12-2001

Temporal column abundances of atmospheric nitrous oxide at the University of Tennessee, Knoxville

John Stewart Hager

To the Graduate Council:

I am submitting herewith a dissertation written by John Stewart Hager entitled "Temporal column abundances of atmospheric nitrous oxide at the University of Tennessee, Knoxville." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Physics.

William E. Blass, Major Professor

We have read this dissertation and recommend its acceptance:

Tom Handler, Marianne Breinig, Robert Compton

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a dissertation written by John Stewart Hager entitled "Temporal Column Abundances of Atmospheric Nitrous Oxide at the University of Tennessee, Knoxville." I have examined the final paper copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Physics.

William E Blass

William E. Blass, Major Professor

We have read this dissertation
and recommend its acceptance:

Thomas Handler

Marianne Breinig

Robert N Compton

Acceptance for the Council:

Vice Provost and Dean of Graduate
Studies

TEMPORAL COLUMN ABUNDANCES
OF
ATMOSPHERIC NITROUS OXIDE
AT THE
UNIVERSITY OF TENNESSEE, KNOXVILLE

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

John Stewart Hager

December 2001

Dedication

This dissertation is dedicated to my family, my brave sons John Paul and Alex, my gorgeous daughters, Sydney and Lauren and my beautiful wife Yolla for teaching me that without courage all other human values are lost, and to my parents, John Hager and Marjorie Hager for their love and patients.

"He who joyfully marches to music in rank and file has already earned my contempt. He has been given a large brain by mistake, since for him the spinal cord would fully suffice." – Albert Einstein

Acknowledgements

I wish to thank all those who helped me on completing my dissertation. I thank Dr. Blass for his aid and guidance. I want to give a special thanks to Forrest Hoffman for his immense help and humor.

I would also like the thank Bob Compton, Marianne Breinig and Tom Handler for serving on my doctoral committee, and for their support and guidance during my graduate education.

I would further like to express my appreciation to Joe Shaw of Environmental Technology Laboratory at NOAA for his help and encouragement during my time in Boulder, Colorado.

I also what to thank Larry Jennings and Allen Ewing for help with the refitting of the controls of the 5-meter Littrow spectrometer and the rest of the Complex System Laboratory crew, Steve Mahan, Larry Senesac, Alberto Rodriguez and technician Gerald McElyea

Abstract

This dissertation reports the detection of real time concentration levels of nitrous oxide in the earth's atmosphere at the University of Tennessee Knoxville, Tennessee and describes the integration of a suntracker with the 5-meter Littrow spectrometric system at the University of Tennessee Complex Systems Laboratory.

Atmospheric nitrous oxide (N_2O) is an important trace gas in the earth's atmosphere. Not only does it have implications to stratospheric ozone depletion, it is an important greenhouse gas. Since the main source of N_2O result from agricultural activities, this study is motivated in part by of the location of the University of Tennessee, which is in a large agricultural area. Tropospheric abundances of N_2O are mostly constant worldwide with only slight local variations due to N_2O sources. This study will quantify any local variations in column abundances of nitrous oxide using ground based solar infrared spectroscopy.

Table of Contents

List of Figures

LIST OF TABLES

Chapter One

Introduction

The importance of nitrogen cycles in the earth's atmosphere has become increasing apparent in recent years. In the most recent issue of Science an article reports on the need to drastically reduce nitrogen pollution (Kaiser 2001). One of the constituents of the nitrogen cycle is N_2O. N_2O is a potent green house gas and is 300 times more powerful than CO_2, which is a gas thought to be the major contributor to the greenhouse effect, on a per molecule basis (Lal 2000). The budget of nitrous oxide (N_2O) in the atmosphere has been under debate since the 1960's. The global estimates of source and sinks of N_2O have undergone continuous revisions and recently have been subject to much debate (Lal 2000). Natural sources of atmospheric N_2O result from biological processes occurring in soil and water. The recent increases of global N_2O are thought to be from anthropogenic sources such as biomass burning, agricultural activities and industrial sources. During the 20th century, a rapid expansion of agricultural land coupled with the use of nitrogen fertilizers, is probably responsible for 80% of the increase in atmospheric N_2O, from ~ 275 ppbv (parts per billion per volume) in 1900 to ~ 294 ppbv in 1970 to projected concentrations of ~317 in 2000 and ~345 in 2020. With the increasing amount of fertilizer application needed to feed an expected additional 1.5

billion people in the next 20 years, an accelerated rate of N_2O accumulation in the atmosphere is calculated for the coming decades (Mosier 2000).

The local interest in agricultural sources is one of the reasons that N_2O was chosen for this study. Sun source atmospheric spectra are generally recorded in dry climates, with many clear days, but without much agricultural activity. It is our thought that because of the location of the University of Tennessee in a large agricultural area one would see a seasonal variation in the column abundances of N_2O during the use of fertilizers in the area.

The only sinks of atmospheric N_2O are natural ones. In the stratosphere, 90% of the N_2O loss occurs due to photo dissociation by high-energy (5 x 10^5 cm^{-1}) ultra-violet photons.

$$N_2O + h\nu \rightarrow N_2 + O \qquad [1.1]$$

A less significant loss mechanism is the reduction due to oxidation:

$$N_2O + O(^1D) \rightarrow NO + NO \text{ (6\%)} \qquad [1.2]$$

$$N_2O + O(^1D) \rightarrow N_2 + O_2 \text{ (4\%)} \qquad [1.3]$$

The major source of nitric oxide (NO) in the stratosphere occurs from the reaction [1.2], involving nitrous oxide emanating from the troposphere. This is followed by a reaction, which proceeds to catalytically destroy ozone by means of the cyclic reactions.

$$NO + O_3 \rightarrow NO_2 + O_2 \qquad [1.4]$$

$$NO_2 + O \rightarrow NO + O_2 \qquad [1.5]$$

$$NO + O_3 \rightarrow NO_2 + O_2 \qquad [1.4]$$

Etc...

Therefore, N_2O is not only a potent greenhouse gas but it also indirectly destroys stratospheric ozone, which is another environmental concern.

Method of remote sensing

Different methods of remote sensing have their strengths and weakness. Ground based infrared remote sensing (GBIRS) of the composition of the atmosphere yields superior data and results compared to earth looking satellites. GBIRS is more economical than airplane and balloon limb spectroscopy of the earth's atmosphere using the sun as a source. Infrared satellite remote sensing detects the spectrum of the earth's atmosphere against the low contrast temperature of the earth's surface. The quality of the observed data depends on the contrasts in temperatures between the layers of the atmosphere and the earth's surface. These contrasts are subtle and sometimes lead to low-resolution data at best. On the contrary, in the case of ground based remote sensing using the sun as a source, there is a large contrast between the temperature of the sun's photosphere and the earth's atmosphere. This enables the acquisition of high resolution, high quality atmospheric spectral data. While balloon and airplane limb spectroscopy of the earth's atmosphere using the sun as a source is marginally higher in resolution than stationary ground based atmospheric spectroscopy, the additional costs of atmospheric measurements by these methods will restrict the amount of atmospheric spectral data accrued. Ground based remote sensing is therefore the logical choice for daily monitoring atmospheric trace gases.

Atmospheric studies using the sun as a source started in the early 20th century. Numerous research groups are involved in sun source atmospheric studies. The University of Denver has one of the largest and most prominent groups in this field. They have developed algorithms that retrieve atmospheric profiles of trace gases using the line shapes. The author helped with the beginning of the Poker Flats, Alaska project at UD during a NOAA fellowship in Boulder, Colorado in the summer of 1998. The undertaking was to couple a suntracker to a high-resolution Bruker spectrometer in order to monitor trace gases in Alaska.

The project to integrate the 5-meter Littrow spectrometer to a suntracker was initiated in 1987 through a Research Incentive Grant to the Department of Physics by the University of Tennessee. The University of Tennessee Complex Systems Laboratory 5-meter Littrow spectrometer is the outcome of a long-term project initiated by Professor Norman Gailar who developed the original optical design of the spectrometer. The spectrometer was originally designed to record laboratory infrared spectra of gas samples. When the 5-meter Littrow spectrometer was no longer the state of the art for lab spectroscopy, it was decided to use it for atmospheric spectra where its' resolution would be comparable if not better than other atmosphere spectrometers. It was designed to have two beams in the sample section in order to have two laboratory experiments functioning at the same time. In this study, the second beam is used to direct the sun's radiation into the spectrometer. This allows the acquisition of an infrared spectrum of the sun's and the earth's atmosphere. The carbon rod source or glow bar, from the original design, can still be used in the first beam as a method to calibrate the spectrometer or to take the infrared absorption spectrum of molecules of interest.

A suntracker (see Chapter 3) was placed on the roof of the Science and Engineering Research building at the University of Tennessee. This instrument directs the sun's radiation to a fixed mirror, so the sun's radiation can be reflected over to a relay mirror and down a four story high optical conduit in the Nielsen Physics Building to the 5-meter Littrow spectrometer. Through a series of transfer optics, radiation is brought into the second beam port of the spectrometer. The new optics of the second beam were kept as close as possible to the original design.

The spectrometer was the subject of extensive study by Donald Jennings in 1974 (Jennings 1974). There have also been modifications since 1974 that enable the system to achieve more closely its theoretical resolution limit. These changes are discussed by Dakhil (Dakhil, 1983).

Goals

- Design and integration of a suntracker with the 5-meter Littrow spectrometric system at the University of Tennessee Complex Systems Laboratory.

- Establish whether column abundances of N_2O could be determined with experimental facilities at the University of Tennessee Complex System Laboratory. Part of this goal is to develop a model atmosphere and to compute synthetic spectra from this model.

- The measurement of temporal changes of N_2O in the earth's atmosphere using the 5-meter Littrow spectrometer.

Chapter Two

Spectroscopy of the Earth atmosphere

As it relates to this dissertation, spectroscopy of the earth's atmosphere was performed by first collecting radiation in order to introduce the light to a spectrometer.

It is beneficial to provide definitions of some of the radiation quantities in order to be consistent throughout this work. The definition and notation of radiation properties vary among references.

The Planck distribution represents an expression describing the spectral energy density of a perfectly absorbing and emitting body. When describing radiation there must be a source of that radiation. An ideal blackbody radiation source is generally employed because it is a perfect emitter; meaning all frequencies of the emitted radiation are at their highest possible intensity, most all other sources are a subset of a blackbody source. Planck's distribution or spectra energy density of states for a blackbody radiator, $\rho(\nu)$, in (cm^{-1}), is:

$$\rho(\nu) = \frac{8\pi h\nu^3}{c^3}\left(\frac{e^{-h\nu/kT}}{1-e^{-h\nu/kT}}\right) \qquad [2.1]$$

The Planck's distribution is used to define the radiation quantities shown in Table 2.1.

Table 2.1 Definition of Radiation Quantities.

Radiation Quantity	**Definition**	**Units**
Spectral Energy Density	$\rho(\nu)$	Joule/Meter^3*Hz
Energy Density	$\rho=\int_{-\infty}^{\infty}\rho(\nu)d\nu$	Joule/Meter^3
Intensity	Total watts of a point source divided by the total radians of a sphere 4π	Watt/Steradian
Spectral Irradiance	$F(\nu)=\frac{1}{\mu_0}\left(\vec{E}\times\vec{B}\right), F(\nu)=c\rho(\nu)$	Watt/Meter^2*Hz
Irradiance	$F=\int_{-\infty}^{\infty}F(\nu)d\nu$	Watt/Meter^2
Spectral Radiance	$I(\nu)=c\frac{d\rho(\nu)}{d\omega}$	Watt/(Meter^2*Steradian*Hz)
Radiance	$I=\int_{-\infty}^{\infty}I(\nu)d\nu$	Watt/(Meter^2*Steradian)

Radiance is independent of the direction of emission and size of the source, therefore it is the most universal quantity. Because it is the most general term it is used frequently in this work.

When we point a focusing device up into the atmosphere, such as a lens or concave mirror, what is the nature of the radiation we collect? Let us first describe how a focusing device collects radiation. Figure 2.1 shows how the aperture defines the solid angle $d\omega_l = \alpha/f^2$. Then the solid angle defines the directions of the radiation that the lens collects.

In most spectrometers there is usually a filtering process, like a filter wheel or a prism. This introduces a limit to the frequencies between ν and $\nu + d\nu$. The detector measures the rate, $\dot{E}$, at which energy passes through the filter. So the radiance at point P is:

$$I_P(\nu) = \frac{\dot{E}}{A d\nu d\omega_l} \qquad [2.2]$$

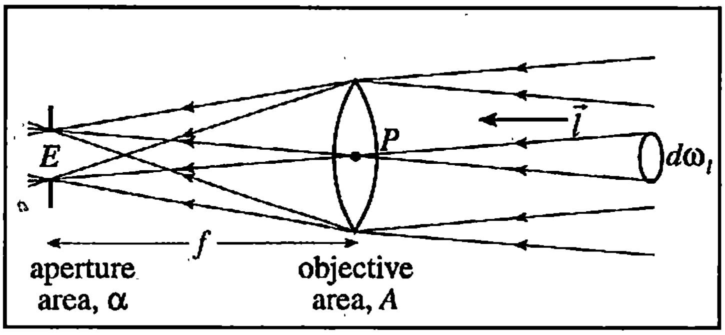


Figure 2.1 The radiance a lens collects

All interaction between radiation and matter results in either extinction or emission of energy (extinction includes both absorption and scattering terms). A light gathering lens or mirror is directed to an arbitrary point in the sky where there will be a change in the radiance due to both extinction and emission. The Schwartzschild's equation relates the rate of change of the radiance to the net emission and extinction.

$$\frac{dI(v)}{dl} = J(v)ne_v - I(v)ne_v \qquad [2.3]$$

The first term on the right is the emission term, the second is the absorption term and l is length. This relation will be explained in more detail later in the chapter, but the solution of this differential equation for the earth's atmosphere is the basic aim of majority of atmospheric physicist. All the large computer models (see chapter 4), attempt to solve this differential equation numerically.

General Properties of the Atmosphere

In order to understand the measurement of atmospheric spectra it is necessary to introduce some the properties of the earth's atmosphere. Some of the general properties presented here are: mixing ratios of atmospheric molecules, atmospheric particles, pressure and density profiles and the thermal organization of the atmosphere.

Chemical Composition of the Atmosphere

The mass of the earth's atmosphere is about 8.8 x 10^{-7} that of the entire planet. Mixing ratios of major trace gases are shown in Figure 2.2. Mixing ratios are column densities given by molecules per unit volume. Biological processes create nitrous oxide and methane and although both disassociate due to photochemical processes in the upper atmosphere, N_2O disassociates far more in the upper atmosphere, so its' mixing ratio will decrease more in the upper atmosphere than CH_4. Water vapor is the single most important thermodynamic gas or greenhouse gas in the lower atmosphere. Water vapor, along with carbon dioxide, are the driving forces of thermodynamics (radiative energy flow) in the earth's atmosphere. Carbon dioxide becomes more important in the middle atmosphere than H_2O.

Atmospheric aerosols

Particles suspended in a gas are called aerosols. Aerosols are one of the main sources of scattering of light in the earth's atmosphere. Apart from scattering from clouds, the most important light scattering process is that of Rayleigh scattering. Rayleigh scattering is of importance to many scattering phenomena of relevance to the atmosphere. For example, Rayleigh scattering describes not only how visible light is scattered by individual molecules, but also how longer wavelength infrared radiation is scattered by aerosols. Rayleigh scattering is proportional to λ^{-4}. Because of this in a vertical column of atmosphere, 40% of the sun's radiation is scattered out of the column in the near ultraviolet while less than 1% is lost in the near infrared (Stephens 1994). The only aerosols that will scatter infrared radiation in the 2500 cm^{-1} region are pollens, some industrial pollutants and cloud droplets while evaporating (Houghton 1986). Most of these particles are short lived and mainly are found in the lower 2 km of the earth's atmosphere. Although scattering may have an effect on data for zenith angles larger than 70 degrees, we, for the most part have avoided that scenario. For this reason it is believed that scattering by aerosols that effect infrared radiation is minimal and will not be taken into account in this study (Stephens 1994).

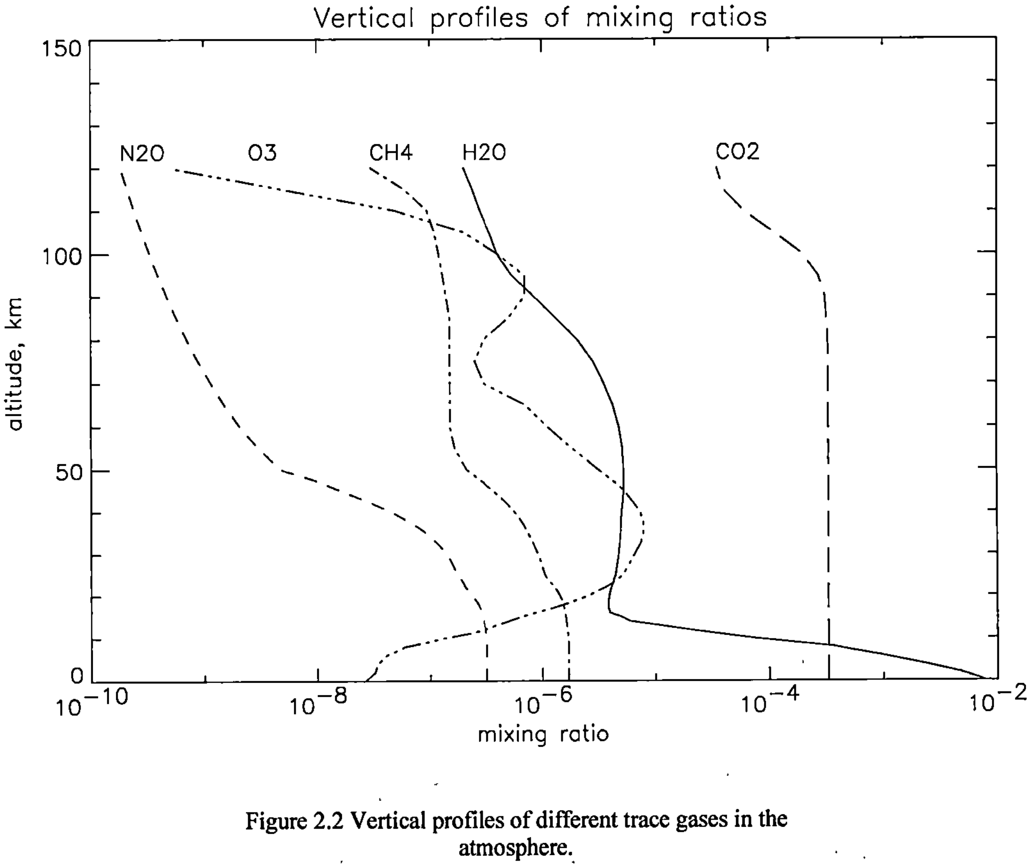


Figure 2.2 Vertical profiles of different trace gases in the atmosphere.

Vertical distribution of Pressure and Density

Due to the earth's gravity the density of the earth's atmosphere decreases with altitude in accord with $dp = -g\rho dz$. The equation for a perfect gas of molecular weight M is

$$\rho = \frac{Mp}{RT}$$

therefore,

$$\frac{dp}{p} = -\frac{dz}{H}$$

$$p \approx p_0 \exp\left(-\frac{z}{H}\right) \qquad [2.4]$$

where ρ is density, p is pressure, R is the ideal gas constant, T is temperature and z is the altitude. The variable $H = RT/Mg$ is called the scale height. It depends on average molecular weight and temperature of a layer of atmosphere. The scale height of the earth's atmosphere varies (6.3 < H < 8.5) with altitude mainly due to temperature fluctuation. The difference in molecular mass of atmospheric molecules does not affect the mixing ratio up to 120 km. The molecular diffusion, due to the weight of the molecule, is a very slow process even at normal pressures. The molecular diffusion of the atmosphere due to molecular weight is negligible all the way up to the upper atmosphere (120 km), due to the mixing of the atmosphere. Past 120 km one starts seeing higher concentrations of hydrogen due to molecular diffusion. Another important observation based on equation [2.4] is that 99% of the mass of the earth's atmosphere lies below 30 km and 50% lies below 5.5 km.

Thermal structure of the atmosphere

The vertical thermal structure of the earth's atmosphere is the main factor for the identifying of the layers of the atmosphere. Figure 2.3 shows the customary partitioning of the atmosphere into four distinctive atmospheric layers, troposphere, stratosphere, mesosphere and thermosphere.

The levels separating these layers are referred to as the tropopause, stratopause, mesopause and thermopause. Figure 2.3 shows that the thermal structure or the change in direction of the temperature marks these levels separating the layers.

The lapse rate is the rate at which the temperature drops with altitude in the troposphere. Some regular features of atmospheric layers include the lapse rate of the troposphere, which has a mean value of 6.5 K/km. This lapse rate in the troposphere is evidence of a main thermodynamic source of the earth's atmosphere, which is the earth's surface heating the atmosphere.

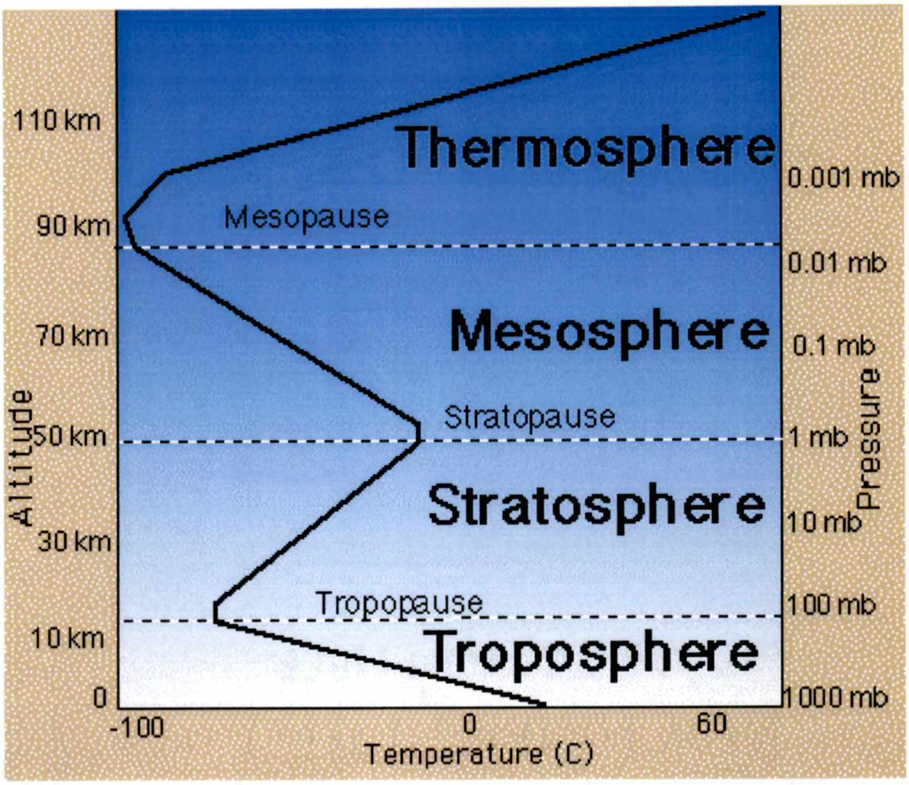


Figure 2.3 The Thermal Structure of the Atmosphere and How it Relates to the Layers of the Atmosphere.

Since the earth's surface absorbs considerably more of the sun's energy than the atmosphere (you can fry an egg on the asphalt in the south in the summer) the surface of the earth heats up the atmosphere faster than the sun's radiation. This explains why the temperature in the lower troposphere is warmer than that of the upper troposphere. It is interesting to note that if the atmosphere were non-reactive to the radiation the lapse rate would be ~10 K/km.

Local thermodynamic equilibrium

Local thermodynamic equilibrium is achieved by the dividing up of a system that is not in thermodynamic equilibrium into sub-systems that can be considered in thermodynamic equilibrium.

An atmosphere where temperature and pressure are a function of position is not in thermal equilibrium. However, it is usually possible to divide the atmosphere up into smaller systems that can be considered to be isobaric and isothermal. These systems or small layers can be treated as if they are in thermodynamic equilibrium. Each of the layers is said to be in 'local thermodynamic equilibrium'. In this work we will use the concept of a stratified atmosphere. A stratified atmosphere is a simple model represented by flat layers, which are in local thermodynamic equilibrium, where state variables do not change horizontally.

The concept of thermodynamic equilibrium is key to atmospheric studies. Each molecule in the earth's atmosphere has a large number of energy states, distributed over translational, rotational, vibrational and electronic energies. If we knew the populations of every quantum state, the energy state of the whole system would be known. This of course is impossible, so we must rely on equilibrium relationships between state population and how they relate to the system as a whole. We must rely on macroscopic state variables such as temperature and pressure. These state variables and Planck's blackbody distributions are derived in part from the Boltzmann distribution, i.e. the statistical mechanics of thermodynamic equilibrium. In the earth's atmosphere there are non-local thermodynamic equilibrium effects. But these are very complicated and mainly deal with atmospheric emission spectra. This subject of non-local thermodynamic equilibrium effects is now being considered by Tony Clough and others and will be left for future consideration (Clough 1992).

Radiation

Radiation from the sun

The sun's disk subtends, on the average, an angle of 32' at the earth's surface; and for most practical purposes sunlight can be regarded as a parallel beam of radiation. An expression for irradiance of the sun's radiation will now be derived. For a specific definition of radiance of the sun at the earth's surface, $s_{\circ}$ is used as the direction of the sun, let $d\omega_{\circ}$ be the solid angle the sun's disk subtends at the earth.

$$\bar{I}_s(\nu) = \int_A I_s(\nu) dA \qquad [2.5]$$

where $\bar{I}_s(\nu)$ is the mean value of the intensity averaged over the sun's disk.

$$F_{s,d} = \int_{\omega_s} I_s(\nu) \cos\left(\vec{d}, s_{\circ}\right) d\omega_s$$

$$F_{s,d} = \bar{I}_s(\nu)\cos\left(\vec{d}, s_\circ\right) d\omega_\circ$$

$F_{s,d}$ is the irradiance of the sun going through a unit area whose unit vector is $\vec{d}$. The $\cos\left(\vec{d}, s_\circ\right)$ term has to be considered on surfaces that are not perpendicular s_o like a mirror on a suntracker. If we let $\cos\left(\vec{d}, s_\circ\right)$ equal to one, unit area is perpendicular to the direction of the sun.

$$F_s = \bar{I}_s(\nu) d\omega_\circ \qquad [2.6]$$

then F_s is the solar irradiance, a positive quantity that is a function of solar distance only. The solar irradiance at the earth's surface is approximately 1368 Watt/meter^2.

Beer-Lambert law

The fundamental law of extinction is the The Beer-Lambert law. It states that the extinction process is linear, independent of the intensity of the radiation and the amount of matter, provided that the state is in thermodynamic equilibrium. In other words, the change in intensity along a path dl is proportional to the amount of matter in the path.

$$\frac{dI(\nu)}{dl} = -e(\nu) n I(\nu) \qquad [2.7]$$

where $e(\nu)$ is the monochromatic extinction coefficient, and n is the number density of the absorbing molecules. The monochromatic extinction coefficient is not only absorption, but includes a scattering term.

$$e(\nu) = k(\nu) + s(\nu) \qquad [2.8]$$

where $k(\nu)$ is the monochromatic absorption coefficient and $s(\nu)$ represents the monochromatic scattering coefficient. The monochromatic extinction coefficient $e(\nu)$ has units of area. Representing the extinction process as a collision between a photon of negligible size and a molecule, as defined by equation [2.7], $e(\nu)$ is the collision cross section (Goody 1995).

Equation [2.7] can be conveniently rearranged and integrated to give

$$I(\nu) = I_0(\nu)\exp(-e(\nu)nl) \qquad [2.9]$$

which one recognizes as the the Beer-Lambert law.

Let us first consider molecular absorption for the moment. The equation

$$k(\nu) = Sf(\nu - \nu_0). \qquad [2.10]$$

is the definition of the absorption coefficient, where S is the line strength and $f(\nu - \nu_0)$ is the line shape function.

The monochromatic transmission function is defined as

$$\mathrm{T}(\nu) = \frac{I(\nu)}{I_0(\nu)} = \exp(-k(\nu)nl). \qquad [2.11]$$

The optical path is defined as

$$\tau_v = \int k(v) n dl \qquad [2.12]$$

The optical path along a line tilted from the vertical by an angle θ, the zenith angle, is

$\tau(l) = \tau_v(z)/\mu$, where $\mu = \cos\theta$ for a stratified atmosphere Figure 2.4. $\tau_v(z)$ looking down into the atmosphere is called optical depth.

Column abundances

We define the total column abundances or optical mass as

$$u(z) = \int \rho_a dz \qquad [2.13]$$

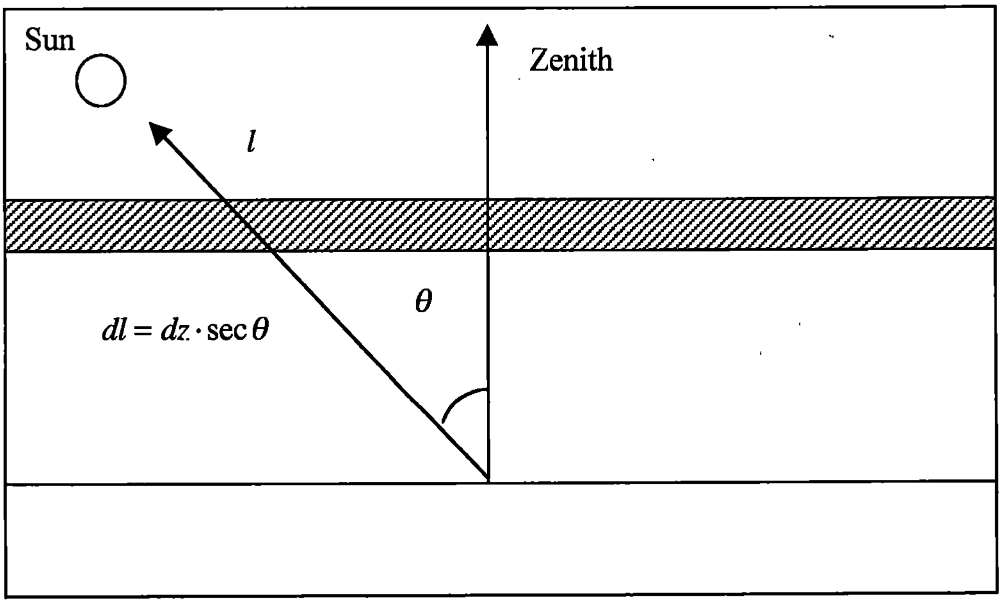


Figure 2.4 Illustrates Zenith Angle and Optical Path

The density ρ_a is number of molecules per volume. Therefore the column abundance is density multiplied by length Δz

$$\rho_a \Delta z \qquad [2.14]$$

Column abundances are sometimes preferable to column densities because strengths of lines are in unit of the reciprocal of column abundances. Column abundances are similar to cross-sections, in the sense that they are in units of number of molecules/area.

$$S_A^B \left[\frac{1}{\rho_a \Delta z} \right]$$

Column abundances are usually quoted in units of molecules per square centimeter.

Schwartzschild's equation of transfer

The earth's atmosphere not only absorbs the sun's radiation, but is also emits its own radiation. The atmosphere emits radiation that is collected by the suntracker. When we take atmospheric absorption spectra, using the sun as a source, we will chose to neglect the emission terms in the Schwartzschild's equation of transfer because of the minimal contribution to the radiance. A heuristic argument is presented below to justify this assumption.

As stated before the Beer-Lambert law is the fundamental law for loss of radiance as a result of interactions between radiation and matter. Thermal emission and scattering are processes that add to this radiance. Both thermal emission and scattering are proportional to the amount of matter in the path, as before with absorption. As a formal definition, we introduce the source function $J(\nu)$ through the equation.

$$\frac{dI(\nu)}{dl} = J(\nu)ne_\nu \qquad [2.15]$$

All interactions between radiation and matter lead to either emission or absorption. We then just add equation [2.9] to obtain the differential equation:

$$\frac{dI(\nu)}{dl} = J(\nu)ne_\nu - I(\nu)ne_\nu \qquad [2.3]$$

This is the Schwartzschild's equation of energy transfer. Schwartzschild's equation of radiative energy transfer can also be considered as a statement of the first law of thermodynamics.

The contribution of emission from the atmosphere can be calculated from the radiance from a 300K and 6000K blackbody. However, we must take in to consideration the distance of the sun from the earth and the solid angle of the emitting radiation. The radiance of 2500 cm^{-1} @300K is $2\cdot 10^{-4}$ $watts/cm^2\cdot sr\cdot \mu m$ and the radiance of 2500 cm^{-1} @6000K is 5 $watts/cm^2\cdot sr\cdot \mu m$. Integrate 5 $watts/cm^2\cdot sr\cdot \mu m$ over the entire surface of the sun and use the inverse square law to show that the irradiance at the earth's surface of 2500 cm^{-1} is $1.08\cdot 10^{-4}$ $watts/cm^2\cdot \mu m$. The solid angle of the disc of the sun is $3\cdot 10^{-2}$ sr. which is very small. Calculating the emission from the atmosphere very close the entrance to the optical system using the same small solid angle subtended by the sun one gets $5 \cdot 10^{-6}$ $watts / cm^2 \cdot \mu m$ for the emission of the atmosphere, Of course the further away the layer of atmosphere is the less the irradiance. Also, at 2500 cm^{-1} the earth's atmosphere emits about 40% of the radiation of a blackbody. All the factors together let us neglect the emission terms in Schwartzschild's equation of transfer. So the problem reduces down to just the Beer-Lambert law, equation [2.9]:

$$I(\nu) = I_0(\nu)\exp(-e(\nu)nl)$$

Atmospheric absorption spectra

In order to develop a program that will calculate synthetic spectra on the earth's atmosphere, one must first work out an algorithm to model the earth's atmosphere.

