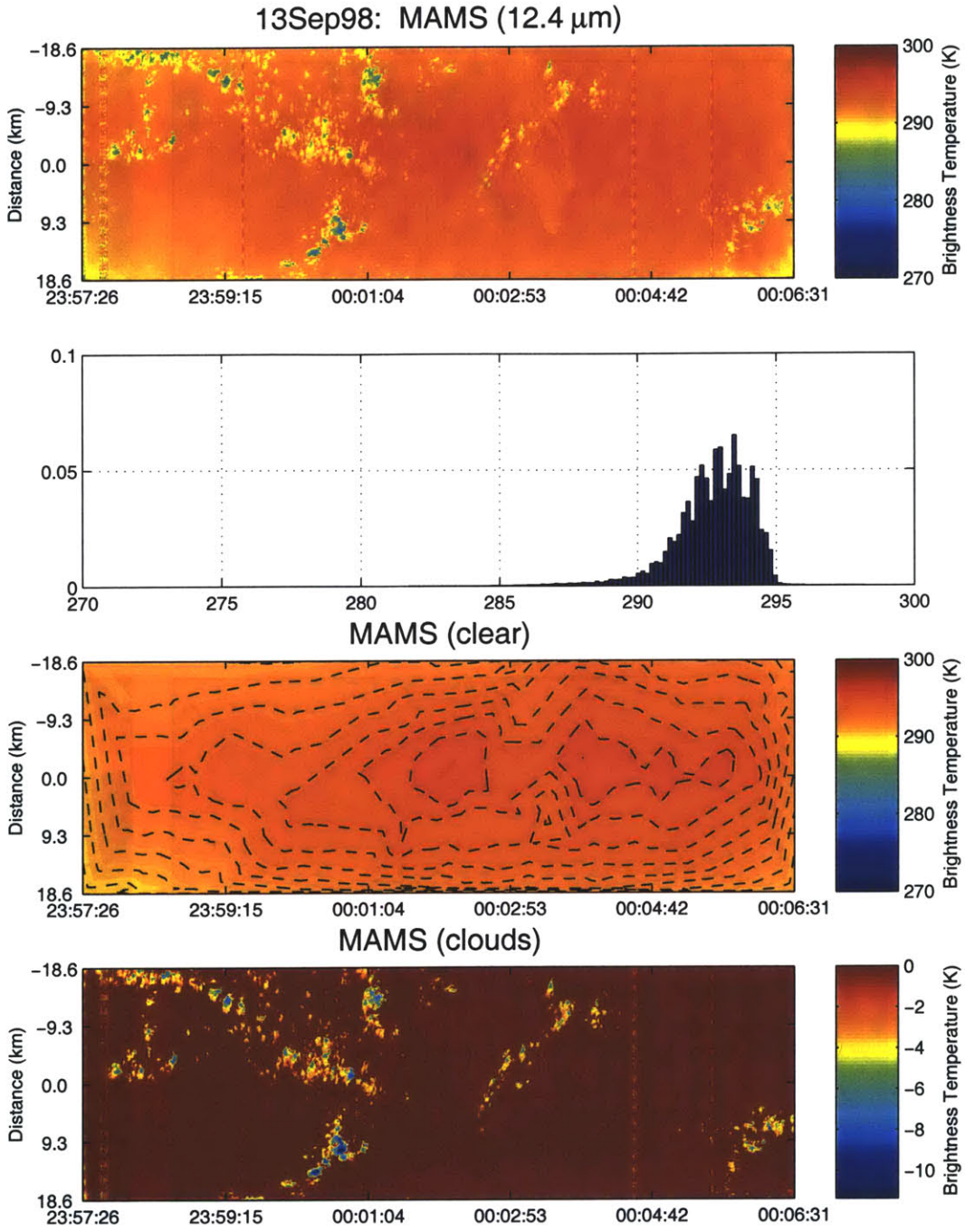

Figure D-11: MAMS imagery from September 13, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

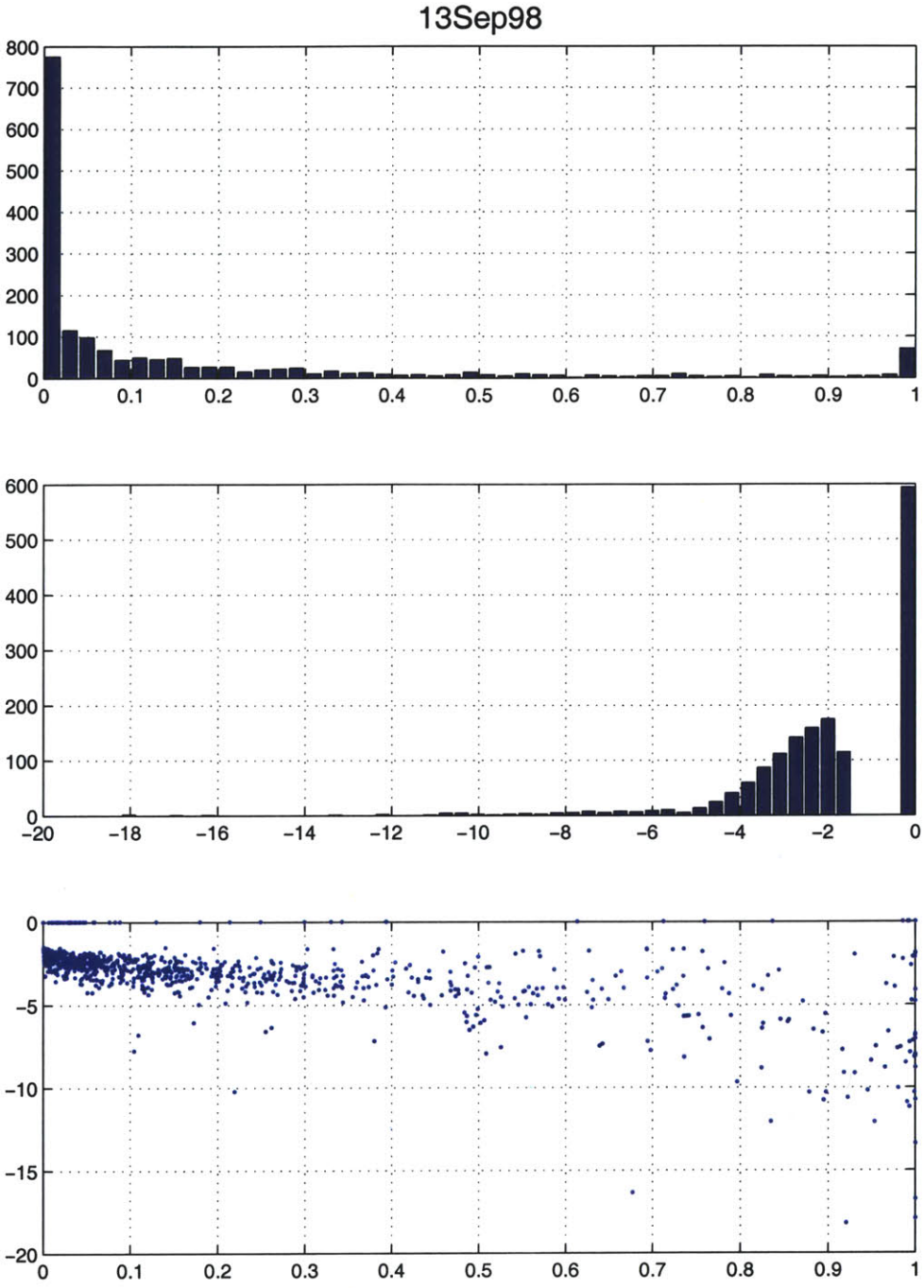

Figure D-12: MAMS-derived cloud statistics from September 13, 1998. The top panel shows a histogram of cloud fraction. The second panel shows a histogram of mean cloud perturbation. A scatterplot of cloud fraction versus mean cloud perturbation is given in the last panel.

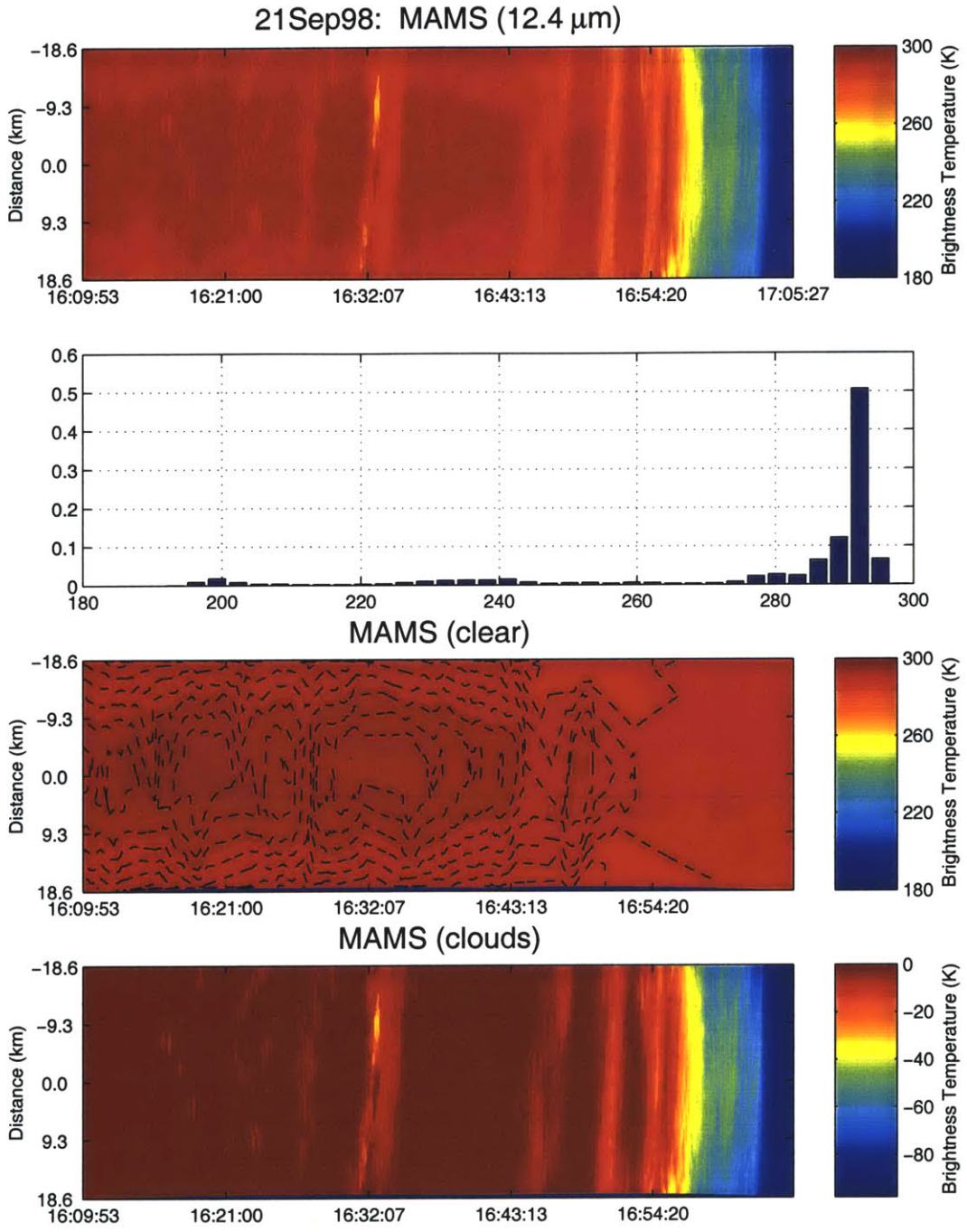

Figure D-13: MAMS imagery from September 21, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

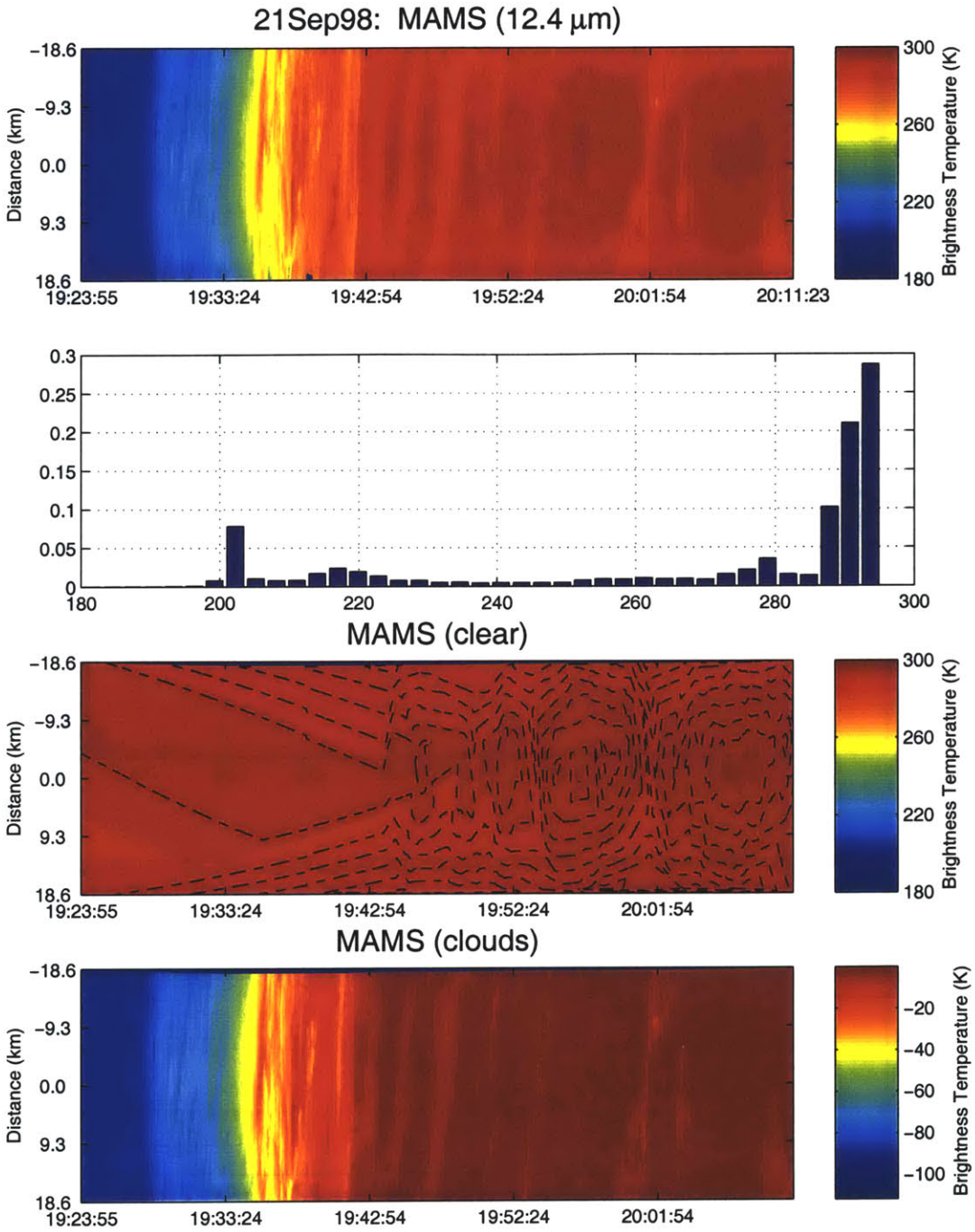

Figure D-14: MAMS imagery from September 21, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

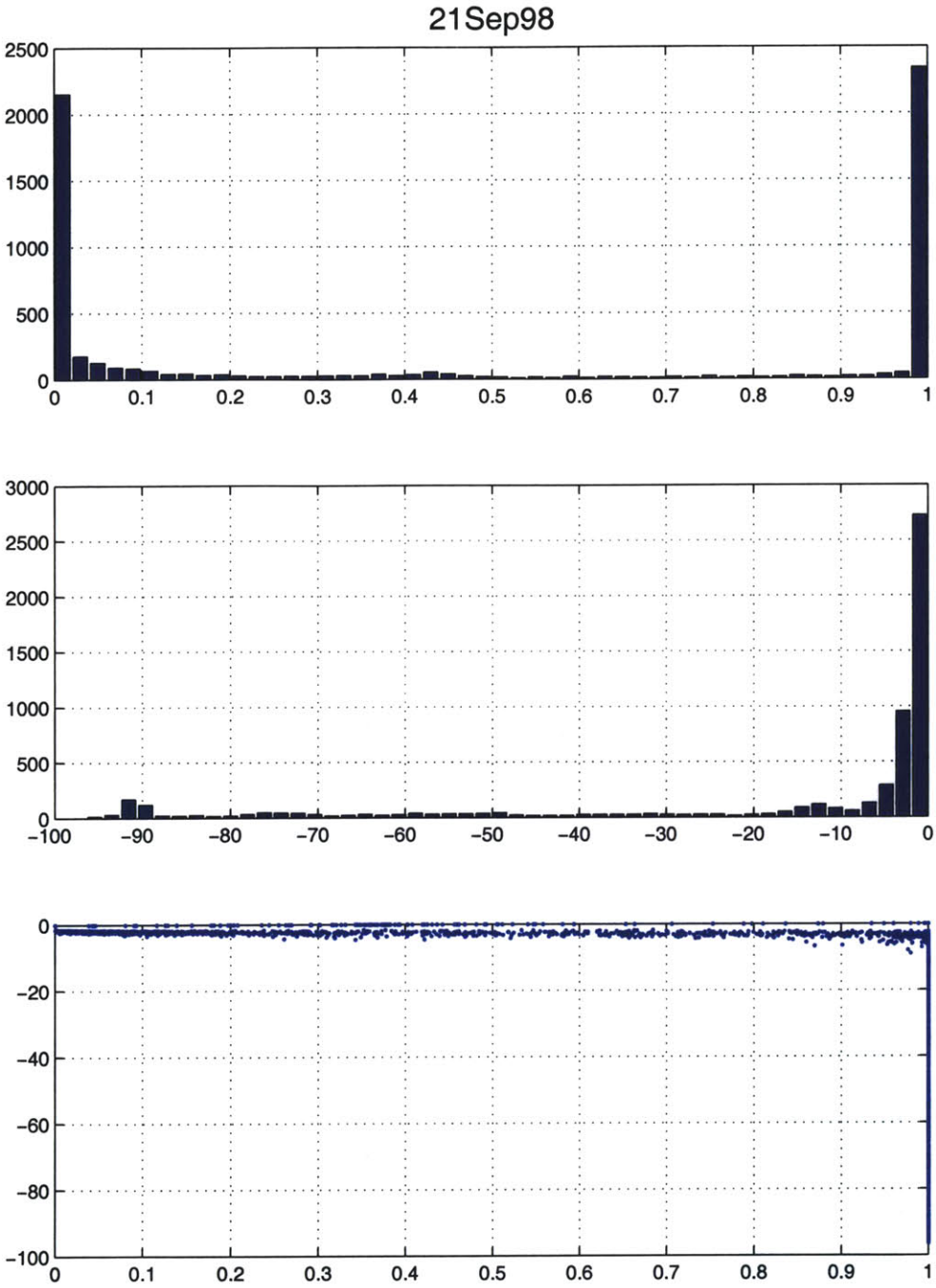

Figure D-15: MAMS-derived cloud statistics from September 21, 1998. The top panel shows a histogram of cloud fraction. The second panel shows a histogram of mean cloud perturbation. A scatterplot of cloud fraction versus mean cloud perturbation is given in the last panel.