The earth's atmosphere must be divided up into layers in order to take advantage of local thermodynamic equilibrium. Each layer is said to be at an average temperature, pressure and density of different atmospheric gases. One has to take into account several different factors in atmospheric absorption spectra as opposes to taking spectra in the lab with a gas cylinder. First we will develop the Beer-Lambert law as it applies to spectra of a mix gases.

A gas may have several absorption lines in a small region of the spectrum. When these lines are close to each other, we cannot add The Beer-Lambert law for each line i.e. $(I(\nu))_1 + (I(\nu))_2 + ...$ A photon can only be absorbed by one molecule at a time. A baseline $I_0(\nu)$ is implied using the the Beer-Lambert law. The baseline will change as you go though a gas of multiple species, thus it is necessary to consider absorption happening in small layers. The first absorption line from one molecule may have already absorbed radiation at a specific frequency before the second molecule does. The baseline for the second molecule is just The Beer-Lambert law for the first.

$$I(\nu) = \left(I_0(\nu)e^{\tau_1}\right)e^{\tau_2} = I_0(\nu)e^{\tau_1 + \tau_2} \qquad [2.16]$$

Therefore we must add up the optical path for each molecule before using the Beer-Lambert law.

$$T(\nu) = \frac{I(\nu)}{I_0(\nu)} = \exp\left(-\sum_{j=1}^{m} k_j(\nu)\rho_j dl\right) \qquad [2.17]$$

where j is the quantum line reference. Equation [2.17] is the Beer-Lambert law for gas samples that have multiple species. When layering the atmosphere we will need to sum up the layers in the same way and they should not be confused.

The optical path is $\tau_\nu = \int k_m(\nu)\rho_a dl$, where $dl = dz \cdot \sec\theta$. If we have n layers, and sum up each layer, we have

$$T(\nu) = \exp\left(-\sum_{i=1}^{n}\sum_{j=1}^{m} k_{ij}(\nu) \cdot \rho_{ij} \cdot z_i \cdot \sec\theta\right) \qquad [2.18]$$

where $k(v)$ is the absorption coefficient. Equation [2.18] is the form of The absorption coefficient can always be written as:

$$k(v) = S \cdot f(v - v_0) \tag{2.19}$$

where S is the strength of the line and $f(v - v_0)$ is the line shape function. The line shape function does not affect the strength of the line so,

$$\int_{-\infty}^{\infty} f(v - v_0) dv = 1 \tag{2.20}$$

and,

$$S = \int_{-\infty}^{\infty} k(v) dv. \tag{2.21}$$

LINE STRENGTHS

The spectral line intensity in units of $\left[cm^{-1}/\left(molecule \cdot cm^{-2}\right)\right]$ and at the reference temperature of 296 K is S_A^B. Then intensity is defined here for a single molecule. The Radiative transfer theory (Penner 1959) for two states of a vibrational-rotational system defines the spectral line intensity as

$$S_A^B = \frac{h v_{AB}}{c} \frac{n_A}{N} \left(1 - \frac{g_A n_B}{g_B n_A}\right) B_{AB} \tag{2.22}$$

where B_{AB} in units of $\left[\mathrm{cm}^3/\left(ergs\cdot s^2\right)\right]$ is the Einstein coefficient for induced absorption, n_A and n_B are the populations of the lower and upper states, respectively, g_A and g_B are the state statistical weights, and N is the molecular number density. The weight includes electronic, vibrational, rotational, and nuclear statistics. The quantity in parentheses represents the effect of stimulated emission.

The Einstein coefficient B_{AB} is related to the weighted transition-moment squared

$$\left|\langle A|\mu|B\rangle\right|^2 \left[Debye^2 = 10^{-36} ergs\cdot cm^3\right].$$

$$\left|\langle A|\mu|B\rangle\right|^2 = \frac{3h^2}{8\pi^3} B_{AB}\cdot 10^{36} \qquad [2.23]$$

Assuming local thermodynamic equilibrium (LTE), the population partition between states is governed by Boltzmann statistics at the ambient temperature T. This allows us to write

$$\frac{g_A n_B}{g_B n_A} = \exp\left(-hc\nu_{AB}/kT\right) \qquad [2.24]$$

and,

$$\frac{n_A}{N} = \frac{g_A \exp\left(-hcE_A/kT\right)}{Q} \qquad [2.25]$$

where E_A is the lower state energy in units [cm^{-1}]. The total internal partition sum Q is given by

$$Q = \sum_m g_m \exp(-hcE_m/kT) \qquad [2.26]$$

where g_m is the degeneracy of state m.

Substituting Equations [2.24], [2.25], [2.26] at T, and introducing I_a, the natural terrestrial isotopic abundance, gives

$$S_A^B = \frac{8\pi^3}{3hc}\frac{I_a g_A}{Q}\nu_{AB}\cdot\exp\left(\frac{-hcE_A}{kT}\right)\cdot\left[1-\exp\left(\frac{-hc\nu_{AB}}{kT}\right)\right]\cdot\left|\langle A|\mu|B\rangle\right|^2\cdot 10^{-36} \qquad [2.27]$$

It should be understood that S_A^B is weighted according to the natural terrestrial isotopic abundances (Rothman, Rinsland et al. 1998).

The strength, in unit of $\left[cm^{-1}/cm\cdot atm\right]$ @300K, without isotropic abundances, is given by:

$$S_A^B = \frac{8\pi^3}{3hc}\frac{N}{Q}\nu_{AB}\cdot\exp\left(\frac{-hcE_A}{kT}\right)\cdot\left[1-\exp\left(\frac{-hc\nu_{AB}}{kT}\right)\right]\cdot\left|\langle A|\mu|B\rangle\right|^2 \qquad [2.28]$$

This is the unit used in the authors atmospheric modeling program. The HITRAN unit of strength has to be converted into this unit in order to use the state variable pressure in atmospheres.

LINE SHAPES

There are three different line-broadening types: natural, collisional or Lorentz and Doppler. Natural line broadening comes from the Heisenberg uncertainty principle written in the form.

$$\Delta t \cdot \Delta E \geq \hbar \qquad [2.29]$$

Equation [2.29] means that an energy determination that has accuracy of ΔE must occupy at least a time interval $\Delta t \approx \hbar / \Delta E$. Due to spontaneous emission a molecule can only absorb a photon for a finite period of time. This will leads to a natural broadening of an absorption line because the energy cannot be infinitely well defined.

Lorentz broadening

Lorentz or Collisional broadening comes from the same principle as natural line broadening. The number of collisions in a gas increases in proportion to the pressure. Instead of the molecules spontaneously emitting, the collisions induced an emission. The average time a molecule stays in an excited state is shorter so the energy becomes less certain causing the absorption line to broaden. The normalize expression for Lorentz or Collisional broadening is:

$$f(\nu - \nu_0) = \frac{b_L}{\pi\left[(\nu - \nu_0)^2 + b_L\right]} \qquad [2.30]$$

where b_L is the collision-broadened half-width. Both natural and Collisional broadening have a lorentzian shape. In the earth's atmosphere, the normal range of pressures encountered, the collision-broadened half-width b_L is linearly dependent on the pressure.

$$b_L = b_{0L}p \qquad [2.31]$$

Doppler broadening

Doppler broadening is caused by the Doppler shift associated with molecular motions. For thermal equilibrium between translational levels, the probability that one velocity component lies between υ and $\upsilon + d\upsilon$, is given by Maxwell's distribution of velocities:

$$p(\nu) = \left(\frac{m}{2\pi kT}\right)^{\frac{1}{2}} \cdot \exp\left(-\frac{m\upsilon^2}{2kT}\right) \qquad [2.32]$$

At non-relativistic speeds the Doppler shift for an emitted photon with a rest-frame frequency, ν_0, is

$$(\nu - \nu_0) = \frac{\nu_0 \upsilon}{c} \qquad [2.33]$$

We combine equations [2.32], [2.33] to eliminate υ.

$$f(\nu - \nu_0) = \frac{1}{b_D}\left[\frac{\ln 2}{\pi}\right]^{\frac{1}{2}} \cdot \exp\left[-\ln 2\left(\frac{(\nu - \nu_0)}{b_D}\right)^2\right] \qquad [2.34]$$

Where b_D is the Doppler broadening half-width and is equal to:

$$b_D = \left(\frac{v_0}{c}\right)\left[\frac{2kN_A T\ln 2}{M}\right]^{\frac{1}{2}} \qquad [2.35]$$

or,

$$b_D = 3.581\times 10^{-7}\cdot v_0\left(\frac{T}{M}\right)^{\frac{1}{2}} \qquad [2.36]$$

where M is the molecular weight N_A is Avogadro's number and T is the temperature of the gas.

Voigt profile

The Voigt profile is a convolution of independent Lorentz and Doppler profiles and is given by

$$f(v - v_0) = \frac{Cy}{\pi}\int_{-\infty}^{\infty}\frac{\exp(-t^2)}{(x-t)^2 + y^2}dt \qquad [2.37]$$

Where C is a normalization constant $C = (1/b_D)[(\ln 2)/\pi]^{1/2}$ and

$$y = (\ln 2)^{\frac{1}{2}}\frac{b_L}{b_D} \qquad [2.38]$$

The relative importance of Doppler broadening compared to Lorentz broadening for N_2O can be appreciated in terms of the ratio

$$\frac{b_D}{b_L} \approx 10^{-5} \frac{\nu_0}{p} \qquad [2.39]$$

where the units of b_D/b_L is cm^{-1}/atm.

Continuum

The distortion of the baseline due to large Lorentzian wings and weak pressure broadened absorption lines is called the continuum. Lorentzian line shape extends a long distance in cm^{-1} from the peak. If we look at a lorentzian line shape in transmission space, the wings would approach complete transmission or 1.0 slowly and extend far from the center of the line. Large absorption lines can contribute as far as 100 cm^{-1} or more from the line centers. It is conventional to consider displacements of less than 25 cm^{-1} from the line centers as line contributions and displacements more than 25 cm^{-1} as continuum contributions. These large line wings due to the large pressure broadening in the lower atmosphere along thousands of smaller pressure broaden lines make up what is called the continuum. The continuum shifts the baseline of spectrum of other atmospheric molecular gases and is very important to the thermal budget of the earth's atmosphere. Some of these large absorption lines that have large lorentzian wings come from carbon dioxide and water vapor.

Water vapor and carbon dioxide are very radiative intense molecules in the lower atmosphere and water vapor can have a mixing ratio that varies wildly from as much as

1×10^{-2} to 1×10^{-5}. Figure 2.5 illustrates the large amount of energy that H_2O and CO_2 absorb from the solar irradiance. H_2O being an asymmetric tri-atomic molecule has three fundamental bands. The ν_2 band is centered at 1595 cm^{-1} and along with the 2349 cm^{-1} ν_2 band of CO_2 they completely absorb most of the radiation from 1400 – 2400 cm^{-1}. This is important because of two strong bands, $\nu_1 + 2\nu_2$ and $2\nu_1$, of N_2O. The $\nu_1 + 2\nu_2$ band center is at about 2462 cm^{-1}, which is still on the line wings of the 2349 cm^{-1} ν_2 band of CO_2. But the $2\nu_1$ band is in the middle of the 2500 cm^{-1} gap or hole in the continuum. This means that the continuum is very small and mostly N_2O and CH_4 absorb the sun's radiation at this part of atmospheric spectrum. This eliminates the need to adjust the baseline due to the continuum and will lead to more accurate abundance results. This latter fact is one of the principal motivations that led to the study of the N_2O lines in this region were chosen for this research.

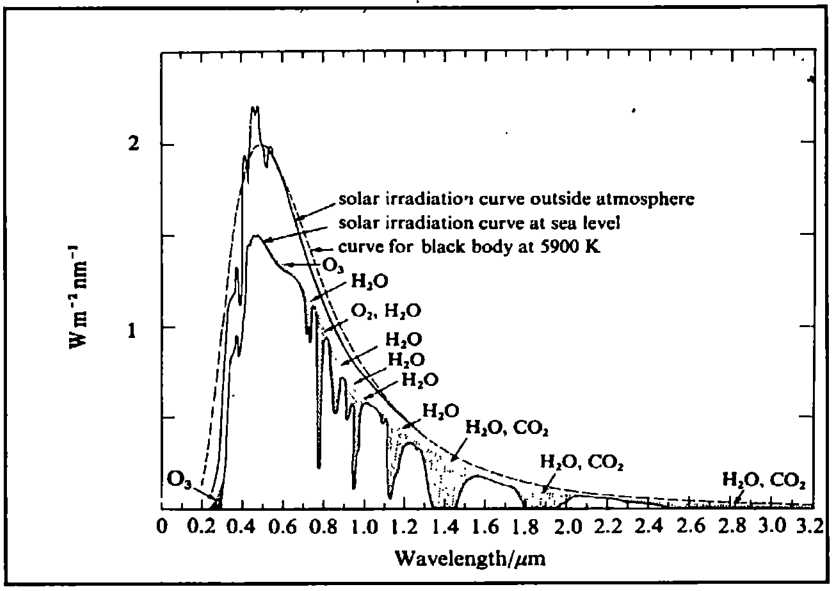


Figure 2.5 Shows the solar irradiation at both outside the atmosphere and at sea level.

Nitrous Oxide

The nitrous oxide molecule in its ground state is linear and structurally asymmetric (NNO). The three fundamental bands are ν_1 (1284.907cm-1), ν_2 (558.767 cm-1) and ν_3 (2223.756 cm-1). Since ν_1 is approximately equal to $2\nu_2$ there is a strong Fermi resonance between the levels. The isotropic abundances are $^{14}N^{14}N^{16}O$ (99.043%), $^{15}N^{14}N^{16}O$ and $^{14}N^{15}N^{16}O$ (0.358%), $^{14}N^{14}N^{18}O$ (0.199%), and $^{14}N^{14}N^{17}O$ (0.040%) (Goody and Yung, 1989).

Quantum mechanics of N₂O

The rotational quantum mechanical aspects of a molecule will be considered first and how that applies to a linear molecule such as N₂O. The vibrational quantum mechanical aspects will be considered secondly. It will be shown how rotation and vibration relate to the temperature of the gas, which is important to calculating the strengths of each absorption line.

The rotation of a molecule involves three moments of inertia. The molecule has two moments of inertia if the molecule is symmetric about one axis the rotation The figure axis is the axis of symmetry.

If $B = \dfrac{h}{8\pi^2 c I_\perp}$, $A = \dfrac{h}{8\pi^2 c I_\parallel}$, $I_\parallel$ is the moment of inertia around the figure axis, then

$I_\perp$ is the moment of inertia perpendicular to the figure axis. The energy levels (in cm^{-1}) are given by

$$E(J)/hc = BJ(J+1) + (A-B)K^2 \qquad [2.40]$$

where, J is the rotational quantum number and K is the angular momentum around the figure axis.

The vibrational energy of a molecule is

$$E(v) = \left(v + \frac{1}{2}\right)\hbar\omega + \text{ anharmonic terms} \qquad [2.41]$$

The selection rules for perpendicular and parallel vibrational-rotational bands are:

Parallel bands:

$\Delta K = 0,\ \Delta J = 0, \pm 1 \text{ if } K \neq 0,\ \ \Delta J = \pm 1, \text{if } K = 0$

Perpendicular bands:

$\Delta K = \pm 1,\ \Delta J = 0, \pm 1$

N_2O is a linear molecule and does not have angular momentum about the figure axis, and K is equal to zero. Because of the selection rules N_2O does not have perpendicular bands and therefore a Q-branch. The infrared spectrum exhibits only the R-branch for $\Delta J = +1$ and P-branch for $\Delta J = -1$. Equation [2.40] reduces to

$$E(J)/hc = BJ(J+1) \qquad [2.42]$$

The energy of a rotating and vibrating molecule is approximated as

$$E(v,J) = \left(v + \frac{1}{2}\right)\hbar\omega + hcBJ(J+1) + \ldots \qquad [2.43]$$

plus some higher order terms of vibration and rotation due to different effects.

The absorption probability of a molecule not only depends on the transition probability as calculated by quantum mechanics but also depends on the number of molecules in the initial state. For vibrational transitions, the probability of being in a certain state is given by the Boltzmann factor $\exp(-E/kT)$. This is not the case in rotational transitional probabilities. The thermal distribution of the rotational levels (unlike that of the vibrational levels) is not simply given by the Boltzmann factor $\exp(-E/kT)$. We have to allow for the fact that, according to quantum theory, each state with total angular momentum J consists of 2J+1 levels which coincide in the absence of an external field; that is the state has a (2J+1)- fold degeneracy. The frequency of its occurrence (its statistical weight) is therefore (2J+1) times that of a state with J=0 (Herzberg 1950). The number of molecules N_J in the rotational level J of the lowest vibrational state at the temperature T is thus proportional to

$$N_J \propto (2J+1)\cdot\exp(-BJ(J+1)\,hc/kT) \qquad [2.44]$$

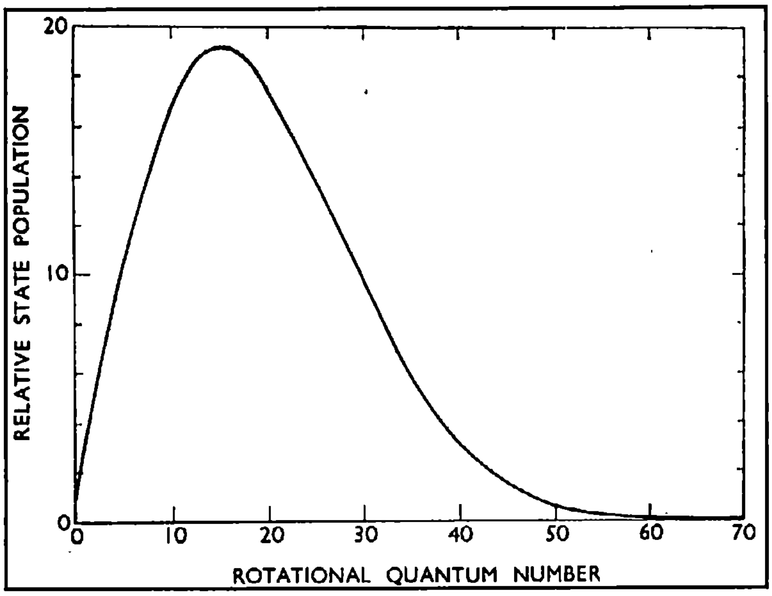


Figure 2.6 Thermal distribution of the Rotational Levels for T=300K and B=10.44 cm^{-1}

The maximum J value lies at

$$J_{\max} = \sqrt{\frac{kT}{2Bhc}} - \frac{1}{2} = 0.5896\sqrt{\frac{T}{B}} - \frac{1}{2} \qquad [2.45]$$

The overall shape of the P and R branches of N_2O at a given temperature are described by equation [2.44] and shown in Figure 2.6 for 300K. When the temperature increases or decreases we can see that the shape of this distribution changes. Therefore the intensities of the absorption lines change considerably with temperature. In order to build a model of the earth's atmosphere the range of temperatures of these intensities must be calculated. The range of temperature is approximately from 210K to 300K.

Chapter Three

The Integration of a Suntracker to the Spectrometer

Introduction

In many types of astronomical research, it is desirable to have the measuring instrument or detector remain stationary while a radiation collector moves with the object under observation, unlike a telescope that moves with the object. In order for the instrument to remain stationary you must reflect the radiation of the astronomical object into the instrument, by a moving mirror. A suntracker is such a device that tracks the sun. At UTCSL a suntracker was built and located on top of the new Science and Engineering Building at the University of Tennessee. The UTCSL Suntracker-2 rotates a mirror around the polar axis to track the sun. The latitude of Knoxville is 36 degrees so this is the angle of the axis of the polar mounted shaft, which rotates at a speed of $2\pi/24$ radians/hour. Figure 3.1 shows the set up with three mirrors on top of the Science and Engineering Building at the University of Tennessee, one tracking mirror and two relay mirrors, then a fourth mirror placed on top of the fourth floor roof of the physics building.

The radiation is bought down a four story high optical conduit to 5-meter Littrow spectrometer.

Suntracker

A Suntracker-1 was initially built on top of the fourth floor roof of the Nielsen Physics Building, but was limited to tracking times due to obstructions of buildings and trees. Building a Suntracker-2 on top of the Science and Engineering Building accomplished two goals. The roof of the Science and Engineering Building is one of the highest points on campus. This solved the obstruction problem of trees and buildings. Also, a new 15" diameter optical pipe was built into the Science and Engineering Building, that travels from the roof to room 606 on the 6th floor of the Science and Engineering Building. A new FTIR spectrometer has been ordered and will be coupled with the Suntracker-2, for future investigations.

The Suntracker-2 is designed by the author and built by the machine shop at the UT Physics department. During the construction of the Science and Engineer Building four pylons were positioned on the roof near the new 15" diameter optical pipe. The Suntracker-2 frame had to be designed into sections small enough to fit through a 5' X 2.5' hatch in the SERF roof. A platform was designed by the author and built by B&D Construction Co. to fit on the top of the pylons.

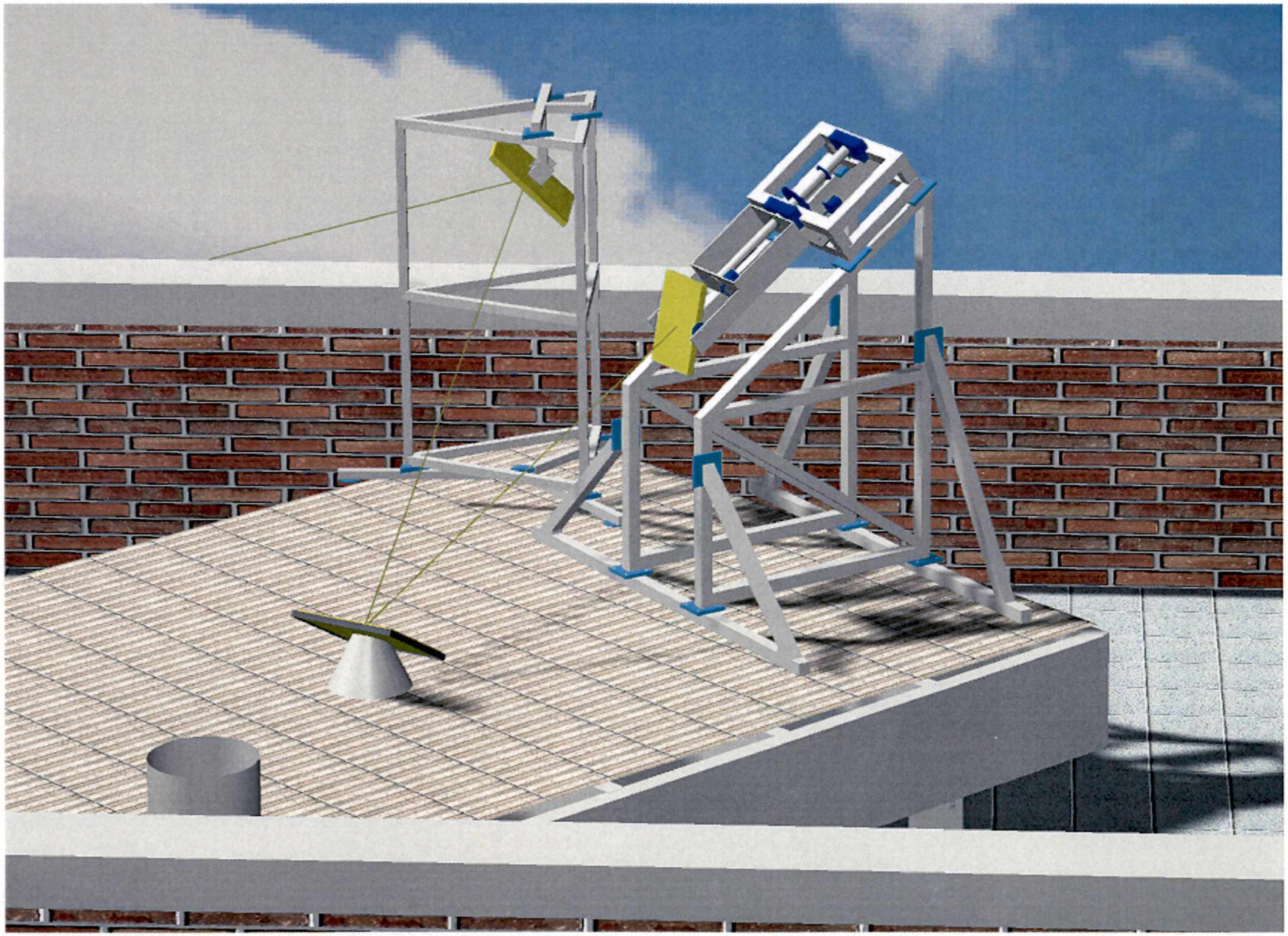


Figure 3.1 The Suntracker-2 with two relay mirrors. The two relay mirrors direct the sun's light down to a third relay mirror at Nielsen Physics Building

The platform is built from prefabricated roofing trusses and is both lightweight and very strong. The sections of the Suntracker-2 were bolted together on top of the platform.

The exact location of the Suntracker-2 is 35° 57' 25" N and 83° 55' 30"W. The axis of the rotation must be 35° 57' 25" off the horizon and parallel with the latitude of 83° 55' 30"W. The slow speed at which the Suntracker-2 must rotate requires a large gear reduction ratio. The average speed of the stepper motors is 1/2 revolution/second. We need a rotation of approximately 1/2 revolution per 12hours. So, the gear reduction ratio is approximately 43,200:1. This large gear reduction is produced by two worm gears in series with each other. Figure 3.2 shows the drive mechanism.

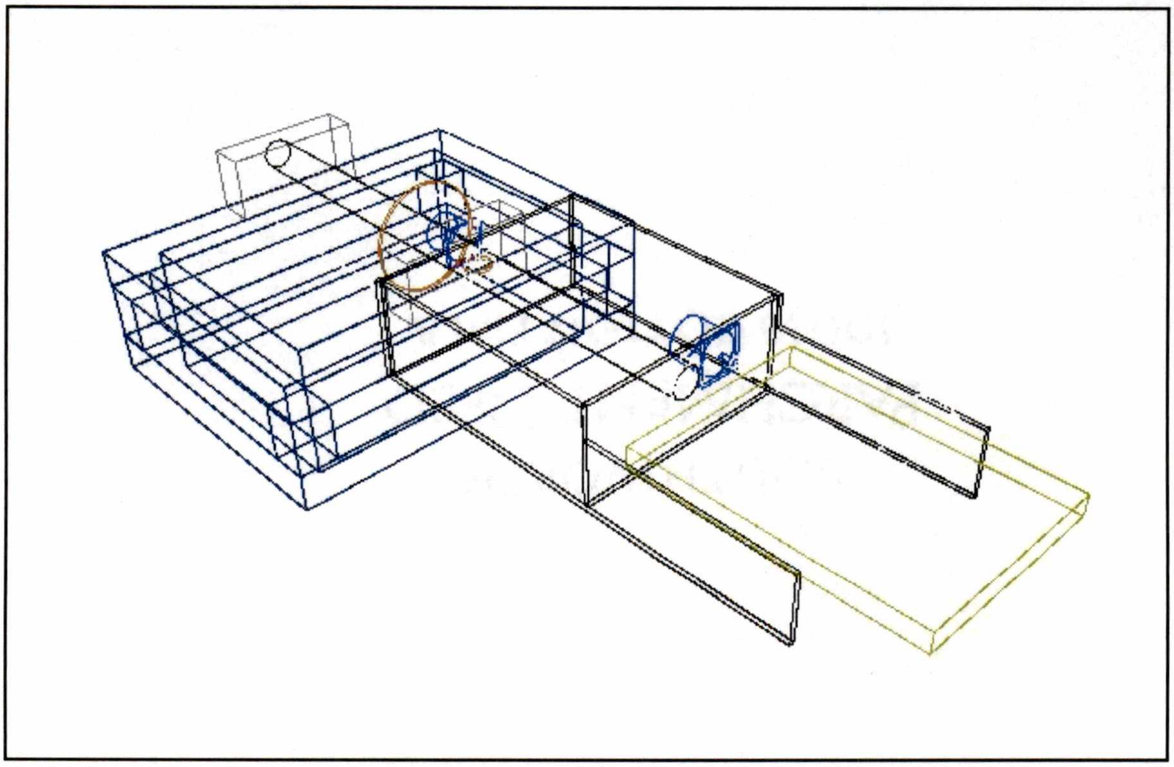


Figure 3.2 Main drive of Suntracker-2

The solstices and the zenith angles of the sun determine the range of optical paths that can be scanned. The summer and winter solstices dictated the maximum azimuthal angles of sunrise and sunset throughout the year. The four major azimuthal directions are, 0° North, 90° East, and 180° south and 270° west. The azimuthal ranges of sunrises at the University of Tennessee are from summer solstice, 63°, to the winter solstice, 117°. The ranges of sunsets are from 243° for the winter solstice to 297° for the summer solstice Figure 3.3.

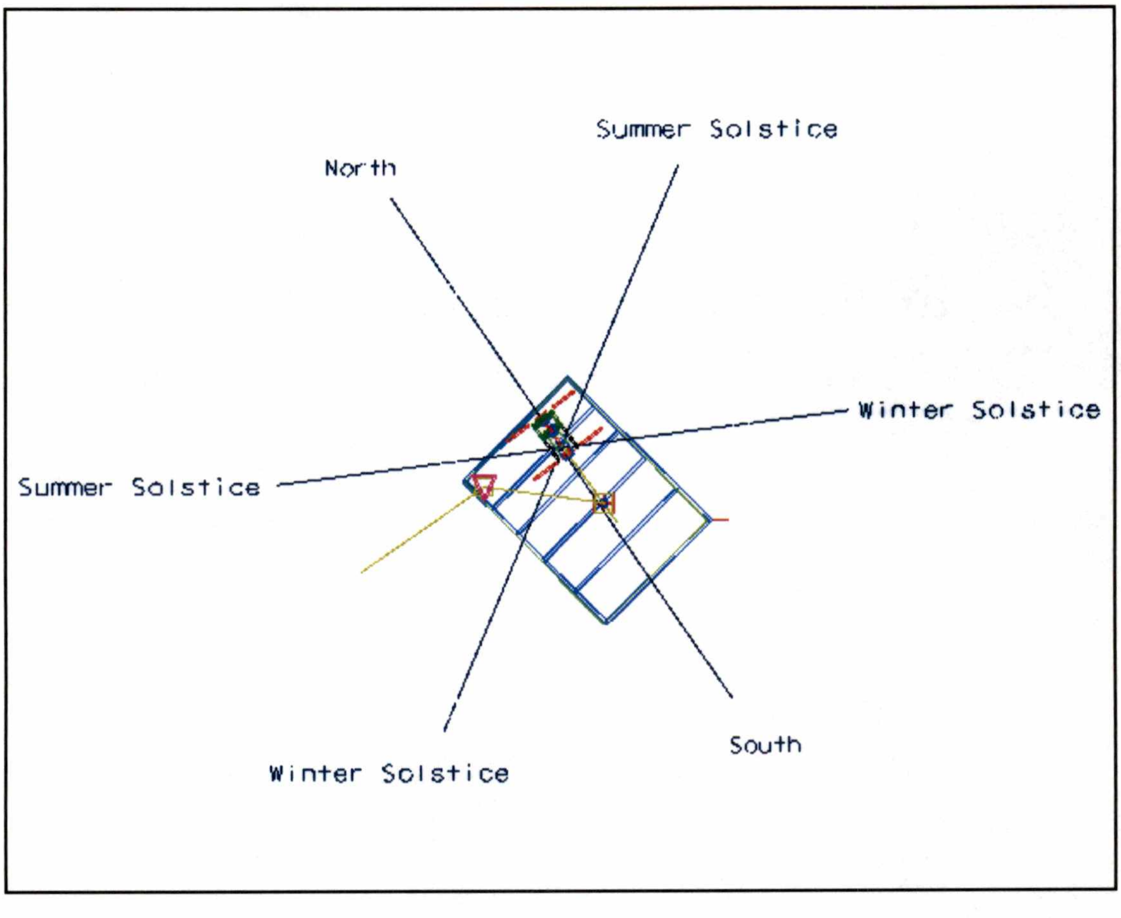


Figure 3.3 Azimuthal Solstice Angles

The third mirror that relays the sun's light over the side of the Science and Engineer Building to the Nielsen Physics Building is designed to be as low as possible on the roof of the Science and Engineering Building, so sunset data can be taken during the times around the equinoxes Figure 3.3

During the recording of atmospheric spectra, the path of the sun's light extends out over east Tennessee and can be considered a hypotenuse of a right triangle. The angle between the path of the sun's light and the ground is $(90^\circ - \theta)$ in Figure 2.4. Therefore, the amount of land that the optical path extends over is just $\sin(\theta)$ for a stratified atmosphere where θ is the zenith angle. The map in Figure 3.4 shows the reach of the optical path as it exits the stratosphere at 50 km in altitude. The larger circle represents an 80° zenith angle (this has been adjusted for the curvature of the atmosphere) and the inner circle represents the amount of land that the optical path extends over at a 60° zenith angle.

The suntracker-2 on top of the Science and Engineering Building is a siderostat configuration. A siderostat is similar to the Kitt Peak McMath Solar Telescope (Pierce, 1964). The center of the rotating mirror that tracks the sun and the center of the secondary mirror are aligned with the earth's polar axis. Therefore, rotating mirror sends the solar radiation down the polar axis towards the South Pole, and the solar radiation is directed to a fixed point in the heavens that is stationary as the earth rotates. Due to the declination of the sun, the only thing that needs daily adjustment is the initial angle of the rotating mirror. Since the declination of the sun changes minutely on an hourly basis, it

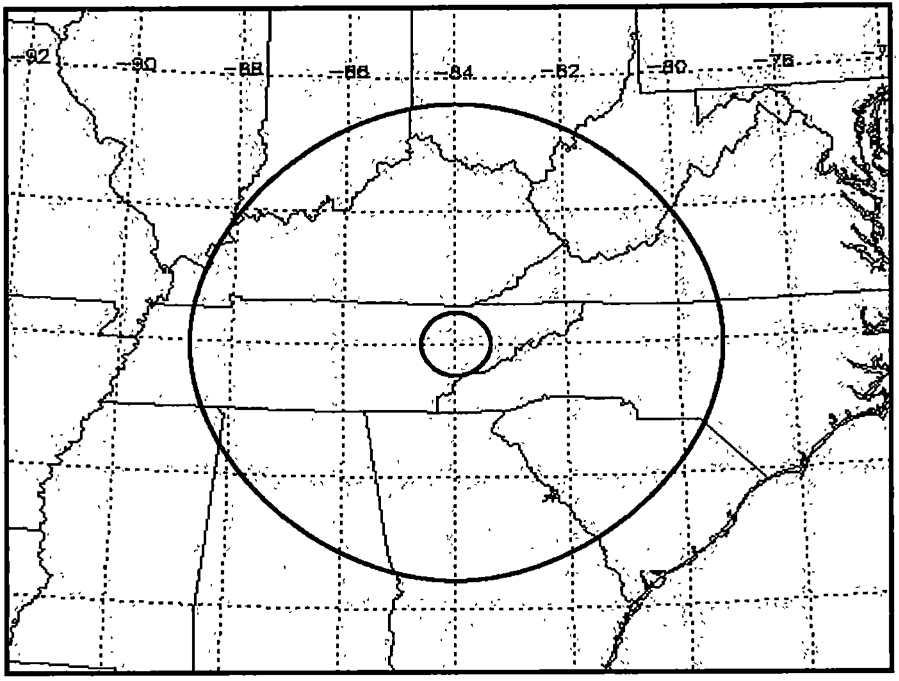


Figure 3.4 Reach of Optical path at 80° and 60° Zenith Angles

is not necessary to adjust the declination of the rotating mirror during the day. The most declination change during any 24-hour period is 0.5°. This maximum change occurs during the equinoxes. Since an average high-resolution scan last for about half an hour, this is acceptable without adjustment during the scan.

Problems with the Suntracker-2

Due to the large distance between the first mirror on the Suntracker-2 and the first focusing mirror of the spectrometer, there are problems with the divergence of the sun's light. First, the edges of the first mirror on the Suntracker-2 will cause Fraunhofer diffraction. This is evident with the alignment procedure, were the visible image almost doubles its size. With Fraunhofer diffraction the spreading of the beam is proportional to the wavelength, thus the infrared beam will spread more than the visible beam. Second, due the large size of the mirrors on the roofs there is bending of the glass due to the weight of the glass. When the sunlight beam reaches the mirror at the top of the optical chase on the Nielsen physics building it has spread out several time it original size. This means the intensity that is gathered by the first focus mirror will be less than if the Suntracker-2 was nearer to the optical chase on the Nielsen physics building.

Path of the sun

Calculation of the path of the sun is important in the design of a suntracker. It is also important to be aware of the region of the earth's atmosphere under examination. The position of the sun can be calculated by a series, as shown in the following

The difference between the ecliptic plane and the equatorial plane of the earth is 23.5°. The solar declination is the angle between a line joining the centers of the sun and the earth to the equatorial plain. It ranges from 23.5° (summer solstice) to -23.5° (winter solstice). The Taylor series

$$\begin{aligned}\delta = (&0.006918 - 0.399912\cos\Gamma + 0.070257\sin\Gamma - 0.006758\cos 2\Gamma \\ &+0.000907\sin 2\Gamma - 0.002697\cos 3\Gamma + 0.00148\sin 3\Gamma)(180/\pi)\end{aligned} \qquad [3.1]$$

is used order to calculate the declination of the sun (Iqbal 1983). Where

$$\Gamma = 2\pi(d_n - 1)365 \qquad [3.2]$$

and d_n are the days since January 1st.

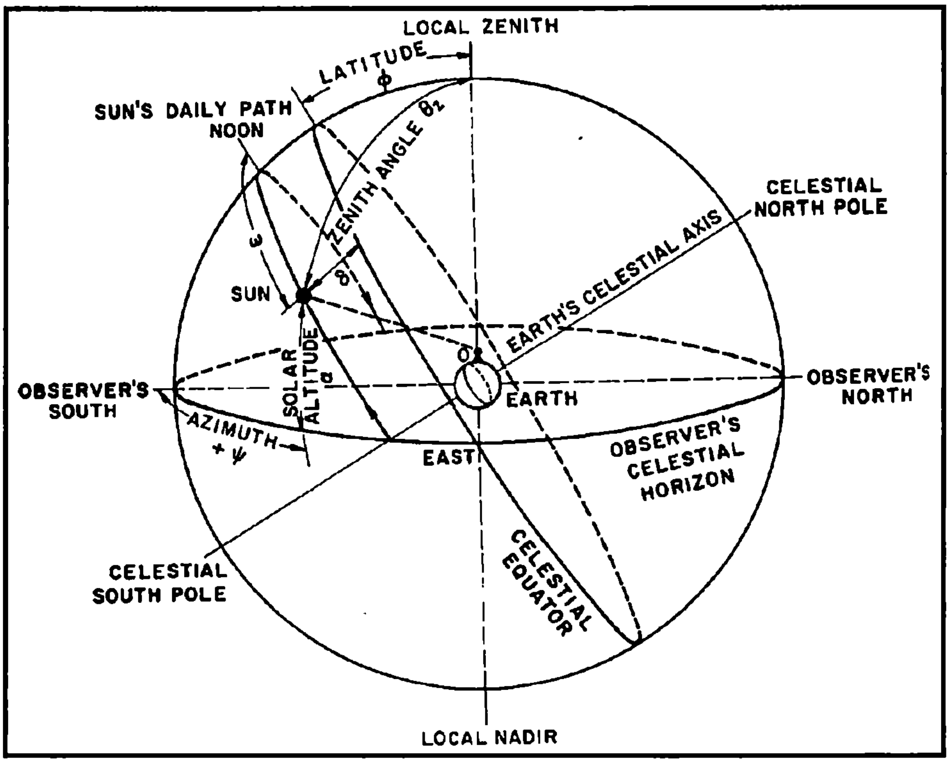


Figure 3.5 Celestial sphere and the sun's coordinated relative to an observer on earth.

Figure 3.5 Shows the path of the sun relative to an observer on earth. The following are the variables used to describe the location of the sun in a chosen frame of reference:

θ_z is the zenith angle.

α is the solar altitude, or solar elevation; α=90°-θ_z.

ω is the hour angle, noon zero, morning positive and 15° equals one hour.

ϕ is the geographic latitude, in degrees, north positive

ψ is the solar azimuth, in degrees, south zero, east positive

δ is the declination of the sun

Using the daily declination of the sun, the latitude of the Science and Engineering Building (35° 57' 25"), the smallest zenith angle and the sun's highest point off the horizon for the summer solstice is 12.5°. Therefore the smallest possible length of an optical path of the sun's radiation at the University of Tennessee is $altitude \cdot \sec(12.5^\circ) = altitude \cdot (1.024)$.

The azimuthal angle and the angle of the sun's altitude are related by

$$\cos\psi = (\sin\alpha\sin\phi - \sin\delta)/\cos\alpha\cos\phi \qquad [3.3]$$

At sunrise, the angles are

$$\cos\psi = -\sin\delta/\cos\phi.$$

The Solar Altitude angle,

$$\sin\alpha = \sin\delta\sin\phi + \cos\delta\cos\phi\cos\omega. \qquad [3.4]$$

This is the equation, along with equation [3.7] that is used in the IDL programs to calculate the zenith angle of the sun.

Time

Solar radiation data are often recorded in terms of local apparent time (LAT). Using standard or daylight savings time proves to be inconvenient. The sun may not be at $(\theta_Z)_{\min}$ when it is noon in standard or daylight savings time, but is always at $(\theta_Z)_{\min}$ when using LAT. This discrepancy is calculated with an equation of time [3.6] and is measured relative to a perfectly uniform terrestrial motion. The information base that is needed in solar observation, like temperature and wind velocity, is given in standard or daylight savings time. Therefore it is desirable to convert local time in to local apparent time or LAT. The equation of time (Iqbal, 1983) is

$$\begin{aligned} E_t = {} & (0.000075 + 0.001868 \cos \Gamma - 0.032077 \sin \Gamma \\ & -0.014615 \cos 2\Gamma \ - \ 0.04089 \sin 2\Gamma)(229.18) \end{aligned} \quad [3.5]$$

E_t in [3.5] is represented in radians without the conversion factor (229.18), which converts E_t into minutes. The maximum error with this series is 0.0025 radians, equivalent to about 35 seconds. Local apparent time for a given standard time can now be calculated by the formula:

$$\text{Local apparent time} \ = \ \text{Local standard time} \ + \ 4(\mathrm{L_s} - L_e) + E_t. \quad [3.6]$$

L_s and L_e are standard longitude and local longitude, respectfully. Standard longitude for the Eastern Time zone is 75° and the local longitude for the Science and Engineering Building is (-83° 55' 30" or -83.925°). The negative longitude denotes the western hemisphere, positive longitude for the eastern hemisphere. The physics building is west of the standard longitude so the longitude correction is negative. The final equation for Local apparent time at the physics building is

$$\text{Local apparent time} = \text{Local standard time} - 35.7\text{minutes} + E_t \qquad [3.7]$$

The incremental value of w or hour angle (noon zero, morning positive and afternoon negative) is 15° per hour using LAT.

Irradiance of the sun on a inclined surface

Irradiance will indicate the rate of solar energy arriving at a surface per unit time and per unit area (Wm^{-2}). As the sun moves through the sky, the angle between it and the secondary mirror obviously changes. Therefore the amount of irradiance also changes. The smaller the angle the more surface of the tracking mirror is pointed towards the sun and therefore it subtends more area. This changes independently from the sun's radiation going through different amounts of the earth's atmosphere. The formula for calculating the irradiance is,

$$F_0 = F_{SC}E_0(\cos\beta\cos\theta_Z + \sin\beta\sin\theta_Z\cos(\psi - \gamma))$$

where F_{SC} is the solar constant and E_0 is the eccentricity correction factor of the earth's orbit.

Alignment

The alignment of the heliostat is more easily done at night. Two different alignments are made. The first is aligning the axis of the Suntracker-2 with the North Star. This is done by looking through the hollow steel shaft and centering the North Star. The tracking mirror mount may have to be removed to do this or one can use a hand mirror.

The second alignment is to make sure the relay mirrors are at their optimal position. This alignment is done with helium-neon lasers. A tripod that sits on top of the Science and Engineering Building roof has a holder for a 0.8 mW helium- neon laser. Another laser is positioned just outside the exit slit of the grating monochrometer and the beam goes backward through the system. The procedure for aligning the heliostat is:

1. The alignment is made easier if one first lines up the returning grating laser with the center of the mirror at the bottom of the conduit.

2. Then, put a white sheet of paper, with a cross in the middle, on the mirror at the bottom of the conduit.

3. Go to the top of the Science and Engineering Building roof and direct the laser to the center of the tracking mirror.

4. Center the laser on the secondary mirror, by adjusting the primary mirror.

5. Using binoculars and while looking at the laser beam's reflection in the secondary mirror, align the laser dot with the cross on the paper, adjusting both the primary and secondary mirrors.

6. After removing the paper, place an aperture near the first focus between M_{1b}' and M_1' of Figure 3.2. There will be a magnified image of the aperture that can be seen with a pair of binoculars from the position of the mirror at the bottom of the optical conduit while looking at the top mirror of the periscope. Align the laser dot on the aperture using the mirror at the bottom of the optical conduit.

The dispersion of the lasers can be a problem. When the heliostat laser reaches the first focus, the laser dot is approximately 4 inches in diameter. That is why it is easier to perform alignments at the foci of the optics.

The 5-Meter Littrow Spectrometer

Introduction

The University of Tennessee Complex Systems Laboratory 5-meter Littrow spectrometer was the subject of extensive study by Don Jennings in his dissertation of 1974. There have also been modifications since 1974 that enable the system to achieve its theoretical resolution limit; these changes are discussed by Dakhil (Dakhil, 1983). The project to integrate the spectrometer to a heliostat was initiated in 1987 through a Research Incentive Grant to the Department of Physics by the University of Tennessee. In order to connect the spectrometer to the Suntracker-2, a 10-inch diameter optical conduit that goes from the roof of the fourth floor of the physics building down to the Complex Systems Laboratory has been constructed. Modifications have been made to the second beam of the 5-meter spectrometer's sample section to bring the sun's radiation in and through the spectrometer. Also, an upgrade to the controlling motor and re-wiring was implemented. The ray trace of the 5-meter spectrometer with the new optics are shown in Figure 3.6

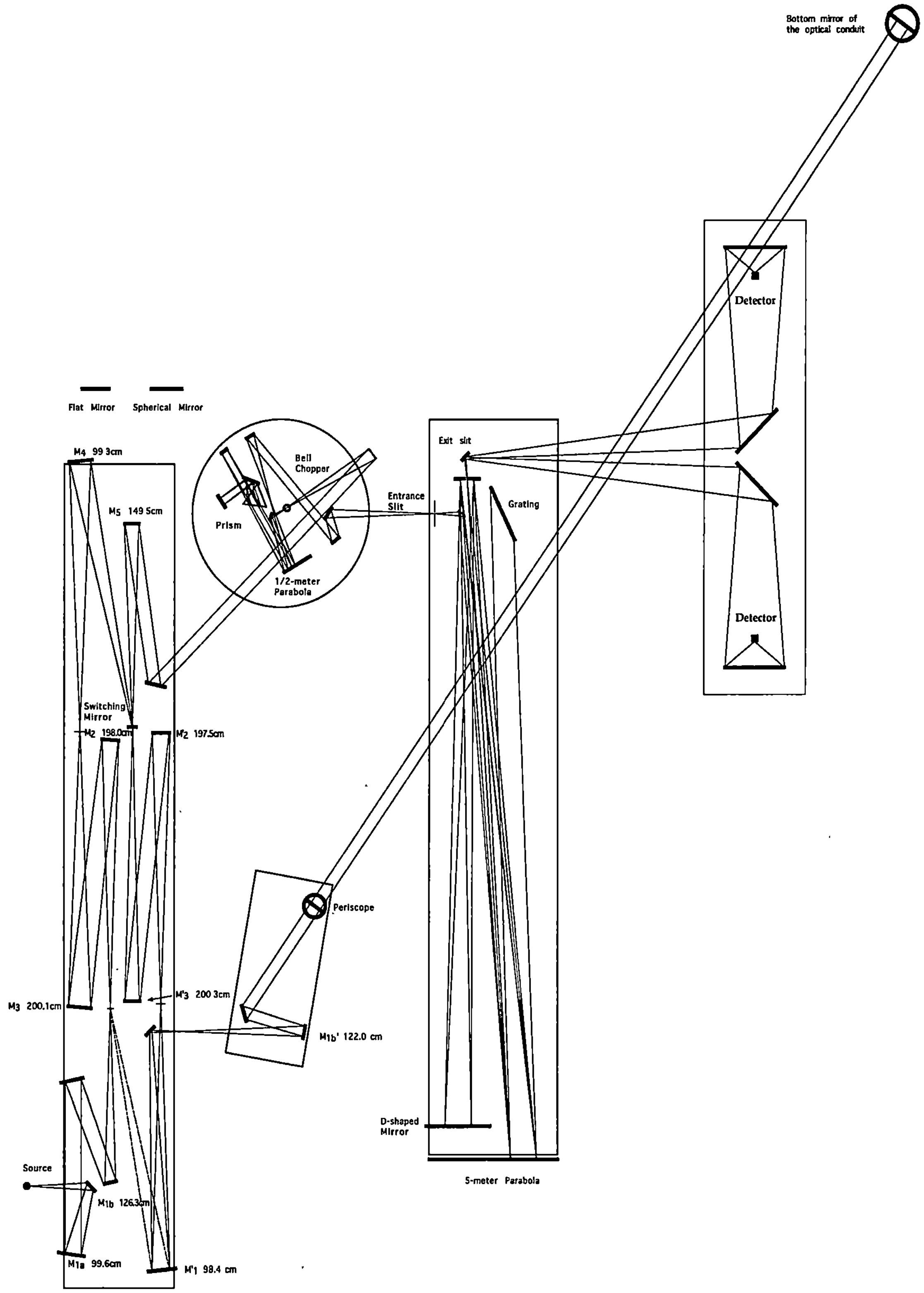


Figure 3.6 The 5-Meter Littrow Spectrometer Integrated with a suntracker.

The spectrometer consists of four main sections:

1) The sample section Figure 3.7

2) The prism predisperser section Figure 3.8

3) The grating monochromator section Figure 3.9

4) The detector section Figure 3.9

There is a vacuum system for the 5-meter spectrometer. All of the sections are connected to a vacuum manifold, which are coupled to the pumping system. There are gate values at all inter-section ports, between each section and the manifold, and between the pumping system and the manifold. There are also small air inlet valves on each section and on the manifold. The system presently functions in the air rather than vacuum for solar studies.

Since the sun's radiation passes through the earth's atmosphere it is not necessary to operate the system in vacuum. Some reasons one might consider to bring the spectrometer down to vacuum are to make it easy to calibrate and to keep the optics clean. It is difficult to keep the pump oil vapor and dust off the mirrors, especially the bottom periscope mirrors.

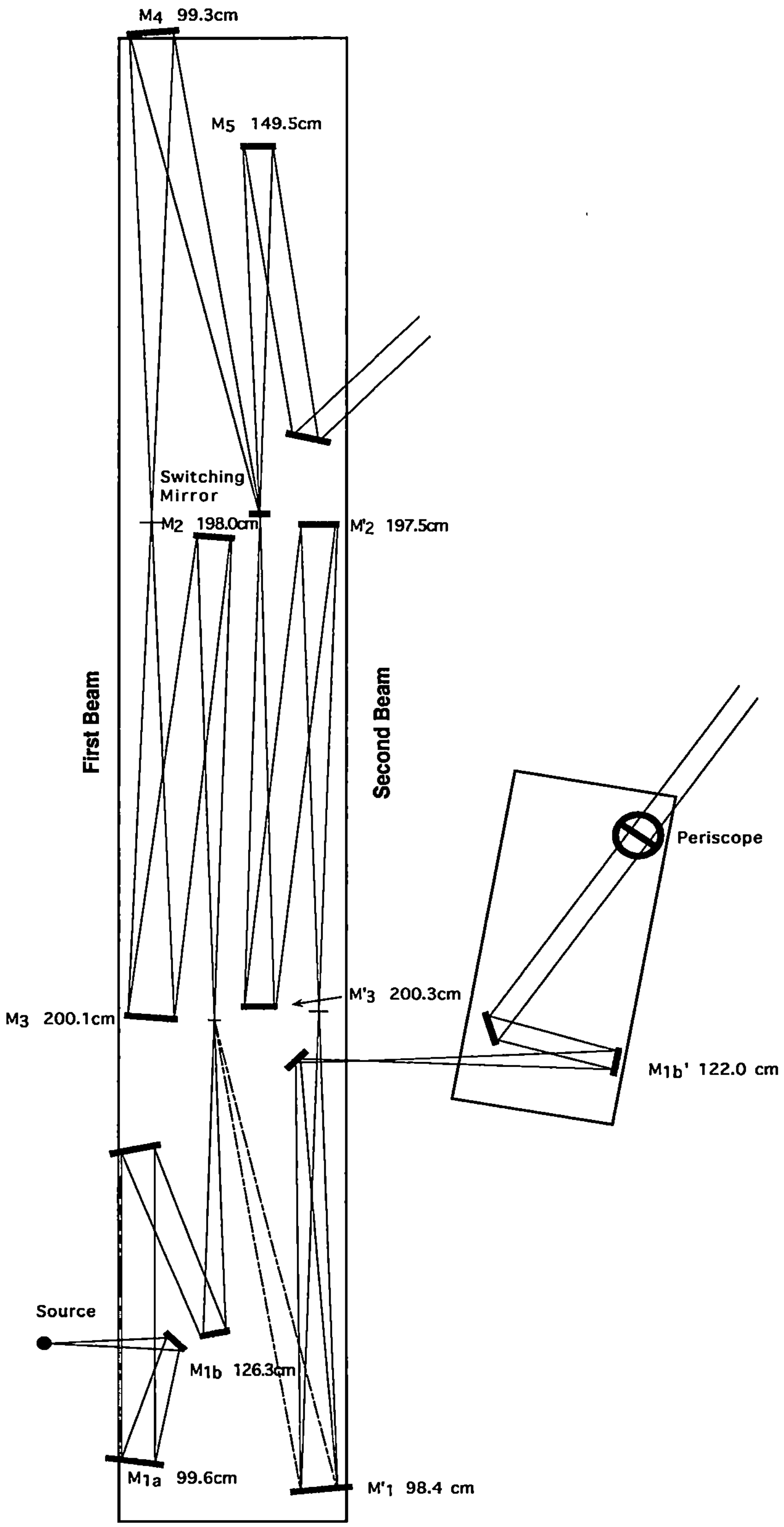


Figure 3.7 The Sample Section.

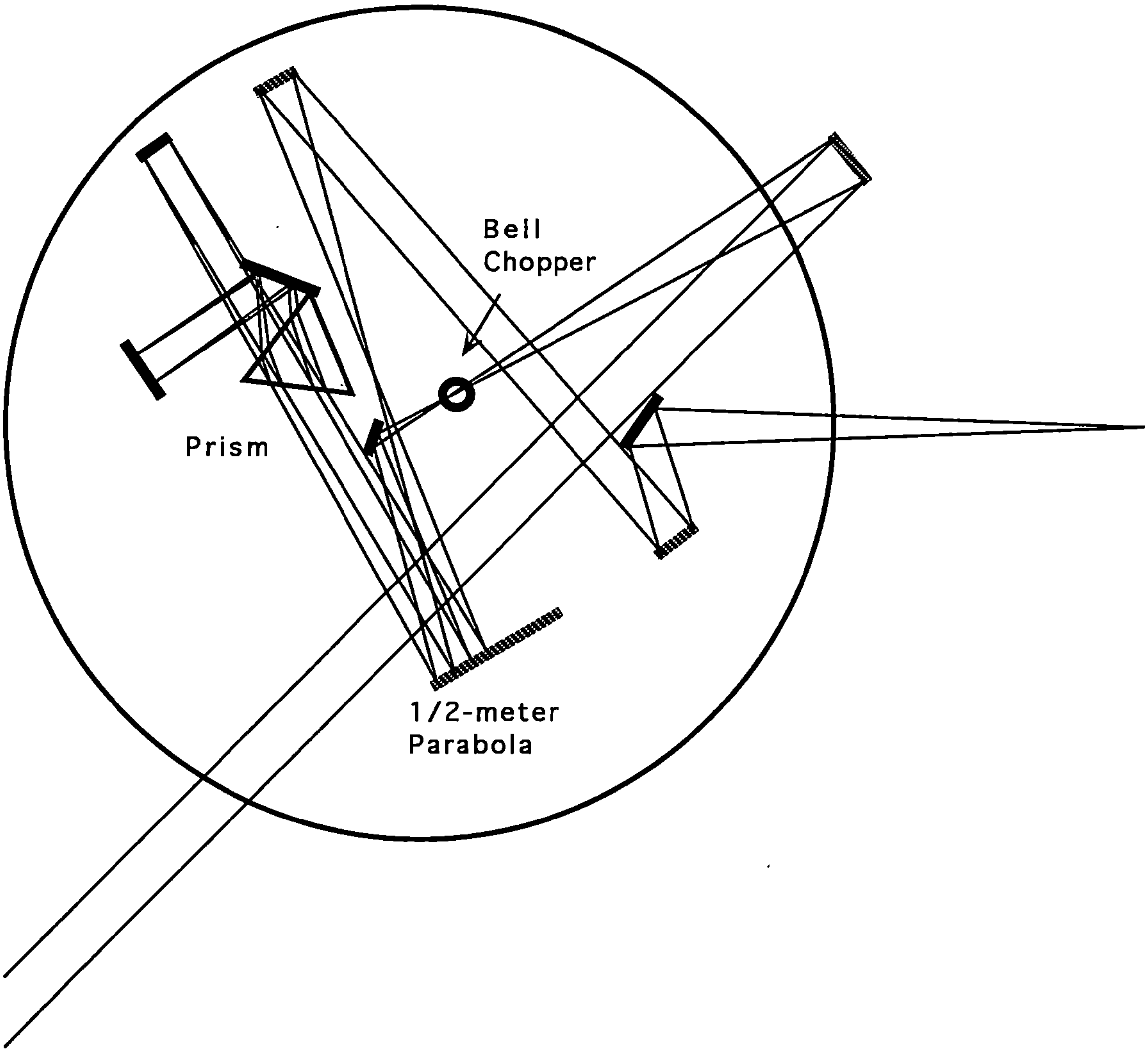


Figure 3.8 The Prism Section

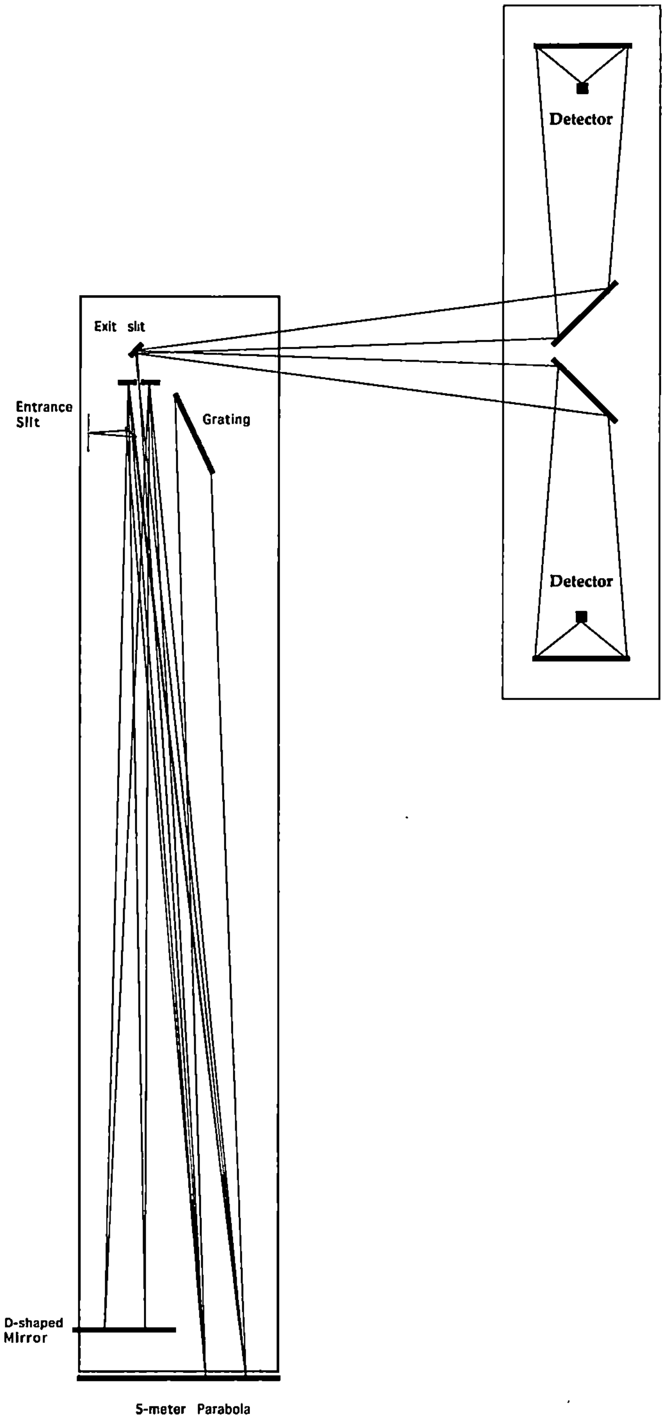


Figure 3.9 The grating monochromator section and the detector section

New interface and upgrades

The 5-meter Littrow spectrometer was inactive for 5 years before the heliostat project was initiated. The original computer system and data acquisition system were replaced. The drive motors, wiring and control interfaces were also upgraded with new technology.

DATA ACQUISITION AND CONTROL

The radiation from the heliostat is brought down through the sample section into the prism section. There, it is interrupted by the chopper at a rate of 1080 Hz and dispersed by the prism predisperser. After being doubly dispersed by the grating monochromator, the signal is detected and fed into a preamplifier. From there the signal goes to a lock-in amplifier and the reference signal comes directly from the chopper (Jennings, 1974; Dakhil, 1983). The amplified signal is then recorded digitally after sampling by an analog to digital (A/D) converter.

The PDP-11 computer acquisition and control system was replaced with an IBM compatible 386DX 25Mhz computer. This computer was fitted with an Alpha Products A-bus motherboard. Two A-Bus cards control the entire spectrometer. An Alpha Products DG-148 Digital I/O card is used for opening and closing the gate valves of the vacuum system. An Alpha Products SC-149 Smart Stepper Controller is used for governing all the stepper motors running the system. One data acquisition card is used, (Alpha Products FA-154 fast 12-bit A/D converter). It is used for sampling the amplified detector signal.

Digital I/O

The controls for the gate values were replaced with solid-state units, so the gate values can be control by computer. The DG-148 card was chosen to control the gate values.

The DG-148 has three 8-bit I/O ports, which may be configured for various modes of operation, input, latched output, strobed input or strobed output. The DG-148 opens and closes the seven gate valves.

Stepper motor card

The SC-149 card is supplemented with PD-123 power driver option. With the PD-123 a 5-amp/phase high torque stepper motor can be powered over long distances. Modifications where made to the PD-123 cards. Notable modifications included the removal of R5 and R6 resistors and the remote relocation of CR1, CR2, CR3 and CR4 LEDs. It should be noted that all grounds on the PD-123 are common. Leads to +5v are connected to the vacant R5 and R6. The schematics of the PD-123 card is in Figure 3.5

Programs

The A-bus system can be programmed through port calls using about any programming language. The decision was made initially to program in Visual Basic. A QuickBasic program that came with the SC-149 was divided in to different sub-routine to make up the Visual Basic program. This program has three main sub-routines; a main sub-routine, that controlled the buttons and the information that goes to the windows; a listening sub-routine that is a loop that constantly checks to see if the SC-149 has any information to communicate; a talking sub-routine that sends commands to the SC-149. All of these sub-routines are also controlled by different forms, buttons, slide bars and other sub-routines (See appendix).

A QuickBasic program that came with the SC-149 was modified by the addition of a data acquisition sub-routine (see appendix).

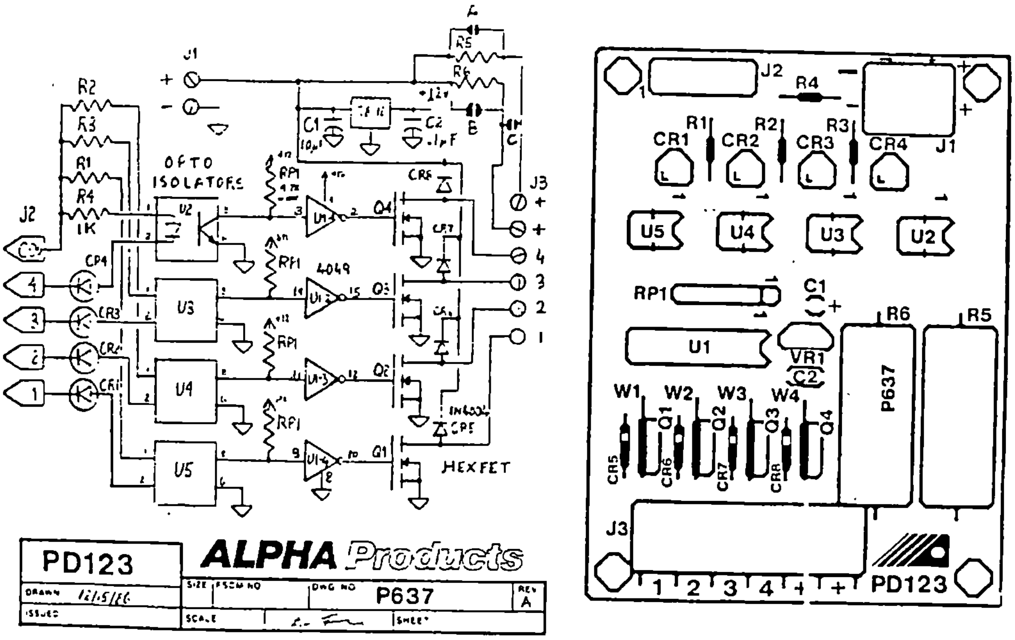


Figure 3.10 PD-123 Card

STEPPER MOTORS

The old servo type motors were replaced on the prism drive, both the entrance and exit slit drives, the course grating drive and on the grating arm drive. They were replaced with Superior Electronic stepper motor. Each motor rotates 1.8 degrees per step at a maximum step rate of 200 steps per second or one revolution per second. A stepper motor also drives the Suntracker-2. The step rate for the Suntracker-2 is about 80.2 steps/sec.

New optics

The original design of the spectrometer has a second beam for running different experiments simultaneously. The idea was to use the second beam to bring the radiation in from the suntracker. This way the sun's radiation could not only be analyzed, but it could be used as a source in which you could put a gas cell in one of the focuses of the second beam. Analysis of infrared regions that are unaffected by atmospheric absorption can be scanned for a particular molecule. The main advantage is to use a 6000K black body source, the sun, instead of a 2700K source, the carbon rod. The decision was made to bring the heliostat radiation in through the port hole 193 cm from the west side of the sample section optical table on the south side of the sample section. If the system is ever brought down to vacuum the steel cover of the port could be fitted with an infrared window. The window would only have to be approximately 5 cm in diameter, because it is near the first focus of the second beam. An infrared filter could be used as a window.