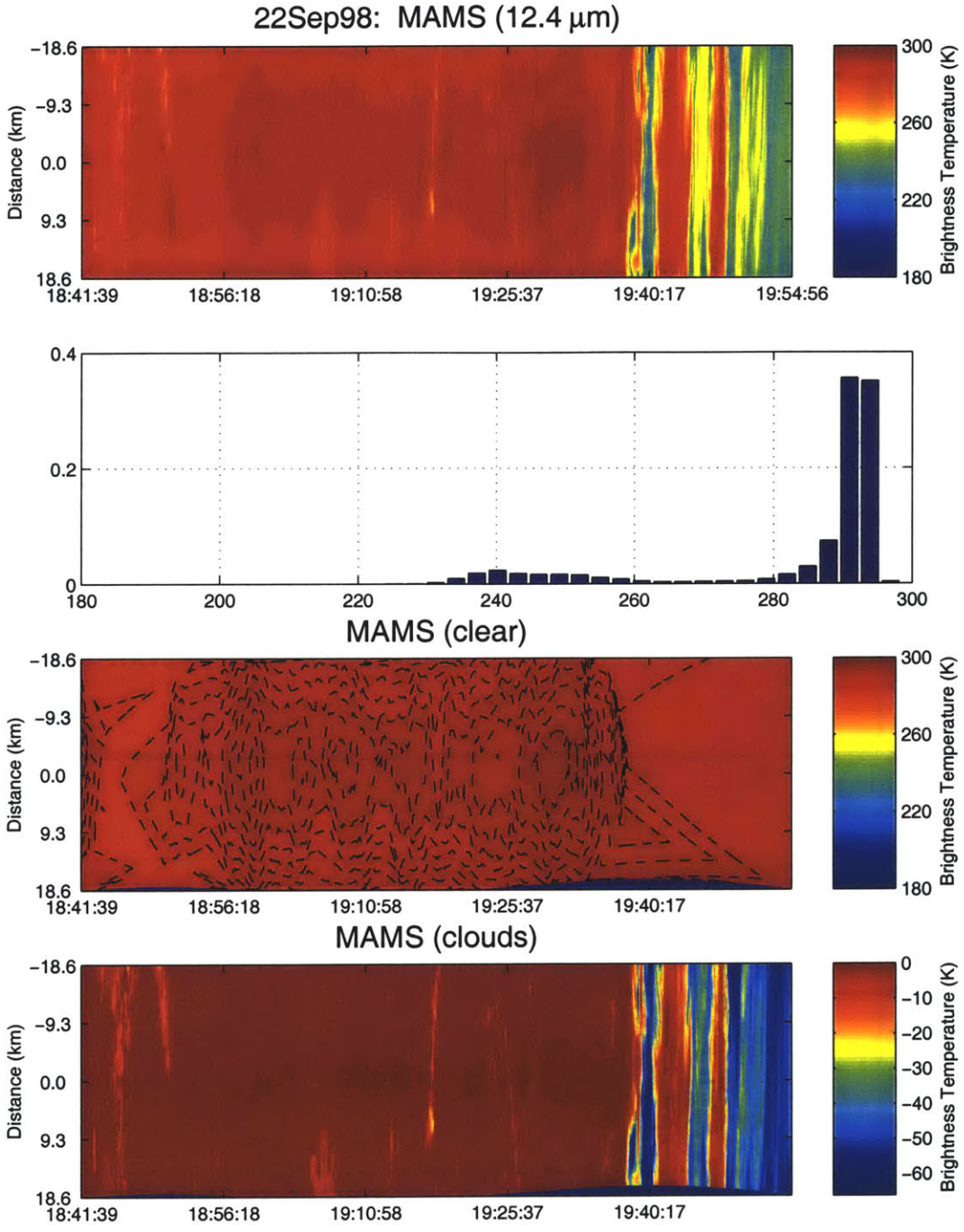

Figure D-16: MAMS imagery from September 22, 1998. The top panel shows MAMS brightness temperatures. A histogram of pixel brightness temperatures for the image is given in the second image. The clear-air background is shown in the third image. Contour lines are drawn every 0.5 K. The clouds (background removed) are shown in the last image.

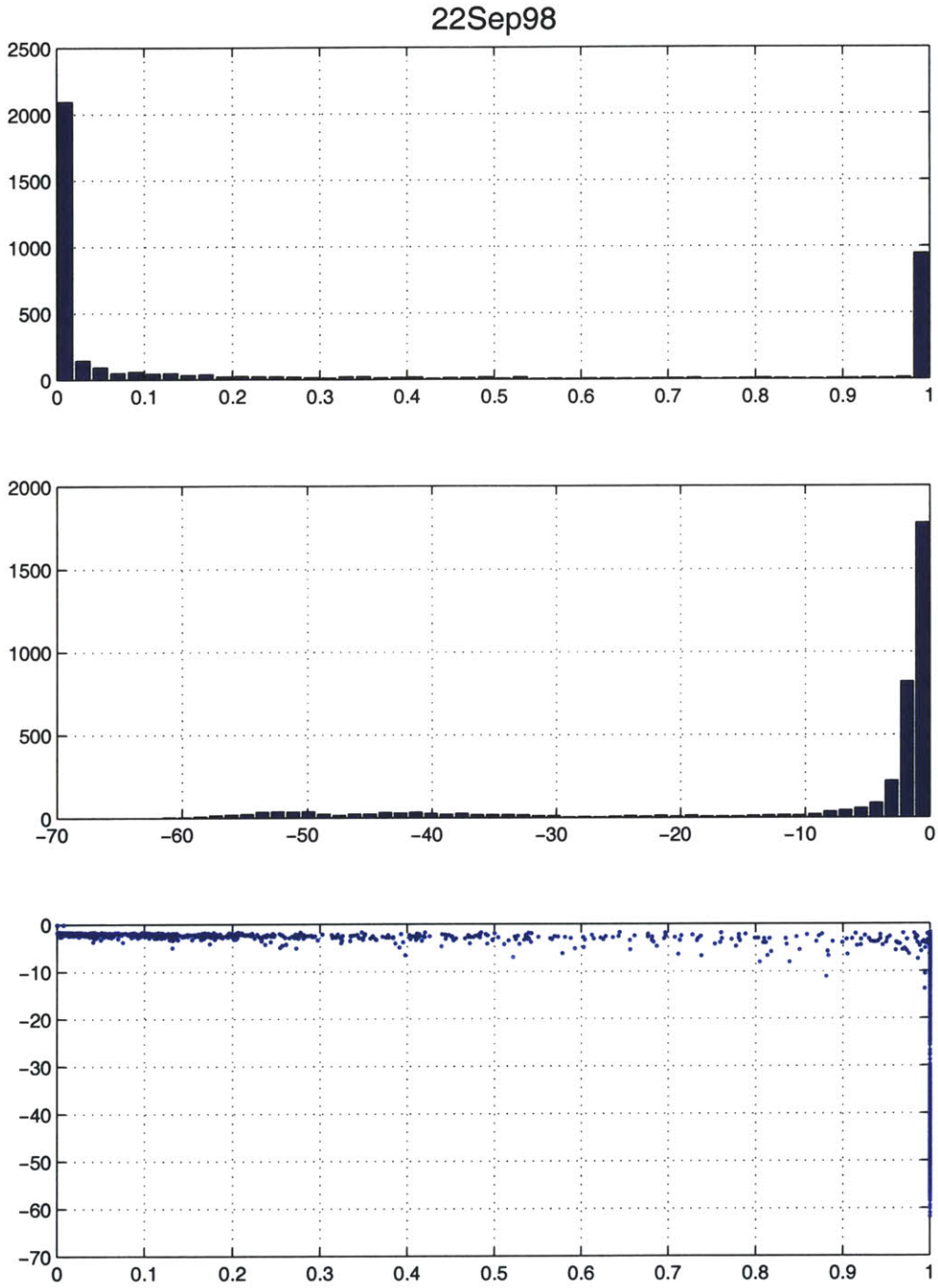

Figure D-17: MAMS-derived cloud statistics from September 22, 1998. The top panel shows a histogram of cloud fraction. The second panel shows a histogram of mean cloud perturbation. A scatterplot of cloud fraction versus mean cloud perturbation is given in the last panel.

Appendix E

Selected Source Code

THIS appendix lists selected MATLAB scripts and functions that were used in the thesis. All software was written by the author, with the following exceptions: 1) ION algorithm code, which was written by Junehee Lee and modified by the author, and 2) NAST-M data display functions, which were based on routines written by M. J. Schwartz. Additional source code not listed here may be obtained from the author: `billb@alum.mit.edu`.