DESCRIPTION OF THE NEW OPTICS

The Suntracker-2 brings the sun's radiation to a secondary mirror that is aligned with the earth's polar axis. The secondary mirror relays the beam to a third mirror that reflects it down to a fourth mirror on top of the fourth floor roof of the Nielsen Physics Bldg. The fourth mirror directs the beam through a 10-inch diameter optical conduit into the laboratory. There it is picked off by a mirror at the bottom of the conduit and sent horizontally across the lab to a periscope. The bottom pick off mirror mount at the end of the optical conduit was designed to send the sun's radiation to any point in the lab. An optical table has been constructed that is on the south side of the sample section. The table holds a periscope, flat mirror, and the first spherical mirror of the second beam. The periscope drops the beam 88.4 cm to the same level as the beam in the sample tank Figure 3.11. The beam is then sent to a optical flat that in turn sends it to a 122 cm focal length spherical mirror. This mirror collects the sun's energy at f/8. The beam is then sent to a optical flat that brings it to mirror M_1' Figure 3.11.

SECOND BEAM

The spectrometer was originally designed to use two beams simultaneously by switching between them. This feature was never implemented. The optics on the second beam reverses the optics of the first, starting after the 125 cm M_1 mirror Figure 3.7 and moving all the way to the switching mirror. The original design used a system of three mirrors to obtain the radiation from the source to the second beam. In order to bring a collimated source into the second beam, matching the 125 cm M_1 mirror and spatially matching the focal length of it and M_2' is all that is needed as shown in Figure 3.7. The pupils, aberrations and foci are all be approximately the same as the original design.

Bottom Conduit Mirror

The mirror at the bottom of the fourth story conduit has a 23.4 cm x 20 cm flat. The mirror mount can rotated and angled up and down in order to direct the radiation anywhere in the lab. It is design to be easily disassemble and cleaned.

New Optical table

A 23x53 inch optical table was constructed with 3/8 inch threaded holes. The holes are 5 1/2 inches apart and are used for tie downs. The table stand is designed to sit on the main platform. The main platform of the spectrometer sits on six air baffles in order to absorb vibration. The new optical table holds the periscope, flat and the 122 cm focal length mirror. The periscope is designed to bring the center of the radiation over the detector tank and down 88.4 cm to the height of 23.7 cm above the optical table Figure 3.11. The periscope is 50 inches tall and has a base 12 inches square.

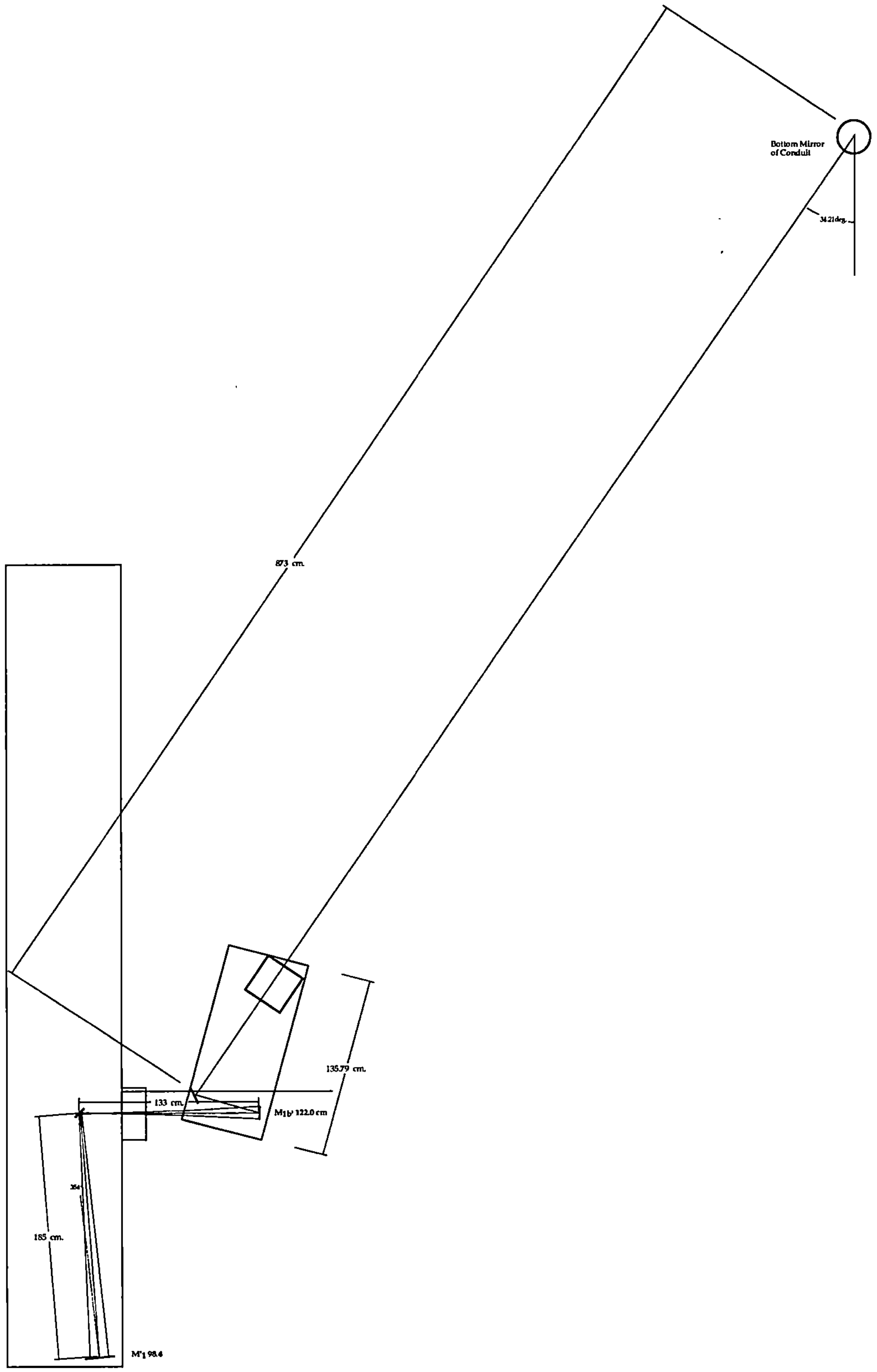


Figure 3.11 New Optics

Spherical mirror

The spherical mirror M_{1b}' has 122 cm focal length and a diameter of 6 inches. It is the first focus of the new optics (Fig. 3.6).

Switching mirror

The switching mirror is on a linear drive axis. The drive axis is not threaded so there is nothing to strip out. The angle of the axis is perpendicular to the normal of the switching mirror. This puts the switching mirror in precise position consistently

Calculations

The new optics together with using the sun as a source required new calculations of the optical characteristics of the 5-meter spectrometer. Pupils, aberrations and the curvature of the slit will cause degradation of the resolution. Calculations were made using the sun as a source instead of the carbon rod and new optics in the second beam instead of the original second beam design.

f-number or f/#

The first spherical mirror M_{1b}' in Figure 3.11 it is 6 inches in diameter, and its' focal length is 48 inches. The f/#, which is proportional to the inverse of the irradiance is given by

$$\mathrm{f/\#} = f/d.$$

(Hecht, 1987), where *d* is the diameter and *f* is the focal length. M_{1b}' collects the sun's radiation at f/# = 8.

Pupils

The diameter of M_{1b}' is the first aperture stop or first exit pupil for the second beam. The aperture stop of the whole system is the grating itself. Since the grating is the most expensive element in the spectrometer, its size should be the limiting aperture in the spectrometric system. The entrance pupil of the whole spectrometer is the final image of the grating at the first focus mirror. The first focus mirror M_{1b}' images the grating as one goes backwards though the spectrometer. Since the grating is the limiting aperture of the spectrometer this image is defined as the entrance pupil. The grating is the limiting aperture of the whole spectrometric system. The intermediate pupils and the entrance pupils are shown in

Table 3.1. The table is fairly self-explanatory. The 'Dis. Between' column is the distance between the mirror and object in the row above and the mirror or object in the row below. The 'virtual' column is the distance the virtual images are in front of the mirror on the same row. The 'Real' column is the distance the real images are in front of the mirror on the same row. The 'Mag.' is the magnification of the object.

In order for the grating to be completely filled with radiation, the real images of the grating, going backward through the optical system, must not be larger than the size of the beam at the point at which the real image is formed.

Table 3.1 Pupils using the grating as the limiting aperture.

	FOCI	Objects	Dis. Between	Virtual	Real	Mag.	Diag. of Grating	side in cm.
Grating							28.73	20.31
			475.00					
5-meter	500.00	475.00		-9500.00		20.00	574.55	406.21
			600.00					
mirror #9	100.00	10100.00		101.00		0.01	5.75	4.06
			100.00					
mirror #8	50.00	-1.00			0.98	-0.01	5.63	3.98
			100.00					
1/2 M Para	50.00	99.02			101.00	1.02	5.75	4.06
			200.00					
1/2 M Para	50.00	99.00			101.02	1.02	5.86	4.14
			130.00					
mirror #6	80.00	28.98		-45.44		-1.57	9.19	6.50
			358.19					
mirror #5	149.55	403.63			237.57	0.59	5.41	3.83
			351.60					
mirror #3'	200.25	114.03		-264.82		-2.32	12.57	8.88
			187.00					
mirror #2'	197.55	451.82			351.03	-1.80	9.76	6.90
			396.00					
mirror #1'	98.40	44.97		-82.80		-1.84	17.98	12.71
			324.30					
mirror #1b'	122.00	407.10			174.21	0.43	7.69	5.44

Note that almost all real images of the grating are very near the following mirror from the imaging mirror. The first real image formed by the parabola is at a collimated part of the beam. The other real images are small compared to the size of the beam.

The only marginal situation is the fourth real image of the grating. We use a collimated beam size of approximately 15 cm. The fourth pupil is imaged 237.57 cm off mirror M_5' which is 88.02 cm in front of the focus of M_3'. The beam size goes from 15 cm at M_3' to 1.1 cm at the focus. Instead of using 1.1 cm, we approximate the size of the sun's image as a point. Then using similar triangles to compute the beam size at the fourth pupil, we will observe that the pupil is small enough to fit inside the beam. As follows, 7.5 cm is to 200 cm as (beam size/2) cm is to 88.02 cm. The beam size at the fourth pupil is ~6.72 cm. The diagonal of the image of the grating at the fourth pupil is 5.41 cm Table 3.2, which is smaller than the beam size. Another marginal situation is the sixth pupil. The sixth pupil is 351.03 cm off M_2'. The beam diameter is 11.7 cm and the pupil diagonal is 9.76 cm, which is smaller than the beam size. The seventh or entrance pupil is 174.21 cm off M_{1b}', which is in the collimated part of the beam.

Curvature of slits

In order to calculate the curvature of the slits using the sun as a source, it is necessary to know the size of the sun's image at the predisperser and monochromator entrance slits.

The sizes of the virtual and real images of the sun are calculated at every focus, including the slits.

Just as a window will cause a pincushion distortion when you look through it at a skewed angle, the prism causes slit image distortion. The distortion of a straight slit is a parabola that curves toward the short wavelength side of the spectra.

An arc of a circle can approximate this parabola. Wadsworth-Littrow optical arrangement was implemented in the prism monochromator in 1977 (Dakhil, 1983). The slit image distortion depends on the angle of incidence on the prism and the focal length of the collimator. Both of these factors are invariant for a given prism in the predisperser using the Wadsworth-Littrow optical arrangement.

When the prism is set at minimum deviation, the formula for this circle is:

$$r_{cp} = \frac{fn^2}{4(n^2-1)} \cot \theta_1' \qquad [3.8]$$

Where f is the focal length of the imaging optics, n is the index of refraction and θ_1' is the angle if incident on the prism. The index of refraction for the LiF prism at 2450 cm-1 is approximately 1.35. The angle of incidence is approximately 30° and the focal length of the imaging optics is 1 meter. The radius of curvature is 96 cm. The size of the sun's image is only 0.64 cm at the entrance slit. So this curvature of the prism entrance slit is not of great concern.

Table 3.2 The Sizes of the Sun’s Image Throughout the 5-meter Spectrometer

	FOCI	Objects	Dis. Between	Virtual	Real	Magnification	Dia. of Sun cm.
Sun							1.40E+11
			1.50E+13				
mirror M1b'	122.00	1.50E+13			122.00	0.00	1.14
			321.20				
mirror M1'	98.40	199.20			194.46	-0.98	1.11
			396.75				
mirror M2'	197.55	202.29		8425.92		-41.65	46.35
			197.00				
mirror M3'	200.25	8622.92			205.01	-0.02	1.10
			351.60				
mirror M5	149.55	146.59		-7403.80		50.51	55.66
			358.19				
mirror M6	80.00	7045.61			80.92	-0.01	0.64
			130.00				
Parabola	50.00	49.08		-2670.94		54.42	34.79
			200.00				
Parabola	50.00	2470.94			51.03	-0.02	0.72
			100.00				
mirror M8	50.00	48.97		-2370.94		48.42	34.79
			100.00				
mirror M9	100.00	2270.94			104.61	-0.05	1.60

The grating also distorts off -axis radiation. It curves the image of the entrance slit of the monochromator just like the prism but towards the longer wavelengths instead of the shorter. Figure 3.6 illustrates the final image of the entrance slit on the exit slit. The formula for calculating this shift is derived and discussed in Jenning's dissertation (Jennings, 1974). The displacement of the top and bottom points of the entrance slit in a double pass monochromator is

$$\Delta x = (\alpha - \delta)^2 2\tan\theta. \qquad [3.9]$$

where Δx is the lateral displacement. The collimator axis and the grating normal define the horizontal plane. The angle α is the angle which the incident and diffracted rays make with the horizontal plane. The angle δ is the angle that the center of inversion (the axis of the D-shaped mirror) makes with the horizontal plane. The angle θ is the angle that the grating normal makes with the collimator axis. Both the top and bottom points are displaced by the same amount in a double pass system.

The size of the sun's image at the entrance slit of the monochromator is 1.6 cm. So Δx at blaze or 65° is 44 mm. Since the diffraction limit of the slits is 110 mm at 2500 cm-1. The lateral slit curvature affects the resolution negligibly.

Data Acquisition

Data acquisition for this project started in December of 2000 and continued on until October of 2001. In all, 227 scans where taken over a 10-month period. Each scan on the average took 15 to 20 minutes and had a range of wavenumbers of approximately 10 to 20 cm^{-1}. The time constant or 1/bandwidth on the average was 400msec to 1.25 second. This kept the noise at an acceptable level, but increased our scan time. The movement of the grating stepper motor on the average was 143 steps/sec or 0.70 revolutions/second. The gearing mechanism reduces the stepper motor down to 2.16 X 10^5:1. This allows the grating to turn at 0.027 cm^{-1}/second at ~ 2500 cm^{-1}. A thorough inspection of the spectra showed a skewing effect at stepper motor speeds faster than 160-170 steps/second. Although, Jennings in his dissertation show that the instrument response function was slightly skewed. The effects of faster scan speed were more than slightly skewed. The stepper motor speed of choice is 143 steps/second (Jennings 1974).

The weather in east Tennessee is not as conducive to solar atmospheric spectroscopy as a desert, but in the spring and fall months in east Tennessee, is when the large cold and dry air masses come down from Canada. This is usually good data acquisition weather. In order to acquire high quality data the sky must be clear of clouds and the atmosphere must be fairly dry. Some of the better days that data was taken are: January 21, 2001; February 18, 2001; March 7, 2001; March 25, 2001; March 27, 2001; April 25, 2001; June 18, 2001.. The data taken in June had some H_2O lines that interfered with the calculation of the column abundances of N_2O, but the N_2O lines that the H_2O lines interfered with were not taken into consideration when calculating column abundances. Problems with other data will be discussed in later section.

Data Acquisition Procedure

The 5-meter spectrometer is controlled from a station on the north side of room 108. It contains the computer terminal, lock-in amplifier and stepper motors speed indicators.

The instrument functions, which are controlled from this station, are:

- Grating position drive
- Grating scan drive
- Prism position drive
- Slits widths drives
- Beam switching drive
- Vacuum system valves

These different drives and valves are discussed in Jenning's dissertation (Jennings, 1974).

The procedure assumes that you are already on a prism order. The procedure for taking atmosphere spectra is as follows

1. Run “5meter4” program on the computer.
2. Power up stepper motor cards
3. Adjust declination and right ascension of primary heliostat mirror on roof.
4. Start heliostat and approximately align the sun’s radiation with the middle of the bottom conduit mirror.
5. Set speed of heliostat motor to 80.6 for Suntracker-2 configuration.
6. Turn on detector
7. Turn on chopper
8. Turn on lock-in amplifier
9. Adjust the speed to the heliostat to get the maximum signal on lock in-amplifier read out.
10. Start grating drive.
11. Push ‘q’ to get out of SC-149 control mode
12. Push ‘t’ to take data
13. Set sampling parameter

14. Name data file, after you name the data file and press return, the program will start taking data.

15. To stop taking data press 'q'

In order to rotate the prism to a specific order:

1. Rotate the prism until you see the red (920 nm) on the entrance slit of the monochrometer.

2. Take a prism profile by starting at the visible and scanning the infrared for about 20 minute at a motor speed of 200.

3. Print out the prism profile and scale the time on it.

4. Go back toward the visible and stop on the desired order by using the time scale.

Once the data had been taken, the programs to read and graph data were KaleidaGraph and Origin.

Chapter Four

Methods

The N_2O project involves the acquisition of the atmospheric spectra of the $2\nu_1$ band of N_2O over an extended period of time at good signal to noise ratio. The data needs to be computer modeled using existing N_2O data (in this case, HITRAN data discussed below). The model spectrum of the N_2O is compared to the observed spectra and parameters are adjusted to bring the modeled data into register with the experimental data. The goal of this step in the analysis is the retrieval of N_2O column densities given an assumed atmospheric temperature profile and other trace gas composition. Retrieving column densities of N_2O over the time period of the data acquisition should allow us to discuss any time variation.

Modeling data

Many people involved with atmospheric physics and related field are involved in atmospheric radiative transfer or transmittance and radiance calculations. There are many applications varying from remote sensing, satellite meteorology, measurements of atmospheric gases and development of climate models. These calculations are carried out over a large spectral range, from UV to microwave. The physics that must be included has resulted in very complex and large computer models. At the heart of each of computer models lies the analysis of the effect of absorbing and emitting gases on radiation.

Some of the most notable computer programs that model atmospheric spectra are, MODTRAN, LBLRTM, FASCODE, GENLN2 and RFM.

All these programs require some kind of database with the necessary information needed to calculate the individual band or line strengths of individual molecules in the earth's atmosphere. The database used in all these programs is HITRAN.

HITRAN

The high-resolution transmission molecular absorptions database (known under the acronym HITRAN) is a compilation of spectroscopic parameters from which a wide variety of computer simulation codes are able to calculate and predict the transmission and emission of radiation in the atmosphere. This database is a prominent and long running effort established by the Air Force at the Air Force Geophysics Laboratory (AFGL) in the late 1960's in response to the requirement of a detailed knowledge of infrared transmission properties of the atmosphere (Rothman, Gamache et al. 1987).

HITRAN has evolved over the last 32 years to included new parameters and new trace molecules in the earth's atmosphere. The version used for our data analysis is HITRAN2000. A more upgraded version of the methane data was used instead of the HITRAN2000 version. The parameters included in the HITRAN2000 are shown in Table 4.1

The authors computer program written to model the atmosphere uses six parameters of the HITRAN database. They are, reference numbers for the molecule and its' isotopes, transition frequency, intensity, transitional probability squared, air-broadened width and the temperature dependence of air width.

The other parameters are useful in atmospheric modeling, but the effects of these parameters are negligible for nitrous oxide and our wavelength section of the spectrum. Some of the reasons for not using these parameters are: the self-broadening halfwidth are of course less accurate the air-broadening halfwidth in our case because the molecules are in the air, there are no 'hot bands' for nitrous oxide and methane (hot bands are absorptions lines whose lower energy state is not the ground state), the two largest absorbers, in our region; there are no significant pressure shifts; and the accuracy indices are all zero for nitrous oxide and methane.

Table 4.1 The parameter included in the HITRAN2000

Molecule number	A reference number use to indicate the molecule in question.
Isotope number	A reference number use to indicate isotope of the molecule in question.
Transition frequency (v_0)cm^{-1}	The frequency of the transition.
Line Intensity - S	The intensity of the transition in $cm^{-1}/(molecule \bullet cm^{-2})$ @296K
|R|2	The transitional probability Squared in Debye^2
Air-broadened width	(b_L) The Air-broadened halfwidth (HWHM) in cm^{-1}/atm @296K
Self-broadened width	(b_L) The Self-broadened halfwidth (HWHM) in cm^{-1}/atm @296K
Lower-state Energy – (E”)	Indicating ‘Hot bands’ in cm^{-1}
Temperature dependence of air width. – (n)	Coefficient of temperature dependence of the air-broadened halfwidth.
Pressure shift - (δ)	Air-broadened pressure shift of line transition in cm^{-1}/atm @296K
Upper vibrational quanta -- (v’)	The upper state global quanta index
Lower vibrational quanta – (v”)	The lower state global quanta index
Upper local quanta - (Q’)	The upper state local quanta index
Lower local quanta – (Q’)	The lower state local quanta index
Error codes	Accuracy indices for frequency, intensity and air-broadened halfwidth
Reference codes	Indices for table of references corresponding to frequency, intensity and halfwidth

Atmospheric modeling – this study

A short introduction to the accepted techniques of atmospheric spectral modeling should enable the reader to better understand the details, which follow in this chapter. The model of the atmosphere can be likened to an onion -- with the layers of the onionskin similar to layers of the atmosphere. While this seems a naive model, it turns out to be useful for nearly all of the scientists working in atmospheric spectral studies.

If we assume local thermodynamic equilibrium (LTE) for each layer and all other parameters are known, the calculation of the total column amounts of N_2O can be determined. If the strengths, temperature and path length are all known quantities the area or line width of the absorption line is proportional the total column amount (Rao 1976).

To model the earth's atmosphere it was broken up into 6 different layers of 5 km each for a total of 30 km. A 30 km atmosphere is chosen because 99% of the mass of the atmosphere lies below 30 km and due the photo-dissociation of N_2O in the stratosphere, the concentration of N_2O drops off drastically above 20 km. The concentration of N_2O at 30 km is 1/10 of its value at 20 km.

Thinner layers do not increase the precision of the fit between the observed and calculated spectra according to a conversation the author had with Frank Murcray at the University of Denver while on a fellowship at NOAA in Boulder.

The program 'atm_model.pro' reads in the save file of 'lineidenity.pro' and takes the sampling rate of the observed spectra and uses the same sampling rate to compute the synthetic spectra. The average pressure, temperature and volume-mixing ratio are computed for each layer. The volume mixing ratios come from an AFGL paper that gives different mixing ratios of atmospheric molecules at different latitudes and seasons. It was decided to employ the U.S. standard atmosphere as a starting point and make adjustments according to the data recorded. Different parameters are entered into 'atm_model.pro' from the day and time the scan was taken. These parameters are barometric pressure, ground temperature and secant values of the slant path.

The parameters used to calculate the optical density in our model are pressure and length. The strengths are calculated with data from the HITRAN database. The strengths in Hitran are given in cm molecule^{-1}. We must convert molecule into atmosphere. In order to convert strengths into cm^{-2} atm^{-1} at 300K, we use a conversion factor of 2.4464 x 10^{19}. These strengths are compared and confirmed with the calculated and observed strengths from A. Levy's paper (Levy, Lacome et al. 1984). The modeling program adjusts the strengths of the line according to temperature. The expression that HITRAN uses for line strength is given in equation [2.27]. The following expression is used to calculate the line intensity $S_A^B(T)$ from the HITRAN quantity $S_A^B\left(T_{ref}\right)$,

$$S_A^B(T) = S_A^B(T_{ref}) \cdot \frac{Q(T_{ref})}{Q(T)} \cdot \frac{\exp(-hcE_A/kT)}{\exp(-hcE_A/kT_{ref})} \cdot \frac{1-\exp(-hc\nu_{AB}/kT)}{1-\exp(-hc\nu_{AB}/kT_{ref})} \qquad [4.1]$$

All constants and variables are defined in Line strengths on page 25.

The broadening coefficients are calculated from the HITRAN database. The Doppler coefficients or half widths are calculated with equation [2.36]. The atomic weights for N_2O and CH_4 are 44.0128 and 16.0465 amu, respectfully. The pressure-broadening halfwidths decrease with altitude as the pressure drops. Equation [2.31] is used to calculate pressure-broadening halfwidths. The pressure-broadening, or air-broadening halfwidths are read from the HITRAN database for each absorption line. But the pressure-broadening half widths also depend on temperature.

The relationship is:

$$b_L(T) = b_L(T_0)\left(\frac{T_0}{T}\right)^n \qquad [4.2]$$

The exponent (n) is the coefficient of temperature dependence of the air-broadened halfwidth in the HITRAN database. This equation [4.2] is computed after equation [2.31]

Slant path calculations are carried out using $\sec\theta$, see Figure 2.4, and do not take into consideration the index of refraction of the earth's atmosphere. This is not a serious source of error for values of θ less than 60°. (Edwards 1992).

The Voigt profile is used to calculate the line shape function. A computer program written in C was translated into IDL and used to calculate the Voigt profiles at all layers.

Once the model is computed it is overlaid with the observed spectra and the difference between the two is taken. The associated errors are then calculated and graphed. Then a variable multiplies the U.S. standard atmosphere mixing ratio for N2O to change the simulated abundance of N_2O in order to minimize the difference.

baselinerough.pro

Research systems, in the 5.x versions of IDL, have developed a Widget Toolkit and Object Graphics in order to make the visualization of data easier. IDL Widget Toolkit and Object Graphics is a set of functions and procedures that can be used to create graphical user interface (GUI) programs. In order to compute baselines and identify absorption lines, it is necessary to visually inspect, mark and manipulate the atmospheric data. IDL Widget Toolkit and Object Graphics allow us to do this.

The author has written approximately 2500 lines of IDL code for this project. Some of their main tasks are to calculate the baseline, calculate wavenumber from data points and calculate areas of the atmospheric absorption lines; identify atmospheric absorption lines, create atmospheric models and calculate synthetic spectra from that model, and more. A brief synopsis of the each program will be given in the following.

The baseline of the N_2O spectrum is difficult to ascertain. Because the source, the sun, cannot be seen without the absorbing gases, the baseline cannot be ascertained with great accuracy. This is one of the advantages of limb spectroscopy from orbital platforms such as ATMOS. The satellite can calibrate the baseline by looking at the sun above the atmosphere, then take data through the atmosphere.

The baseline for the monochromatic transmission function is the line of 0% absorption and 100% transmission.

Monochromatic transmission function: $T(\nu)=\dfrac{I(\nu)}{I_0(\nu)}$,

whereas for the Monochromatic absorption function it is the opposite, i.e.

Monochromatic absorption function: $A(\nu)=1-T(\nu)$,

The baseline is difficult to ascertain since the raw spectral data from the 5-meter Littrow spectrometer exhibit long period variations in the baseline signal. This is mainly due to the entrance slit to the monochromator and the predisperser.

The reason for a predisperser is to separate the orders of the grating, so that they do not overlap. There are no overlapping orders using the LiF prism and the 31.5-lines/mm grating at spectra ranges around 2500 cm^{-1}. The resolving power of the prism predisperser is $R=\lambda/\Delta\lambda=2t(dn/d\lambda)$ (Jennings 1974). The entrance slit to the predisperser is in the bell chopper and the exit slit of the predisperser is the same slit as the entrance slit to the monochromator Figure 3.8. The resolving power of the predisperser dictates the spectral range of the monochromator for a stationary prism. The monochromator scans the radiance that comes through the entrance slit of the monochromator. As the monochromator scans the radiance close to either edge of the entrance slit, a loss of energy is observed at that frequency. This, in turn, exhibits a downward variation in the baseline of the spectra. An example if this is data taken on March 7th, 2001 in Figure 4.1

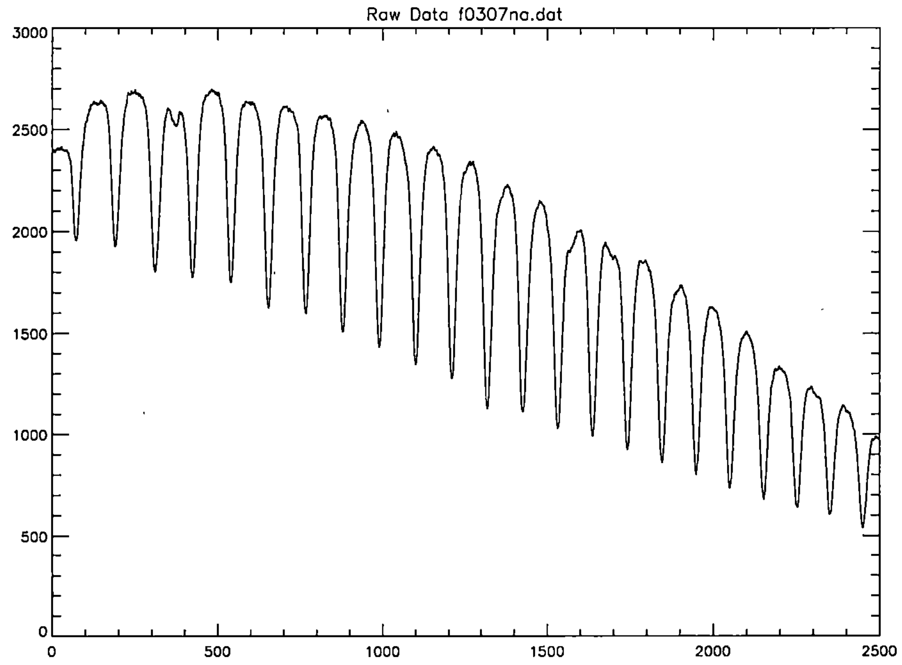


Figure 4.1 Variation of the baseline due to the predisperser and the monochromator entrance slit.

It is seen in Figure 4.1 that the energy has dropped off by a factor of two during the scan. In this particular instant, Figure 4.1, we are starting the scan towards the middle of the entrance slit and scanning towards the higher wavenumbers.

In order to obtain the total absorption line or zero transmittance line, in the beginning of each data session, an obstruction, was put into the beam of the sun's radiation and the zero adjustment on the lock-in amplifier is tuned to read exactly zero on the A/D converter. Once the total absorption level is determined it is possible to roughly calculate the baseline of the spectrum using the troughs between the absorption peaks.

It is known from other atmospheric data, that the baseline of the $2\nu_1$ band of N_2O is close to the bottoms of certain troughs. The IDL program 'baselinerough.pro' is used to pick key points near certain troughs in the spectra. These troughs are parts of the spectrum with known low concentrations of small non-N_2O absorption lines (Park 1987). Some of these troughs are between P32-31, P28-27, P24-23, P21-22, P16-15 and P5-4 of the P-branch of the $2\nu_1$ band of N_2O. Points near these troughs were consistently chosen for the baseline approximation for each data set. The points are then fitted with a line function. The line fitting routine selected is a spline curve fitting routine. The points on the baseline curve are then subtracted from the raw data points then divided by the same point on the baseline curve. The two are subtracted to get the spectrum into transmission space, see Figure 4.2

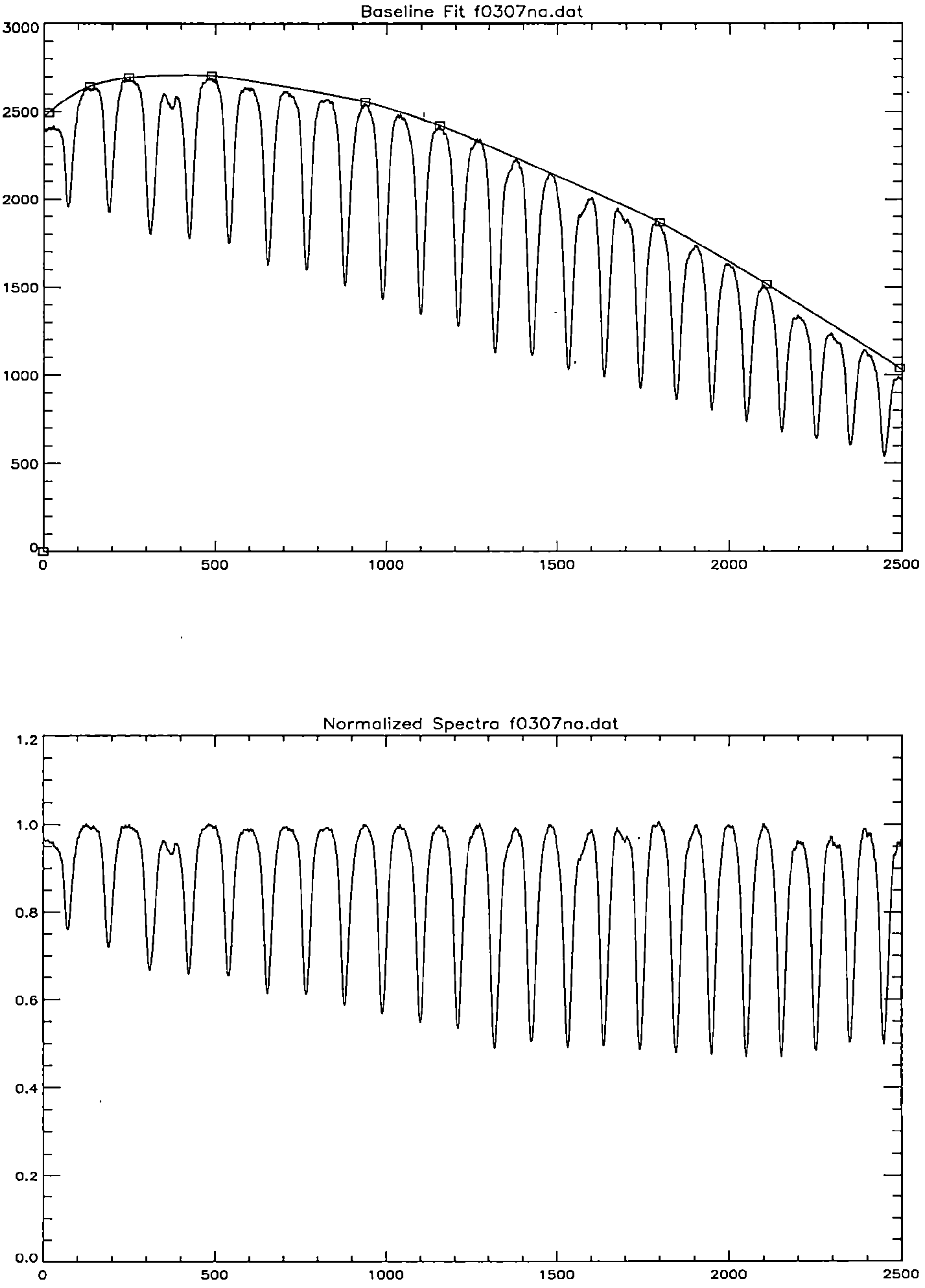


Figure 4.2 Baseline correction of program 'baselinerough.pro'

The IDL program ‘baselinerough.pro’ has been written in order normalize the baseline of the raw atmospheric spectra. Other IDL programs fit the observed data to modeled atmospheric spectra.