E.1 NAST-M calibration and validation software

```
function [Tb, TH, TC, TS] = calib3wls(count_data, rtd_data, W, count_filter_length, eta)

% function for calibrating NAST-M counts using 3 calibration sources
% The calibration line is calculated using a weighted least-squares solution
% Corrections for spillover are implemented using wjb eta model
%
% usage:      Tb = calib3wls(count_data, rtd_data, eta, W, count_filter_length)
%
% inputs:
%                                                                              10
% count_data    structure with the following fields:
%               count_data.counts    raw counts (17 x 25 x N)
%               count_data.time      timestamp matrix (25 x N)
%               count_data.hcal_index index of spots to be used as "hot" cal
```

```
%                                   if more than one spot is given, we average
%           count_data.ccal_index index of spots to be used as "cold" cal
%                                   if more than one spot is given, we average
%           count_data.scal_index index of spots to be used as "sky" cal
%                                   if more than one spot is given, we average
% rtd_data  structure with the following fields:                                  20
%           rtd_data.rtdfc        interpolated, filtered, and corrected rtd values
%                                   (N x 29)
%           rtd_data.time         timestamp (1 x N)
%           rtd_data.wgtH         weight of RTD's to be used for "hot" cal (17 x 30)
%                                   NOTE: the extra weight is the value for deltaH
%           rtd_data.wgtC         weight of RTD's to be used for "cold" cal (17 x 30)
%                                   NOTE: the extra weight is the value for deltaC
%           rtd_data.wgtS         weight of RTD's to be used for "sky" cal (17 x 30)
%                                   NOTE: the extra weight is the value for deltaS
% W                               matrix of weights for weighted least-squares (17x3x3)  30
%                                   ordering is [H C S]
% count_filter_length             length input to filtfilt
% eta (optional) structure with the following fields:
%           eta.eta_C             eta values for ambient load (17 x 2)
%                                 first column is nadir contribution, second is zenith
%           eta.eta_H             eta values for heated load (17 x 2)
%                                 first column is nadir contribution, second is zenith
%           eta.T_N               nadir temperatures (17 x N)
%           eta.T_Z               zenith temperatures (17 x N)
%           eta.time              timestamps for T_N and T_Z                       40
%
% outputs:
%
% Tb        calibrated brightness temperatures (17 x 25 x N)
% TH        thermometric temperature used for "hot" load (17 x N)
% TC        thermometric temperature used for "cold" load (17 x N)
% TS        brightness temperature used for "sky" load (17 x N)
%
% internal paramaters:
%                                                                                 50
% written by WJB 2/3/00
```

```
%
% REVISION HISTORY:
%
%

N = size(count_data.counts);

if (nargin < 5)
        fprintf('Using ideal eta case. . . .\n');                    60
        eta.eta_C = zeros(17, 2);
        eta.eta_H = zeros(17, 2);
        eta.T_N = zeros(17, N(3));
        eta.T_Z = zeros(17, N(3));
        eta.time = zeros(1, N(3));
end

Tb = zeros(N(1), N(2), N(3));

% calibration counts                                                 70
CH = squeeze(mean(count_data.counts(:,count_data.hcal_index,:),2));
CC = squeeze(mean(count_data.counts(:,count_data.ccal_index,:),2));
CS = squeeze(mean(count_data.counts(:,count_data.scal_index,:),2));

if (count_filter_length > 1)
   % filter the calibration counts
   CH = filtfiltcol(ones(1,count_filter_length)/count_filter_length,1,CH')';
   CC = filtfiltcol(ones(1,count_filter_length)/count_filter_length,1,CC')';
   CS = filtfiltcol(ones(1,count_filter_length)/count_filter_length,1,CS')';
end                                                                  80

% timestamp for calibration counts
t_CH = mean(count_data.time(count_data.hcal_index,:),1);
t_CC = mean(count_data.time(count_data.ccal_index,:),1);
t_CS = mean(count_data.time(count_data.scal_index,:),1);

% thermometric calibration data
TH = rtd_data.wgtH * [rtd_data.rtdfc'; ones(1,size(rtd_data.rtdfc,1))];
```

```
TC = rtd_data.wgtC * [rtd_data.rtdfc'; ones(1,size(rtd_data.rtdfc,1))];
TS = rtd_data.wgtS * [rtd_data.rtdfc'; ones(1,size(rtd_data.rtdfc,1))];

fprintf('\n');

% loop over the spots
for spots = 1:N(2)
    fprintf('Calibrating spot %d of %d. . .\r', spots, N(2));

    % interpolate calibration data to scene
    CH_ = interp1(t_CH', CH', count_data.time(spots,:)','linear','extrap')';
    CC_ = interp1(t_CC', CC', count_data.time(spots,:)','linear','extrap')';
    CS_ = interp1(t_CS', CS', count_data.time(spots,:)','linear','extrap')';
    TH_ = interp1(rtd_data.time', TH', count_data.time(spots,:)','linear','extrap')';
    TC_ = interp1(rtd_data.time', TC', count_data.time(spots,:)','linear','extrap')';
    TS_ = interp1(rtd_data.time', TS', count_data.time(spots,:)','linear','extrap')';
    TZ_ = interp1(eta.time', eta.T_Z', count_data.time(spots,:)','linear','extrap')';
    TN_ = interp1(eta.time', eta.T_N', count_data.time(spots,:)','linear','extrap')';

    % loop over the scans
    for scans = 1:N(3)
        % loop over the frequencies
        for freqs = 1:N(1)
            % calibrate with interpolated parameters
            % correct ambient and hot loads
            TC_c = eta.eta_C(freqs,1) * TN_(freqs,scans) + ...
                eta.eta_C(freqs,2) * TZ_(freqs,scans) + ...
                (1 - eta.eta_C(freqs,1) - eta.eta_C(freqs,2)) * TC_(freqs,scans);
            TH_c = eta.eta_H(freqs,1) * TN_(freqs,scans) + ...
                eta.eta_H(freqs,2) * TZ_(freqs,scans) + ...
                (1 - eta.eta_H(freqs,1) - eta.eta_H(freqs,2)) * TH_(freqs,scans);
            A = [[CH_(freqs,scans); CC_(freqs,scans); CS_(freqs,scans)] ones(3,1)];
            b = [[TH_c; TC_c; TS_(freqs,scans)]];
            gb = inv(A' * squeeze(W(freqs,:,:)) * A) * A' * squeeze(W(freqs,:,:)) * b;
            Tb(freqs,spots,scans) = count_data.counts(freqs,spots,scans) * gb(1) + gb(2);
        end
    end
```

```
end

fprintf('Calibrating spot %d of %d. . .Done\n', N(2), N(2));
```

```
% Script to calculate RTD weights (heated load) using data from
% March 29, 1999 WINTEX flight
% WJB

% downwelling (sky data) for 19.42 km
T_Z = [3.2265 3.5223 4.6198 11.4832 15.6511 28.9500 37.4925 69.5931]';
T_Z_118 = [4.35 4.7093 5.2818 6.3674 8.4721 14.7136 37.8203 91.4319]';

% load data here (code omitted)
% need the following:                                                    10
%
% t          (radiometric timestamps)
% trtd       (rtd timestamps)
% x          (radiometric counts) (17 x 25 x N)
% rtdfc      (rtd data, filtered and corrected) (29 x N)

CAL_RANGE = 1120:1220;

% fill in sturctures for calib2 (two-point cal)
x = x(1:16, :, CAL_RANGE);                                               20
count_data.counts = x;
count_data.time = t(:, CAL_RANGE);
count_data.hcal_index = [24:25]; % amb load
count_data.ccal_index = [1:2]; % sky cal

rtd_data.wgtH = ones(16,1) * [zeros(1, 23) .2 * ones(1,5) 0 0];
rtd_data.wgtC = [zeros(16, 29) [T_Z; T_Z_118]];
rtd_data.time = trtd(CAL_RANGE)';
rtd_data.rtdfc = rtdfc(CAL_RANGE,:);
                                                                         30
[Tb, TH, TA] = calib2(count_data, rtd_data, 4);
```

```
% thermodynamic temps
hot_load_rtds = mean(rtdfc(CAL_RANGE,9:15));
% radiometric temps (use only 118.75 +/- 3.5)
hot_load_rad = mean(mean(Tb(9, 3:4, :)));

w = pinv([hot_load_rtds; ones(1,7)]) * [hot_load_rad; 1];
```

```
% Script to calculate antenna beam coupling coefficients (eta's)
% from laboratory measurements of liquid nitrogen
% WJB

% load data here (code omitted)
% need the following:
%
% t         (radiometric timestamps)
% trtd      (rtd timestamps)
% x         (radiometric counts) (17 x 25 x N)
% rtdfc     (rtd data, filtered and corrected) (29 x N)
% SCAN_RANGE (range of useful scans)

% crop data
t = t(:, SCAN_RANGE);
trtd = trtd(SCAN_RANGE)';
x = x(:, :, SCAN_RANGE);
rtdfc = rtdfc(SCAN_RANGE, :);

% The spot pattern is [S 24H | N 24A] * 4
%   A/N, A/A, N/A, N/N

% Assemble "superscans"
NUM_SCANS = floor(size(x,3)/8)*8;
if (size(x,3) ~= NUM_SCANS)
  y = x(:, :, 1:NUMSCANS);
else
  y = x;
```

```
end

fprintf('Using %d scans of data\n', NUM_SCANS);

y = reshape(y, 17, 200, NUM_SCANS/8);
t = reshape(t, 200, NUM_SCANS/8);
T_N_77 = 79.5;

% fill in sturctures for calib3wls
count_data.counts = y;
count_data.time = t;
count_data.hcal_index = [86:89]; % Ambient - mean of four spots
count_data.ccal_index = 176;  % Nadir LN2 #1
count_data.scal_index = 26;   % Nadir LN2 #2

rtd_data.wgtH = ones(17,1) * [zeros(1, 23) .2 * ones(1,5) 0 0];
rtd_data.wgtC = [zeros(17, 29) ones(17,1) * T_N_77];
rtd_data.wgtS = [zeros(17, 29) ones(17,1) * T_N_77];
rtd_data.time = trtd;
rtd_data.rtdfc = rtdfc;

% not enough data to filter calibration counts
COUNT_FILTER_LENGTH = 1;

% Weight matrix - need to weight 2 cal looks equally with am look
for i = 1:17, W(i,:,:) = diag([sqrt(2) 1 1]);, end;

% call calibration routine
[Tb, TH, TC, TS] = calib3wls(count_data, rtd_data, W, COUNT_FILTER_LENGTH);

% Use 54-GHz data to figure out temperature of absorbers....
T_N_A = mean(Tb(:, [76 126], :), 3);
pm_N_A = (max(max(T_N_A(1:8,:))) - min(min(T_N_A(1:8,:))))/2;
T_N_A = mean(mean(T_N_A(1:8,:)));
fprintf('Using nadir ambient temperature of %g K +- %g K\n', T_N_A, pm_N_A);
T_S_A = mean(Tb(:, [51 1], :), 3);
pm_S_A = (max(max(T_S_A(1:8,:))) - min(min(T_S_A(1:8,:))))/2;
```

```
T_S_A = mean(mean(T_S_A(1:8,:)));
fprintf('Using zenith ambient temperature of %g K +- %g K\n', T_S_A, pm_S_A);
T_S_77 = mean(Tb(:, [101 151], :), 3);
T_S_77 = mean(T_S_77(1:8,:)');
fprintf('54-GHz zenith measurements of LN2: %g\n', T_S_77);
T_S_77 = mean(T_S_77);

angles_h = 40.2:1.2:67.8;
angles_am = fliplr(angles_h);

% calculate the eta's for the heated targets....
A = [[T_N_A T_S_77 1]; [T_N_A T_S_A 1]; [T_N_77 T_S_A 1]; [T_N_77 T_S_77 1]];
Ai = inv(A' * A);

for i = 1:size(Tb,1) % loop over channels
  TH_prime = mean(Tb(i, [(2:25) (2:25)+50 (2:25)+100 (2:25)+150], :), 3);
  for j = 1:24 % loop over angles
    eta_H_hat3(i, j, :) = Ai*A'*reshape(TH_prime( (0:24:72) + j), 4, 1);
    eta_H_hat3(i, j, 3) = eta_H_hat3(i, j, 3) / (1 - eta_H_hat3(i, j, 1) - eta_H_hat3(i, j, 2));
  end
end

% calculate the eta's for the ambient targets....
A = [[T_N_A T_S_77]; [T_N_A T_S_A]; [T_N_77 T_S_A]; [T_N_77 T_S_77]] - mean(TH(1,:));
Ai = inv(A' * A);

for i = 1:size(Tb,1) % loop over channels
  TA_prime = mean(Tb(i, [(27:50) (27:50)+50 (27:50)+100 (27:50)+150], :), 3) - mean(TH(1,:));
  for j = 1:24 % loop over angles
    eta_A_hat(i, j, :) = Ai*A'*reshape(TA_prime( (0:24:72) + j), 4, 1);
  end
end
```

```
% Script to tune lab measurements of antenna beam coupling
% coefficients (eta's) using coincident AMSU observations on March 15, 1999
%
% WJB

% AMSU temperature retrieval
mar15r = ...
 [266.4880 265.1500 257.5120 251.7450 247.1640 242.1180 237.8270 233.5290 229.3660 ...
  225.6180 219.0610 214.9770 212.7880 211.3430 211.1300 211.1620 211.1510 211.4750 ...
  212.1850 212.2240 212.5570 212.9830 213.3050 213.2660 213.0850 213.2170 213.2850 ...
  213.4240 213.4600 213.5200 214.0940 216.0040 218.2880 220.8070 223.3010 226.4920 ...
  230.1150 233.5270 236.7200 239.8500 243.1790 246.7280 250.1370 253.1500 256.0460 ...
  258.5420 260.9290 263.1190 265.2270 267.1790 268.9410 270.6240 272.1090 273.1310 ...
  274.1660 275.1130 275.7710 276.4100 276.1900 275.9310 275.3180 274.7950 273.9860 ...
  273.1350 272.2730 271.5060]';

% AMSU water vapor retrieval
H2Oprofile = ...
 [      0       0       0       0       0       0       0       0       0 ...
        0       0       0       0       0       0       0       0       0 ...
        0       0  0.0009  0.0011  0.0012  0.0014  0.0014  0.0016  0.0018 ...
   0.0020  0.0022  0.0024  0.0033  0.0046  0.0065  0.0091  0.0123  0.0185 ...
   0.0290  0.0637  0.0889  0.1219  0.1694  0.2380  0.3264  0.4284  0.5525 ...
   0.6869  0.8421  1.0094  1.1958  1.3958  1.5979  1.8148  2.0289  2.2128 ...
   2.4122  2.6105  2.8240  3.0444  3.1015  3.1452  3.1194  3.1064  3.0253 ...
   2.9318  3.1864  3.0509]';

% AMSU pressure grid
pres_am_mar15 = ...
 [ 1   2   3   4   5   6   7   8   9  10 ...
  15  20  30  40  50  60  70  80  90 100 ...
 110 120 130 140 150 160 170 180 190 200 ...
 220 240 260 280 300 320 340 360 380 400 ...
 425 450 475 500 525 550 575 600 625 650 ...
 675 700 725 750 775 800 825 850 875 900 ...
 925 950 975 1000 1025 1050]';
```

```
% set surface parameters from buoy 45007
Ts=276.45; % from buoy 3.3C
Ps = 1018;                                                              40
emis=3;    % use fastem to calculate surface emisivity
alt=std76mb(49.91); % ~20.58 km

% use hydrostatic equation to calculate altitudes
if (Ps < 1000)
   pres_am_mar15(64) = Ps;
   pres_am_mar15(65:66) = [];
   H2Oprofile(65:66)=[];
   mar15r(64)=Ts;
   mar15r(65:66)=[];                                                    50
   H = ([mar15r(2:64)] + mar15r(1:63))/2;
   H = H * 8.3143 / (28.8 * 9.8);
   avgP = exp((log(pres_am_mar15(2:64,:)) + log(pres_am_mar15(1:63)))/2);
   diffP = diff(pres_am_mar15);
   dz = H .* (diffP ./ avgP); % km
   height = flipud(cumsum(flipud(dz)));
   height(64) = 0;
else
   pres_am_mar15(65) = Ps;
   pres_am_mar15(66) = [];                                              60
   H2Oprofile(66)=[];
   mar15r(65)=Ts;
   mar15r(66)=[];
   H = ([mar15r(2:65)] + mar15r(1:64))/2;
   H = H * 8.3143 / (28.8 * 9.8);
   avgP = exp((log(pres_am_mar15(2:65,:)) + log(pres_am_mar15(1:64)))/2);
   diffP = diff(pres_am_mar15);
   dz = H .* (diffP ./ avgP); % km
   height = flipud(cumsum(flipud(dz)));
   height(65) = 0;                                                      70
end

profile_mar15_tbarray=[height mar15r pres_am_mar15 H2Oprofile];
```

```
% call radiative transfer routines (see fastem.f)
% upwelling...
[Tb54r_,Tb118r_]=findTbfastem(profile_mar15,Ts,alt,emis,0,0);
% downwelling...
[Tb118s,Tb54s]=skytb(alt,profile_mar15);

% load data here (code omitted)
% need the following:
%
% t          (radiometric timestamps)
% trtd       (rtd timestamps)
% x          (radiometric counts) (17 x 25 x N)
% rtdfc      (rtd data, filtered and corrected) (29 x N)


CAL_RANGE = 2800:2880;
RANGE = 11:70;

% crop data
x = x(1:16, :, CAL_RANGE);

% set structures for calib3wls
count_data.counts = x;
count_data.time = t(:, CAL_RANGE);
count_data.hcal_index = [3:4]; % hot load
count_data.ccal_index = [24:25]; % amb load
count_data.scal_index = [1:2]; % sky cal

% rtd weights
w = [0.1568 0.1435 0.1402 0.1330 0.1412 0.1449 0.1402]';

rtd_data.wgtH = ones(16,1) * [zeros(1, 8) w' zeros(1,15)];
rtd_data.wgtC = ones(16,1) * [zeros(1, 23) .2 * ones(1,5) 0 0];

% TIGR profile set used to compute zenith data statistics (code omitted)
% values for 19.42 km
```

```
T_Z_1942 = [3.2263 3.5300 4.6601 11.6876 15.8043 29.1484 37.6094 69.5979]';
T_Z_118_1942 = [4.2479 4.7093 5.2818 6.3674 8.4721 14.7136 37.7476 91.1371]';
sigma_Z_1942 = [0.0152 0.0262 0.2728 1.7239 1.8919 2.9926 2.6230 0.7413]';
sigma_Z_118_1942 = [0.0701 0.1233 0.1888 0.3120 0.5464 1.2050 3.1599 5.0097]';

% values for 20.58 km

T_Z_2058 = [3.1179 3.3423 4.2023 10.1784 12.7607 23.6231 28.9200 53.7995]';
T_Z_118_2058 = [4.0544 4.3702 4.7626 5.5082 6.9596 11.3126 28.1133 71.4057]';
sigma_Z_2058 = [0.0107 0.0228 0.2167 1.4821 1.4786 2.4243 1.9871 0.3000]';
sigma_Z_118_2058 = [0.0493 0.0867 0.1331 0.2206 0.3887 0.8748 2.4879 4.8909]';

% interpolate to actual altitude
T_Z = interp1([19.42; 20.58], [T_Z_1942'; T_Z_2058'], alt,'linear','extrap')';
T_Z_118 = interp1([19.42; 20.58], [T_Z_118_1942'; T_Z_118_2058'], alt,'linear','extrap')';
sigma_Z = interp1([19.42; 20.58], [sigma_Z_1942'; sigma_Z_2058'], alt,'linear','extrap')';
sigma_Z_118 = interp1([19.42; 20.58], [sigma_Z_118_1942'; sigma_Z_118_2058'], alt,'linear','extrap')';

rtd_data.wgtS = [zeros(16, 29) [T_Z; T_Z_118]];
rtd_data.time = trtd(CAL_RANGE)';
rtd_data.rtdfc = rtdfc(CAL_RANGE,:);

% First-pass calibration to get T_N ...

% 54:
%  Assume 2K error on H load and 1.5K error on A load due to spillover
%  Assume 0.5K error on H due to gradient
% 118:
%  Assume 0.2K error on H
%  Assume 0.1K error on A

C = zeros(14, 3, 3);
W = C;
dT_54 = [.21 .13 .12 .16 .13 .15 .18 .18];
dT_118 = [.19 .23 .21 .25 .28 .34 .61 .89];
for i = 1:8
  C(i,:,:) = diag([dT_54(i)^2 + 4.25 dT_54(i)^2 + 2.25 dT_54(i)^2 + sigma_Z(i)^2]);
```

```
end
for i = 1:8                                                                    150
  C(i+8,:,:) = diag([dT_118(i)^2 + 0.04 dT_118(i)^2 + 0.01 dT_118(i)^2 + sigma_Z_118(i)^2]);
end
for i = 1:16
  W(i, :, :) = inv(squeeze(C(i, :, :)));
end

[Tb, TH_, TA_, TS_, CH_, CA_, CS_] = calib3wls(count_data, rtd_data, W, 4);
Tb54_uncorrected = Tb(1:8, :, :); % First pass

T_N = squeeze(mean(Tb(1:8, 7:21, :), 2));                                      160
TN = T_N(:, RANGE);

TZ = T_Z * ones(1, length(TN));

% sky data...
TH = squeeze(mean(TH_(1:8, 1:2, RANGE),2));
TA = squeeze(mean(TA_(1:8, 1:2, RANGE),2));
CH = squeeze(mean(CH_(1:8, 1:2, RANGE),2));
CA = squeeze(mean(CA_(1:8, 1:2, RANGE),2));
CZ = squeeze(mean(CS_(1:8, 1:2, RANGE),2));                                    170

C = squeeze(mean(x(1:8, 1:2, RANGE),2));
Tb54_u_sky = mean(((TH - TA) .* (C - CA) ./ (CH - CA) + TA)')';
a1 =    mean(((TN - TH) .* (C - CA) ./ (CH - CA))')';
a2 =    mean(((TZ - TH) .* (C - CA) ./ (CH - CA))')';
a3 =    mean(((TN - TA) .* (CH - C) ./ (CH - CA))')';
a4 =    mean(((TZ - TA) .* (CH - C) ./ (CH - CA))')';
A = [a1 a2 a3 a4];

% nadir data...                                                                180
TH = squeeze(mean(TH_(1:8, 14, RANGE),2));
TA = squeeze(mean(TA_(1:8, 14, RANGE),2));
CH = squeeze(mean(CH_(1:8, 14, RANGE),2));
CA = squeeze(mean(CA_(1:8, 14, RANGE),2));
CZ = squeeze(mean(CS_(1:8, 14, RANGE),2));
```

```
C = squeeze(mean(x(1:8, 14, RANGE),2));
Tb54_u_nadir = mean(((TH - TA) .* (C - CA) ./ (CH - CA) + TA)')';
b1 =   mean(((TN - TH) .* (C - CA) ./ (CH - CA))')';
b2 =   mean(((TZ - TH) .* (C - CA) ./ (CH - CA))')';                    190
b3 =   mean(((TN - TA) .* (CH - C) ./ (CH - CA))')';
b4 =   mean(((TZ - TA) .* (CH - C) ./ (CH - CA))')';
B = [b1 b2 b3 b4];

% First guess for eta:
% eta_H and eta_A are the lab measurements of the etas
eta_first_guess = [eta_H eta_A]';

delta_Tb54_sky = Tb54s - Tb54_u_sky;
Tb54r = Tb54r_(:,10); % nadir                                           200
delta_Tb54_nadir = Tb54r - Tb54_u_nadir;
delta_Tb54 = [delta_Tb54_sky delta_Tb54_nadir]';

% solve minimization numerically.
% if no solution feasible, relax constraints

for i = 1:8
  [delta_eta(:,i), fval, exitflag] = ...
    fmincon(inline('norm(x)'), eta_first_guess(:,i), -eye(4), eta_first_guess(:,i)-.001, ...
    [A(i,:); B(i,:)], delta_Tb54(:,i) - [A(i,:); B(i,:)] * eta_first_guess(:,i));   210
  if (exitflag < 0)
    pcounter = 1;
    ncounter = 1;
    pertp = [dT_54(i) 0]';
    pertn = -pertp;
    fprintf('***Beginning relaxation for channel %d\n', i);
  end
  while (exitflag < 0)
    if (mod(pcounter+ncounter,2))
      pert = pertp * pcounter;                                          220
      pcounter = pcounter + 1;
    else
```

```
    pert = pertn * ncounter;
    ncounter = ncounter + 1;
  end
  % try again ...
  [delta_eta(:,i), fval, exitflag] = ...
    fmincon(inline('norm(x)'), eta_first_guess(:,i), -eye(4), eta_first_guess(:,i)-.001, ...
    [A(i,:); B(i,:)], delta_Tb54(:,i) - [A(i,:); B(i,:)] * eta_first_guess(:,i)+pert);
  if (exitflag > 0)                                                                    230
    fprintf(' SUCCESS!! Relaxation %d with perturbation of %g.\n\n', pcounter+ncounter, pert(1));
  else
    fprintf(' Finished relaxation %d with perturbation of %g. Still trying. . .\n\n', pcounter+ncounter, pert(1));
  end
  if ((ncounter+pcounter) > 250)
    fprintf(' No solution found after %d relaxations.\n\n', ncounter+pcounter);
    break
  end
end
```

```
% Script to apply eta corrections to March 25, 1999 WINTEX data
% WJB

alt = 20.25;

% estimated statistics
sigma_N = [10 7 3 3 3 3 2 2]';
sigma_H = 0.5 * ones(8,1);
sigma_A = 0.1 * ones(8,1);
                                                                                       10
% sky data from TIGR
T_Z_1942 = [3.2263 3.5300 4.6601 11.6876 15.8043 29.1484 37.6094 69.5979]';
T_Z_118_1942 = [4.2479 4.7093 5.2818 6.3674 8.4721 14.7136 37.7476 91.1371]';
sigma_Z_1942 = [0.0152 0.0262 0.2728 1.7239 1.8919 2.9926 2.6230 0.7413]';
sigma_Z_118_1942 = [0.0701 0.1233 0.1888 0.3120 0.5464 1.2050 3.1599 5.0097]';

T_Z_2058 = [3.1179 3.3423 4.2023 10.1784 12.7607 23.6231 28.9200 53.7995]';
T_Z_118_2058 = [4.0544 4.3702 4.7626 5.5082 6.9596 11.3126 28.1133 71.4057]';
```

```
sigma_Z_2058 = [0.0107 0.0228 0.2167 1.4821 1.4786 2.4243 1.9871 0.3000]';
sigma_Z_118_2058 = [0.0493 0.0867 0.1331 0.2206 0.3887 0.8748 2.4879 4.8909]';

T_Z = interp1([19.42; 20.58], [T_Z_1942'; T_Z_2058'], alt)';
T_Z_118 = interp1([19.42; 20.58], [T_Z_118_1942'; T_Z_118_2058'], alt)';
sigma_Z = interp1([19.42; 20.58], [sigma_Z_1942'; sigma_Z_2058'], alt)';
sigma_Z_118 = interp1([19.42; 20.58], [sigma_Z_118_1942'; sigma_Z_118_2058'], alt)';

% thermal noise stats
dT_54 = [.21 .13 .12 .16 .13 .15 .18 .18];
dT_118 = [.19 .23 .21 .25 .28 .34 .61 .89];

% load data here (code omitted)
% need the following:
%
% t         (radiometric timestamps)
% trtd      (rtd timestamps)
% x         (radiometric counts) (17 x 25 x N)
% rtdfc     (rtd data, filtered and corrected) (29 x N)

CAL_RANGE = 1500:1700;
RANGE = 51:150;

% fill in sturctures for calib2 (two-point cal)
x = x(1:16, :, CAL_RANGE);
count_data.counts = x;
count_data.time = t(:, CAL_RANGE);
count_data.hcal_index = [3:4]; % hot load
count_data.ccal_index = [24:25]; % amb load
count_data.scal_index = [1:2]; % sky cal

% rtd weights
w = [0.1568 0.1435 0.1402 0.1330 0.1412 0.1449 0.1402]';

rtd_data.wgtH = ones(16,1) * [zeros(1, 8) w' zeros(1,15)];
rtd_data.wgtC = ones(16,1) * [zeros(1, 23) .2 * ones(1,5) 0 0];
rtd_data.wgtS = [zeros(16, 29) [T_Z; T_Z_118]];
```

```
rtd_data.time = trtd(CAL_RANGE)';
rtd_data.rtdfc = rtdfc(CAL_RANGE,:);

% First-pass calibration to get T_N ...

% 54:
% Assume 2K error on H load and 1.5K error on A load due to spillover
% Assume 0.5K error on H due to gradient
% 118:
% Assume 0.2K error on H
% Assume 0.1K error on A

C = zeros(16, 3, 3);
W = C;
for i = 1:8
  C(i,:,:) = diag([dT_54(i)^2 + 4.25 dT_54(i)^2 + 2.25 dT_54(i)^2 + sigma_Z(i)^2]);
end
for i = 1:8
  C(i+8,:,:) = diag([dT_118(i)^2 + 0.04 dT_118(i)^2 + 0.01 dT_118(i)^2 + sigma_Z_118(i)^2]);
end
for i = 1:16
  W(i, :, :) = inv(squeeze(C(i, :, :)));
end

[Tb, TH, TA, TS] = calib3wls(count_data, rtd_data, W, 4);
Tb118_new = Tb(9:16, :, :);     % Final product

Tb54_uncorrected = Tb(1:8, :, :); % First pass

T_N = squeeze(mean(Tb(1:8, 7:21, :), 2));
T_N_mean = mean(T_N(:, RANGE)')';
TA_mean = mean(TA(1,RANGE));
TH_mean = mean(TH(1,RANGE));

% Need expected errors in TA and TH after correction
% load tuned etas with expected errors (code omitted)
```

```
sigma_A = sqrt( T_N_mean.^2 .* sigma_eta_A(:, 1).^2 + T_Z.^2 .* sigma_eta_A(:, 2).^2 + ...
TA_mean.^2 .* (sigma_eta_A(:, 1).^2 + sigma_eta_A(:, 2).^2) + sigma_A.^2);

sigma_H = sqrt( T_N_mean.^2 .* sigma_eta_H(:, 1).^2 + T_Z.^2 .* sigma_eta_H(:, 2).^2 + ...
TH_mean.^2 .* (sigma_eta_H(:, 1).^2 + sigma_eta_H(:, 2).^2) + sigma_H.^2);

% Now, calibrate with corrections
count_data.counts = x(1:8, :, :);                                                    100
rtd_data.wgtH = ones(8,1) * [zeros(1, 8) w' zeros(1,15)];
rtd_data.wgtC = ones(8,1) * [zeros(1, 23) .2 * ones(1,5) 0 0];
rtd_data.wgtS = [zeros(8, 29) T_Z];

C = zeros(8, 3, 3);
W = C;
for i = 1:8
  C(i,:,:) = diag([dT_54(i)^2 + sigma_H(i)^2 dT_54(i)^2 + sigma_A(i)^2 dT_54(i)^2 + sigma_Z(i)^2]);
end
for i = 1:8                                                                          110
  W(i, :, :) = inv(squeeze(C(i, :, :)));
end

eta.T_N = T_N;
eta.T_Z = T_Z * ones(1, length(T_N));

eta.eta_C = eta_A;
eta.eta_H = eta_H;
eta.time = mean(t(7:21,CAL_RANGE),1);
                                                                                     120
[Tb54_corrected, TH, TA, TS] = calib3wls(count_data, rtd_data, W, 4, eta);
```

```
% Script to generate NAST-M equivalent Tb's from AMSU: March 26, 1999
% WJB

% AMSU temperature retrieval
```

```
mar25r = ...
  [265.9870 260.7420 251.0490 244.2440 238.9910 234.1420 230.1000 226.2930 222.8360 ...
   219.6710 215.0550 212.5960 211.9200 210.6610 210.2610 210.2570 210.4540 211.3600 ...
   212.6520 213.6390 214.5100 215.3150 216.0390 216.5430 216.9600 217.5330 218.1110 ...
   218.6200 219.0790 219.4740 219.7040 221.0670 222.1690 223.0830 224.0140 225.8450 ...
   228.3120 230.8910 233.3160 235.8290 238.5130 241.6020 244.5930 247.4100 250.1190 ...
   252.6370 255.0450 257.3440 259.5640 261.7190 263.6780 265.6020 267.3200 268.7200 ...
   270.1200 271.4190 272.5730 273.7060 274.2010 274.6850 274.9110 275.1120 275.0640 ...
   274.9100 274.8250 274.7210]';


% Water vapor profile (g/cm^3) from x3260112 raob
H2Oprofile = ...
  [0      0      0      0      0      0      0      0      0      ...
   0      0      0      0      0      0      0      0      0      ...
   0      0      0      0      0      0      0      0      0      ...
   0      0      0      0      0.0011 0.0028 0.0041 0.0065 0.0102 ...
   0.0109 0.0190 0.0360 0.0753 0.1101 0.1548 0.1733 0.2471 0.3641 ...
   0.3735 0.4592 0.5565 0.5157 0.5402 0.5114 0.3480 0.1712 0.3422 ...
   0.8850 1.1296 1.3084 1.4626 1.9406 2.3058 2.5464 2.5980 2.6173 ...
   2.6173 2.6173 2.6173]';

% AMSU pressure grid
pres_am_mar25 = ...
  [ 1   2   3   4   5   6   7   8   9   10 ...
   15  20  30  40  50  60  70  80  90  100 ...
  110 120 130 140 150 160 170 180 190 200 ...
  220 240 260 280 300 320 340 360 380 400 ...
  425 450 475 500 525 550 575 600 625 650 ...
  675 700 725 750 775 800 825 850 875 900 ...
  925 950 975 1000 1025 1050]';

Ts=275.35; % from AVHRR
Ps=1030;  % from buoy

% use hydrostatic equation to calculate altitudes
if (Ps < 1000)
```

```
   pres_am_mar25(64) = Ps;
   pres_am_mar25(65:66) = [];
   H2Oprofile(65:66)=[];
   mar25r(64)=Ts;
   mar25r(65:66)=[];
   H = ([mar25r(2:64)] + mar25r(1:63))/2;
   H = H * 8.3143 / (28.8 * 9.8);
   avgP = exp((log(pres_am_mar25(2:64,:)) + log(pres_am_mar25(1:63)))/2);
   diffP = diff(pres_am_mar25);                                                    50
   dz = H .* (diffP ./ avgP); % km
   height = flipud(cumsum(flipud(dz)));
   height(64) = 0;
else
   pres_am_mar25(65) = Ps;
   pres_am_mar25(66) = [];
   H2Oprofile(66)=[];
   mar25r(65)=Ts;
   mar25r(66)=[];
   H = ([mar25r(2:65)] + mar25r(1:64))/2;                                          60
   H = H * 8.3143 / (28.8 * 9.8);
   avgP = exp((log(pres_am_mar25(2:65,:)) + log(pres_am_mar25(1:64)))/2);
   diffP = diff(pres_am_mar25);
   dz = H .* (diffP ./ avgP); % km
   height = flipud(cumsum(flipud(dz)));
   height(65) = 0;
end


profile_mar25_tbarray=[height mar25r pres_am_mar25 H2Oprofile];                    70


emis=3;
alt=20.25;
% slight angular offset
offset=-1.5;
[Tb54fastem_tbarray,Tb118fastem_tbarray]=findTbfastem_tbarray(profile_mar25_tbarray,Ts,alt,emis,0,offset);
```

```
% Script to make nice displays of the NAST-M window channels
% Based heavily on the code written by Mike Schwartz
%
% WJB

% load data here (code omitted)
% need the following:
%
% t         (radiometric timestamps)
% trtd      (rtd timestamps)                                    10
% x         (radiometric counts) (17 x 25 x N)
% rtdfc     (rtd data, filtered and corrected) (29 x N)

% broad range used to define baseline; not very critical
RANGE=3500:4600;

% range used for eye retrieval
EYE=3710:3745;

% range used for image plots                                    20
EYE1=3695:3830;

% range used for eye perturbations
CLEAR=3895:3904;

% need to define colormap to use all colors
% note that ghostview-based pdf converters aren't real happy with this
colormap(ones(110,3));
map = colormap('jet');
colormap(ones(290,3));                                          30
% use blue for all values below 180 K
map = [zeros(180,3); map];

% put up figure
f=figure(1)
clf
orient landscape
```

```
% move figure so it doesn't go off-screen
foo = get(f,'position');
set(f,'position',[foo(1) foo(2)-300 foo(3) foo(4)])
wysiwyg

[h, colbar_ax]=show8strip2_contour(Tbf,[1 9],EYE1,0,[190 280],[4 4],map)
set(colbar_ax,'position',[0.075 0.0500   0.0200    0.8500])
axes(colbar_ax)
ylabel('Brightness Temperature (K)','fontsize',14)
ax=getsub;
set(ax(1), 'xtickla', hrminsec(t(get(ax(1), 'xtick'))))
set(ax(2), 'xtickla', hrminsec(t(get(ax(2), 'xtick'))))
title_ax = text(29, 283.5, 'Microwave Images of Hurricane Bonnie','fontsize',16)


function [hand, colbar_ax]=show8strip2_contour(Tb, chs, r, rbase, Tran, Interp, cmap)
% Tb is a 3-dim matrix [channels x angle x records]
%      there need not be eight channels. More or less are allowed.
% chs  channels to display 1..8 (THIS COULD BE SELECTED BY RANGE IN Tb)
% r    scans to show
% rbase range of scans to set as zero baseline
%      (use 0 or '' for no baseline subtraction)
% Tran range of temperatures to scale to colormap
% Interp [xinterp yinterp] interpolates data for smoothing between values
% cmap optional colormap (tweaked zebra is default)

if exist('chs')~=1, chs=[7:-1:1 8]; end;
if chs == ':', chs=[7:-1:1 8]; end;
if exist('r')~=1, r=1:size(Tb,3); end;
if r == ':', r=1:size(Tb,3); end;

if exist('Interp')~=1, Interp=[1 1]; end;
if exist('cmap')~=1, cmap=olicolors(1); end;
if exist('Tran')~=1, Tran=[-100 10]; end;
if length(Tran)~=2, Tran=[-100 10]; end;
if exist('rbase')~=1, rbase=1:size(Tb,3); end;
if ~isempty(rbase) & rbase==0, rbase=''; end;
```

```
Nch=length(chs);

% set up some dimensions
top = .1;
bot = .05;                                                          80
separ = .05;
lmar = .17;
rmar = .08;

wid = (1 - (Nch-1)*separ -top-bot)./Nch ;

for i=1:Nch,
  axes('Position', [lmar bot + (i-1) .*(wid+separ) 1-lmar-rmar wid ]);
  hand(i)=showstrip_contour(squeeze(Tb(chs(i), 5:23, :)), r, rbase, Tran, Interp, cmap,i);
                                                                    90
  if (i == 1)
    title('50.3 GHz', 'fontsize', 14)
  else
    title('118.75 \pm 3.5 GHz', 'fontsize',14)
  end
end

colbar_ax = colbar_wjb([.1 bot .02 1-top-bot], Tran)

                                                                    100
function hand=showstrip_contour(ch, r, rbase, Tran, Interp, cmap, i)
% display a strip chart of channel data
%
% ch is a matrix of calibrated brightnesses, typically [14,L] or [16,L]
% r is a range of columns to be displayed
% rbase is a range of columns to use as a zero baseline
%  IF rbase is the same length as the scan width, it is take to be the baseline itself
% Tran is [minT maxT] of temperatures to map to extremes of the colormap
% Interp is [rowfactor columnfactor] for interpolation (default [1 1])
% cmap is optional colormap                                         110
```

```
if size(ch,1)==16, r1=1:14;
else r1=1:size(ch,1); end;

if exist('cmap')~=1, cmap='zebra'; end;
if isstr('cmap'), colormap(cmap);cmap=colormap; end;
if exist('rbase')~=1,
  BASE=zeros(length(r1),1);
elseif length(rbase)==0, BASE=zeros(length(r1),1);                    120
elseif length(rbase)==length(r1), BASE=rbase(:);
else BASE= median(ch(r1,rbase)')';
end


if exist('Interp')~=1, Interp=[1 1]; end;
if Interp(1)==1 & Interp(2)==1,
  X = ceil((((ch(r1,r)- BASE *ones(1,length(r)))-Tran(1))./(Tran(2)-Tran(1))).* length(cmap));
else
  X = ch(r1,r)- BASE *ones(1,length(r)) ;                             130

  if (0)  % mike's old code
    X = interp2(1:C, (1:R)', X, (Interp(2):(C*Interp(2)))./Interp(2), (Interp(1):(R*Interp(1)))'./Interp(1));
  end
  % code to unwarp the swath
  [R,C]=size(X);
  angle_dis = 10 * tan([(-7.2*9):7.2:(7.2*9)] * pi/180);
  bo_diddly = linspace(min(angle_dis), max(angle_dis), Interp(1) * (R-1) + 1);
  X = interp2(1:C, angle_dis', X, (Interp(2):(C*Interp(2)))./Interp(2), bo_diddly', 'spline');
end                                                                   140

r_all = interp1(1:C, r, (Interp(2):(C*Interp(2)))./Interp(2));

hand=image(r_all, bo_diddly,X);

hold on

if (i == 1) % 50.3
```

```
   contour(r_all, bo_diddly, X, [240:4:280], 'k-'); % ~ 10K per contour
   xlabel('Time (UTC) August 26, 1998','fontsize',14)
else        % 118.75 +/- 3.5
   contour(r_all, bo_diddly, X, [240:4:280], 'k-'); % ~ 10K per contour
   contour(r_all, bo_diddly, X, [190:5:235], 'k-'); % ~ 10K per contour
end

ylabel('Distance (km)','fontsize',14)
set(gca,'DataAspectRatio',[1 0.9 1]) % get scale right - 1 scan ~ 1.1 km



function Cax=colbar_wjb(pos, Tran)

if exist('pos')~=1, pos=[.05 .1 .02 .7]; end;
ax=gca;
map=colormap;
mapsize=size(map,1);
if exist('Tran'),
 CLim = Tran;
else
  CLim=get(gca, 'CLim')
end;
delete(findobj(gcf, 'userdata', 'colbar'));
Cax=axes('Position', pos, 'userdata', 'colbar');
set(Cax, 'userdata', 'colbar')
step=(CLim(2)-CLim(1))/(mapsize-1);
image(1:2, CLim(1):step:CLim(2), (ones(2,1)*(CLim(1):step:CLim(2)))');
set(gca, 'YDir', 'Normal', 'XTickLabels', [], 'XTick', [], 'Box', 'on', 'TickDir', 'out', 'userdata', 'colbar');
set(gcf, 'CurrentAxes', ax);
```