The program ‘baselinerough.pro’ operates further to determine the location of the peaks of each line. The peaks of the absorption lines are then calculated using a routine that reads in the normalize spectrum and monitors a change in the sign of the derivative. After baselinerough.pro calculates the peak values in data points it shut down and the program ‘lineidenity.pro’ initiates.

Lineidenity.pro

The next step is to mark each absorption line peak with the correct J value for the $2\nu_1$ transition. This is done using an interactive interface to mark usually the P28 absorption line. P28 is picked because it is easy to identify from the large methane lines lying between P28 and P29. Also, peaks that are not assigned to $2\nu_1$ transitions of N_2O are removed. Next each of the absorption line peaks are assigned a wavenumber value from HITRAN. These points are used to interpolate the wavenumber values of the data points between the peaks. A paper by Robert A. Toth (Toth 1991) has confirmed the $2\nu_1$ values of the P-branch transitions that are given by HITRAN. Then the program asks for the starting time of the scan and the sampling rate. One can use the sampling rate to calculate the total time of each scan, this total time is then added to the starting time to give the finishing time. Then the program 'alt2.pro' calculates the zenith angle of the sun of every absorption peak in the spectra. The formulas used in 'alt2.pro' are in Chapter 3. Then 'lineidenity.pro' interpolates between peaks in order to obtain the time and zenith angle of each data point.

Calculating the zenith angle for each data point is crucial for modeling the data. During times around noon, the zenith angle changes very little, however late afternoon the zenith angle can change a couple of degrees in a single scan. Such variation can change the value of $\sec(\theta)$, in the monochromatic transmission function, as much as 10%, which is very significant.

The program 'lineidenity.pro' then calculate the area under each line. This is done by finding points half way in between each peak and integrating from one trough to another. It re-plots and writes the data to a text and save file. The program 'lineidenity.pro' has a pull-down menu that allows you to print or save the plot as a postscript file. The program 'lineidenity.pro' shuts down and 'atm_model.pro' starts up. The 'atm_model.pro' is the atmospheric modeling and fitting program and is described in the next chapter.

Chapter Five

Analysis and Results

Infrared absorption spectroscopic data for N_2O in the earth's atmosphere using Suntracker-1 integrated with the 5-meter Littrow spectrometer were acquired on 14 dates from January 2001 to September 2001. These dates were primarily chosen because of clear conditions for observation.

The spectra were processed as discussed below. The initial continuum levels were chosen, using the software systems discussed in Chapter 4. The parameters used in the atmospheric modeling program were based upon the U. S. standard atmosphere profile of N_2O mixing ratios as a function of altitude, a standard temperature profile (adjusted to the local ground level temperature), a standard pressure profile, and pressure broadening coefficients from the HITRAN database.

The simulated spectrum (described in detail below) was then compared with the observed spectrum. After adjusting for CH_4 and N_2O concentrations and continuum parameters, the simulation was repeated until a good representation of the observed spectrum was obtained. The N_2O concentration derived from this iteration is represented as a fractional multiplier relative to the U. S. standard atmosphere profile of N_2O mixing- ratio profile.

As a check on this determination, the experimental data was compared with a U. S. standard atmosphere simulation (with appropriate choices of CH_4 concentration and continuum parameters) on a line-by-line basis described on page 109.

The relative concentration of N_2O as retrieved for each data run on the simulation basis and on the line-by-line basis were averaged and used to compute a daily average relative concentration.

Fitting observed data with calculated data

Figure 5.1 exhibits several simulations for a diverse set of data runs. The over-plotted simulated and observed spectra are difficult to visually separate except in very localized positions. The difference between the observed data and the simulated spectrum is shown below the spectrum pane.

The program 'atm_model.pro' overlays a calculated spectrum with the observed spectrum and it adjusts the baseline of the observed spectra. The rough baseline chosen by inspection in the program 'baselinerough.pro' is adjusted principally by the use of the troughs of the calculated spectra (maximum transmittance points). By visual inspection, the calculated lines are manually adjusted to be close to the observed lines. Since the troughs change very little with a change in column abundances compared to the peaks of absorption spectra, the troughs are averaged and then the observed spectrum troughs are adjusted to the calculated spectrum toughs. This provides a reasonably accurate continuum for line fitting purposes. This method of adjusting the baseline is similar to the method chosen by Y.S. Chang and others (Chang 1978).

In the next step, 'atm_model.pro' adjusts the column abundances of the calculated spectra to minimize the point-by-point standard deviation between the observed and calculated spectra. The model spectrum is calculated using the methods and parameters presented on page 79 and following. The resulting spectrum is then overlaid with the observed spectrum and adjustments to the column abundances are preformed. The process is iterated until the standard deviation of the fit does not improve.

The end result of this fitting process is a fractional relative abundance of N_2O as compared to the U. S. Standard atmosphere. It is presumed that this calculation yields a more accurate fractional abundance that does the ad hoc line-by-line method of retrieval. The line-by-line method does provide a check on the consistency of the simulation method of retrieval.

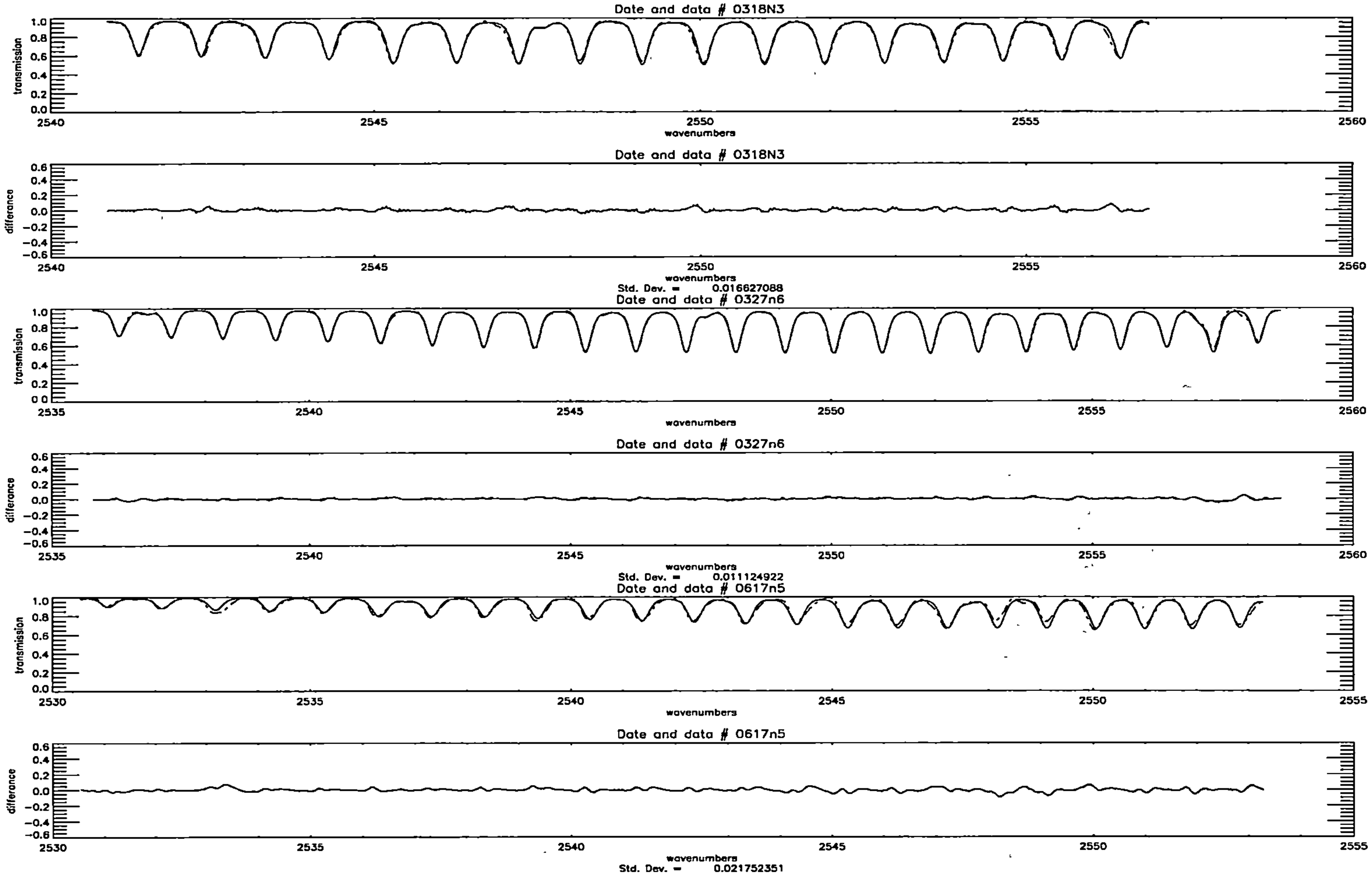


Figure 5.2 Fits of the observed spectra with the calculated spectra, with the variance below each fit.

The data set

Data were taken on 14 days from February through September when the sky was clear enough to provide acceptable data. The data set is summarized in table 5.1. The table components include the date, scan number, relative concentrations, average relative concentrations, standard deviation, variance and average zenith angles. Each P(J) absorption line has its own row, so comparisons can be made between lines of approximately the same strength.

Line-by-line retrievals

The area under each absorption line of the observed data and the area under each absorption line of the U.S. standard atmosphere data was measured and compared. This was carried out by dividing the observed and calculated spectrum into line pairs (one observed and one calculated). The area of the two paired lines were integrated and the fractional difference was determined. The results are shown in Figure 5.2. This comparison indicates that some of the lines are consistently far from the mean. These lines are presumed to have contributions that "atm_model.pro" does not take into consideration. Since the only molecules included in the model are N_2O and CH_4 these additional contributions must come from other molecules i.e. H_2O, SO_2, OCS and isotopes of N_2O and CH_4. On the other hand, the ad hoc line-by-line method depends upon several marginally defensible assumptions and variations in table 5.1 are not thought to be fundamental. Some of the lines that consistently deviated from the model

Table 5.1 Data sets summarized

		Relative Concentrations	Variance						
Date and scan #	0125n2			0125n3			0125n4		
Average Relative Concentrations	1.010058045			1.002051234			0.949281394		
Standard Deviation	0.204065561			0.195797503			0.221452862		
Average Zenith Angle	72.05549041			73.08798441			75.65521983		
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s	31	0.984478414	-0.025579631	31	0.988241732	-0.013809502			
P-Branch line #'s	30	1.294882417	0.284824371	30	1.272878408	0.270827174	30	1.198407054	0.249125659
P-Branch line #'s	29	0.949519634	-0.060538411	29	0.938515484	-0.06353575	29	0.874655128	-0.074626267
P-Branch line #'s	28	0.811351955	-0.19870609	28	0.80856961	-0.193481624	28	0.774781942	-0.174499452
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									
P-Branch line #'s									

Table 5.1 Data sets summarized continued

0125n5			0203n3			0203n4			0203n5		
0.999322414			0.870590031			0.861897528			0.86839813		
0.148208544			0.08293809			0.07726866			0.095842019		
77.16626274			72.04962531			72.74114329			73.52759148		
30	1.104121685	0.104799271									
29	0.894523144	-0.104799271	29	0.929236114	0.058646083	29	0.916534722	0.054637194			
			28	0.811943948	-0.058646083	28	0.807260334	-0.054637194	28	0.800627589	-0.067770541
									27	0.936168671	0.067770541

Table 5.1 Data sets summarized continued

0203na			0218n2			0218n6			0218n7		
0.796101809			1.130651236			1.009032249			0.990028739		
0.060137663			0.169222876			0.075081699			0.137351513		
77.4210018			52.06552537			57.03978945			58.16521817		
			35	0.883017361	-0.247633874						
			34	1.022218227	-0.108433008						
			33	1.215683699	0.085032463						
			32	1.124821663	-0.005829573						
			31	1.200430751	0.069779515						
			30	1.414696097	0.284044862						
29	0.849873543	0.053771734	29	1.05369103	-0.076960206	29	1.06212306	0.053090811			
28	0.718068242	-0.078033566				28	0.955941498	-0.053090751	28	0.892906547	-0.097122192
27	0.836292088	0.040190279							27	1.087150931	0.097122192
26	0.780173242	-0.015928566									

Table 5.1 Data sets summarized continued

0218n9			0218na			0218nb			0218nc		
1.016404748			0.895385683			0.909548581			0.858065784		
0.080785774			0.082371235			0.073554046			0.088471547		
59.51653936			69.28643856			69.88211515			70.62195005		
29	1.030627131	0.014222383	29	0.927987337	0.032601655	29	0.922954917	0.013406336	29	0.911821902	0.053756118
28	0.905198872	-0.111205876	28	0.801703513	-0.09368217	28	0.830213487	-0.079335093	28	0.755955219	-0.102110565
27	1.098860502	0.082455754	27	0.956466138	0.061080456	27	0.975477397	0.065928817	27	0.906420112	0.048354328
26	1.030932307	0.014527559									

Table 5.1 Data sets summarized continued

0307n1			0307n2			0307n4			0307n5		
1.111473441			1.11052382			1.118405819			1.072503567		
0.086062752			0.057762627			0.088432521			0.080503739		
45.44811557			46.62036068			48.11960371			48.98005072		
29	1.056627035	-0.054846406	29	1.098904967	-0.011618853						
28	0.963094175	-0.148379266	28	0.989619851	-0.120903969				28	0.959186137	-0.11331743
27	1.154708505	0.043235064	27	1.162072539	0.051548719				27	1.088412642	0.015909076
26	1.098006845	-0.013466597	26	1.099750876	-0.010772943				26	1.028680086	-0.043823481
25	1.190458536	0.078985095	25	1.159378409	0.048854589	25	1.210680485	0.092274666	25	1.145231605	0.072728038
24	1.1009835	-0.010489941	24	1.109192729	-0.001331091	24	1.141449213	0.023043394	24	1.098255277	0.02575171
23	1.216435075	0.104961634	23	1.167729378	0.057205558	23	1.153492212	0.035086393	23	1.127460837	0.054957271
			22	1.09754169	-0.01298213	22	1.076827049	-0.04157877	22	1.062466383	-0.010037184
						21	1.227039099	0.10863328	21	1.19763267	0.125129104
						20	1.071581125	-0.046824694	20	1.073074579	0.000571012
						19	1.117305636	-0.001100183	19	1.100459695	0.027956128
						18	0.948871851	-0.169533968	18	0.916678786	-0.15582478

Table 5.1 Data sets summarized continued

0307n6			0307n7			0307n8			0307n9		
1.082431793			1.099315286			1.043097734			1.045089722		
0.084152989			0.091025874			0.077727795			0.123927414		
50.26789993			50.42870626			53.64236141			55.31658942		
									30	1.375274777	0.330185056
			29	1.057139039	-0.042176247				29	0.980233729	-0.064855993
28	0.959829092	-0.122602701	28	0.936282277	-0.163033009	28	0.942028046	-0.101069689	28	0.902391315	-0.142698407
27	1.135150909	0.052719116	27	1.184341908	0.085026622	27	1.13008678	0.086989045	27	1.061618924	0.016529202
26	1.042979836	-0.039451957	26	1.057593465	-0.041721821	26	1.033305883	-0.009791851	26	1.037798405	-0.007291317
25	1.1596452	0.077213407	25	1.178235292	0.078920007	25	1.138643503	0.095545769	25	1.099138379	0.054048657
24	1.121470213	0.03903842	24	1.103970528	0.004655242	24	1.077464104	0.034366369	24	1.0192132	-0.025876522
23	1.141916752	0.059484959	23	1.177644014	0.078328729	23	1.133944154	0.090846419	23	1.096398711	0.05130899
22	1.054757595	-0.027674198				22	1.051906586	0.008808851	22	1.007000685	-0.038089037
21	1.219266057	0.136834264				21	1.156433344	0.113335609	21	1.115811467	0.070721745
20	1.040715456	-0.041716337				20	1.014908552	-0.028189182	20	0.990040183	-0.055049539
19	1.087060213	0.00462842				19	1.068051696	0.024953961	19	1.042340875	-0.002748847
18	0.943959594	-0.138472199				18	0.886889458	-0.156208277	18	0.858905554	-0.186184168
						17	0.982315302	-0.060782433			
						16	1.049378991	0.006281257			
						15	0.982041359	-0.061056376			
						14	0.999067962	-0.044029772			

Table 5.1 Data sets summarized continued

0307na			0318n1			0318N3			0325n9		
1.012580633			1.051126719			1.029349446			1.162264705		
0.117988512			0.071357839			0.077888481			0.266659588		
58.45941992			45.38470291			49.20923162			43.13262835		
30	1.415452003	0.40287137							30	1.451358557	0.289093852
29	1.040460587	0.027879953							29	1.109510422	-0.052754283
28	0.895876825	-0.116703808							28	0.925925136	-0.236339569
27	1.087543011	0.074962378									
26	0.997283459	-0.015297174									
25	1.075971246	0.063390613				25	1.125645518	0.096296072			
24	1.030950069	0.018369436	24	1.079922318	0.0287956	24	1.091657281	0.062307835			
23	1.089458346	0.076877713	23	1.1480515	0.096924782	23	1.137583017	0.108233571			
22	0.993944526	-0.018636107	22	1.101910233	0.050783515	22	1.051318526	0.02196908			
21	1.105503917	0.092923284	21	1.180709243	0.129582524	21	1.163199425	0.133849978			
20	0.980442226	-0.032138407	20	1.047158241	-0.003968477	20	1.012225032	-0.017124414			
19	1.043341398	0.030760765	19	1.098318696	0.047191978	19	1.108776331	0.079426885			
18	0.85114187	-0.161438763	18	0.946516752	-0.104609966	18	0.888949156	-0.14040029			
17	0.925374091	-0.087206542	17	1.011569858	-0.039556861	17	0.960343003	-0.069006443			
16	1.001826882	-0.010753751	16	1.09617734	0.045050621	16	1.046818614	0.017469168			
15	0.919591546	-0.092989087	15	1.035649419	-0.0154773	15	0.984920681	-0.044428766			
14	0.957428753	-0.05515188	14	1.004905105	-0.046221614	14	1.002405286	-0.02694416			
13	0.952656627	-0.059924006	13	0.971407473	-0.079719245	13	0.966012061	-0.063337386			
12	0.976587951	-0.035992682	12	1.047599673	-0.003527045	12	1.052204251	0.022854805			
11	0.910779774	-0.101800859	11	0.945879042	-0.105247676	11	0.924039185	-0.105310261			
						10	0.966769874	-0.062579572			
						9	1.01607573	-0.013273716			

Table 5.1 Data sets summarized continued

0325na			0325nb			0325nc			0325nd		
1.116064191			1.023971438			1.021485448			1.066150188		
0.127304181			0.096134387			0.071171455			0.065404087		
43.57025228			45.81344485			47.58638566			48.92993546		
30	1.455929875	0.339865685									
29	1.08906436	-0.026999831	29	0.993226826	-0.030744612	29	1.028851151	0.007365704			
28	0.969865143	-0.146199048	28	0.87891978	-0.145051658	28	0.898303986	-0.123181462	28	0.95022583	-0.115924358
27	1.159311295	0.043247104	27	1.174088359	0.15011692	27	1.062709212	0.041223764	27	1.135078192	0.068928003
26	1.062301636	-0.053762555	26	1.035240769	0.011269331	26	1.012468457	-0.009016991	26	1.027202964	-0.038947225
25	1.178773403	0.062709212	25	1.075439811	0.051468372	25	1.098256111	0.076770663	25	1.118523836	0.052373648
24	1.095816612	-0.020247579	24	1.060241103	0.036269665	24	0.999794543	-0.021690905	24	1.086400986	0.020250797
23	1.172824025	0.056759834	23	1.115137577	0.091166139	23	1.102200031	0.080714583	23	1.146462679	0.08031249
22	1.074930429	-0.041133761	22	0.998690724	-0.025280714	22	0.927298725	-0.094186723	22	1.031917572	-0.034232616
21	1.186601281	0.07053709	21	1.105190635	0.081219196	21	1.063487411	0.042001963	21	1.123228908	0.057078719
20	1.056054115	-0.060010076	20	0.970783353	-0.053188086				20	1.002200961	-0.063949227
19	1.070804715	-0.045259476	19	1.042052865	0.018081427				19	1.040260196	-0.025889993
18	0.936557889	-0.179506302	18	0.838646352	-0.185325086						

Table 5.1 Data sets summarized continued

0327n1			0327n2			0327n5			0327n6		
1.164375067			1.090032339			1.050104856			1.052859426		
0.119748116			0.088994563			0.110729583			0.119338848		
35.86994184			37.35586442			45.07951336			47.83183715		
30	1.403612614	0.239237547							30	1.345132351	0.292272925
29	1.082543254	-0.081831813	29	1.092435956	0.002403617				29	1.06315124	0.010291815
28	0.957398534	-0.206976533	28	0.955229104	-0.134803236				28	0.948769093	-0.104090333
27	1.166634798	0.002259731	27	1.117317557	0.027285218				27	1.143504739	0.090645313
26	1.109014511	-0.055360556	26	1.075072646	-0.014959693	26	1.112596869	0.062492013	26	1.072390795	0.019531369
25	1.201148868	0.036773801	25	1.191519737	0.101487398	25	1.17318058	0.123075724	25	1.145391345	0.092531919
24	1.137929559	-0.026445508	24	1.105717063	0.015684724	24	1.106575847	0.05647099	24	1.095654845	0.04279542
23	1.194876432	0.030501366	23	1.197667837	0.107635498	23	1.183425426	0.13332057	23	1.152241111	0.099381685
22	1.110672474	-0.053702593	22	1.093441367	0.003409028	22	1.05525732	0.005152464	22	1.093032956	0.040173531
21	1.279918551	0.115543485	21	1.202252746	0.112220407	21	1.192096233	0.141991377	21	1.1615659	0.108706474
			20	1.058537722	-0.031494617	20	1.061707139	0.011602283	20	1.036207676	-0.01665175
			19	1.080482006	-0.009550333	19	1.078414202	0.028309345	19	1.049166203	-0.003693223
			18	0.910713851	-0.179318488	18	0.935003281	-0.115101576	18	0.907090843	-0.145768583
						17	1.013804674	-0.036300182	17	1.008611798	-0.044247627
						16	1.07984519	0.029740334	16	1.090522766	0.037663341
						15	1.038209915	-0.011894941	15	1.011188984	-0.041670442
						14	1.029280424	-0.020824432	14	1.048098326	-0.0047611
						13	0.989697814	-0.060407043	13	1.008527756	-0.04433167
						12	1.066571116	0.01646626	12	1.07602036	0.023160934
						11	0.961113572	-0.088991284	11	0.953347087	-0.099512339
						10	1.021747231	-0.028357625	10	1.003819108	-0.049040318
						9	1.016944885	-0.033159971	9	1.001301885	-0.051557541
						8	1.187485933	0.137381077	8	1.164691329	0.111831903
						7	0.699139297	-0.350965559	7	0.689200103	-0.363659322

Table 5.1 Data sets summarized continued

0327n7			0327n8			0404n6			0404n7		
1.058692455			1.063815117			1.02646482			1.107318163		
0.103871271			0.103590019			0.111883812			0.185040131		
51.27253079			54.94615143			39.71841746			41.2998635		
									35	0.865741134	-0.241577029
			34	1.105779767	0.04196465				34	1.153756857	0.046438694
			33	1.188898087	0.12508297				33	1.626546741	0.519228578
			32	1.189477682	0.125662565				32	1.140374899	0.033056736
			31	1.132751942	0.068936825				31	1.170272112	0.062953949
30	1.40439117	0.345698714	30	1.407387614	0.343572497				30	1.453432322	0.346114159
29	1.0305655	-0.028126955	29	1.047714829	-0.016100287				29	1.109546661	0.002228498
28	0.949537873	-0.109154582	28	0.960045397	-0.10376972				28	0.963963926	-0.143354237
27	1.113063574	0.054371119	27	1.08597517	0.022160053				27	1.254091144	0.146772981
26	1.043669224	-0.015023232	26	1.012514114	-0.051301003				26	1.035791516	-0.071526647
25	1.177446365	0.11875391	25	1.109886765	0.046071649				25	1.17704916	0.069730997
24	1.095038772	0.036346316	24	1.077357054	0.013541937	24	1.098946452	0.072481632	24	1.107006311	-0.000311852
23	1.146687627	0.087995172	23	1.130414963	0.066599846	23	1.166695118	0.140230298	23	1.168646216	0.061328053
22	1.041985869	-0.016706586	22	1.057289481	-0.006525636	22	1.058162332	0.031697512	22	1.038169622	-0.06914854
21	1.145107865	0.08641541	21	1.137479067	0.07366395	21	1.077720761	0.051255941	21	1.090536714	-0.016781449
20	1.009708405	-0.048984051	20	1.024783254	-0.039031863	20	1.000307679	-0.026157141	20	0.9958992	-0.111418962
19	1.071120381	0.012427926	19	1.064011097	0.00019598	19	1.114070058	0.087605238	19	1.117660761	0.010342598
18	0.877530515	-0.18116194	18	0.905239165	-0.158575952	18	0.805699468	-0.220765352	18	0.804277897	-0.303040266
17	0.997493804	-0.061198652	17	0.958657682	-0.105157435	17	0.90858531	-0.11787951	17	0.911629319	-0.195688844
16	1.041024208	-0.017668247	16	1.055150747	-0.00866437	16	1.097697139	0.071232319	16	1.12330246	0.015984297
15	1.002091765	-0.05660069	15	0.979838371	-0.083976746	15	0.936765194	-0.089699626	15	0.945989728	-0.161328435
14	1.016421676	-0.04227078	14	0.968061805	-0.095753312						
13	1.000358582	-0.058333874	13	0.983842731	-0.079972386						
12	1.062699318	0.004006863	12	1.030665398	-0.033149719						
11	0.973583519	-0.085108936	11	0.947402179	-0.116412938						
10	0.999160051	-0.059532404	10	0.985583842	-0.078231275						
9	0.998632789	-0.060059667	9	1.003984094	-0.059831023						
8	1.152604103	0.093911648	8	1.172810674	0.108995557						

Table 5.1 Data sets summarized continued

0425n9			0425nb			0425nc			0425nd		
1.164010525			1.164605856			1.118181944			1.129050851		
0.091016576			0.129699707			0.080157198			0.082395747		
22.57783855			23.62199295			26.22751459			28.71574124		
			30	1.524896622	0.360290766						
29	1.137845516	-0.026165009	29	1.139838099	-0.024767756	29	1.134823561	0.016641617	29	1.136045933	0.006995082
28	0.977817118	-0.186193407	28	1.001791596	-0.16281426	28	0.965016544	-0.1531654	28	1.010985851	-0.118065
27	1.261334181	0.097323656	27	1.243950129	0.079344273	27	1.206368327	0.088186383	27	1.256283998	0.127233148
26	1.122909307	-0.041101217	26	1.10477078	-0.059835076	26	1.089449883	-0.028732061	26	1.095330238	-0.033720613
25	1.254628062	0.090617537	25	1.194163203	0.029557347	25	1.219606757	0.101424813	25	1.22365725	0.0946064
24	1.136336565	-0.02767396	24	1.184137344	0.019531488	24	1.143044472	0.024862528	24	1.142026663	0.012975812
23	1.238470435	0.07445991	23	1.235594153	0.070988297	23	1.255063295	0.136881351	23	1.200206399	0.071155548
22	1.145041466	-0.018969059	22	1.113000035	-0.051605821	22	1.126293778	0.008111835	22	1.119077325	-0.009973526
21	1.261760354	0.097749829	21	1.245857239	0.081251383	21	1.220071673	0.101889729	21	1.202051044	0.073000193
20	1.103961825	-0.060048699	20	1.073668003	-0.090937853	20	1.087605	-0.030576944	20	1.081027031	-0.04802382
			19	1.169561744	0.004955888	19	1.186969995	0.068788052	19	1.174268246	0.045217395
			18	0.978508472	-0.186097383	18	0.97362107	-0.144560874	18	0.970982134	-0.158068717
			17	1.067326307	-0.097279549	17	1.061406493	-0.056775451	17	1.042650104	-0.086400747
			16	1.1920259	0.027420044	16	1.141144991	0.022963047	16	1.152117848	0.023066998
						15	1.066543937	-0.051638007			
						14	1.078260064	-0.03992188			
						13	1.077270031	-0.040911913			
						12	1.094714761	-0.023467183			

Table 5.1 Data sets summarized continued

0425ne			0425nf			0425ng			0425nh		
1.121149898			1.116667986			1.104007483			1.132293701		
0.119697943			0.120833427			0.139509529			0.133769065		
30.91590156			35.01296867			37.84992326			42.11887314		
						35	0.845185637	-0.258821845	35	0.923874438	-0.208419263
						34	1.192451358	0.088443875	34	1.09237504	-0.039918661
						33	1.348199487	0.244192004	33	1.335585713	0.203292012
32	1.14242208	0.021272182				32	1.063195705	-0.040811777	32	1.142059088	0.009765387
31	1.196527004	0.075377107	31	1.172659636	0.05599165	31	1.261392832	0.157385349	31	1.239502192	0.10720849
30	1.530860305	0.409710407	30	1.518863678	0.402195692	30	1.49070096	0.386693478	30	1.491894484	0.359600782
29	1.159783483	0.038633585	29	1.105346322	-0.011321664	29	1.08909452	-0.014912963	29	1.098988652	-0.033305049
28	0.957834661	-0.163315237	28	1.010166407	-0.106501579	28	0.92977351	-0.174233973	28	0.940898538	-0.191395164
27	1.183050632	0.061900735	27	1.215135098	0.098467112	27	1.139527082	0.0355196	27	1.156464934	0.024171233
26	1.069365025	-0.051784873	26	1.092303276	-0.02436471	26	0.993808806	-0.110198677	26	1.057996869	-0.074296832
25	1.194535613	0.073385715	25	1.179268003	0.062600017	25	1.181250334	0.077242851	25	1.146776438	0.014482737
24	1.13369751	0.012547612	24	1.11765182	0.000983834	24	1.112046957	0.008039474	24	1.090631127	-0.041662574
23	1.219976544	0.098826647	23	1.215454936	0.09878695	23	1.188242555	0.084235072	23	1.173047781	0.04075408
22	1.109036922	-0.012112975	22	1.102371216	-0.01429677	22	1.098959565	-0.005047917	22	1.059284687	-0.073009014
21	1.196909189	0.075759292	21	1.186032176	0.06936419	21	1.141438723	0.03743124	21	1.144117236	0.011823535
20	1.06844151	-0.052708387	20	1.055866241	-0.060801744	20	1.055721998	-0.048285484	20	1.06846571	-0.063827991
19	1.142678261	0.021528363	19	1.149339318	0.032671332	19	1.144667149	0.040659666	19	1.087027669	-0.045266032
18	0.956030548	-0.16511935	18	0.943705857	-0.172962129	18	0.881746292	-0.22226119			
17	1.044252992	-0.076896906	17	1.009463072	-0.107204914	17	1.023161292	-0.08084619			
16	1.149575114	0.028425217	16	1.101534605	-0.015133381	16	1.142840385	0.038832903			
15	1.039359093	-0.081790805	15	1.004546881	-0.112121105	15	1.015348196	-0.088659286			
14	1.028258681	-0.092891216	14	1.048741102	-0.067926884	14	1.029071569	-0.074935913			
13	1.058356762	-0.062793136	13	1.041757941	-0.074910045	13	1.046536803	-0.057470679			
12	1.07415843	-0.046991467	12	1.063153625	-0.053514361	12	1.081816077	-0.022191405			
11	1.010189414	-0.110960484									

Table 5.1 Data sets summarized continued

0426n2			0426p1			0617n1			0617n2		
1.195991755			1.132316947			1.031625628			1.074249506		
0.177723587			0.130039826			0.134534821			0.23144041		
25.20810724			24.1515306			51.80038644			51.14564776		
36	1.181158304	-0.01483345									
35	0.908714652	-0.287277102	35	0.982721508	-0.149595439				35	0.857016742	-0.217232764
34	1.125584841	-0.070406914	34	1.047780037	-0.08453691				34	1.112843037	0.038593531
33	1.366785645	0.170793891	33	1.387669325	0.255352378				33	1.799978852	0.725729346
32	1.180414438	-0.015577316	32	1.190799594	0.058482647				32	1.10368371	0.029434204
31	1.259445071	0.063453317	31	1.237851143	0.105534196				31	1.044047594	-0.030201912
30	1.482820392	0.286828637	30	1.399614453	0.267297506				30	1.476477742	0.402228236
29	1.063009739	-0.132982016	29	1.088065624	-0.044251323	29	1.067593217	0.035967588	29	1.086513042	0.012263536
			28	0.993488252	-0.138828695	28	0.98838371	-0.043241918	28	0.922404468	-0.151845038
			27	1.213490844	0.081173897	27	1.251059055	0.219433427	27	1.201533914	0.127284408
			26	1.058497787	-0.07381916	26	1.028655529	-0.002970099	26	0.998993874	-0.075255632
			25	1.210490346	0.078173399	25	1.142239928	0.1106143	25	1.09920907	0.024959564
			24	1.123488426	-0.008828521	24	1.052185416	0.020559788	24	1.030693769	-0.043555737
			23	1.224333405	0.092016459	23	1.131267428	0.0996418	23	1.112632632	0.038383126
			22	1.111600161	-0.020716786	22	1.022416472	-0.009209156	22	1.004081607	-0.070167899
			21	1.183014512	0.050697565	21	1.016449571	-0.015176058	21	0.991862059	-0.082387447
			20	1.039457679	-0.092859268	20	0.939698815	-0.091926813	20	0.952318549	-0.121930957
			19	1.096573472	-0.035743475	19	1.083611131	0.051985502	19	1.06569314	-0.008556366
			18	0.91165638	-0.220660567	18	0.717012525	-0.314613104	18	0.688921869	-0.385327637
			17	1.013432264	-0.118884683	17	0.844454825	-0.187170804	17	0.84146136	-0.232788146
						16	1.157730699	0.12610507	16	1.094624877	0.020375371

Table 5.1 Data sets summarized continued

0617n3			0617n4			0617n5			0617n6		
1.083099246			1.063338041			1.125737309			1.091053128		
0.21700938			0.164312094			0.220540226			0.143717527		
44.29341501			38.62935393			34.50690195			29.50953845		
35	0.80069679	-0.282402456				35	0.799544573	-0.326192737			
34	1.11628437	0.033185124				34	1.176107407	0.050370097			
33	1.871338487	0.788239241				33	1.824384928	0.698647618			
32	1.227965593	0.144866347	32	1.20836401	0.145025969	32	1.158914566	0.033177257			
31	1.171306133	0.088206887	31	1.070444226	0.007106185	31	1.158080339	0.03234303			
30	1.491460085	0.408360839	30	1.445275068	0.381937027	30	1.525599837	0.399862528			
29	1.120303631	0.037204385	29	1.124270916	0.060932875	29	1.152892232	0.027154922	29	1.126429558	0.03537643
28	0.989335716	-0.09376353	28	1.020614147	-0.042723894	28	1.023961425	-0.101775885	28	0.987668395	-0.103384733
27	1.276083112	0.192983866	27	1.364754796	0.301416755	27	1.311421633	0.185684323	27	1.423601747	0.332548618
26	1.064918518	-0.018180728	26	1.075337768	0.011999726	26	1.055544257	-0.070193052	26	1.077139735	-0.013913393
25	1.187497497	0.104398251	25	1.21557045	0.152232409	25	1.220005631	0.094268322	25	1.22572875	0.134675622
24	1.09916997	0.016070724	24	1.110399365	0.047061324	24	1.10908401	-0.016653299	24	1.177041888	0.08598876
23	1.164820194	0.081720948	23	1.191948533	0.128610492	23	1.214612961	0.088875651	23	1.220008731	0.128955603
22	1.067564607	-0.015534639	22	1.036291122	-0.027046919	22	1.042280674	-0.083456635	22	1.111664653	0.020611525
21	1.060854077	-0.022245169	21	1.082677722	0.019339681	21	1.072148204	-0.053589106	21	1.089575768	-0.001477361
20	0.997837543	-0.085261703	20	0.997339308	-0.065998733	20	1.000269413	-0.125467896	20	1.036265969	-0.054787159
19	1.138493419	0.055394173	19	1.148553967	0.085215926	19	1.160081744	0.034344435	19	1.201624632	0.110571504
18	0.743977606	-0.33912164	18	0.740580499	-0.322757542	18	0.77846092	-0.34727639	18	0.767811954	-0.323241174
17	0.885384262	-0.197714984	17	0.89634639	-0.166991651	17	0.906776011	-0.218961298	17	0.941756785	-0.149296343
16	1.14963758	0.066538334	16	1.152723551	0.089385509	16	1.17850244	0.052765131	16	1.191475868	0.10042274
15	0.950996757	-0.132102489	15	0.952910304	-0.110427737	15	0.968628526	-0.157108784	15	0.974538684	-0.116514444
14	0.98257494	-0.100524306	14	0.971995234	-0.091342807	14	1.002344966	-0.123392344	14	1.001815081	-0.089238048
13	1.055664778	-0.027434468	13	1.069642663	0.006304622	13	1.052312493	-0.073424816	13	1.103481889	0.012428761
12	1.00499177	-0.078107476	12	0.990157008	-0.073181033				12	0.981325984	-0.109727144
11	0.974918187	-0.108181059	11	0.924073398	-0.139264643						
10	0.952779472	-0.130319774	10	0.937190533	-0.126147509						
9	1.152101994	0.069002748	9	1.160414934	0.097076893						
8	1.201491952	0.118392706	8	1.19222486	0.128886819						
7	0.696689129	-0.386410117	7	0.696776748	-0.366561294						
6	0.864585578	-0.218513668	6	0.855712414	-0.207625628						
5	1.114357829	0.031258583	5	1.14087224	0.077534199						

0617n7			0926n1			0926n3			0926n4		
1.069821477			1.091751337			1.12737751			1.094238997		
0.147552848			0.217502505			0.144843847			0.141841933		
34.59776008			39.69935669			40.52994106			44.10525705		
			36	1.132333994	0.040582657						
			35	0.767330229	-0.324421108						
			34	1.021342993	-0.070408344						
			33	1.340779781	0.249028444						
32	1.066259384	-0.003562093	32	1.025979519	-0.065771818						
31	1.222928166	0.153106689	31	1.17459178	0.082840443	31	1.079493642	-0.047883868	31	1.227655649	0.133416653
30	1.437987328	0.368165851	30	1.465741277	0.37398994	30	1.481843114	0.354465604	30	1.462191463	0.367952466
29	1.105359912	0.035538435	29	1.028458357	-0.06329298	29	1.081538558	-0.045838952	29	1.096254349	0.002015352
28	0.969225049	-0.100596428	28	0.869204044	-0.222547293	28	0.90667659	-0.22070092	28	0.937410116	-0.15682888
27	1.29128623	0.221464753				27	1.180066228	0.052688718	27	1.155017614	0.060778618
26	1.103526831	0.033705354				26	1.05318749	-0.074190021	26	1.094997048	0.000758052
25	1.210487723	0.140666246				25	1.116378427	-0.010999084	25	1.169115663	0.074876666
24	1.097180963	0.027359486				24	1.141293526	0.013916016	24	1.074606061	-0.019632936
23	1.158737063	0.088915586				23	1.141585708	0.014208198	23	1.18228209	0.088043094
22	1.044714451	-0.025107026				22	1.091711521	-0.035665989	22	1.072885752	-0.021353245
21	1.080509424	0.010687947							21	1.17112124	0.076882243
20	0.985627115	-0.084194362							20	1.047854185	-0.046384811
19	1.129698515	0.059877038							19	1.1357131	0.041474104
18	0.74949038	-0.320331097							18	0.839550316	-0.25468868
17	0.889729202	-0.180092275							17	0.94364053	-0.150598466
16	1.13393116	0.064109683							16	1.064975858	-0.029263139
15	0.925803542	-0.144017935							15	0.926791906	-0.16744709
14	0.961485565	-0.108335912									
13	1.054028869	-0.015792608									
12	0.972357273	-0.097464204									
11	0.945720136	-0.124101341									

Table 5.1 Data sets summarized continued

0926n5			0926n8			0926n9		
1.100069642			1.08807826			0.938795209		
0.143153325			0.166108057			0.155898497		
46.18083012			55.56015064			58.85310404		
37	1.3082546	0.208184958	37	1.333139658	0.245061398			
36	0.998478532	-0.10159111	36	1.124360561	0.036282301			
35	0.919225216	-0.180844426	35	0.909179389	-0.178898871	35	0.783812642	-0.154982567
34	1.188058138	0.087988496	34	1.168297887	0.080219626	34	1.018820763	0.080025554
33	1.398090243	0.298020601	33	1.506910682	0.418832421	33	1.310127378	0.371332169
32	1.066598535	-0.033471107	32	1.15923059	0.071152329	32	1.019260526	0.080465317
31	1.179222107	0.079152465	31	1.153117895	0.065039635	31	0.987536609	0.0487414
30	1.463762283	0.363692641	30	1.415900707	0.327822447	30	1.348119259	0.40932405
29	1.103449106	0.003379464	29	1.147176862	0.059098601	29	0.981979012	0.043183804
28	0.914952517	-0.185117126	28	0.947612286	-0.140465975	28	0.815916836	-0.122878373
27	1.136703014	0.036633372	27	1.172593832	0.084515572	27	0.998260856	0.059465647
26	1.028331876	-0.071737766	26	1.036460161	-0.051618099	26	0.855129063	-0.083666146
25	1.116346002	0.01627636	25	1.074046731	-0.014031529	25	0.970781028	0.031985819
24	1.067522287	-0.032547355	24	1.028375745	-0.059702516	24	0.902642012	-0.036153197
23	1.171358705	0.071289063	23	1.108555198	0.020476937	23	0.93541038	-0.003384829
22	1.016829967	-0.083239675	22	0.974408746	-0.113669515	22	0.888041914	-0.050753295
21	1.138744593	0.038674951	21	1.077943087	-0.010135174	21	0.953327239	0.01453203
20	1.033764839	-0.066304803	20	0.997297227	-0.090781033	20	0.852636516	-0.086158693
19	1.115819931	0.015750289	19	1.062950969	-0.025127292	19	0.914851546	-0.023943663
18	0.921857655	-0.178211987	18	0.799559891	-0.288518369	18	0.728965104	-0.209830105
17	0.964330912	-0.13573873	17	0.890230179	-0.197848082	17	0.752272367	-0.186522841
16	1.082166433	-0.017903209	16	1.020621061	-0.067457199	16	0.9012869	-0.037508309
15	0.967735231	-0.132334411	15	0.917829692	-0.170248568	15	0.795521677	-0.143273532

were P28, P17 and P18. The latter two are thought to be influenced by H_2O, since they are most affected during dates of high H_2O concentrations.

The daily relative abundances come from the averages of all the scans on that date. They are calculated by taking the mean of abundances of all the lines, in the line-by-line method for a scan. Some dates had more scans than others, but weights were not use to reflect this in Figure 5.3

The average daily relative abundances are calculated using the mean of abundances of that date. Data taken at zenith angle larger than 70° were thrown out. Several factors that spoil the data are no longer negligible at zenith angle larger than 70°. Path lengths were calculated using the stratified atmospheric model, not taking into consideration the curvature of the earth's atmosphere. Therefore an observation angle larger than 70° the path length is over-estimated leading to an under-estimation of column abundances. The first two data points in Figure 5.4 are not used due to their high average zenith angles but are included here for completeness. Figure 5.4 shows an increase in N_2O during the summer months for the line-by-line method.

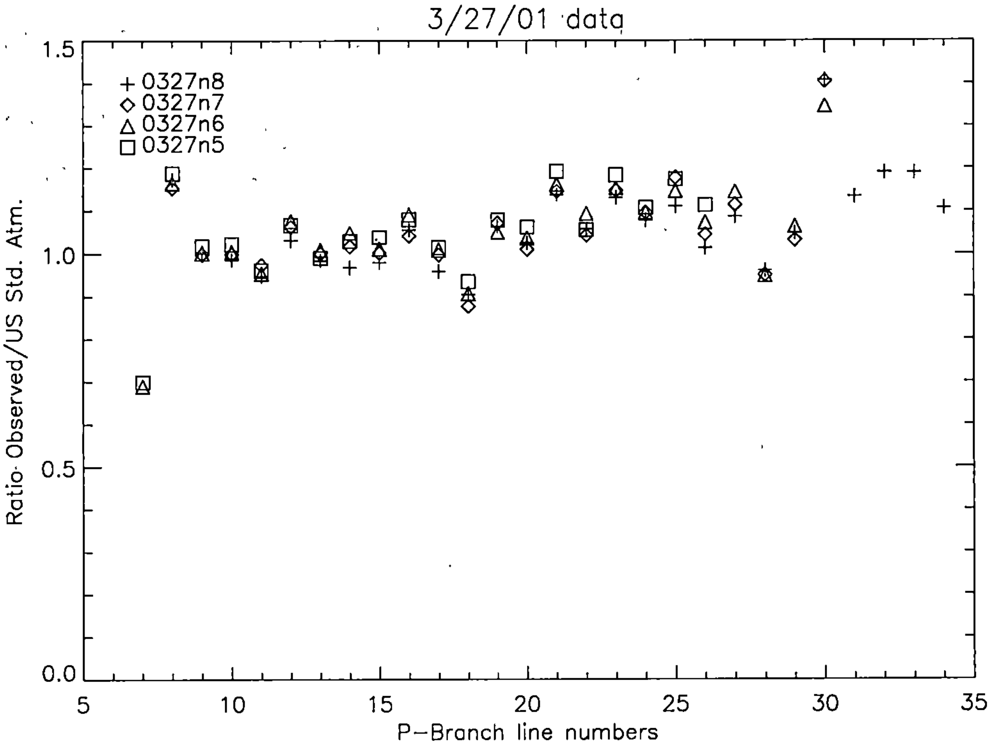


Figure 5.2 Line by line areas compared with U.S. Standard atmosphere line areas.

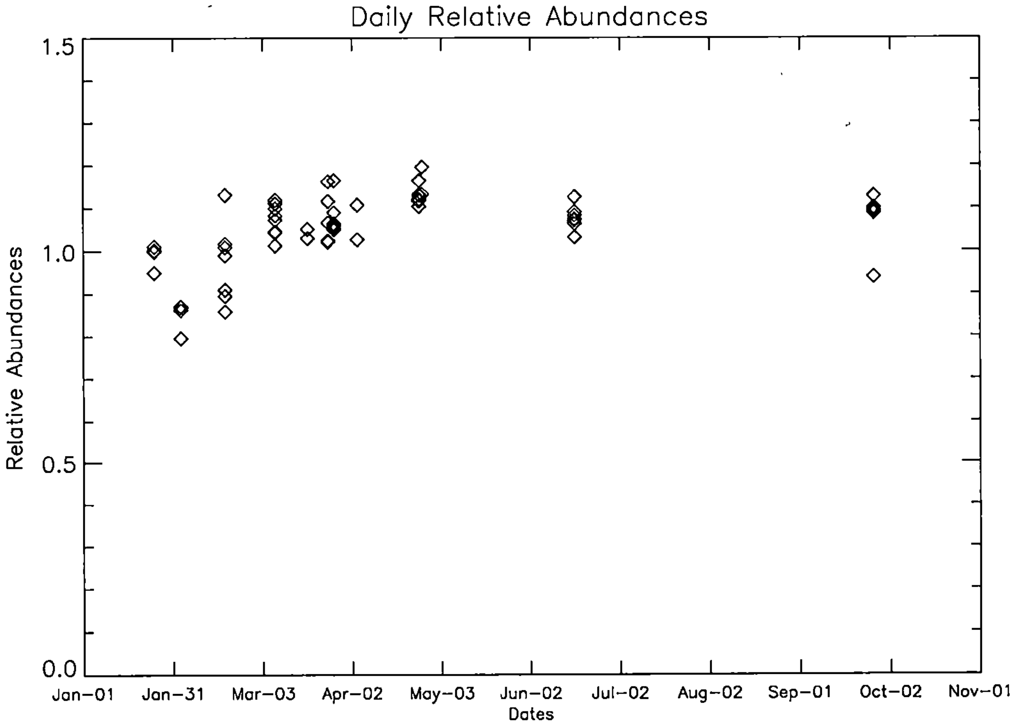


Figure 5.3 Relative abundances of each scan.

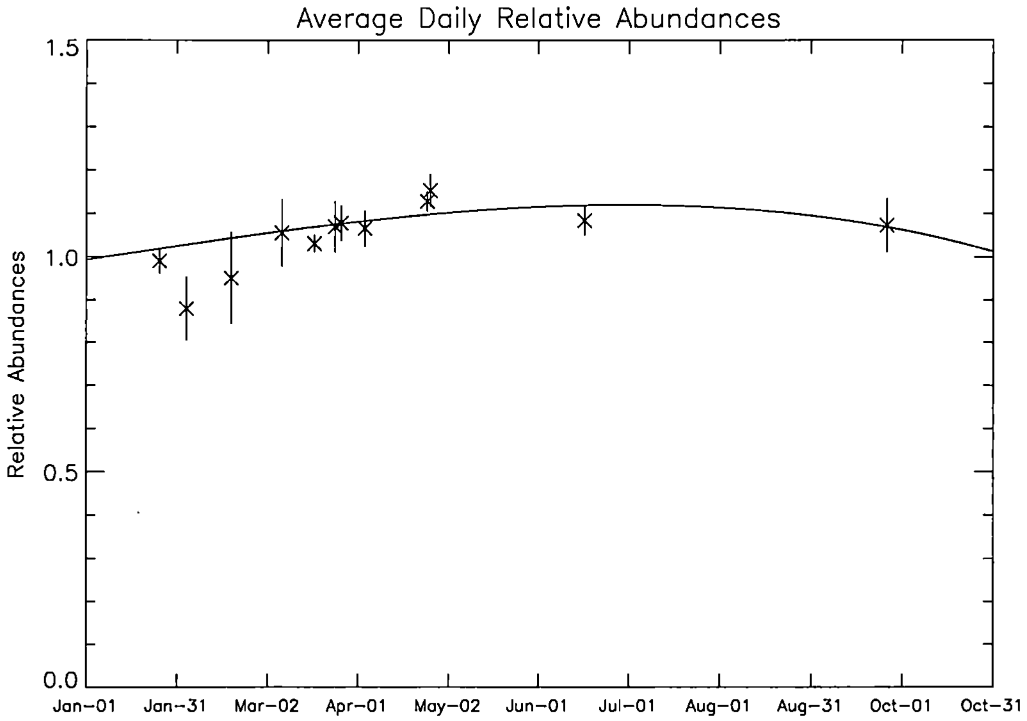


Figure 5.4 Average daily relative abundances.

Retrievals by Simulation

The simulated column abundances were retrieved directly by the atmospheric modeling program 'atm_model.pro'. These abundances correspond to the amount of N_2O that minimized the standard deviation of the fit of the total scan. Figure 5.5 illustrates the amount that the U. S. Standard column abundances of N_2O that each simulated spectra was increased or decreased in order to match the observed spectra. If no adjustments were made, the relative abundance would be 1.0.

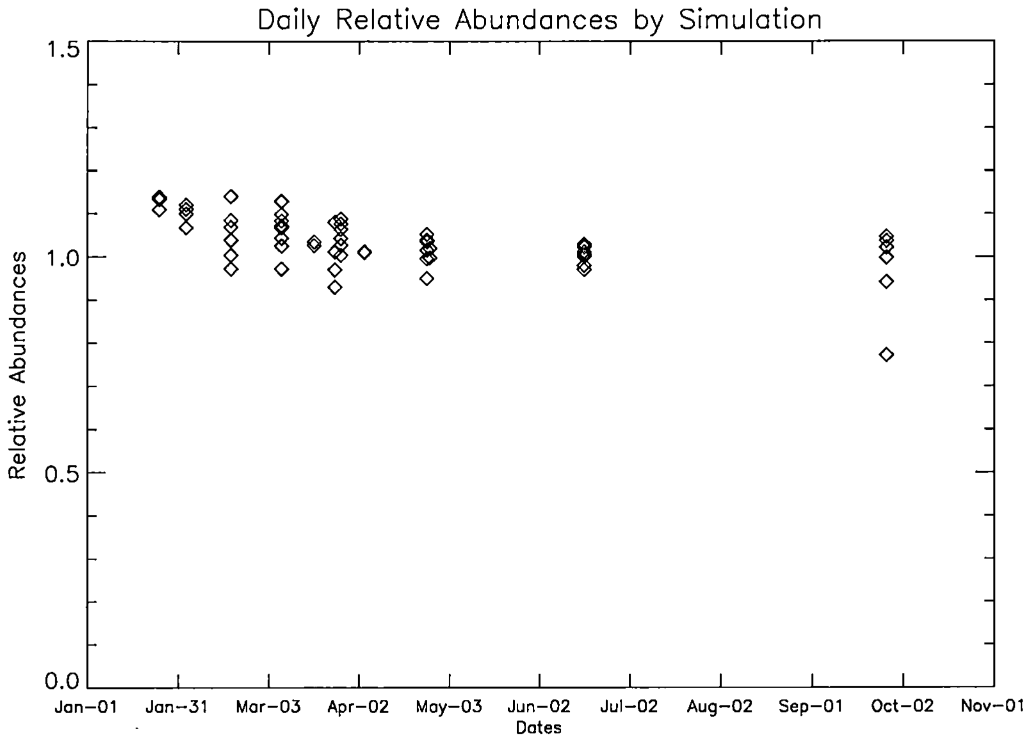


Figure 5.5 Daily relative abundances by simulation.

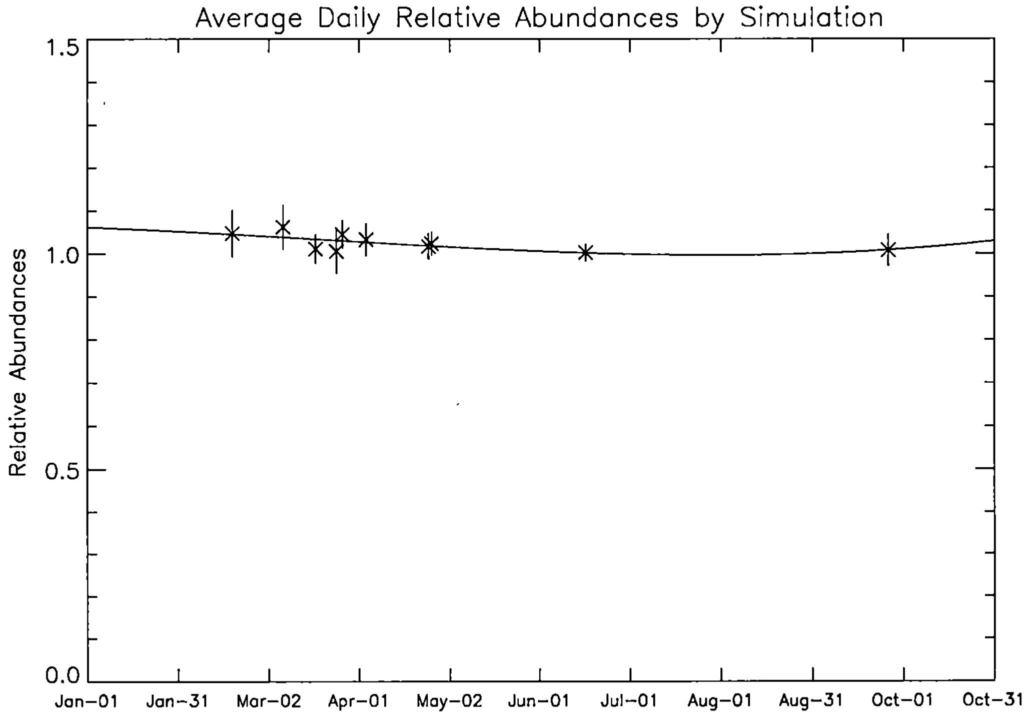


Figure 5.6 Average daily relative abundances by simulation

Figure 5.6 shows the daily average and the standard deviation of the amount of U. S. standard column abundances for N_2O that each simulated spectra was increased or decreased in order the match the observed spectra.

Figure 5.4 and Figure 5.6 shows the results of the mean relative abundances from the line-by-line and simulated methods respectively. It is obvious in both figures that the column abundances of N_2O in east Tennessee are higher than U.S. Standard Atmosphere. U.S. Standard atmosphere column abundances of N_2O are the highest of all the atmospheric data sets supplied by AFGL (updated in 2000). Therefore the relative abundances in East Tennessee are high as compared to the rest of the nation. The tropospheric concentration of the U. S. standard atmosphere is 320 ppbv. The relative abundances are converted into parts per billion are tabulated in Table 5.2.

Table 5.2 Relative abundances converted into ppbv.

Relative Abundances by Simulation	Parts per Billion per Volume
1.04729	335.132
1.06238	339.963
1.01100	323.520
1.00538	321.721
1.04497	334.391
1.03280	330.495
1.01696	325.428
1.02321	327.426
1.00233	320.744
1.00759	322.430

The weather conditions at the time of the scan were noted during the observations. For example, careful records were kept on the amount of clouds and haze that was observed during the scan, along with the local temperature, pressure and relative humidity. Scans that might have appeared acceptable were weighted to reflect the questionable weather conditions. For example, some scans had to be deleted due to water-vapor trails from commercial aircraft. A digital camera with a wide-angle lens will be used in the future to better track weather conditions.

Chapter Six

Conclusions

Goals of this work

- Design and implementation of an improved polar axis suntracker. This goal was partially achieved, but vibration problems and the large distance between the suntracker and the 5-meter Littrow spectrometer lead to the use of the original suntracker. The main purpose of the new suntracker was to eventually integrate it to a spectrometer in the Science and Engineering Building at the University of Tennessee. These problems were foreseen and were solved by using the original suntracker on top of the 4th floor roof of the Nielsen Physics building for this project. The vibration problems were addressed by using vibrational absorbing pads on the platform on top of the Science and Engineering Building. The residual beam divergence problem was not resolved leading to the use of the original suntracker for the spectral acquisition of this project.

- Establish whether column abundances of N_2O could be determined with experimental facilities at the University of Tennessee Complex System

Laboratory and develop an algorithm to compute a synthetic spectrum using an atmospheric model. This goal was achieved and the result is positive.

- Determine the local column abundances of N_2O over an extended period of time, in this instance, about 8 months. This goal was achieved as discussed below.

N_2O variation varies little on a seasonal basis worldwide (Lal 2000). N_2O has a long lifetime in the troposphere. At the outset, it was known that very little abundance variation should be observed. The results, both line by line and by simulation agree with this conclusion. (This is so, even though the line-by-line and simulation abundance variations are not in complete agreement.)

It is not clear which results are the most accurate – simulation or line by line. The presumption, at this point, is that the simulation results are more precise and possibly more accurate. The reason for the bias is that the simulation method, while subject to some uncertainties, makes fewer assumptions in the determination of relative abundances and "fits" the entire spectral data run with a three parameter set.

Figure 6.1 displays the N_2O average relative abundances for 10 different dates. The range of variation is consistent with the expectations of atmospheric science. It is not clear that the variation shown in Figure 6.1 is significant. However, small variation of N_2O abundances in the troposphere is not inconsistent with expectations (Wallace and Livingston 1990). Also, a recent paper on the study of variations of atmospheric nitrous oxide is quoted, " The N_2O concentrations observed from March 1996 to February 1999 increased at an average rate of 0.64 ppb/yr. The observed data also suggest that there is a weak annual cycle of N_2O concentration, increasing in the autumn and winter and decreasing in the spring and summer, with a peak-to-peak amplitude of at most 0.3 ppb"(Tohjima 2000). The results of the latter paper are consistent with the results of this work. The increases of concentrations in N_2O are larger in this work, but may be due to the agricultural nature of east Tennessee as compared to Hateruma, Japan (Tohjima 2000).

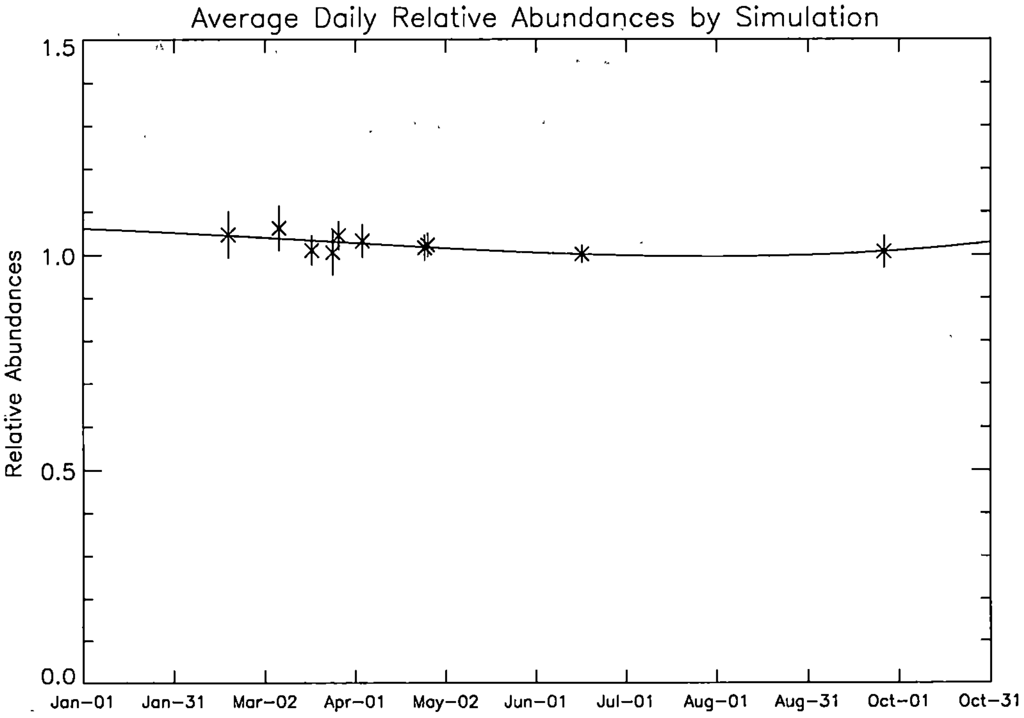


Figure 6.1 Average daily relative abundances of simulated spectra.

The average of all 10 different dates is 1.0254 with a standard deviation of 0.0206. The total column abundance of the U.S. standard atmosphere for N_2O is 6.378 x 10^{18} molecules/cm^2. Therefore, it is the final conclusion of this work that the local N_2O column abundance averaged over an 8-month period is 1.0254 more than the U.S. standard atmosphere for N_2O, or 6.540 x 10^{18} molecules/cm^2, or 328.128 ppbv. The result is based on determination made on separate 62 spectra recorded over the 8-month period and analysis of the spectra using the simulation method.

Future projects

The acquisition of a new FTS Bomem DA8 will allow higher resolution spectroscopy of the atmosphere to be carried out. Since it is easier to see the change in the absorption at the higher resolution, this will make the modeling the data easier.

To illustrate this point Figure 6.2 and Figure 6.3 show the change in concentration of N_2O of 10% of the same two spectral lines for the 5-meter Littrow spectrometer in its present state and the new Bomem DA8. In other words, if there were suddenly 10% more of N_2O in the earth's atmosphere then Figure 6.2 and Figure 6.3 would illustrate the changes in the lines. As one can see the change is easier to detect and quantify on the DA8.

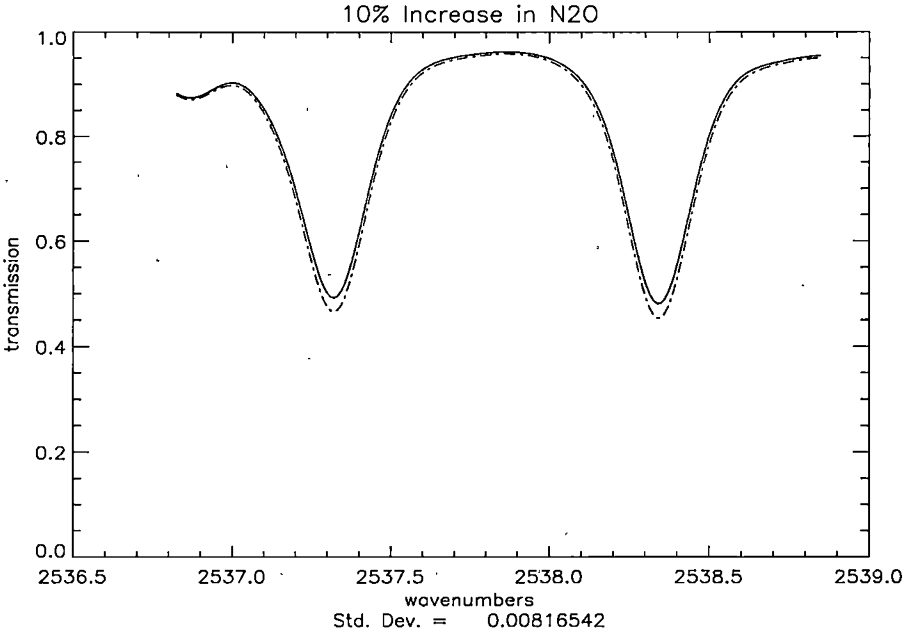


Figure 6.2 The dotted line represents a 10% increase in N_2O for the 5-meter Littrow spectrometer

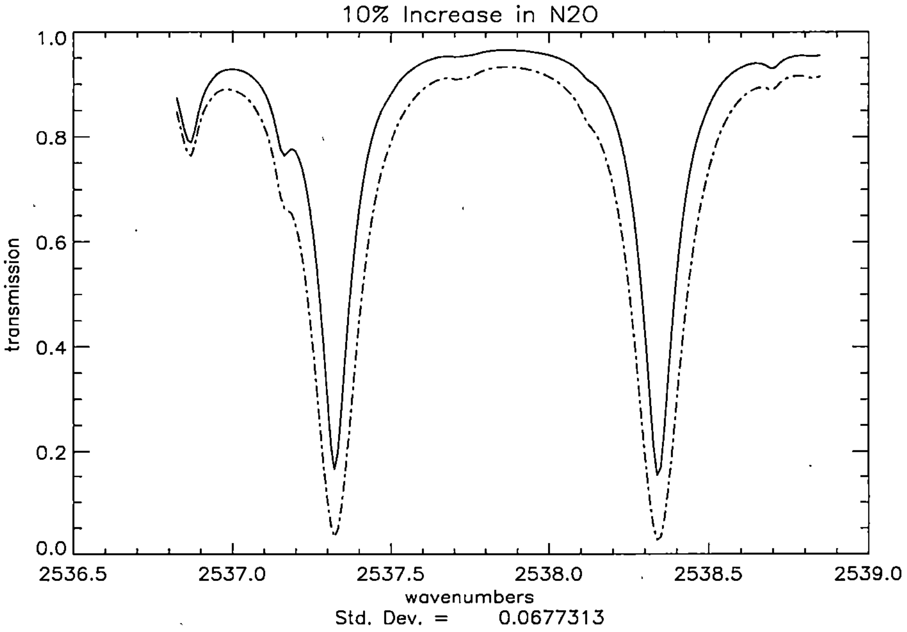


Figure 6.3 The dotted line represents a 10% increase in N_2O for the Bomem DA8

References

Chang, T. S., Shaw, J. H., Calvert, J. G. (1978). "Determination of Abundances of Gases from Solar Spectra." Journal of Quantitative Spectroscopy & Radiative Transfer **19**: 599-605.