E.2 ION

```
function [S, noise, order] = ion(x, iter)
%
%    ION Iterative Order and Noise estimation algorithm
%
```

```
%       [S, NOISE, ORDER] = ion(X, ITER)
%
%       X is an m-by-n data matrix. m represents the number of
%       observations, and n represents the number of variables.
%
%       ITER is the desired number of iterations of the ION algorithm.
%       We have found that ITER = 10 is generally more than sufficient.
%
%       S (n-by-1 vector) is the noise variance vector.
%       NOISE (m-by-n matrix) is the retrieved noise sequence(s).
%       ORDER is the estimated number of independent signals.
%
%       Also see SOEBS, SCREEORDER, FIXEIG, NEBEM2.
%


%
%       This version of the ION algorithm is written in MATLAB 5.2. and
%       includes four subroutines: SOEBS, SCREEORDER, FIXEIG, NEBEM2.
%
%
%           The Iterated Order and Noise estimation (ION) algorithm was
%       developed by Junehee Lee and David H. Staelin at the
%       Massachusetts Institute of Technology (MIT) under sponsorship of
%       the MIT Leaders for Manufacturing Program. It is documented in
%       Junehee Lee's Ph.D. thesis, MIT Department of Electrical Engineering
%       and Computer Science, March, 2000.
%
%           ION iteratively estimates the noise variance associated with
%       each variable of a multivariate data set, and the order of the
%       underlying signal set. No apriori information about the data set
%       is required provided that the input data is properly arrayed in
%       a matrix. If the data set is to be used subsequently for linear
%       regression to predict target variables, these target variables
%       should also be included in the input matrix. This version of ION
%       employs the scree plot to estimate order and the EM algorithm to
%       estimate noise. Any of several other published methods can
%       alternatively be used to estimate order or noise within the
```

```
%       structure of the ION algorithm.
%
%           The order, noise, and noise variance estimates produced by the
%       ION algorithm can be used in a variety of ways. For example, they
%       can offer improved noise estimates, principal components, Wiener
%       filtering, and linear regressions based on limited training data
%       sets. The greatest benefits are achieved for cases where the number
%       of variables is very large, typically more than a minimum of ten
%       or twenty, and at least three times the order of the underlying
%       signal; cases where the training set is small also typically yield
%       greater benefits. These benefits have exceeded 10 dB in some test
%       cases.
%
%
%       (c) Copyright 2000 M.I.T.
%
%       Permission is hereby granted, without written agreement or
%       royalty fee, for Hewlett Packard Corporation (HP) to use, copy,
%       modify, and distribute within HP this software and its
%       documentation for any purpose, provided that the above copyright
%       notice and the following three paragraphs appear in all copies of
%       this software.
%
%       In no event shall M.I.T. be liable to any party for direct,
%       indirect, special, incidental, or consequential damages arising
%       out of the use of this software and its documentation, even if
%       M.I.T. has been advised of the possibility of such damage.
%
%       M.I.T. specifically disclaims any warranties including, but not
%       limited to, the implied warranties of merchantability, fitness
%       for a particular purpose, and non-infringement.
%
%       The software is provided on an "as is" basis and M.I.T. has no
%       obligation to provide maintenance, support, updates, enhancements,
%       or modifications.
```

```
                                                                          80
[Nobs, Nvar] = size(x);
G = ones(Nvar,1); S = ones(Nvar,1);

for it = 1 : iter
Sx = cov(x * inv(diag(sqrt(S))));
[evect, evals] = eig(Sx);
[evect, evals] = fixeig(evect, evals);

   est_p = screeorder(evals)                                              90

   [G, est_signal,exp_lv]=nebem2(x* inv(diag(sqrt(S))) , est_p ,10);
   S_old = S;
   S = S.* G;
end

est_signal = est_signal * diag(sqrt(S_old));
noise = x - est_signal;
order = est_p;
                                                                         100
```

```
function [evect,evals] = fixeig(evect,evals)

%      FIXEIG [NewEvect, NewEvals] = fixeig(EVECT, EVALS)
%
%           EVALS is a vector of eigenvalues.
%           EVECT is a matrix whose column is the eigenvectors.
%           The first column of EVECT is the eigenvector corresponding
%           to the first element of EVALS.
%
%           NewEvals is the eigenvalue in descending order.                10
%           NewEvect is the re-ordered eigenvector matrix.
%
```

```
%
%       (c) Copyright 2000 M.I.T.
%
%       Permission is hereby granted, without written agreement or
%       royalty fee, for Hewlett Packard Corporation (HP) to use, copy,
%       modify, and distribute within HP this software and its
%       documentation for any purpose, provided that the above copyright
%       notice and the following three paragraphs appear in all copies of
%       this software.
%
%       In no event shall M.I.T. be liable to any party for direct,
%       indirect, special, incidental, or consequential damages arising
%       out of the use of this software and its documentation, even if
%       M.I.T. has been advised of the possibility of such damage.
%
%       M.I.T. specifically disclaims any warranties including, but not
%       limited to, the implied warranties of merchantability, fitness
%       for a particular purpose, and non-infringement.
%
%       The software is provided on an "as is" basis and M.I.T. has no
%       obligation to provide maintenance, support, updates, enhancements,
%       or modifications.



evals = diag(evals);
evals_temp = evals;

dim = size(evals);
for i = 1:dim(1)
   mx = max(evals);
   for j = 1:dim(1)
      if mx == evals(j,1)
         loc = j;
      end
   end
   evals_temp(i) = mx;
```

```
   evect_temp(:,i) = evect(:,loc);
   evals(loc,1) = (-1) * (abs(evals(loc,1)));
end
evals = evals_temp;
evect = evect_temp;
[evect,evals];
```

```
function [a,b,R_SQUARED]=lr1(x,Y)
%
%  LR1      [a,b,R_squared]=lr1(x,Y)
%           Multi-dimensional linear regression
%           x is a m by 1 vector
%           Y is a m by n matrix
%           a and b is ther coefficient which fits the data points on
%
%                x=a+Y*b
%                                                                    10
%           If Y and x is not normalized, this command will NOT normalize
%           them before the regression (as opposed to LR).
%
%           R_squared is the variability of x explained by Y
%            (in terms of percentage)
%


%
%       (c) Copyright 2000 M.I.T.
%                                                                    20
%       Permission is hereby granted, without written agreement or
%       royalty fee, for Hewlett Packard Corporation (HP) to use, copy,
%       modify, and distribute within HP this software and its
%       documentation for any purpose, provided that the above copyright
%       notice and the following three paragraphs appear in all copies of
%       this software.
```

```
%
%       In no event shall M.I.T. be liable to any party for direct,
%       indirect, special, incidental, or consequential damages arising
%       out of the use of this software and its documentation, even if
%       M.I.T. has been advised of the possibility of such damage.
%
%       M.I.T. specifically disclaims any warranties including, but not
%       limited to, the implied warranties of merchantability, fitness
%       for a particular purpose, and non-infringement.
%
%       The software is provided on an "as is" basis and M.I.T. has no
%       obligation to provide maintenance, support, updates, enhancements,
%       or modifications.
```

```
mY=mean(Y);
mx=mean(x);
new_Y=Y-ones(size(Y,1),1)*mY;
new_x=x-mx;
b=regress1(new_x,new_Y);
a=ones(size(x,1),1)*mx-mY*b;

Sxx=(x-mean(x))'*(x-mean(x));
Rss=(x-a-Y*b)'*(x-a-Y*b);
R_SQUARED=(Sxx-Rss)/Sxx*100;
```

```
function [S, estimated_signal,Exp_lv] = nebem2(x, p, it)
%
%
%  [S, estimated_signal,Exp_lv] = nebem2(x, p, it)
%
%  NEBEM2 Noise Estimation through EM algorithm
%
```

```
%           x is an m-by-n data matrix. m represents the number of
%           observations, and n represents the number of variables.
%                                                                    10
%           p is the number of latent variables.
%
%           'it' represents how many iterations will be performed for EM.
%
%           S is the estimated noise variances for each variables.

%
%       (c) Copyright 2000 M.I.T.
%
%       Permission is hereby granted, without written agreement or   20
%       royalty fee, for Hewlett Packard Corporation (HP) to use, copy,
%       modify, and distribute within HP this software and its
%       documentation for any purpose, provided that the above copyright
%       notice and the following three paragraphs appear in all copies of
%       this software.
%
%       In no event shall M.I.T. be liable to any party for direct,
%       indirect, special, incidental, or consequential damages arising
%       out of the use of this software and its documentation, even if
%       M.I.T. has been advised of the possibility of such damage.   30
%
%       M.I.T. specifically disclaims any warranties including, but not
%       limited to, the implied warranties of merchantability, fitness
%       for a particular purpose, and non-infringement.
%
%       The software is provided on an "as is" basis and M.I.T. has no
%       obligation to provide maintenance, support, updates, enhancements,
%       or modifications.

                                                                     40

[m, n] = size(x);
A_est = randn(n,p);
% A_est = ones(n,p);
```

```
G_est = diag(0.5*ones(n,1));

S = [ ];
for repeat = 1 : it

% E - STEP                                                              50

   Exp_lv = x * (G_est \ A_est) / (A_est' * (G_est \ A_est) + eye(p,p));

   Exp_lv2 = m *eye(p,p)/ (A_est' * pinv(G_est) * A_est + eye(p,p)) + . . .
((A_est' * (G_est \ A_est) + eye(p,p)) \ A_est') * (G_est \ x') * x * . . .
(G_est \ A_est) /( A_est' * (G_est \ A_est) + eye(p,p));

% M - STEP

   A_est = (x'* Exp_lv) / Exp_lv2;                                       60
   G_est = [ ];

   for index_j = 1:n
     G_est_j = (x(:,index_j)' * x(:,index_j) - A_est(index_j,:) * Exp_lv' *x(:,index_j)) / m;
     G_est = [G_est; G_est_j];
   end

   G_est = diag(G_est);
end
                                                                         70
S = diag(G_est);
estimated_signal = Exp_lv * A_est' ;
```

```
function b= regress1(y,X,alpha)
% REGRESS1 Performs multiple linear regression using least squares.
%      b = REGRESS1(y,X) returns the vector of regression coefficients, B.
%      Given the linear model: y = Xb,
%      (X is an nxp matrix, y is the nx1 vector of observations.)

%      References:
```

```
%       [1] Samprit Chatterjee and Ali S. Hadi, "Influential Observations,
%       High Leverage Points, and Outliers in Linear Regression",
%       Statistical Science 1986 Vol. 1 No. 3 pp. 379-416.
%       [2] N. Draper and H. Smith, "Applied Regression Analysis, Second
%       Edition", Wiley, 1981.

%    B.A. Jones 3-04-93
%    Copyright (c) 1993 by The MathWorks, Inc.
%    $Revision: 1.4 $ $Date: 1993/10/04 12:26:29 $

if nargin < 2,
   error('REGRESS requires at least two input arguments.');
end

% Check that matrix (X) and left hand side (y) have compatible dimensions
[n,p] = size(X);
[n1,collhs] = size(y);
if n~=n1,
   error('The number of rows in Y must equal the number of rows in X.');
end

if collhs ~= 1,
   error('Y must be a vector, not a matrix');
end

% Find the least squares solution.
[Q R]=qr(X);
b = R\Q'*y;
```

```
function p = screeorder(evals)
%
%  SCREEORDER Automatic Order Estimation Based on the Scree Plot.
%
%           EVALS is a vector of eigenvalues in decending order.
%
```

```
Nvar = length(evals);

log_evals = log10(abs(evals));

[slope, intersect] = soebs(evals);
est_eig = intersect + [1: Nvar]' * slope;
L = log_evals - est_eig;

% The following changed from .4 and .6 to .1 and .2 - works better
% for some NAST-I data
```

```
begins = floor(Nvar * 0.1);
ends = ceil(Nvar * 0.2);
p = min(find(L - 20 * std(L(begins : ends)) < 0)) -1 ;
```

```
function [slope, intersect] = soebs(eigenvalues)
%
%       SOEBS  [slope, intersect] = soebs(eigenvalues)
%              Signal Order Estimation by Scree plot
%              EIGENVALUES are the eigenvalues of the
%              covariance matrix of data in descending order.
%              SLOPE and INTERSECT are the slope and intesect of
%              the linear line which fits best the noise eigenvalues
%              when the scree plot is in logarithmic y-axis.
%
%
%       (c) Copyright 2000 M.I.T.
%
%       Permission is hereby granted, without written agreement or
%       royalty fee, for Hewlett Packard Corporation (HP) to use, copy,
%       modify, and distribute within HP this software and its
%       documentation for any purpose, provided that the above copyright
%       notice and the following three paragraphs appear in all copies of
%       this software.
%
%       In no event shall M.I.T. be liable to any party for direct,
%       indirect, special, incidental, or consequential damages arising
%       out of the use of this software and its documentation, even if
%       M.I.T. has been advised of the possibility of such damage.
%
%       M.I.T. specifically disclaims any warranties including, but not
```

```
%       limited to, the implied warranties of merchantability, fitness
%       for a particular purpose, and non-infringement.
%                                                                        30
%       The software is provided on an "as is" basis and M.I.T. has no
%       obligation to provide maintenance, support, updates, enhancements,
%       or modifications.


n = length(eigenvalues);
logeig = log10(eigenvalues);

% The following changed from .4 and .6 to .1 and .2 - works better
% for some NAST-I data                                                   40
begins = floor(n * 0.1);
ends = ceil(n * 0.2);

[intersect, slope, foo] = lr1(logeig(begins:ends),[begins:ends]');
intersect = intersect(1);
```