Clough, S. A., Worsham, R. D. (1992). "Advanced Spectral Modeling Development." Defence Information Systems Agency.

Edwards, D. P. (1992). "Genln2: A General Line-by-Line Atmosheric Transmittance and Radiance Model." NCAR Technical Note **NCAR/TN-367+STR**: p 39.

Goody, R. M. (1989). Atmospheric Radiation. New York, New York, Oxford University Press.

Goody, R. M. (1995). Principles of Atmospheric Physics and Chemistry. New York, Oxford University Press.

Herzberg, G. (1950). Spectra of Diatomic Molecules. Toronto, New York, London, D. Van Nostrand Company, Inc.

Houghton, J. T. (1986). The Physics of Atmospheres, Cambridge University Press.

Iqbal, M. (1983). An Introduction to Solar Radiation. Toronto-New York-London, Academic Press.

Jennings, D. E. (1974). The UTMSL 5-meter Littrow Spectrometric system; An Analysis of n2 , CHD3. Physics. Knoxville, The University of Tennessee.

Kaiser, J. (2001). "The Other Global Pollutant: Nitrogen Proves Tough to Curb." Science **294**(9 November): 1268-1269.

Lal, S., Sheel, Varun (2000). "A study of the atmospheric photochemical loss of N2O based on trace gas measurments." Chemosphere - Global Change Science **2**: 455-463.

Levy, A., N. Lacome, et al. (1984). "Measurement of N2O Line Strengths from High-Resolution Fourier-Transform Spectra." Journal of Molecular Spectroscopy **103**(1): 160-175.

Mosier, A. K., Carolien (2000). "Potential impact on the global atmospheric N2O budget of the increased nitrogen input required to meet future global food demands." Chemosphere - Global Change Science **2**: 465-473.

Park, J. H., Rothman, L. S., Rinsland, C. P. (1987). Atlas of Absorption Lines from 0 to 17900 cm-1. NASA Reference Publication 1188.

Penner, S. S. (1959). Quantitative Molecular Spectroscopy and Gas Emissivities. Reading, MA, Addison_Wesley.

Rao, K. N. (1976). Molecular Spectrocopy Modern Research: Volume II, Academic Press.

Rothman, L. S., R. R. Gamache, et al. (1987). "The HITRAN Database - 1986 Edition." Applied Optics **26**(19): 4058-4097.

Rothman, L. S., C. P. Rinsland, et al. (1998). "The HITRAN molecular spectroscopic database and HAWKS (HITRAN Atmospheric Workstation): 1996 edition." Journal of Quantitative Spectroscopy & Radiative Transfer **60**(5): 665-710.

Stephens, G. L. (1994). Remote Sensing of the Lower Atmosphere. Oxford, Oxford University Press.

Tohjima, Y. M., Maksyutov, S. (2000). "Variations in atmospheric nitrous oxide observed at Hateruma monitoring station." Chemosphere - Global Change Science **2**: 435-443.

Toth, R. A. (1991). "Line frequency measurments and analysis of N2O between 900 and 4700 cm-1." Applied Optics **30**: 5289-5315.

Wallace, L. and W. Livingston (1990). "Spectroscopic Observations of Atmospheric Trace Gases over Kitt Peak .2. Nitrous-Oxide and Carbon-Monoxide from 1979 to 1985." Journal of Geophysical Research-Atmospheres **95**(D10): 16383-16390.

Appendixes

Noise Reduction of a Thermal Detector Using Wavelets and Neural Nets

INTRODUCTION

In early 1996, NOAA/Environmental Technology Laboratory and University of Wisconsin – Madison measured infrared atmospheric radiances with two independent Fourier Transform Infrared (FTIR) spectro-radiometers on the NOAA R/V *Discoverer* in the Tropical Pacific. NOAA/ETL using a thermal detector took data in small time increments at the same time a group at the University of Wisconsin was taking data with a HgCdTe detector with a much higher signal to noise ratio. The thermal detector data is denoised with two different methods, Neural Nets and Wavelets. The University of Wisconsin data was used as pseudo-ground truth to compare with the denoised NOAA data.

DESCRIPTION OF THE INSTRUMENTS

The two FTIR systems are similar, both being based on Bomem interferometers. The University of Wisconsin deployed an AERI (Atmospheric Emitted Radiance Interferometer) system that they modified. They allowed observation at different angles measured from zenith. These included 0, 35, 45, 55, 65, 75, 105, 115, 125, 135, 145 degrees. This allows a three-minute zenith scan every 20 minutes on the average. The AERI measured the down welling emission spectrum between 550 and 3000 cm^{-1} at a spectral resolution of 0.5 cm^{-1} (unapodized). The AERI was housed in an enclosure that is shielded from sun and rain on the forward port deck.

The NOAA ETL FTIR records only a down welling spectrum at the zenith. NOAA data have a three-minute zenith dwell, obtained every 7 minutes, covering from 500 to 2000 cm^{-1} with a spectral resolution of 1 cm^{-1} (with Hanning apodization). Nine hundred and eighteen of the scans are taken with a room temperature thermal detector, which had a dell time of approximately 25 minutes. The interferometer is housed with a microwave radiometer in an environmentally controlled sea container, looking through an infrared window to the beam steering mirror and calibration sources mounted in an insulated box on the outside wall. During this cruise, the NOAA FTIR system sat on the lower aft deck. The distance between the two FTIRs was approximately 60 meters horizontal and 5 meters vertical.

ORGANIZATION OF DATA

The University of Wisconsin zenith looking spectrum is used as pseudo-ground truth for the denoising project. The data from NOAA/ETL and University of Wisconsin examined for the dates and times the data is taken, this allow a determination of which data from University of Wisconsin coincides with the NOAA/ETL thermal data. Table 1 shows how the ETL thermal data coincides with the University of Wisconsin data. The University of Wisconsin cycled through 11 different observing angles, but they always use zenith or 0 degrees in every observing cycle. Thus, the University of Wisconsin only acquired zenith scans every 7th scan or every 20 minutes. The NOAA scans repeated every 7 minutes.

NOAA/ETL data	University of Wisconsin zenith data
4/01-1909/2038 Thermal data 1964-2004 Times taken 08:10/18:51 40 scans	From 960401C1.CDR Times taken 08:28/18:40 28 scans
4/04-2386/2549 Thermal data 2425-2431 Times taken 06:18/08:02 6 scans	From 960404C1.CDR None taken
4/05-2550/2674 Thermal data 2557-2561, 2584-2590 Times taken 01:32/02:25, 05:32/09:09	From 960405C1.CDR Times taken 01:33/02:30 4 scans Times taken 6:04/7:39 6 scans
4/07-2881/3042 Thermal data 2916-2925 Times taken 05:32/09:09 9 scans	From 960407C1.CDR Times taken 5:27/8:57 5 scans
4/08-3074/3194 Thermal data 3074-3084 Times taken 04:52/07:32 10 scans	From 960408C1.CDR Times taken 4:54/6:11 5 scans
4/10-3263/3366 Thermal data 3293-3302 Times taken 05:35/08:04 9 scans	From 960410C1.CDR Times taken 4:47/8:01 8 scans
4/11-3367/3480 Thermal data 3449-3461 Times taken 11:22/21:15 12 scans	From 960407C1.CDR Times taken 5:27–8:57 5 scans

Table 1

Not all spectra from NOAA are taken at exactly the same time as that of the University of Wisconsin spectra. Also, the dwell time of the thermal detector was much longer than the University of Wisconsin HgCdTe detector, since theoretically the two spectra should have the same intensity. The spectra from both instruments were integrated from 856 to 1350 cm^{-1} and plotted verses time to get the intensity between those wavenumbers. Those values are chosen because they include the ozone band center at approximately 1060 cm^{-1}. There are several periods of time that the NOAA and University of Wisconsin spectra have the same intensity. Likewise there were also times where the NOAA and University of Wisconsin spectrum are taken at approximately the same time yet the intensities did not coincide. It was concluded that there may have been an error in calibration. April the first has the most matches of any day and is shown Figure 1.

The thermal data and the HgCdTe detector data that had approximately the same intensity or the saw the same distance up in to the atmosphere are used.

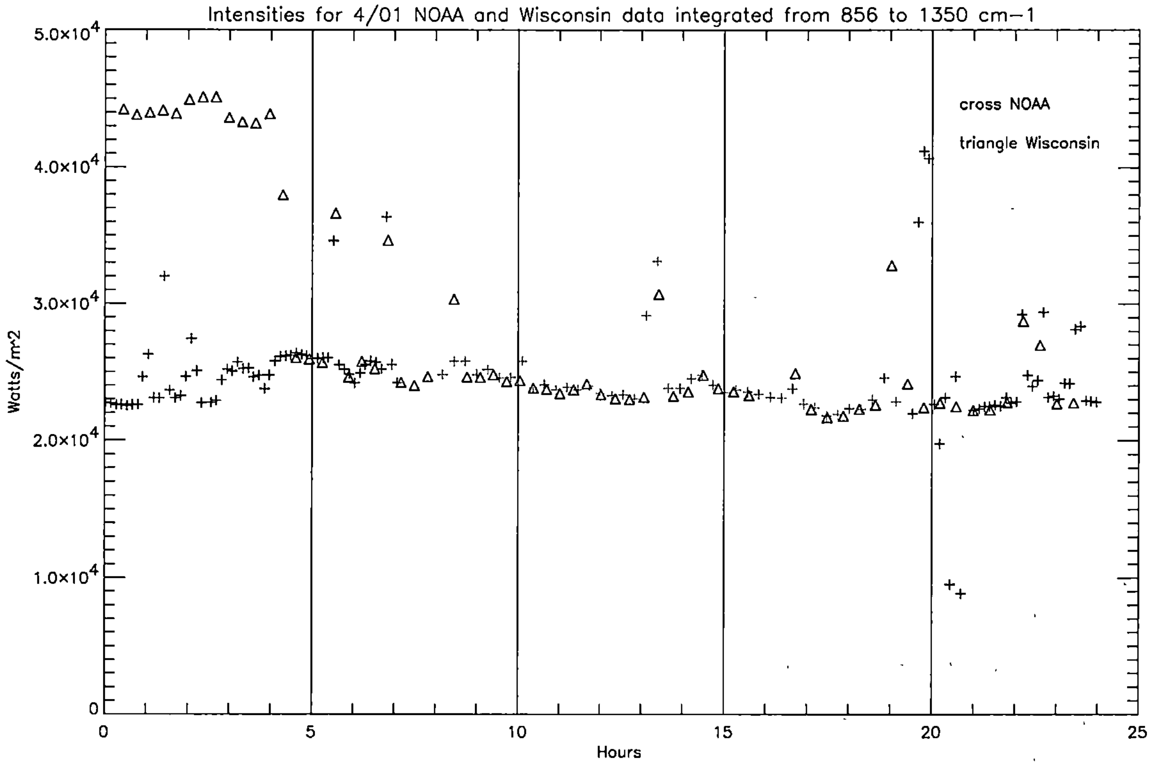


Figure 1 The gray crosses are the thermal data.

NOISE REDUCTION USING NEURAL NETS

Why use neural nets for noise reduction? The main use of a neural net is to train a computer to make educated guesses. We have 78 scans of the University of Wisconsin zenith data that was taken at the same time of the ETL thermal detector scans, which allows us to use the University of Wisconsin data to build supervised training sets. The neural net can then learn the instrument response and the noise of the NOAA radiometer (as compared to the University of Wisconsin instrument response and noise). In other words, we have the advantage of having the answers to the problem. Once the neural net is trained it is a generalized method of denoising data of that instrument, including in the best of all worlds, data that the neural net has not seen before.

Neural nets are very tolerant of unusual noise distributions. Many algorithms that are mathematically optimal make assumptions that, for example, the contaminating noise follows a particular distribution. But this is not always the case and neural nets are very tolerant of unusual noise distributions. Neural nets are also fast once they are trained.

A feed forward backpropagation (FFBP) neural network with the generalized delta rule for error correction is used. The structure of the net is 1024 input nodes, 16 hidden nodes and 1024 output nodes. The data is normalize for the neural net. The net is trained on ten data sets using the University of Wisconsin data as a training set. The training sets are picked from several different days.

The noise reduction results were fair. The neural nets not only reduce the noise it also increased the resolution. The resolution enhancement is a by-product of the training sets. Blass and Mahan have shown that a FFBP network can be trained to enhance resolution with reasonable generalization capabilities.[1] The University of Wisconsin spectra are taken twice the resolution of the ETL spectra. The main problem is the net perceived baseline of the spectrum. The neural nets had trouble learning how deep in to the

[1] Blass and Mahan, *Image Restoration and Super-Resolution by Novel Application of a Neural Network*, Proceedings of a Workshop held at The Space Telescope Science Institute, Baltimore, Maryland, 18-19 November, 1993

atmosphere the spectrum was seeing or optical depth. The neural nets had no trouble denoising data that it had not seen before, where the neural nets had been trained on data with similar optical depths. Figure 2 shows these results.

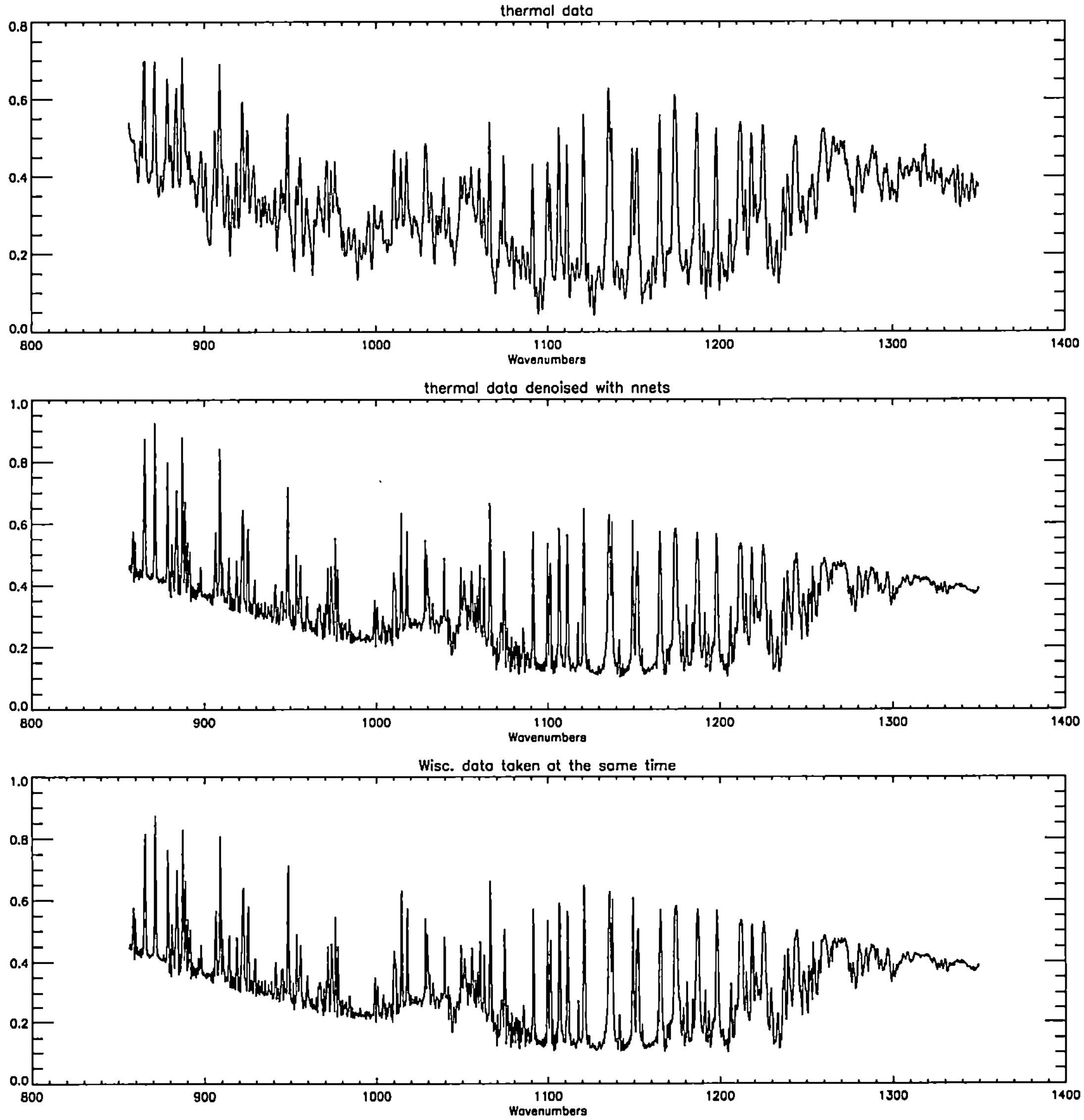


Figure 2

NOISE REDUCTION USING WAVLETS

Wavelet transforms are a natural tool to use in noise reduction because of its use of scales. The wavelet transform is a function of time and scale. Wavelet transforms, for a fixed value of scale, represents the detail contained in the signal at that scale. The wavelets that are used are the Daubechies' wavelets. The use the DAUB12 wavelet, which is relatively smooth and localized, seemed to work the best of all the Daubechies' wavelets. The noise from the thermal data looks to be both high and low frequency or white noise. This made it difficult to tell what coefficients to eliminate and which ones to keep. There is an assumption that the detail of the noise would have small coefficients in the wavelet transform whereas the true signal would have the largest coefficients. Based upon this premise the smallest coefficients are removed. As you can see in figure 3, the noise had large coefficients and is not remove by eliminating the small ones

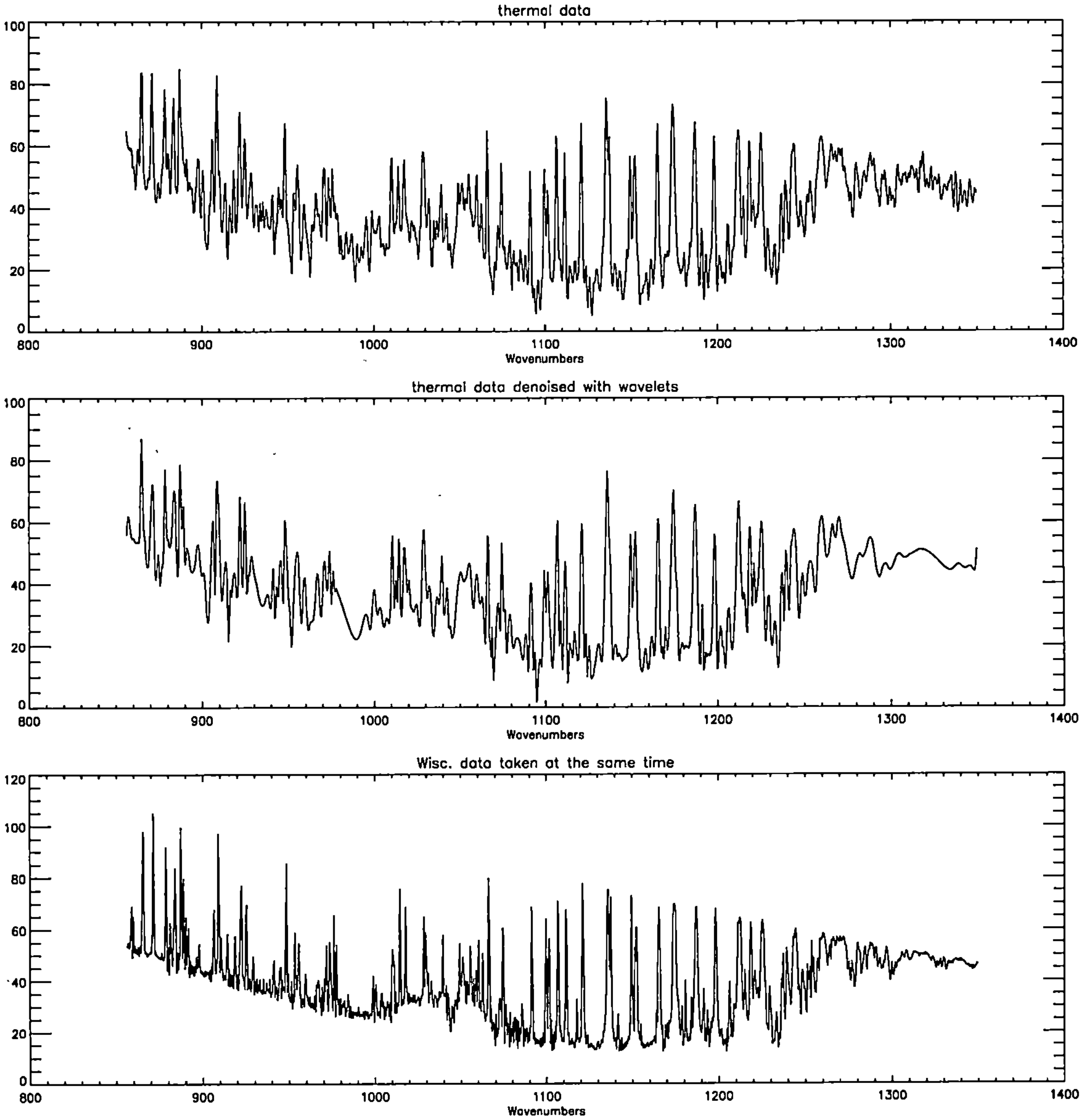

thermal data
100
80
60
40
20
0
800
900
1000
1100
1200
1300
1400
Wavenumbers
thermal data denoised with wavelets
100
80
60
40
20
0
800
900
1000
1100
1200
1300
1400
Wavenumbers
Wisc. data taken at the same time
120
100
80
60
40
20
0
800
900
1000
1100
1200
1300
1400
Wavenumbers

Figure 3

Conclusion

Neural nets show much promise in these algorithms. They lock in on the lines resulting in a higher signal to noise ratio on the level of the HgCdTe, the set back being, the Neural Net's inability to lock in on the base lines the spectra. The trails carried out using neural nets seem promising, especially if the problems associated with the baselines in spectra can be overcome.

The wavelet transforms where less successful. The main problem is the contribution of white noise from the thermal detector. It seemed the wavelet transform had trouble doing away with the low frequency noise.

Radiative equilibrium

In this section we explain be beginning of the math it takes to derive the radiative equations of transfer. This can give an idea of the large amount of approximations it takes when one is deriving equations for the earth's atmosphere.

Local thermodynamic equilibrium lets us derive formulas of radiative transfer that govern how the layers of the stratified atmosphere exchange energy. Consider radiation passing through an infinitesimal cylinder of absorbing matter. In order to have local thermodynamic equilibrium, the cylinder must be in radiative equilibrium with the surrounding matter. Meaning it must have a net exchange of heat with its surrounding equal to zero.

Consider a cylinder with an end of area dA and height of ds with light approaching perpendicularly in direction $\vec{s}$. All the photons with in a distance $cdt = ds$ (and therefore in a volume ($dA \cdot c \cdot dt$)) pass through the plane in the interval dt. If the spectra energy density in the field is $\rho(\nu)$, the total electromagnetic energy passing through the plane in that interval is, $\rho(\nu) \cdot dA \cdot c \cdot dt$. The energy flux density $F(\nu)$ is the energy per unit time per unit area, and so $F(\nu) = \dfrac{\rho(\nu) \cdot A \cdot c \cdot dt}{A \cdot dt} = c \cdot \rho(\nu)$, ((m/s)*(j/m^3)=j/s*m^2). The total amount of energy in the cylinder made up of photons traveling in the $\vec{s}$ direction within a solid angle $d\omega_s$ is that which crosses dA in time dt:

$$I(\nu) \cdot dA \cdot d\nu \cdot d\omega_s \cdot dt = \frac{I(\nu)}{c} dA \cdot d\nu \cdot d\omega_s \cdot ds \qquad [5.1]$$

If the volume of the cylinder is $dA \cdot ds$ then, $\left(I(\nu)/c\right)d\nu d\omega$ must be the energy density. Integrate over all directions we find for the energy density in the frequency range ν to $\nu + d\nu$.

$$\rho(\nu)d\nu = \frac{d\nu}{c}\int I(\nu)d\omega_s \qquad [5.2]$$

$$\rho(\nu) = \frac{4\pi}{c}\bar{I}(\nu) \qquad [5.3]$$

$$\bar{I}(\nu) = \frac{1}{4\pi}\int I(\nu)d\omega_s \qquad [5.4]$$

The quantity $\bar{I}(\nu)$ is the mean radiance of the radiation field.

The divergence of the irradiance equals the rate at which energy is added to the electromagnetic field per unit volume, i.e. the rate at which energy is lost by the matter. Let $h(\nu)$ be the rate per unit volume at which heat is gained by the matter from radiation in unit frequency range. So $h(\nu)$ is equal to the negative of the divergence of the irradiance.

$$h(\nu) = \rho\dot{q} = -\nabla \cdot \vec{F}(\nu) \qquad [5.5]$$

where q is heat and ρ is the density of the matter. $h(\nu)$ is equal to $\rho\dot{q}$ if the matter in the cylinder does not cross the boundaries of the cylinder.

$$\vec{F}(\nu) = \hat{x}F_x(\nu) + \hat{y}F_y(\nu) + \hat{z}F_z(\nu)$$

$$\nabla \cdot \vec{F}(\nu) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z}$$

$$\text{so, } F(\nu) = \int I(\nu) d\omega_s$$

$$\nabla \cdot \vec{F}(\nu) = \int d\omega_s \left[\frac{\partial I(\nu)}{\partial x} \cos(\mathrm{s},x) + \frac{\partial I(\nu)}{\partial y} \cos(\mathrm{s},y) + \frac{\partial I(\nu)}{\partial z} \cos(\mathrm{s},z) \right]$$

$$\nabla \cdot \vec{F}(\nu) = \int d\omega_s \frac{dI(\nu)}{ds} = -h(\nu) \qquad [5.6]$$

Then meaning of equation [5.6] is that the heat loss per unit volume from radiation traveling in the $\vec{s}$ direction is $(dI(\nu)/ds)d\omega_s$ or $(dI(\nu)/ds)d\omega_s \cdot dA \cdot ds$ is the energy lost by matter in the small cylinder $dAds$ per second per unit frequency range. For radiative equilibrium $h(\nu)$ is equal to zero (Goody 1989). The equation [5.6] is used to derive the equations of energy transfer between layers of a stratified atmosphere.

QuickBasic and Visual Basic 5-Meter Spectrometer data acquisition and control programs

QuickBasic data acquisition and control program '5METER'

```
x = 2000
CLS
CLEAR 1000
' clears buffer
P = 512 + 10
' Port number
LOCATE , , 1
= INKEY$: COLOR (7)
' looks for keyboard input then goes on.
PRINT A$; : COLOR (15)
GOSUB 510
GOSUB 610
GOTO 100
' It goes through the 100-510-610-620-630 loop
' until something has been typed form the keyboard.
REM
IF A$ = "" THEN RETURN
IF A$ = "q" THEN GOTO 640
FOR y = 1 TO LEN(A$)
x = ASC(MID$(A$, y, 1))
' reads a$ one letter at a time, and converts it to ASCII code.
IF x = 33 OR x = 35 OR x = 36 THEN 550
' emergency charaters-  33 or ! - stops everthing immediately, must reset
' 35 or # - cancels current command line  36 or $ - Decelerated and stops
' all motors.
IF INP(P + 1) > 127 THEN GOSUB 620: GOTO 540
' checks to see if the sc 159 has any thing to say.  It usually sends out
' 95 and goes to 156 if it has someing to say.  Remember GOSUB 620: GOTO 540
' is the THEN statement block
OUT P, x: NEXT y: RETURN
' outputs ASCII code one letter at a time to the card
IF (INP(P + 1) AND 2) = 0 THEN A$ = "e" + CHR$(13): GOSUB 500
IF (INP(P + 1) AND 64) = 0 THEN PRINT CHR$(INP(P));
z = INP(P + 1)
FOR i = 1 TO 3: NEXT i
RETURN

P = 512
PRINT
PRINT
PRINT
PRINT "t) Take data"
PRINT "r) Return to SC-149 Command Mode"
PRINT "q) Quit to DOS"
LINE INPUT "? "; f$
PRINT
IF f$ = "t" THEN
GOTO 645
ELSEIF f$ = "q" THEN
GOTO 800
ELSEIF f$ = "r" THEN
GOTO 10
ELSE
GOTO 640
END IF
INPUT "Input sampling rate "; x
LINE INPUT "Input prism file name "; c$
= ".dat"
= c$ + d$
OPEN e$ FOR OUTPUT AS #1
PRINT
OUT P, 0: d = INP(P)
H = INP(P + 1)
L = INP(P)
K = (L + (H * 256))
PRINT USING "#####"; K
WRITE #1, K
PRINT
FOR z = 0 TO x: NEXT z
= INKEY$
IF A$ = "q" THEN 750
```

```
GOTO 720
CLOSE #1
GOTO 640
END
```

Visual Basic control program

```
Declare Function Inp Lib "inpout.dll" (ByVal Port%) As Integer
Declare Sub out Lib "inpout.dll" (ByVal Port%, ByVal Value%)
Dim s2(35), f2(35), y2(35), u(100), X(35), Y(35) As Double
Global reply, step3, step4
Global flag As String


Sub getcalfile (s2(), f2(), numinfi)
Debug.Print numinfi;
Rem Open "c:\cal.dta" For Input As #1
Rem Input #1, numinfi
Rem For conter = 1 To numinfi
Rem Input #1, s2(conter), f2(conter)
Rem Next conter
Rem 235 Close #1
End Sub


Sub listening (flagal, reply)
'Debug.Print "called by: "; flagal
For mqe = 1 To 3000: Next mqe
reply = ""
937 If (Inp(522 + 1) And 64) = 0 Then
reply = reply + Chr$(Inp(522))
Else
GoTo 938
End If
For mqe = 1 To 50: Next mqe
GoTo 937
938 If Right$(reply, 1) = Chr$(13) Then
reply = Left$(reply, Len(reply) - 1)
If flagal = 0 Then form1!receive.Text = reply
If flagal = 1 Then
```

```
form1!Text1.Text = Int(Val(reply) / 41#)
helio!heliopos.Text = reply
step3 = reply
End If
If flagal = 2 Then
form1!Text2.Text = Int(Val(reply) / 41#)
step4 = reply
Rem tdlfrm!tdlposbox.Text = reply
End If
If flagal = 3 Then
form1!Text3.Text = reply
helio!heliospdbox.Text = reply
End If
End If
939 End Sub


Sub Main ()
form1.Show      'Optionally, you can load and display a form.
Rem For dlay = 1 To 10: Next dlay
Call getcalfile(s2(), f2(), A)
Rem Call SPLINE(s2(), f2(), A, 1E+30, 1E+30, y2())
Do While DoEvents()
ww = ww + 1
If ww = 500 Then
outptr$ = "pos3"
'  Debug.Print "from1"
Call talking(outptr$)
For mq = 1 To 1000: Next mq
Call listening(1, resp)
End If
If ww = 1000 Then
outptr$ = "sp3"
Call talking(outptr$)
For mq = 1 To 1000: Next mq
' Debug.Print "from3"
Call listening(3, resp)
End If
If ww = 1500 Then
outptr$ = "pos4"
Call talking(outptr$)
For mq = 1 To 1000: Next mq
'  Debug.Print "from2"
```

```
Call listening(2, resp)
Rem resp1 = Val(resp)
'  Debug.Print resp1, resp
Rem Call SPLINT(s2(), f2(), y2(), A, resp1, resp2)
Rem resp3 = Str$(resp2)
Rem form1!text3.Text = resp3
Rem tdlfrm!Text2.Text = resp3
ww = 0
End If
'Place idle-loop code here. These statements are processed
'whenever the system has free time
'Debug.Print "i'm in the idle loop " + r
'r = r + 1
' if r > 32000 Then r = 0
'let's listen to the sc149 since we are not doing anything
' Call listening(0)
Rem Call heliogetpos
Loop
End Sub


Sub SPLINT (XA(), YA(), Y2A(), N, X, Y)
KLO = 1
KHI = N
50200    If KHI - KLO > 1 Then
K = (KHI + KLO) / 2
If XA(K) > X Then
KHI = K
Else
KLO = K
End If
GoTo 50200
End If
H = XA(KHI) - XA(KLO)
If H = 0# Then Exit Sub
A = (XA(KHI) - X) / H
B = (X - XA(KLO)) / H
Y = A * YA(KLO) + B * YA(KHI)
Y = Y + ((A ^ 3 - A) * Y2A(KLO) + (B ^ 3 - B) * Y2A(KHI)) * (H ^ 2) / 6#
End Sub


Sub talking (motorcmd)
```

```
'Debug.Print "command received: "; motorcmd
If motorcmd = "" Then 32001
For mqe = 1 To 5000: Next mqe
motorcmd = motorcmd + Chr$(13)
For yyy = 1 To Len(motorcmd)
For w = 1 To 3500:  Next w: Rem Delay here
xxx = Asc(Mid$(motorcmd, yyy, 1))
' Debug.Print x, Chr$(x)
551 out 522, xxx: Next yyy: GoTo 32001
32001 Rem ****** time to quit talking to sc149******
End Sub
```