E.3 Neural network temperature profile retrieval

```
% NNET_CLEAR.m
%
% Script for estimating temperature profile from AIRS/AMSU-A/MHS
% observations. Uses MATLAB neural network toolbox.
%
% Bill Blackwell


[errcode,hostname] = system('hostname');
hostname(end) = [];                                                      10

% training/validation profiles
```

```
NUM_TRAINING = 10000;
TRAINING = round(linspace(1, 11997, NUM_TRAINING));
VALIDATION = 1:11997;
VALIDATION(TRAINING) = [];

% Number of PPC coefficients to use
TEMP_KEEPERS = 30;

% Number of nodes in hidden layer
NUM_NODES1 = 20;

% Maximum number of training epochs
MAX_EPOCHS = 100;

% load training data
fprintf('*** Starting to load data. . .\n');
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/Airs_Freq
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/NoiseVector
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/Anvec
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/Mnvec
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/p66

iwd = '/usr/dicke1/bill/AIRS_cloud_clearing/airs_radiances/';
awd = '/usr/dicke1/bill/AIRS_cloud_clearing/amsu_radiances/';
mwd = '/usr/dicke1/bill/AIRS_cloud_clearing/mhs_radiances/';

noise = [anvec mnvec noise_vector];

load /usr/dicke1/bill/AIRS_cloud_clearing/airs_radiances/airs.clear.night.ir04.1013.Ncl.mat
airs_clear = x(:, 1:11997);
clear x
mean_airs_clear = mean(airs_clear(:, TRAINING)')';
airs_clear = diag(1./noise_vector) * (airs_clear - mean_airs_clear * ones(1, length(airs_clear)));

Crr = airs_clear(:, TRAINING) * airs_clear(:, TRAINING)' / (length(TRAINING)-1) + eye(2371);
[evects, evals] = eigs(Crr, NAPC_KEEPERS);
```

```
% load temperature data
load /usr/dicke1/bill/AIRS_cloud_clearing/profiles/cld2lr.mat tsurf t
tsurf = tsurf(1:11997)';
mean_tsurf = mean(tsurf(TRAINING));
tsurf = tsurf - mean_tsurf;

% Use only 0-15 km data
t = t(20:64,1:11997);
mean_t = mean(t(:, TRAINING)')';
t = t - mean_t * ones(1,11997);
t = [t; tsurf];
Ctt = t(:, TRAINING) * t(:, TRAINING)' / (length(TRAINING)-1);

Ctr = t(:, TRAINING) * airs_clear(:, TRAINING)' / (length(TRAINING)-1);

% Linear regression (just to check)
L = Ctr / Crr;
lin_err = sqrt(diag(Ctt - L * Ctr'));
lin_est = L * (airs_clear(:, VALIDATION) + randn(size(airs_clear(:, VALIDATION))));

MSE = mean(lin_err.^2);

% Compute PPC coefficients
[ppc_evects, ppc_evals] = eigs(L * Ctr', TEMP_KEEPERS);
[U, S, V] = svd(ppc_evects' * L);
clear Crr
V = V(:, 1:TEMP_KEEPERS);
pcs = V' * airs_clear;

% Normalize data
s_pcs = std(pcs(:, TRAINING)')';
pcs = diag(1./s_pcs) * pcs;
Snn = sqrtm(diag(1./s_pcs) * V' * diag(noise_vector.^2) * V * diag(1./s_pcs));

% Compute nine retrievals separately
levels_1 = [1 7 12 17 22 27 32 37 42];
levels_2 = [6 11 16 21 26 31 36 41 46];
```

```
for j = 1:length(levels_1)

fprintf('--- Preparing neural network. . .\n');                                    90
nnet=newff(minmax(pcs')',[NUM_NODES1 levels_2(j) - levels_1(j) + 1],{'tansig','purelin'},'trainlm');
% set pertinent intialization parameters and reinitialize
nnet.layers{2}.initFcn = 'initwb';
nnet.layers{1}.initFcn = 'initwb';
nnet.inputWeights{1}.initFcn='rands';
nnet.layerWeights{2,1}.initFcn = 'rands';
nnet.biases{2}.initFcn='rands';
nnet.biases{1}.initFcn='rands';
nnet=init(nnet);
nnet.IW{1}=0.5.*nnet.IW{1};                                                       100

% set pertinent training parameters
nnet.trainParam.max_fail=300;
nnet.trainParam.show = 250;
nnet.trainParam.epochs = 1;
nnet.trainParam.goal = 1e-10;
nnet.trainParam.searchFcn = 'srchgol';
nnet.trainParam.minstep = 0;
nnet.trainParam.time = inf;
nnet.performFcn='mse';                                                            110

TV.T = t(levels_1(j):levels_2(j),VALIDATION);

% For graphical output
fig = figure;

nnet_best = zeros(size(nnet));
validation_error_best = inf;
clear training_error validation_error
for i = 1:MAX_EPOCHS                                                              120
  % Change noise every epoch
  foo = pcs(:,TRAINING) + Snn * randn(size(pcs(:,TRAINING)));
  TV.P = pcs(:,VALIDATION) + Snn * randn(size(pcs(:,VALIDATION)));
```

```
  [nnet,tr]=train(nnet, foo, t(levels_1(j):levels_2(j),TRAINING), [], [], [], TV);
  validation_error(i) = tr.tperf(end);
  training_error(i) = tr.perf(end);
  if (validation_error(i) < validation_error_best)
    nnet_best = nnet;
    validation_error_best = validation_error(i);
    fprintf(' NEW minimum found! (Chunk %d of %d: Training error = %g, Validation Error = %g)\n', ... 130
    j, length(levels_1), training_error(i), validation_error(i));
  end
  figure(fig)
  clf
  plot(1:i, training_error)
  grid on; hold on
  plot(1:i, validation_error,'r')
  drawnow
  if (rem(i, 10) == 0)
    eval(['save traingdx_out_' hostname '_sim_run1_' num2str(i) ...                    140
    '.mat nnet nnet_best tr training_error validation_error']);
  end
end

eval(['save nn_AIRS_clear_' num2str(j) ' nnet_best . . .
training_error validation_error']);

end
```

E.4 Neural network cloud clearing

```
% Script to estimate clear-column radiances (derived from AIRS, AMSU, and MHS)
%
% Training data: airs.cld2lr.night.ir04.1013.[1 3 7 9].Nc2.mat
%                amsu.cld2lr.night.mw075.1013.[1 3 7 9].Nc2.mat
%                mhs.cld2lr.night.mw075.1013.[1 3 7 9].Nc2.mat
%
% Validation data: airs.cld2lr.night.ir04.1013.5.Nc2.mat
```

```
%                amsu.cld2lr.night.mw075.1013.5.Nc2.mat
%                mhs.cld2lr.night.mw075.1013.5.Nc2.mat

% Number of radiance PC's from warmest spot
nn_param.WARM_KEEPERS = 30;

% Number of PC's from each pixel pair
nn_param.DIFF_KEEPERS = 5;

% Numper of clear-column PC's to estimate
nn_param.OUT_KEEPERS = 30;

% Number of nodes in first hidden layer
nn_param.NUM_NODES1 = 60;

% Number of nodes in second hidden layer
nn_param.NUM_NODES2 = 45;

% IR channels to estimate
nn_param.IR_OUTPUT_CHANNELS = [101:400 1901:2000 floor(linspace(2001, 2089, 75)) 2090:2114];

% Number of surface PC's to filter
nn_param.SURFACE_FILTER_PCS = 0;

% Change the noise after this many epochs
nn_param.NOISE_ROTATE = 5;

% Save output after this many epochs
nn_param.SAVE_OUTPUT = 50;

% Size of each batch of training vectors
nn_param.TRAIN_LENGTH = 5000;

% Training algorithm
nn_param.TRAIN_ALG = 'trainscg_wjb';

% load training stats
```

```
load /usr/dicke1/bill/AIRS_cloud_clearing/neural_nets/NN_STATS_092501

% load training data
CHUNKS = [1 3 7 9];
NUM_CHUNKS = length(CHUNKS);
                                                                    50
fprintf('*** Starting to load data. . .\n');
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/Airs_Freq
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/NoiseVector
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/Anvec
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/Mnvec
load /usr/dicke1/bill/AIRS_cloud_clearing/matlab/p66
% The following are 50 15-micron channels with peaks between levels
% 56-60 (inclusive) which corresponds to ~ 1-2km
load /usr/dicke1/bill/AIRS_cloud_clearing/neural_nets/airs_warmspot_detect_channels
iwd = '/usr/dicke1/bill/AIRS_cloud_clearing/airs_radiances/';                60
awd = '/usr/dicke1/bill/AIRS_cloud_clearing/amsu_radiances/';
mwd = '/usr/dicke1/bill/AIRS_cloud_clearing/mhs_radiances/';

noise = [anvec mnvec noise_vector];

load /usr/dicke1/bill/AIRS_cloud_clearing/airs_radiances/airs.clear.night.femis.1013.Ncl.mat
airs_clear = x;
clear x
airs_clear_ = airs_clear - mean_airs_clear * ones(1, length(airs_clear));
clear airs_clear                                                            70

if (nn_param.SURFACE_FILTER_PCS > 0)
   fprintf('*** Using surface filter: Removing %d surface PC''s.\n', nn_param.SURFACE_FILTER_PCS);
   load napcs_2_surf.mat evects
   surface_filter = eye(2391) - diag(noise) * evects(:, 1:nn_param.SURFACE_FILTER_PCS)
      ... * evects(:, 1:nn_param.SURFACE_FILTER_PCS)' * diag(1./noise);
else
   fprintf('*** NOT USING SURFACE FILTER ***\n');
   surface_filter = eye(2391);
   nn_param.SURFACE_FILTER_PCS = [];                                        80
end
```

```
for chunk_loop = 1:NUM_CHUNKS

ifile = ['airs.cld2lr.difsurf.night.ir04.1013.' num2str(CHUNKS(chunk_loop)) '.Nc2.mat'];
fprintf(['Loading ' iwd ifile '\n']);
eval(['load ' iwd ifile]);
airs_cloudy = x;
clear x

% AIRS
airs_cloudy_ = airs_cloudy - mean_airs_cloudy * ones(1, length(airs_cloudy));
clear airs_cloudy

% AMSU-A
afile = ['amsu.cld2lr.difsurf.night.mw075.1013.' num2str(CHUNKS(chunk_loop)) '.Nc2.mat'];
fprintf(['Loading ' awd afile '\n']);
eval(['load ' awd afile]);
amsu_cloudy = a;
clear a
amsu_cloudy_ = amsu_cloudy - mean_amsu_cloudy * ones(1,length(amsu_cloudy));

% Average blocks of nine to get 50-km AMSU
amsuavg_cloudy_ = zeros(15, 1333);
for i = 1:1333
amsuavg_cloudy_(:, i) = mean(amsu_cloudy_(:, (9*i-8):(9*i))')';
end
clear amsu_cloudy
bar = vertcat(ones(9,1) * (1:1333));
amsuavg_cloudy_ = amsuavg_cloudy_(:, bar);
clear bar

% MHS
mfile = ['mhs.cld2lr.difsurf.night.mw075.1013.' num2str(CHUNKS(chunk_loop)) '.Nc2.mat'];
fprintf(['Loading ' mwd mfile '\n']);
eval(['load ' mwd mfile]);
mhs_cloudy = m;
clear m
```

```
mhs_cloudy_ = mhs_cloudy - mean_mhs_cloudy * ones(1,length(mhs_cloudy));
clear mhs_cloudy                                                          120

% surface clear
foo = surface_filter * [amsuavg_cloudy_; mhs_cloudy_; airs_cloudy_];

amsuavg_cloudy_ = foo(1:15, 5:9:end);
mhs_cloudy_ = foo(16:20,:);
airs_cloudy_ = foo(21:end,:);

% Find warmest spot
% Use 50 15-micron channels with peaks between levels                     130
% 56-60 (inclusive) which corresponds to ~ 1-2km
fprintf('*** Finding warmest spot...\n');

diff_8_airs_cloudy = zeros(2371, 8*1333);
for i = 1:1333
foo1 = airs_cloudy_(AIRS_WARMSPOT_DETECT_CHANNELS,(9*i-8):9*i);
% add noise here
foo1 = foo1 + diag(noise_vector(AIRS_WARMSPOT_DETECT_CHANNELS)) * ...
randn(length(AIRS_WARMSPOT_DETECT_CHANNELS), 9);
foo1 = mean(foo1); % not optimal, but probably pretty close              140
ind1(i) = find(foo1 == max(foo1));  % warmest spot
ass = 1:9;
ass(ind1(i)) = [];
ind2(i) = 9*(i-1) + ind1(i);
  for j = 1:8
    diff_8_airs_cloudy(:, 8*(i-1) + j) = airs_cloudy_(:, 9*(i-1) + ass(j)) - airs_cloudy_(:, 9*(i-1) + ind1(i));
  end
end
mhs_8_cloudy_ = mhs_cloudy_;
mhs_8_cloudy_(:, ind2) = [];                                              150
warmest_airs_cloudy_ = airs_cloudy_(:, ind2);
clear airs_cloudy_
warmest_mhs_cloudy_ = mhs_cloudy_(:, ind2);
warmest_all_rad = [amsuavg_cloudy_; warmest_mhs_cloudy_; warmest_airs_cloudy_];
```

```
% Estimate clear radiance
load regress_coef_2 regress_coef_2
eval(['est_airs_clear_' num2str(CHUNKS(chunk_loop)) ' = regress_coef_2 * warmest_all_rad;']);

% Compute ouput PC's                                                    160
eval(['err_airs_clear' num2str(CHUNKS(chunk_loop)) ' = est_airs_clear_' num2str(CHUNKS(chunk_loop)) ...
' - airs_clear_(:, CHUNKS(chunk_loop):9:end);']);
load pcs_TEMP_ONLY_norm_3 evects
evects_pcs_3 = evects;
clear evects
%load std_err_airs_clear
eval(['pc_err_airs_clear' num2str(CHUNKS(chunk_loop)) ...
' = evects_pcs_3(:, 1:nn_param.OUT_KEEPERS)'' *
diag(1./std_err_airs_clear(nn_param.IR_OUTPUT_CHANNELS)) * err_airs_clear' ...
num2str(CHUNKS(chunk_loop)) '(nn_param.IR_OUTPUT_CHANNELS,:);']);     170


% Assemble Projected PC's of warmest spot
fprintf('*** Assembling %d Projected PC''s of warmest spot...\n', nn_param.WARM_KEEPERS);
load ppcs_1 evects ppc_regress_coef
evects_ppc = evects;
clear evects
pcs_warmest = evects_ppc(:, 1:nn_param.WARM_KEEPERS)' * ppc_regress_coef * warmest_all_rad;

% Assemble 8 IR deltas                                                  180
fprintf('*** Assembling %d PC''s of AIRS differences...\n', nn_param.DIFF_KEEPERS);
load pcs_2 evects
evects_pcs_2 = evects;
clear evects
pcs_diff_8_airs_cloudy = zeros(nn_param.DIFF_KEEPERS*8, 1333);
foo = evects_pcs_2(:, 1:nn_param.DIFF_KEEPERS)' * ...
diag(1./(noise_vector*sqrt(2))) * diff_8_airs_cloudy;
for i = 1:1333
  for j = 1:8
    pcs_diff_8_airs_cloudy((nn_param.DIFF_KEEPERS*(j-1)+1):(nn_param.DIFF_KEEPERS*j), i) = ... 190
foo(:, 8*(i-1) + j);
  end
```

```
end
clear foo diff_8_airs_cloudy

% Assemble other 8 15-km MHS pixels
fprintf('*** Assembling MHS pixels. . .\n');
pcs_mhs_8_cloudy = zeros(40, 1333);
for i = 1:1333
  for j = 1:8                                                                    200
    pcs_mhs_8_cloudy( (5*(j-1)+1):(5*j) , i) = mhs_8_cloudy_(:, 8*(i-1) + j);
  end
end

eval(['data' num2str(CHUNKS(chunk_loop)) ...
' = [pcs_warmest; pcs_diff_8_airs_cloudy; pcs_mhs_8_cloudy];']);
fprintf('--- FINISHED assembling data chunk %d (%d of %d).\n', ...
CHUNKS(chunk_loop), chunk_loop, NUM_CHUNKS);

end % looping over num_chunks                                                    210

data = [data1 data3 data7 data9];
pc_err_airs_clear = [pc_err_airs_clear1 pc_err_airs_clear3 pc_err_airs_clear7 pc_err_airs_clear9];

% Normalize to unit variance....
std_data = std(data')';
std_pc_err_airs_clear = std(pc_err_airs_clear')';

% calculate noise stats
cov_pcs_warmest_noise = (evects_ppc(:, 1:nn_param.WARM_KEEPERS)' * ppc_regress_coef * ...    220
diag(noise)) * (evects_ppc(:, 1:nn_param.WARM_KEEPERS)' * ppc_regress_coef * diag(noise))';
cov_pcs_warmest_noise_sqrt = sqrtm(cov_pcs_warmest_noise);
std_pcs_warmest_noise = sqrt(diag(cov_pcs_warmest_noise));
std_diff_noise = diag(eye(nn_param.DIFF_KEEPERS));
std_mhs_noise = mnvec';

cov_pc_err_airs_clear_noise = (evects_pcs_3(:, 1:nn_param.OUT_KEEPERS)' * ...
regress_coef_2(nn_param.IR_OUTPUT_CHANNELS,:) * diag(noise)) * (evects_pcs_3(:, 1:nn_param.OUT_KEEPERS)' *
regress_coef_2(nn_param.IR_OUTPUT_CHANNELS,:) * diag(noise))';
```

```
std_pc_err_airs_clear_noise = sqrt(diag(cov_pc_err_airs_clear_noise));
cov_pc_err_airs_clear_noise_sqrt = sqrtm(cov_pc_err_airs_clear_noise);

cov_err_airs_clear_noise = regress_coef_2(nn_param.IR_OUTPUT_CHANNELS,:) * diag(noise) * ...
(regress_coef_2(nn_param.IR_OUTPUT_CHANNELS,:) * diag(noise))';
% need to take real part because of numerical error
cov_err_airs_clear_noise_sqrt = real(sqrtm(cov_err_airs_clear_noise));


std_data_noise = [std_pcs_warmest_noise; std_diff_noise; std_diff_noise; std_diff_noise; std_diff_noise; ...
std_diff_noise; std_diff_noise; std_diff_noise; std_diff_noise; std_mhs_noise; std_mhs_noise; ...
std_mhs_noise; std_mhs_noise; std_mhs_noise; std_mhs_noise; std_mhs_noise; std_mhs_noise ];

eval(['save std_data_difsurf_surfclear' num2str(nn_param.SURFACE_FILTER_PCS) '_' ...
num2str(nn_param.WARM_KEEPERS) ' std_data std_data_noise']);
eval(['save std_pc_err_airs_clear_TEMP_ONLY_difsurf_surfclear' num2str(nn_param.SURFACE_FILTER_PCS) '_'
num2str(nn_param.WARM_KEEPERS) ' std_pc_err_airs_clear std_pc_err_airs_clear_noise']);


clear err_airs_clear* est_airs_clear_* pc_err_airs_clear1 pc_err_airs_clear2 pc_err_airs_clear3 ...
pc_err_airs_clear4 airs_clear_ regress_coef_2 warmest_airs_cloudy_ warmest_all_rad

fprintf('*** Loading/assembling test set. . .\n');
CHUNKS = [5];
NUM_CHUNKS = length(CHUNKS);

for chunk_loop = 1:NUM_CHUNKS
   ifile = ['airs.cld2lr.difsurf.night.ir04.1013.' num2str(CHUNKS(chunk_loop)) '.Nc2.mat'];
   fprintf(['Loading ' iwd ifile '\n']);
   eval(['load ' iwd ifile]);
   airs_cloudy = x;
   clear x
   airs_cloudy_ = airs_cloudy - mean_airs_cloudy * ones(1, length(airs_cloudy));

   clear airs_cloudy
%  airs_cloudy_ = airs_cloudy_ + diag(noise_vector) * randn(2371, 11997);
%  airs_cloudy_ = diag(noise_vector) * randn(2371, 11997);
```

```
afile = ['amsu.cld2lr.difsurf.night.mw075.1013.' num2str(CHUNKS(chunk_loop)) '.Nc2.mat'];
fprintf(['Loading ' awd afile '\n']);
eval(['load ' awd afile]);                                                    270
amsu_cloudy = a;
clear a
% Average blocks of nine to get 50-km AMSU
amsuavg_cloudy = zeros(15, 1333);

for i = 1:1333
   amsuavg_cloudy(:, i) = mean(amsu_cloudy(:, (9*i-8):(9*i))')';
end

amsuavg_cloudy_ = amsuavg_cloudy - mean_amsu_cloudy * ones(1,1333);           280
clear amsuavg_cloudy amsu_cloudy
mfile = ['mhs.cld2lr.difsurf.night.mw075.1013.' num2str(CHUNKS(chunk_loop)) '.Nc2.mat'];
fprintf(['Loading ' mwd mfile '\n']);
eval(['load ' mwd mfile]);
mhs_cloudy = m;
clear m
mhs_cloudy_ = mhs_cloudy - mean_mhs_cloudy * ones(1,length(mhs_cloudy));
clear mhs_cloudy

% surface clear                                                               290
bar = vertcat(ones(9,1) * (1:1333));
amsuavg_cloudy_ = amsuavg_cloudy_(:, bar);
clear bar
foo = surface_filter * [amsuavg_cloudy_; mhs_cloudy_; airs_cloudy_];

amsuavg_cloudy_ = foo(1:15, 5:9:end);
mhs_cloudy_ = foo(16:20,:);
airs_cloudy_ = foo(21:end,:);
clear foo
                                                                              300
% Find warmest spot
fprintf('*** Finding warmest spot. . .\n');
diff_8_airs_cloudy = zeros(2371, 8*1333);
```

```
for i = 1:1333
   foo1 = airs_cloudy_(AIRS_WARMSPOT_DETECT_CHANNELS,(9*i-8):9*i);
   foo1 = foo1 + diag(noise_vector(AIRS_WARMSPOT_DETECT_CHANNELS)) * ...
   randn(length(AIRS_WARMSPOT_DETECT_CHANNELS), 9);
   foo1 = mean(foo1);
   ind1(i) = find(foo1 == max(foo1));   % warmest spot                    310
   ass = 1:9;
   ass(ind1(i)) = [];
   ind2(i) = 9*(i-1) + ind1(i);

   for j = 1:8
      diff_8_airs_cloudy(:, 8*(i-1) + j) = airs_cloudy_(:, 9*(i-1) + ass(j)) - ...
      airs_cloudy_(:, 9*(i-1) + ind1(i));
   end

end                                                                        320

ind3 = ones(9,1) * [1:1333];
ind3 = ind3(:);
all_rad_ = [amsuavg_cloudy_(:, ind3); mhs_cloudy_; airs_cloudy_];
mhs_8_cloudy_ = mhs_cloudy_;
mhs_8_cloudy_(:, ind2) = [];
warmest_airs_cloudy_ = airs_cloudy_(:, ind2);
clear airs_cloudy_
warmest_mhs_cloudy_ = mhs_cloudy_(:, ind2);
warmest_all_rad = [amsuavg_cloudy_; warmest_mhs_cloudy_; warmest_airs_cloudy_];   330
% Estimate clear radiance
load regress_coef_2 regress_coef_2
eval(['est_airs_clear_' num2str(CHUNKS(chunk_loop)) ' = regress_coef_2 * warmest_all_rad;']);
eval(['est_airs_clear_all_' num2str(CHUNKS(chunk_loop)) ' = regress_coef_2 * all_rad_;']);
clear all_rad_
% average blocks of nine - old method
%eval(['for i = 1:(size(est_airs_clear_all_' num2str(CHUNKS(chunk_loop)) ' , 2)/9)'])

for i = 1:(size(est_airs_clear_all_5 , 2)/9)
   eval(['airs_clear_est_old(:,i) = mean(est_airs_clear_all_' ...                340
```

```
    num2str(CHUNKS(chunk_loop)) '(:, (9*i-8):(9*i))'')'';']);
  end

  eval(['clear est_airs_clear_all_' num2str(CHUNKS(chunk_loop))])
  % Compute ouput PC's
load /usr/dicke1/bill/AIRS_cloud_clearing/airs_radiances/airs.clear.night.femis.1013.Ncl.mat
airs_clear = x;
clear x
airs_clear_ = airs_clear - mean_airs_clear * ones(1, length(airs_clear));
clear airs_clear                                                              350

  eval(['err_airs_clear' num2str(CHUNKS(chunk_loop)) ' = est_airs_clear_' ...
  num2str(CHUNKS(chunk_loop)) ' - airs_clear_(:, CHUNKS(chunk_loop):9:end);']);
  eval(['pc_err_airs_clear' num2str(CHUNKS(chunk_loop)) ...
  ' = evects_pcs_3(:, 1:nn_param.OUT_KEEPERS)'' * ...
diag(1./std_err_airs_clear(nn_param.IR_OUTPUT_CHANNELS)) * err_airs_clear' ...
  num2str(CHUNKS(chunk_loop)) '(nn_param.IR_OUTPUT_CHANNELS,:);']);
  % Assemble Projected PC's of warmest spot
  fprintf('*** Assembling %d Projected PC''s of warmest spot. . .\n', nn_param.WARM_KEEPERS); 360
  load ppcs_1 evects ppc_regress_coef
pcs_warmest = evects(:, 1:nn_param.WARM_KEEPERS)' * ppc_regress_coef * warmest_all_rad;
  % Assemble 8 IR deltas
  fprintf('*** Assembling %d PC''s of AIRS differences. . .\n', nn_param.DIFF_KEEPERS);
  pcs_diff_8_airs_cloudy = zeros(nn_param.DIFF_KEEPERS*8, 1333);
  foo = evects_pcs_2(:, 1:nn_param.DIFF_KEEPERS)' * ...
  diag(1./(noise_vector*sqrt(2))) * diff_8_airs_cloudy;

  for i = 1:1333
    for j = 1:8                                                               370
      pcs_diff_8_airs_cloudy( (nn_param.DIFF_KEEPERS*(j-1)+1):...
      (nn_param.DIFF_KEEPERS*j) , i) = foo(:, 8*(i-1) + j);
    end
  end

  clear foo diff_8_airs_cloudy
```

```
  % Assemble other 8 15-km MHS pixels
  fprintf('*** Assembling MHS pixels. . .\n');
  pcs_mhs_8_cloudy = zeros(40, 1333);                                          380

  for i = 1:1333
    for j = 1:8
      pcs_mhs_8_cloudy( (5*(j-1)+1):(5*j) , i) = mhs_8_cloudy_(:, 8*(i-1) + j);
    end
  end

  eval(['data' num2str(CHUNKS(chunk_loop)) ...
  ' = [pcs_warmest; pcs_diff_8_airs_cloudy; pcs_mhs_8_cloudy];']);
  fprintf('--- FINISHED assembling data chunk %d (%d of %d).\n'...           390
  , CHUNKS(chunk_loop), chunk_loop, NUM_CHUNKS);

end % looping over num_chunks

% Normalize to unit variance....
TV_P_clean = data5;
TV_T_clean = pc_err_airs_clear5;

clear surface_filter data5 pc_err_airs_clear5
                                                                               400
fprintf('--- Preparing neural network. . .\n');
nnet=newff(minmax(data')',[nn_param.NUM_NODES1 nn_param.NUM_NODES2 nn_param.OUT_KEEPERS],...
{'tansig','tansig','purelin'},nn_param.TRAIN_ALG);
% set pertinent intialization parameters and reinitialize
nnet.layers{2}.initFcn = 'initwb';
nnet.layers{1}.initFcn = 'initwb';
nnet.inputWeights{1}.initFcn='rands';
nnet.layerWeights{2,1}.initFcn = 'rands';
nnet.biases{2}.initFcn='rands';
nnet.biases{1}.initFcn='rands';                                                410
nnet=init(nnet);
nnet.IW{1}=0.5.*nnet.IW{1};

% set pertinent training parameters
```

```
nnet.trainParam.max_fail=30;
nnet.trainParam.show = ceil(nn_param.NOISE_ROTATE/2);
nnet.trainParam.epochs = nn_param.NOISE_ROTATE;
nnet.trainParam.goal = 0.1;
nnet.trainParam.searchFcn = 'srchgol';
nnet.trainParam.minstep = 0;
nnet.trainParam.time = inf;
nnet.performFcn='mse';

if 1
fprintf('\n\n### WARNING - Loading previously trained net. . . ###\n\n');
load traingdx_wjb_out_warm30_v9_run1_2.mat
nn_param.WARM_KEEPERS = 30;
nn_param.DIFF_KEEPERS = 5;
nn_param.OUT_KEEPERS = 30;
nn_param.NUM_NODES1 = 60;
nn_param.NUM_NODES2 = 45;
nn_param.IR_OUTPUT_CHANNELS = [101:400 1901:2000 floor(linspace(2001, 2089, 75)) 2090:2114];
nn_param.SURFACE_FILTER_PCS = 0;
nn_param.NOISE_ROTATE = 5;
nn_param.SAVE_OUTPUT = 50;
nn_param.TRAIN_LENGTH = 5000;
nn_param.TRAIN_ALG = 'trainscg_wjb';
nnet.trainFcn=nn_param.TRAIN_ALG;
nnet.trainParam.max_fail=30;
nnet.trainParam.show = ceil(nn_param.NOISE_ROTATE/2);
nnet.trainParam.epochs = nn_param.NOISE_ROTATE;
nnet.trainParam.goal = 0.1;
nnet.trainParam.searchFcn = 'srchgol';
nnet.trainParam.minstep = 0;
nnet.trainParam.time = inf;
nnet.performFcn='mse';
START = length(training_error) + 1;
else
START = 1;
end
```

```
fprintf('*** Starting training . . .\n');
% new noise

fig = figure;

addpath /usr/dickel/bill/AIRS_cloud_clearing/matlab
clear err_airs_clear5
clear warmest_airs_cloudy_
airs_clear_ = airs_clear_(nn_param.IR_OUTPUT_CHANNELS,5:9:end);
est_airs_clear_5 = est_airs_clear_5(nn_param.IR_OUTPUT_CHANNELS,:);
clear airs_clear_est_old

for i = START:Inf
    % add noise to training set
    foo3 = randperm(length(data));
    pcs_warmest_noise = cov_pcs_warmest_noise_sqrt * ...
    randn(nn_param.WARM_KEEPERS, nn_param.TRAIN_LENGTH);
    diff_noise = randn(nn_param.DIFF_KEEPERS*8, nn_param.TRAIN_LENGTH);
    mhs_noise = diag([mnvec mnvec mnvec mnvec mnvec mnvec mnvec]) * ...
    randn(40, nn_param.TRAIN_LENGTH);
    additive_noise2 = cov_pc_err_airs_clear_noise_sqrt * ...
    randn(nn_param.OUT_KEEPERS, nn_param.TRAIN_LENGTH);
    additive_noise1 = [pcs_warmest_noise; diff_noise; mhs_noise];
    foo1 = diag(1./sqrt(std_data.^2 + std_data_noise.^2)) * ...
    (data(:, foo3(1:nn_param.TRAIN_LENGTH)) + additive_noise1);
    foo2 = diag(1./sqrt(std_pc_err_airs_clear.^2 + std_pc_err_airs_clear_noise.^2)) * ...
    (pc_err_airs_clear(:, foo3(1:nn_param.TRAIN_LENGTH)) + additive_noise2);

    % add noise to validation set
    pcs_warmest_noise = cov_pcs_warmest_noise_sqrt * ...
    randn(nn_param.WARM_KEEPERS, length(TV_P_clean));
    diff_noise = randn(nn_param.DIFF_KEEPERS*8, length(TV_P_clean));
    mhs_noise = diag([mnvec mnvec mnvec mnvec mnvec mnvec mnvec]) * ...
    randn(40, length(TV_P_clean));
    additive_noise2 = cov_pc_err_airs_clear_noise_sqrt * ...
    randn(nn_param.OUT_KEEPERS, length(TV_P_clean));
    additive_noise1 = [pcs_warmest_noise; diff_noise; mhs_noise];
```

```
    TV.P = diag(1./sqrt(std_data.^2 + std_data_noise.^2)) * ...
    (TV_P_clean + additive_noise1);                                          490
    TV.T = diag(1./sqrt(std_pc_err_airs_clear.^2 + std_pc_err_airs_clear_noise.^2)) * ...
    (TV_T_clean + additive_noise2);

[nnet,tr]=train(nnet, foo1, foo2, [], [], [], TV);

fprintf('*** Calculating cloud clearing error. . .');
pc_err_air_clear_est = diag(sqrt(std_pc_err_airs_clear.^2 + ...
std_pc_err_airs_clear_noise.^2)) * sim(nnet, TV.P);
err_airs_clear_est = diag(std_err_airs_clear(nn_param.IR_OUTPUT_CHANNELS)) * ...
evects_pcs_3(:, 1:nn_param.OUT_KEEPERS) * pc_err_air_clear_est;          500
est_airs_clear_5 = regress_coef_2(nn_param.IR_OUTPUT_CHANNELS,:) * warmest_all_rad + ...
cov_err_airs_clear_noise_sqrt * randn(length(nn_param.IR_OUTPUT_CHANNELS),length(TV_P_clean));
airs_clear_est = est_airs_clear_5 - err_airs_clear_est;
err = airs_clear_est' - airs_clear_';
err9 = sqrt(mean(err.^2));
for j = 1:length(nn_param.IR_OUTPUT_CHANNELS)
   err9T(j) = rad2brit(freq(nn_param.IR_OUTPUT_CHANNELS(j)), ...
   brit2rad(freq(nn_param.IR_OUTPUT_CHANNELS(j)), 290) + err9(j)) - 290;
end
fprintf('Error = %g K\n\n', mean(err9T));                                510
cloud_clearing_validation_error(i) = mean(err9T);

training_error(i) = tr.perf(end);
validation_error(i) = tr.tperf(end);
figure(fig)
clf
plot(training_error)
grid on; hold on
plot(validation_error,'r')
plot(cloud_clearing_validation_error,'g')                                520
drawnow
if (rem(i, nn_param.SAVE_OUTPUT * 2) == 0)
  eval(['save ' nn_param.TRAIN_ALG '_out_warm30_v9_run2_1.mat nnet tr training_error . . .
  validation_error cloud_clearing_validation_error err9T nn_param']);
elseif (rem(i, nn_param.SAVE_OUTPUT) == 0)
```

```
  eval(['save ' nn_param.TRAIN_ALG '_out_warm30_v9_run2_2.mat nnet tr training_error ...
  validation_error cloud_clearing_validation_error err9T nn_param']);
end

                                                                                    530

end
```

Appendix F

List of Symbols and Acronyms

Constants

ϵ_o	=	permittivity of free space = 8.854×10^{-12} F/m
μ_o	=	permeability of free space = $4\pi \times 10^{-7}$ H/m
π	=	3.1415926535 ...
e	=	2.7182818285 ...
c	=	speed of light in a vacuum = 2.99793×10^8 m/s
g	=	mean acceleration of gravity at Earth's surface = 9.80665m/s^2
h	=	Planck's constant = 6.6252×10^{-34} J·s
k	=	Boltzmann's constant = 1.38046×10^{-23} J/K
R	=	gas constant = 8.3143 J/(K·mole)

Greek Symbols

ϵ	permittivity
ε	surface emissivity
η	antenna beam coupling coefficient
η_Z^A	contribution from zenith port while viewing ambient load
η_Z^H	contribution from zenith port while viewing heated load
η_N^A	contribution from nadir port while viewing ambient load
η_N^H	contribution from nadir port while viewing heated load
κ	absorption coefficient
Γ	adiabatic lapse rate, reflection coefficient
ρ	mass density
ρ_c	cloud-top reflectivity (IR only)
ρ_s	surface reflectivity
σ_D^2	error variance on downwelling $\mathbf{B}_{obs} - \mathbf{B}_{sim}$ residual
σ_U^2	error variance on upwelling $\mathbf{B}_{obs} - \mathbf{B}_{sim}$ residual
τ	optical depth (thickness)
τ^*	opacity

Roman Symbols

b	radiometer baseline voltage
$\mathbf{B}$	MW brightness temperature vector
$\mathbf{B}^{clr}$	clear-air MW brightness temperature vector
$\mathbf{B}^{cld}$	cloudy MW brightness temperature vector
$\mathbf{B}_{sim}$	simulated MW brightness temperature vector
$\mathbf{B}_{obs}$	observed MW brightness temperature vector
$\mathbf{B}_{obs}^{clr}$	observed, clear-air MW brightness temperature vector
$\mathbf{B}_{obs}^{cld}$	observed, cloudy MW brightness temperature vector
$\mathbf{B}_{sim}^{clr}$	simulated, clear-air MW brightness temperature vector
$\mathbf{B}_{sim}^{cld}$	simulated, cloudy MW brightness temperature vector
C	radiometer output voltage (counts)
c_p, c_v	specific heat at constant pressure, volume
g	radiometer gain
H	scale height
$\mathbf{R}$	IR radiance vector
$\mathbf{R}^{clr}$	clear-air IR radiance vector
$\mathbf{R}^{cld}$	cloudy IR radiance vector
$\mathbf{R}_{sim}$	simulated IR radiance vector
$\mathbf{R}_{obs}$	observed IR radiance vector
$\mathbf{R}_{obs}^{clr}$	observed, clear-air IR radiance vector
$\mathbf{R}_{obs}^{cld}$	observed, cloudy IR radiance vector
$\mathbf{R}_{sim}^{clr}$	simulated, clear-air IR radiance vector
$\mathbf{R}_{sim}^{cld}$	simulated, cloudy IR radiance vector
T	temperature
V	volume
$\mathbf{w}$	weights for heated-load RTDs which minimize error due to thermal gradient

Calligraphic Symbols

$\mathcal{T}$	Transmittance, transmission coefficient

Acronyms

AIRS	Atmospheric InfraRed Sounder
AMSU	Advanced Microwave Sounding Unit
CAMEX	Convection And Moisture EXperiment
ER-2	NASA U-2 high-altitude environmental research aircraft
IF	Intermediate Frequency
ION	Iterated Order-Noise
IR	InfraRed
LLSE	Linear Least-Squares Estimator
LO	Local Oscillator
MHS	Microwave Humidity Sounder
MAMS	Multispectral Atmospheric Mapping Sensor
MW	Microwave
NAPC	Noise-Adjusted Principal Components
NAST-I	NPOESS Aircraft Sounder Testbed–Infrared
NAST-M	NPOESS Aircraft Sounder Testbed–Microwave
NN	Neural Network
NPOESS	National Polar-orbiting Operational Environment Satellite System
PC	Principal Components
PPC	Projected Principal Components
RAOB	RAdiosonde OBservation
RMS	Root-Mean-Square
RTD	Resistive Temperature Device
UTC	Univeral Time Code
WINTEX	WINter EXperiment

References

[1] J. Lee and D. H. Staelin. Iterative signal-order and noise estimation for multivariate data. *IEE Electronics Letters*, 37(2):134–135, 2001.

[2] D. H. Staelin. Passive remote sensing at microwave wavelengths. *Proceedings of the IEEE*, 57(4):427–439, April 1969.

[3] W. J. Blackwell. Atmospheric temperature profile estimation from infrared and microwave spectral radiance observations using principal components analysis. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, September 1995.

[4] D. P. Wylie, W. P. Menzel, H. M. Woolf, and K. I. Stabala. Four years of global cirrus cloud statistics using HIRS. *Journal of Climate*, 7(12):1972–1986, 1994.

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

[6] G. L. Stephens. *Remote Sensing of the Lower Atmosphere.* Oxford University Press, New York, New York, 1994.

[7] J. T. Houghton. *The Physics of Atmospheres.* Cambridge University Press, 1986.

[8] K. N. Liou. *An Introduction to Atmospheric Radiation.* Academic Press, Orlando, Florida, 1980.

[9] D. H. Staelin, A. W. Morgenthaler, and J. A. Kong. *Electromagnetic Waves.* Prentice Hall, 1994.

[10] J. M. Wallace and P. V. Hobbs. *Atmospheric Science: An Introductory Survey.* Academic Press, New York, New York, 1977.

[11] B. J. Mason. *The Physics of Clouds.* Oxford University Press, Oxford, 1971.

[12] J. S. Marshall and W. Palmer. The distribution of raindrops with size. *Journal of the Atmosphere*, 5:165–166, 1948.

[13] C. Elachi. *Introduction to the Physics and Techniques of Remote Sensing.* Wiley, New York, New York, 1987.

[14] D. Deirmendjian. *Electromagnetic Scattering on Spherical Polydispersions.* American Elsevier Publishing Co., Inc., New York, New York, 1969.

[15] A. J. Gasiewski. Microwave radiative transfer in hydrometeors. *Atmospheric Remote Sensing by Microwave Radiometry*, M. A. Janssen, Ed., Chapter 3, July 1993.

[16] H. S. Chen. *Space Remote Sensing Systems: An Introduction.* Academic Press, 1985.

[17] J. D. Kraus. *Radio Astronomy.* Cygnus-Quaser Books, second edition, 1986.

[18] W. L. Smith. Satellite techniques for observing the temperature structure of the atmosphere. *Bulletin of the American Meteorological Society*, 53(11):1074–1082, November 1972.

[19] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.

[20] C. D. Rodgers. Information content and optimisation of high spectral resolution measurements. *SPIE, Optical Spectroscopic Techniques and Instrumentation for Atmospheric and Space Research II*, 2830:136–147, 1996.

[21] C. D. Rodgers. *Inverse Methods for Atmospheric Sounding.* World Scientific, 2000.

[22] M. A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE*, 37(2):233–243, 1991.

[23] S. Tan and M. Mavrovouniotis. Reducing data dimensionality through optimizing neural-network inputs. *AIChE*, 41(6):1471–1480, 1995.

[24] A. J. Slone. Improved remote sensing data analysis using neural networks. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, September 1995.

[25] A. A. Green, B. Berman, P. Switzer, and M. D. Craig. A transformation for ordering multispectral data in terms of image quality with implications for noise removal. *IEEE Transactions on Geoscience and Remote Sensing*, 26:65–74, 1988.

[26] J. B. Lee, A. S. Woodyatt, and M. Berman. Enhancement of high spectral resolution remote-sensing data by a noise-adjusted principal components transform. *IEEE Transactions on Geoscience and Remote Sensing*, 28:295–304, 1990.

[27] J. H. Lee. *Blind Noise Estimation and Compensation for Improved Characterization of Multivariate Processes.* PhD thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2000.

[28] G. E. Backus and J. F. Gilbert. Uniqueness in the inversion of inaccurate gross earth data. *Phil. Trans. Roy. Soc. London*, 266:123–1922, 1970.

[29] K. M. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 4(5):359–366, 1989.

[30] M. Stinchcombe and H. White. Universal approximation using feedforward networks with non-sigmoid hidden layer activation functions. *Proceedings of the International Joint Conference on Neural Networks*, 1:613–617, 1989.

[31] A. Gallant and H. White. There exists a neural network that does not make avoidable mistakes. *Proceedings of the International Joint Conference on Neural Networks*, 1:657–664, 1988.

[32] D. E. Rumelhart, G. Hinton, and R. Williams. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol. 1: Foundations.* D. E. Rumelhart and J. L. McClelland (Eds.). MIT Press, Cambridge, MA, 1986.

[33] D. Cousins and W. L. Smith. NPOESS airborne sounder testbed - interferometer. *SPIE Proceedings*, 3756:323–331, 1997.

[34] W. J. Blackwell, J. W. Barrett, P. W. Rosenkranz, M. J. Schwartz, and D. H. Staelin. NPOESS Aircraft Sounder Testbed-Microwave (NAST-M): Instrument description and initial flight results. *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, July 2000.

[35] W. J. Blackwell, F. W. Chen, R. V. Leslie, P. W. Rosenkranz, M. J. Schwartz, and D. H. Staelin. NPOESS Aircraft Sounder Testbed-Microwave (NAST-M): Results from CAMEX-3 and WINTEX. *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, July 2000.

[36] W. J. Blackwell, J. W. Barrett, F. W. Chen, R. V. Leslie, P. W. Rosenkranz, M. J. Schwartz, and D. H. Staelin. NPOESS aircraft sounder testbed-microwave (NAST-M): Instrument description and initial flight results. *IEEE Transactions on Geoscience and Remote Sensing*, 39(11):2444–2453, November 2001.

[37] R. V. Leslie. Temperature profile retrievals with the NAST-M passive microwave spectrometer. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, June 2000.

[38] M. Barabanov and V. Yodaiken. Real-time linux. *Linux Journal*, March 1996.

[39] G. E. Peckham. An optimum calibration procedure for radiometers. *International Journal of Remote Sensing*, 10(1):227–236, 1989.

[40] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing.* Prentice Hall, 1989.

[41] M. S. Hersman and G. A. Poe. Sensitivity of the total power radiometer with periodic absolute calibration. *IEEE Transactions on Microwave Theory and Techniques*, 29(1):32–40, 1981.

[42] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization.* Academic Press, London, 1981.

[43] P. W. Rosenkranz. Retrieval of temperature and moisture profiles from AMSU-A and AMSU-B measurements. *IEEE Transactions on Geoscience and Remote Sensing*, 39(11):2429–2435, 2001.

[44] P. W. Rosenkranz. Absorption of microwaves by atmospheric gases. *Atmospheric Remote Sensing by Microwave Radiometry*, M. A. Janssen, Ed., Chapter 2, July 1993.

[45] K. Lamkaouchi, E. Berge, and W. Ellison. New permittivity data for sea water (30-100 GHz). *ESA report 11197/94/NL/CD*, 1997.

[46] H. F. Teh. Development and use of an antenna gain pattern test system for the NAST-M microwave radiometer at ~54 GHz. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, June 2001.

[47] T. Mo. Prelaunch calibration of the advanced microwave sounding unit-A for NOAA-K. *IEEE Transactions on Geoscience and Remote Sensing*, 44(8):1460–1469, 1996.

[48] M. J. Schwartz. *Observation and modeling of atmospheric oxygen millimeter-wave transmittance.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1998.

[49] D. J. Schwab, G. A. Leshkevich, and G. C. Muhr. Satellite measurements of surface water temperature in the Great Lakes - Great Lakes CoastWatch. *Journal of Great Lakes Research*, 18(2):247–258, 1992.

[50] D. R. Hearn. *Fourier Transform Interferometry.* MIT Lincoln Laboratory Technical Report 1053, October 1999.

[51] M. W. Kelly, M. J. Gazarik, and R. Marino. Noise performance of the NPOESS airborne sounder testbed interferometer (NAST-I) sounder: Flight and model results. *Proceedings of the SPIE Earth Observing Systems IV, Denver*, 3750, July 1999.

[52] D. Cousins and M. J. Gazarik. *NAST Interferometer Design and Characterization: Final Report.* Project Report NOAA-26, MIT Lincoln Laboratory, July 1999.

[53] H. H. Aumann and C. Miller. Atmospheric infrared sounder (AIRS) on the Earth Observing System. *SPIE Proceedings*, 2583:332–343, 1995.

[54] R. W. Saunders. Note on the advanced microwave sounding unit. *Bulletin of the American Meteorological Society*, 74:2211–2212, 1993.

[55] R. W. Saunders, T. J. Hewison, S. J. Stringer, and N. C. Atkinson. The radiometric characterization of AMSU-B. *IEEE Transactions on Microwave Theory and Techniques*, 43(4):760–771, 1995.

[56] J. Susskind, C. Barnet, and J. Blaisdell. Determination of atmospheric and surface parameters from simulated AIRS/AMSU/HSB sounding data: retrieval and cloud clearing methodology. *Advances in Space Research*, 21(3):369–384, 1998.

[57] M. D. Goldberg and L. M. McMillin. Methodology for deriving deep-layer temperatures from combined satellite infrared and microwave observations. *Journal of Climate*, 12:5–20, 1999.

[58] J. Escobar. *Base de Données pour la Restitution de Paramètres Atmosphériques à L'échelle Globale-Etude sur L'inversion per Réseaux de Neurones des Données des Sondeurs Verticaux Atmosphériques Satellitaires Présents et à venir.* PhD thesis, Ecole Polytechnique, France, 1993.

[59] H. E. Motteler, et al. Comparison of neural networks and regression based methods for temperature retrievals. *Applied Optics*, 34(24):5390–5397, August 1995.

[60] S. Y. Lee. Personal Communication, 1998.

[61] H. H. Aumann, et al. *AIRS Algorithm Theoretical Basis Document, Level 1B, Part 1: Infrared Spectrometer.* JPL D-17003, 1999.

[62] L. L. Strow, H. E. Motteler, R. G. Benson, S. E. Hannon, and S. De Souza-Machado. Fast computation of monochromatic infrared atmospheric transmittances using compressed look-up tables. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 59:481–493, 1998.

[63] H. H. Aumann, et al. *AIRS Algorithm Theoretical Basis Document, Level 1B, Part 2: Microwave Instruments.* JPL D-17005, 1996.

[64] P. W. Rosenkranz. A rapid atmospheric transmittance algorithm for microwave sounding channels. *IEEE Transactions on Geoscience and Remote Sensing*, 33(5):1135–1140, September 1995.

[65] P. W. Rosenkranz. Improved rapid transmittance algorithm for microwave sounding channels. *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, pages 728–730, June 1998.

[66] W. L. Smith. An improved method for calculating tropospheric temperature and moisture fro satellite radiometer observations. *Monthly Weather Review*, 96:387–396, 1968.

[67] M. T. Chahine. Remote sounding in cloudy atmospheres II: multiple cloud formations. *J. Atmos. Sci.*, 34:744–757, 1977.

[68] A. J. Gasiewski, J. W. Barrett, P. G. Bonanni, and D. H. Staelin. Aircraft-based radiometric imaging of tropospheric temperature and precipitation using the 118.75-GHz oxygen resonance. *Journal of Applied Meteorology*, 29(7):620–632, July 1990.

[69] M. S. Spina, M. J. Schwartz, D. H. Staelin, and A. J. Gasiewski. Application of multilayer feedforward neural networks to precipitation cell-top altitude estimation. *IEEE Transactions on Geoscience and Remote Sensing*, 36(1):154–162, January 1998.

[70] M. J. Schwartz, J. W. Barrett, P. W. Fieguth, P. W. Rosenkranz, M. S. Spina, and D. H. Staelin. Observations of thermal and precipitation structure in a tropical cyclone by means of passive microwave imagery near 118 GHz. *Journal of Applied Meteorology*, 35(5):671–678, May 1996.

[71] A. J. Gasiewski and D. H. Staelin. Statistical precipitation cell parameter estimation using passive 118-GHz O_2 observations. *Journal of Geophysical Research*, 94:18367–18378, 1989.

[72] D. H. Staelin and F. W. Chen. Precipitation observations near 54 and 183 GHz using the NOAA-15 satellite. *IEEE Transactions on Geoscience and Remote Sensing*, 38(5):2322–2332, 2000.

[73] F. W. Chen and D. H. Staelin. Validation of AIRS/AMSU/HSB precipitation estimates. *IEEE Transactions on Geoscience and Remote Sensing*, (submitted for publication).

[74] A. Chedin, N. A. Scott, C. Wahiche, and P. Moulinier. The improved initialisation inversion method: A high-resolution physical method for temperature retrievals from the TIROS-N series. *Journal of Climate and Applied Meteorology*, 24:128–143, 1985.

[75] N. C. Grody, C. M. Hayden, W. C. Shen, P. W. Rosenkranz, and D. H. Staelin. Typhoon June winds estimated from scanning microwave spectrometer measurements at 55.45 GHz. *Journal of Geophysical Research*, 24:3689–3695, 1979.

[76] C. S. Velden. Observational analyses of north atlantic tropical cyclones from NOAA polar-orbiting satellite data. *Journal of the Applied Meteorology*, 28:59–70, 1989.

[77] S. Q. Kidder. Satellite analysis of tropical cyclones using the advanced microwave sounding unit (amsu). *Bulletin of the American Meteorological Society*, 101(6):1241–1259, 2000.

[78] F. W. Chen. Personal communication. 2001.

[79] P. W. Rosenkranz. Radiative transfer solution using initial values in a scattering and absorbing atmosphere with surface reflection. *IEEE Transactions on Geoscience and Remote Sensing*, (submitted for publication).

[80] A. J. Gasiewski and D. H. Staelin. Numerical modeling of passive microwave O_2 observations over precipitation. *Radio Science*, 25(3):217–235, 1990.

[81] G. J. Ledlovec, W. P. Menzel, R. J. Atkinson, and G. S. Wilson. *The Multispectral Atmospheric Mapping Sensor (MAMS): Instrument Description, Calibration, and Data Quality.* NASA TM-86565, 1986.

[82] W. J. Blackwell and D. H. Staelin. Cloud flagging and clearing using high-resolution infrared and microwave sounding data. *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, June 2002.

[83] A. R. Kerr, N. J. Bailey, and N. Horner. A study of materials for a broadband millimeter-wave quasi-optical vacuum window. *NRAO Electronics Division internal report no. 292*, 1992.

[84] A. Stogryn. Electromagnetic scattering from rough, finitely conducting surfaces. *Radio Science*, 415(2), 1967.

[85] E. A. Vinogradov, E. M. Dianov, and N. A. Irasova. Dieletric characteristics of liquid nitrogen at λ=2.3mm. *Soviet physics - technical physics*, 11:983, 1967.