Interactive Data Language (IDL), data visualization and analysis programs

baselinerough.pro

```
;********** baselinerough.pro ********************
;********** use this program to rough fix baseline from the 5-meter spectrometer
; baselinerough is the last program


PRO plotinteractive_doplot, pstate


case 1 of

total((*pstate).newy) ne 0.0: begin
   plot, (*pstate).xl, (*pstate).newy, color=0, background=255,$
   title='Normalized Spectra '+ (*pstate).fname, pos = [0.05,0.05,0.95,0.95]
   end
total((*pstate).func) ne 0.0: begin
   plot, (*pstate).xl,  (*pstate).yl,  color=0, background=255, title='Baseline Fit '+
(*pstate).fname,$
   pos = [0.05,0.05,0.95,0.95]
   plots, (*pstate).posx, (*pstate).posy, psym=6
   plots, (*pstate).xl, (*pstate).func, color=0
   end

else: begin
plot, (*pstate).xl, (*pstate).yl, PSYM = (*pState).psym, color=0, background=255,$
title='Raw Data '+ (*pstate).fname, pos = [0.05,0.05,0.95,0.95]
end
endcase

END

;****************************************

PRO plotinteractive_print, event
widget_control, event.top, get_uvalue = pstate
oktoprint = dialog_printersetup()
if oktoprint then begin
oldDev = !d.name
```

```
set_plot, 'PRINTER'
plotinteractive_doplot, pstate
device, /close_document
set_plot, oldDev
wset, (*pState).winID
endif
END

PRO plotinteractive_postscript, event
widget_control, event.top, get_uvalue = pstate

fname=strmid((*pstate).fname,0,7)
devkeywords = ps_form(CANCEL = cancel, GROUP_LEADER = event.top, filename='BAS'+fname +
'.ps')
;***************************************
if (not cancel) then begin
oldDev = !d.name
set_plot, 'ps'
device, _extra = devkeywords
plotinteractive_doplot, pState
device, /close
set_plot, oldDev
wset, (*pState).winID
endif
END


PRO plotinteractive_table, event
widget_control, event.top, get_uvalue = pState

; get the value of the data in the table
widget_control, event.id, get_value = ydata

; reset the ydata in the state structure
(*pState).yl = ydata
plotinteractive_doplot, pState

END

PRO plotinteractive_resize, event
widget_control, event.top, get_uvalue = pstate

; to propertly resize a draw widget, the geometry of the bases
; surrounding the draw must be determined and subtracted.
statusg = widget_info((*pState).status, /GEOMETRY)
controlg = widget_info((*pState).controlbase, /GEOMETRY)
tlbg = widget_info(event.top, /GEOMETRY)

; subtract out the xpadding of the tlb for the xsize
newx = event.x - (2*tlbg.xpad)

; subtract the size and padding of the bases
newy = event.y - statusg.scr_ysize - (2*statusg.ypad) - $
             controlg.scr_ysize - (2*controlg.ypad) - $
             (2*tlbg.ypad)
```

```
; resize the draw widget
widget_control, (*pState).draw, XSIZE = newx, YSIZE = newy

; re-plot
plotinteractive_doplot, pState

END


PRO plotinteractive_sym, event

; get the pointer to the data structure from the UVALUE
; (data slot) of the top-level base
widget_control, event.top, get_uvalue = pstate

; store the new linestyle
(*pstate).psym = event.index

; replot
plotinteractive_doplot, pstate

END

;*****************************************************************************

;*** Plotinteractive_draw put the current cursor position in that status window
; and lets you pick the points of the baseline and stores them in (*pstate).posx,
;(*pstate).posy ***

PRO plotinteractive_draw, event
widget_control, event.top, GET_UVALUE = pState
m=(*pstate).m

; check for motion events
if (event.type eq 2) then begin
; convert the coordinates from device to data
datac = convert_coord(event.x, event.y, /DEVICE, /TO_DATA)
str = strtrim(datac[0],2)+','+strtrim(datac[1],2)
widget_control, (*pstate).status, SET_VALUE = str
endif

if (event.type eq 0) then begin
; convert the coordinates from device to data
datac = convert_coord(event.x, event.y, /DEVICE, /TO_DATA)
print, datac[0],datac[1]
plots, datac[0],datac[1], color=0, psym=6, /data
(*pstate).posx(m)=datac[0]
(*pstate).posy(m)=datac[1]
;*** m can go to 15 or 15 line fitting points ***
m=m+1
(*pstate).m=m
;widget_control, event.top, SET_UVALUE = pState
endif

END
```

```
;*** plotinteractive_fit, fits the baseline points in plotinteractive_draw ***


PRO plotinteractive_fit, event

widget_control, event.top, get_uvalue = pstate

; button on - fits baseline data point
if (event.select) then begin
m=(*pstate).m
xpos=(*pstate).posx
ypos=(*pstate).posy
xpos=xpos(0:m-1)
ypos=ypos(0:m-1)
(*pstate).posx=xpos
(*pstate).posy=ypos

func=SPLINE(xpos,ypos,(*pstate).x1,5.0)

;polyf=POLY_FIT(xpos, ypos, 3, yfit)
;func=polyf(0)+polyf(1)*(*pstate).x1+polyf(2)*(((*pstate).x1)^2)  $
;+polyf(3)*(((*pstate).x1)^3)

oplot, (*pstate).x1, func, color=0
(*pstate).func=func
;widget_control, event.top, SET_UVALUE = pState
; button off - flop neg sym to pos
endif else begin
nele=n_elements((*pstate).newy)
(*pstate).m=0
(*pstate).posx=fltarr(15)
(*pstate).posy=fltarr(15)
(*pstate).func=fltarr(nele)
print, 'Reset Baseline'
plotinteractive_doplot, pstate

endelse

END


PRO plotinteractive_subtract, event
widget_control, event.top, get_uvalue = pstate

;subtract baseline
if (event.select) then begin

y2=(1.0-((*pstate).func-(*pstate).y1)/(*pstate).func);-1.0
(*pstate).newy=y2
plotinteractive_doplot, pstate
plot, (*pstate).newy, color=0, background=255

end else begin
nele=n_elements((*pstate).newy)
(*pstate).newy=fltarr(nele)
plotinteractive_doplot, pstate
```

```
;oplot, (*pstate).posx,(*pstate).posy, psym=6, color=0
;oplot, (*pstate).x1, (*pstate).func, color=0
endelse

END

PRO plotinteractive_write, event
widget_control, event.top, GET_UVALUE = pstate
posx=(*pstate).posx
posy=(*pstate).posy
func=(*pstate).func
newy=(*pstate).newy
m=(*pstate).m
x1=(*pstate).x1
y1=(*pstate).y1
file='S'+(*pstate).fname
fname=strmid(file,0,8)
save, posx, posy, func, newy, m, x1, y1, filename=fname+'.sav'
openw, 1 , file
for n=0,N_ELEMENTS((*pstate).newy)-1 do begin
printf, 1, (*pstate).newy(n)
endfor
close,/all
widget_control, event.top, /destroy
END


;***************************************************************************************
;*********  Main Program  *************

PRO baselinerough

;fname='John Paul:RSI:IDL 5.3:raw 5meter data:1/21:r0121n2.dat'
;fname='Macintosh G3:RSI:IDL 5.1:raw 5meter data:1/21:r0121n2.dat'
;fpath='Macintosh HD:RSI:IDL 5.1:f0425ne.dat'
;fname='f0425ne.dat'
fpath=dialog_pickfile(path='John Paul:RSI:IDL 5.3:Baseline data:', get_path=path,$
title = 'Pick a raw data *.dat file', filter = '*.dat')
print, fpath
print, path
fname=strmid(fpath,strlen(path))
print, fpath
print, fname
data1=READ_ASCII(fpath)
nele=N_ELEMENTS(data1.field1)
x1=findgen(nele)
y1=data1.field1


; top-level base - add resize events
tlb = widget_base(TITLE = 'Suntracker Data Analysis', /COLUMN, $
/TLB_SIZE_EVENTS, MBAR = menubase)

fileMenu = widget_button(menubase, VALUe = 'File', /MENU)
printButton = widget_button(fileMenu, VALUE = 'Print', $
EVENT_PRO = 'plotinteractive_print')
```

```
psbutton = widget_button(fileMenu, VALUE = 'PostScript', $
EVENT_PRO = 'plotinteractive_postscript')

; main draw widget
status = widget_label(tlb, VALUE = '', /DYNAMIC_RESIZE)
draw = widget_draw(tlb, XSIZE = 1050, YSIZE = 600, $
/BUTTON_EVENTS, /MOTION_EVENTS, EVENT_PRO = 'plotinteractive_draw')

; create a row base at the bottom of the display to hold
; the widgets to control the plot
controlbase = widget_base(tlb, /ROW)

; define a hexagon for symbol 8
usersym, [1, 0.5, -0.5, -0.5, 0.5, 1], [0,1,1,0,-1,-1]
symbols = ['None','Plus sign (+)','Asterisk (*)','Period (.)','Diamond', $
          'Triangle','Square','X','Hex']
symboldrop = widget_droplist(controlbase, VALUE = symbols, $
TITLE = 'Symbol', EVENT_PRO = 'plotinteractive_sym')

; create a nonexclusive base to hold the toggle button for
; symbol connections
connectbase = widget_base(controlbase, /NONEXCLUSIVE, /ROW)
fitbutton = widget_button(connectbase, VALUe = 'Fit Baseline Points', $
EVENT_PRO = 'plotinteractive_fit')
subtractbutton = widget_button(connectbase, VALUe = 'Subtract Baseline', $
EVENT_PRO = 'plotinteractive_subtract')
writebutton = widget_button(controlbase, VALUe = 'Write data', $
EVENT_PRO = 'plotinteractive_write')

; create a new tlb for the interactive plot editor
tlb2 = widget_base(TITLE = 'Plot Editor', /ROW, GROUP_LEADER = tlb)
plotdatatable = widget_table(tlb2, XSIZE = n_elements(y1), YSIZE = 1, $
    X_SCROLL_SIZE = 10, Y_SCROLL_SIZE = 1, $
/EDITABLE, /RESIZEABLE_ROWS, /RESIZEABLE_COLUMNS, $
VALUE = y1, EVENT_PRO = 'plotinteractive_table', $
ROW_LABELS = string(x1,FORMAT = '(f6.2)'))

; realize the top-level bases
widget_control, tlb, /REALIZE
;widget_control, tlb2, /REALIZE

; get the window ID of the draw widget
widget_control, draw, GET_VALUE = winID

; set the draw widget to be the current window and draw the plot
; with all the default settings
wset, winID

m=0
posx=fltarr(30)
posy=fltarr(30)
func=fltarr(nele)
newy=fltarr(nele)

; create a structure of data for the application
state = {x1:x1, y1:y1, winID:winID, psym:0, $
status:status, controlbase:controlbase, draw:draw, m:m, $
posx:posx, posy:posy, func:func, newy:newy, fname:fname}

; create a pointer to the state structure and put that pointer
; into the UVALUE (data slot) of the top-level base
pstate = ptr_new(state, /no_copy)

plotinteractive_doplot, pstate

widget_control, tlb, SET_UVALUE = pState
widget_control, tlb2, SET_UVALUE = pState

; call xmanager to start the event handling procedure
Xmanager, 'baselinerough', tlb, EVENT_HANDLER = 'plotinteractive_resize'
END
```

Baselinefine.pro

```
; program for the P-branch of the 2nu1 band of N2O
;
; This program takes a peak value file in the format pkx, pky and changes the x-axis
values
; to wavenumbers.  It calculates the area of each peak and stores it into a text file
;
; must compile "alt2.pro'
;

PRO plotinteractive_doplot, pstate

;********* contribution **************
; The case statement goes from top to bottom, first true statement it executes
case 1 of

total((*pstate).newx) ne 0.0: begin
   plot, (*pstate).newx, (*pstate).newy, color=0, background=255, title='Normalized
Spectra '+ (*pstate).fname,$
   pos = [0.05,0.1,0.95,0.95], subtitle= 'Wavenumbers'
   n_ele= n_elements((*pstate).pkx)

   for u = 0, n_ele-1 do begin
      xyouts, (*pstate).pkx(u), ((*pstate).pky(u)-0.01), $
      strcompress(string(fix((*pstate).pktable(u))), /remove_all), /data, color = 0
   endfor
   end
```

```
total((*pstate).newy) ne 0.0: begin
   plot, (*pstate).x1, (*pstate).newy, color=0, background=255, title='Normalized Spectra
'+ (*pstate).fname,$
   pos = [0.05,0.05,0.95,0.95]
   plots,(*pstate).pkx, (*pstate).pky, psym=4, color = 0
   end
total((*pstate).func) ne 0.0: begin
   plot, (*pstate).x1,  (*pstate).y1,  color=0, background=255, title='Baseline Fit  '+
(*pstate).fname,$
   pos = [0.05,0.05,0.95,0.95]
   plots, (*pstate).posx, (*pstate).posy, psym=6
   plots, (*pstate).x1, (*pstate).func, color=0
   end

else: begin
plot, (*pstate).x1, (*pstate).y1, PSYM = (*pState).psym, color=0, background=255,$
title='Raw Data '+ (*pstate).fname, pos = [0.05,0.05,0.95,0.95]
end

endcase

END
;****************************************

PRO plotinteractive_print, event
widget_control, event.top, get_uvalue = pstate
oktoprint = dialog_printersetup()
if oktoprint then begin
oldDev = !d.name
set_plot, 'PRINTER'
plotinteractive_doplot, pstate
device, /close_document
set_plot, oldDev
wset, (*pState).winID
endif
END

PRO plotinteractive_postscript, event
widget_control, event.top, get_uvalue = pstate
;********* my contribution **************
; Changes the filename in the ps_form function to 'S'+fname + '.ps'
fname=strmid((*pstate).fname,0,8)
devkeywords = ps_form(CANCEL = cancel, GROUP_LEADER = event.top, filename='NOR'+fname +
'.ps')
filname='NOR'+fname + '.ps'
print, 'Postscript file: '+filname
;****************************************
if (not cancel) then begin
oldDev = !d.name
set_plot, 'ps'
device, _extra = devkeywords
plotinteractive_doplot, pState
device, /close
set_plot, oldDev
wset, (*pState).winID
endif
```

```
END


PRO plotinteractive_table, event
widget_control, event.top, get_uvalue = pState

; get the value of the data in the table
widget_control, event.id, get_value = pktable

; reset the ydata in the pstate structured varible
(*pState).pktable = pktable
plotinteractive_doplot, pState

END

PRO plotinteractive_resize, event
widget_control, event.top, get_uvalue = pstate

; to propertly resize a draw widget, the geometry of the bases
; surrounding the draw must be determined and subtracted.
statusg = widget_info((*pState).status, /GEOMETRY)
controlg = widget_info((*pState).controlbase, /GEOMETRY)
tlbg = widget_info(event.top, /GEOMETRY)

; subtract out the xpadding of the tlb for the xsize
newx = event.x - (2*tlbg.xpad)

; subtract the size and padding of the bases
newy = event.y - statusg.scr_ysize - (2*statusg.ypad) - $
            controlg.scr_ysize - (2*controlg.ypad) - $
            (2*tlbg.ypad)

; resize the draw widget
widget_control, (*pState).draw, XSIZE = newx, YSIZE = newy

; re-plot
plotinteractive_doplot, pState

END


PRO plotinteractive_sym, event

; get the pointer to the data structure from the UVALUE
; (data slot) of the top-level base
widget_control, event.top, get_uvalue = pstate

; store the new linestyle
(*pstate).psym = event.index

; replot
plotinteractive_doplot, pstate

END

;*****************************************************************
```

```
;*****************************************************************

PRO plotinteractive_draw, event
widget_control, event.top, GET_UVALUE = pState


; check for motion events
if (event.type eq 2) then begin
; convert the coordinates from device to data
datac = convert_coord(event.x, event.y, /DEVICE, /TO_DATA)
str = strtrim(datac[0],2)+','+strtrim(datac[1],2)
widget_control, (*pstate).status, SET_VALUE = str
endif

END



PRO plotinteractive_write, event
widget_control, event.top, GET_UVALUE = pstate

;***************************
; plotinteractive_write: removes bad peak data, calculates x-axis in
; wavenumbers, fills in P line values, reverses scans with
; increaseing P line values, calculates zenith angles, calculates area
; of each peak
;

;***** variables ************
; x1= x-axis data points
; pkx= peak x-axis data points
; pky= peak y-axis data points
; pkx3= peak x-axis data points with bad data removed
; n_ele4=n_elements(pkx3)
; pktable= peak x-axis data points
; n_ele=n_elements(pktable)
; pktable2= line # that you put in
; pktable3= peak x-axis wave# points
; pkhitran= 2 dimensional array of P branch lines,
; row 0->(0-40),row 1->(wave# of each line)
; n_ele2=n_elements(pkhitran)/2
; newx= the x-axis in wave#
; newy= the y-axis in absorption units
; n_ele3=n_elements(newy)
; dpkhitran= 2 dimensional array of P branch troughs,
; row 0->(0-40),row 1->(wave# of each trough),
; dpkhitran(1,0) is the wave# in between P0 and P1, dpkhitran(0,0) is 0
; zenith= zenith angle of the sun of each line
; secant= the secant of zenith
; shour2= hour start time in 1/100
; shour= hour of start time (integer)
; smins= minutes of start time (integer)
; ehour2= hour end time in 1/100
; ehour= hour of end time (integer)
; emins= minutes of end time (integer)
; pky3= y value of peaks in absorption units (neg. values)
; pkhour2= hour of peak time in 1/100
; pkhour= hour of peak time (integer)
; pkmins= minutes of peak time (integer)
; normz= area of each line normalize to 1 air mass
;*******************************

;save, newx, newy, intg2, pktable2, pktable3, dpkhitran, zenith, secant, shour2, $
;ehour2, pky3, pkhour, pkhour2, pkmins, normz, filename=file2+'.sav'


newy=(*pstate).newy
newx=(*pstate).newx
x1=(*pstate).x1
pkx=(*pstate).pkx
pky=(*pstate).pky
pktable=(*pstate).pktable
restore, 'pkdat.sav'
pkhitran=pkdata2
n_ele=n_elements(pktable)
n_ele2=n_elements(pkhitran)/2
n_ele3=n_elements(newy)
pktable2=dblarr(n_ele)
pkx3=pkx
pky3=pky
k=0

;********** get rid of negative elements (non-N2O lines) in pktable,
; and take out the same element in pkx, pky **************
for m=0,n_ele-1 do begin
if pktable(m) ge 0.0 then begin
   pktable2(k) = pktable(m)
   pkx3(k)  = pkx(m)
   pky3(k)  = pky(m)
   k=k+1
endif
endfor
print, k
pktable2=pktable2(0:k-1)
pkx3=pkx3(0:k-1)
pky3=pky3(0:k-1)
print, pktable2
n_ele4=n_elements(pkx3)
pktable3=pktable2

; ******* checks for adjacent marked lines, fills in P line values
; reverses scans with increaseing P line values *************
switch0 = 0
for n=0,k-2 do begin
if pktable2(n) ne 0.0 then begin
   case 1 of

      pktable2(n+1) eq 0.0 : begin
         result = dialog_message('You must have two adjacent lines marked')
         widget_control, event.top, /destroy
      end
      pktable2(n) gt pktable2(n+1): begin
         for t= 0,n_ele4-1 do begin
            pktable2(t) =pktable2(n)+n-t
         endfor
         goto, jump
      end

      pktable2(n) lt pktable2(n+1): begin
         pkx3=reverse(pkx3)
         pkx4=pkx3
         for z=0,n_ele4-1 do begin
            pkx3(z)=n_ele3-pkx4(z)
         endfor
         pky3=reverse(pky3)
         newy=reverse(newy)
         for t= 0,n_ele4-1 do begin
            pktable2(t) =pktable2(n)-n+t
         endfor
         pktable2=reverse(pktable2)
         switch0 = 1
         goto, jump
      end
      else : print, 'its a zero'
   endcase
endif
endfor

jump  : print, 'jump'

print, pktable2
(*pstate).pktable=pktable2
widget_control, (*pstate).pdt, SET_VALUE = pktable2
; Assigns wavenumbers to x-axis peak values of pktable2
for u= 0,n_ele4-1 do begin
pktable3(u) = pkhitran(1,pktable2(u))
endfor

;******* change (*pstate).pkx-y so you plot only the peaks that you assigned
(*pstate).pkx=pktable3
(*pstate).pky=pky3

;******* change the x-axis in to wavenumbers *******


;******  changing the front x-axis in to wavenumbers ******
lfit=ladfit(pkx3(0:1),pktable3(0:1), /double)
for h=0,pkx3(0)-1 do begin
newx(h)=lfit(1)*xl(h) + lfit(0)
endfor


;******  changing the middle x-axis in to wavenumbers ******
for g=0,n_ele4-2 do begin
lfit2 =ladfit(pkx3(g:g+1),pktable3(g:g+1),/double)
for f=pkx3(g),pkx3(g+1)-1 do begin
   newx(f)=lfit2(1)*xl(f)+lfit2(0)
```

```
endfor
endfor

;******  changing the end x-axis in to wavenumbers *******
for e = pkx3(k-1), n_ele3-1 do begin
newx(e) = lfit2(1)* x1(e)+lfit2(0)
endfor
;******************************************


(*pstate).newy=newy
(*pstate).newx=newx

;********** calculate the zenith angle for each peak *********

plotinteractive_doplot, pstate

; calculate the wavenumbers in between peaks with Hitran P line values
dpkhitran = dblarr(2,n_ele2-1)
for p = 0,n_ele2-2 do begin
dpkhitran(0,p) = pkhitran(0,p)
dpkhitran(1,p) = ((pkhitran(1,p)-pkhitran(1,p+1))/2.0)+pkhitran(1,p)
endfor

n_ele5 = n_elements(dpkhitran)/2
xblk=dblarr(n_ele5, 500)
yblk=dblarr(n_ele5, 500)
xblk2 = dblarr(500)
yblk2 = dblarr(500)

intg=dblarr(n_ele4)
intg2=dblarr(n_ele4)

; ******** Divide data into blocks around each peak and integrate
; ****** dpkhitran(1,s+1) is smaller than dpkhitran(1,s)

for s=0,n_ele5-2 do begin
w=0
for r=0,n_ele3-1 do begin
   if newx(r) gt dpkhitran(1,s+1) and newx(r) lt dpkhitran(1,s) then begin
      xblk(s,w)=newx(r)
      yblk(s,w)=newy(r)
      w=w+1
   endif
endfor
endfor

; reduce 'xblk' 2-D blocks to 'xblk2' 1-D blocks to integrate
b=0
for d= pktable2(0),pktable2(n_ele4-1),-1 do begin
xblk2 = dblarr(500)
yblk2 = dblarr(500)

for c= 0, 499 do begin
   if xblk(d,c) ne 0.0 then begin
   xblk2(c)=xblk(d,c)
```

```
    yblk2(c)=yblk(d,c)
    endif
endfor

; get rid of the elements with 0.0 values
xn=n_elements(uniq(xblk2))
yn=n_elements(uniq(yblk2))
if yn gt xn then yn=yn-1
if xn gt yn then xn=xn-1
; we have to use xn-2 because 'uniq' function says the last element is unique
; which it is not.  So it give one extra element or index
xblk2=xblk2(0:xn-2)
yblk2=yblk2(0:yn-2)

intg2(b)= int_tabulated(xblk2,yblk2)
b=b+1
endfor


print, ' '
for a= 0, n_ele4-1 do begin
print, pktable2(a), intg2(a)
print, ' '
endfor

; get filenames and make filenames
file=(*pstate).fname
fname=strmid(file,0,8)
fname2=strmid(file,1,7)
file2='I'+fname2
file3=file2+'.dat'

;************* calculate the zenith angle using  alt2 **********
; get month and day from file name
month=float(strmid(file,2,2))
day= float(strmid(file,4,2))

; read in from keyboard start and end times
valid = 0
on_ioerror, bad_num
read, shour, smins, prompt = 'Put in start time (hour, mins) ' + fname2 +':  '
read, smplrt, prompt = 'Put in sampling rate ' + fname2 +':  '
valid = 1
bad_num  :  if not valid then result  =  dialog_message('Unexpected hour and mins encountered',/error)
; message, 'Unexpected hour and mins encountered'

hour = dblarr(n_ele3)
shour2 =shour + smins/60.0
;ehour2 = ehour + emins/60.0
case smplrt of

1000: ehour2 = (n_ele3/446.0)/60.0 + shour2

1500: ehour2 = (n_ele3/301.0)/60.0 + shour2
```

```
2000: ehour2 = (n_ele3/228.0)/60.0 + shour2

2500: ehour2 = (n_ele3/184.0)/60.0 + shour2

3500: ehour2 = (n_ele3/130.0)/60.0 + shour2

4000: ehour2 = (n_ele3/114.0)/60.0 + shour2

8000: ehour2 = (n_ele3/57.0)/60.0 + shour2

else: result = dialog_message('Unexpected sampling rate encountered',/error)

endcase

ehour = floor(ehour2)
emins = (ehour2-ehour) * 60.0

;***** reverses start and end times if scan was reversed *****
if switch0 eq 1 then begin
h = ehour2
dhour = (shour2 - ehour2)/n_ele3
print, ' hours reversed'
endif else begin
h = shour2
dhour = (ehour2 - shour2)/n_ele3
endelse

;***** assigns times to each data point
for n = 0, n_ele3-1 do begin
hour(n) = h
h = h +dhour
endfor

pkhour = dblarr(n_ele4)
pkmins = dblarr(n_ele4)
altitude = dblarr(n_ele4)
;***** assigns times to each peak
pkhour = hour(pkx3)

pkhour2 = floor(pkhour)
for m = 0, n_ele4-1 do begin

pkmins(m) = (pkhour(m) - pkhour2(m)) * 60.0
;print, pkmins(m)
altitude(m) = alt2(month, day, pkhour2(m), pkmins(m))

endfor

zenith=(90.0-altitude)

secant = 1.0/cos(zenith*!DTOR)
; normalize area for each peak
normz = intg2/secant

; save .sav file write text files
;'I' files: P data # value, area, start time, end time, zenith value, normalize area,
; 'NW' files: newx, newy

save, newx, newy, intg2, pktable2, pktable3, dpkhitran, pkhitran, zenith, secant, shour2,
$
ehour2, pky3, pkhour, pkhour2, pkmins, normz, filename=file2+'.sav'

for ac= 0, n_ele4-1 do begin
print, fix(pktable2(ac)), intg2(ac), pkhour(ac), zenith(ac), normz(ac)
endfor

openw, 1 , file3
for aa= 0, n_ele4-1 do begin
printf, 1, fix(pktable2(aa)), intg2(aa), pkhour(aa), zenith(aa), normz(aa)
endfor
close,/all

file4= 'NW' + fname2 + '.dat'
openw, 2 , file4
for ab= 0, n_ele3-1 do begin
printf, 2, newx(ab), newy(ab)
endfor
close,/all
END

pro plotinteractive_exit, event

widget_control, event.top, /destroy

end


;*****************************************************************************************

PRO baselinefine

;********* Read in the position of peaks from Origin file *********
ppath=dialog_pickfile(path='John Paul:RSI:IDL 5.3:Baseline data:', get_path=path,$
title='Choose peak position file or Pf_.dat file', filter = '*.dat')
;ppath=dialog_pickfile(path='Macintosh G3:RSI:IDL 5.1:Baseline data:', get_path=path,$
;title='Choose peak position file or Pf_.dat file', filter = '*.dat')
peakdata = READ_ASCII(ppath)

pkx2 = peakdata.field1(0:0,0:*)
pky2 = peakdata.field1(1:1,0:*)
duck= n_elements(pkx2)
pkx = dblarr(duck)
pky = dblarr(duck)
for n = 0, duck-1 do begin
pkx(n)=pkx2(0,n)
pky(n)=pky2(0,n)
endfor

;pky = double(sfdpky(0,*))
pktable = dblarr(n_elements(pkx2))
```

```
;*************** Restore all variables ***********
print, ppath
print, path
pname=strmid(ppath,strlen(path))
print, pname
fname= 'S' + strmid(pname,1,7) + '.sav'
print, fname
; i.e. fname is equal to Sf0125n5.sav
restore, fname

nele=N_ELEMENTS(newy)
x1=findgen(nele)
y1=newy
newx=dblarr(nele)


; top-level base - add resize events
tlb = widget_base(TITLE = 'Suntracker Data Analysis', /COLUMN, $
/TLB_SIZE_EVENTS, MBAR = menubase)



fileMenu = widget_button(menubase, VALUe = 'File', /MENU)
printButton = widget_button(fileMenu, VALUE = 'Print', $
EVENT_PRO = 'plotinteractive_print')
psbutton = widget_button(fileMenu, VALUE = 'PostScript', $
EVENT_PRO = 'plotinteractive_postscript')

; main draw widget
status = widget_label(tlb, VALUE = '', /DYNAMIC_RESIZE)
draw = widget_draw(tlb, XSIZE = 1050, YSIZE = 600, $
/BUTTON_EVENTS, /MOTION_EVENTS, EVENT_PRO = 'plotinteractive_draw')

; create a row base at the bottom of the display to hold
; the widgets to control the plot
controlbase = widget_base(tlb, /ROW)

; define a hexagon for symbol 8
usersym, [1, 0.5, -0.5, -0.5, 0.5, 1], [0,1,1,0,-1,-1]
symbols = ['None','Plus sign (+)','Asterisk (*)','Period (.)','Diamond', $
           'Triangle','Square','X','Hex']
symboldrop = widget_droplist(controlbase, VALUE = symbols, $
TITLE = 'Symbol', EVENT_PRO = 'plotinteractive_sym')

; create a nonexclusive base to hold the toggle button for
; symbol connections
connectbase = widget_base(controlbase, /NONEXCLUSIVE, /ROW)

;writebutton = widget_button(controlbase, VALUe = 'Write data', $
;EVENT_PRO = 'plotinteractive_write')

; create a new tlb for the interactive plot editor
tlb2 = widget_base(TITLE = 'Plot Editor', /ROW, GROUP_LEADER = tlb)

plotdatatable = widget_table(tlb2, XSIZE = n_elements(pkx), YSIZE = 1, $
    X_SCROLL_SIZE = 10, Y_SCROLL_SIZE = 1, $
/EDITABLE, /RESIZEABLE_ROWS, /RESIZEABLE_COLUMNS, $
VALUE = pktable, EVENT_PRO = 'plotinteractive_table', $
ROW_LABELS = ['pk position', 'P Quantum no.'],$
COLUMN_LABELS = STRING(pkx, format= '(i5)'))

wtext=widget_text(tlb2, value='Input neg. # for non N2O peaks',xsize=10,/wrap)

writebutton2 = widget_button(tlb2, VALUe = 'Write data', $
EVENT_PRO = 'plotinteractive_write')

exitbutton2 = widget_button(controlbase, VALUe = 'EXIT', $
EVENT_PRO = 'plotinteractive_exit')


; realize the top-level bases
widget_control, tlb, /REALIZE
widget_control, tlb2, /REALIZE


; get the window ID of the draw widget
widget_control, draw, GET_VALUE = winID

; set the draw widget to be the current window and draw the plot
; with all the default settings
wset, winID


; create a structure of data for the application
state = {x1:x1, y1:y1, winID:winID, psym:0, pdt:plotdatatable, $
status:status, controlbase:controlbase, draw:draw, $
pkx:pkx, pky:pky, func:func, newx:newx, newy:newy, fname:fname,$
pktable:pktable}

; create a pointer to the state structure and put that pointer
; into the UVALUE (data slot) of the top-level base
pstate = ptr_new(state, /no_copy)

plotinteractive_doplot, pstate
plots, pkx, pky, psym=4, color = 0
widget_control, tlb, SET_UVALUE = pState
widget_control, tlb2, SET_UVALUE = pState

; call xmanager to start the event handling procedure
Xmanager, 'baselinefine', tlb, EVENT_HANDLER = 'plotinteractive_resize'

END
```

Alt2.pro

```
; calculates the altitude of the sun at a certain date and time

function alt2, month, day, hour, mins

year = 2001
;month = 6
;day = 18
;hour = 14
;mins = 51
longtd = -83.9217
lattd = 35.9541

;year = 1997
;month = 8
;day = 7
;hour = 11
;mins = 0
;longtd = -1.91667
;lattd = 52.5

If month lt 4 then begin
hour = hour + 5.0
endif else begin
;daylight savings time
hour = hour + 4.0
endelse


d = double(julday(month, day, year, hour, mins)-2451545)

;print, 'julday', d

LST2 = (280.46061837 + 360.98564736629 * d + longtd)
LST = LST2 mod 360.0

L2 = 280.461 + 0.9856474 * d
L = (L2 mod 360.0) + 360.0

g2 = 357.528 + 0.9856003 * d
g = (g2 mod 360.0) + 360.0

lambda = L +1.915 * sin (g*!DTOR) + 0.02 * sin(2*g*!DTOR)
epsilon = 23.439 - 0.0000004 * d
Y = cos(epsilon*!DTOR) * sin(lambda*!DTOR)

X = cos(lambda*!DTOR)

a = atan(Y/X)*!RADEG
```

```
If X lt 0.0 then begin
  alpha = a + 180.0
  goto, jump
endif

If Y lt 0 and X gt 0 then begin
   alpha = a + 360
endif else begin
   alpha = a
endelse

jump: delta = asin(sin(epsilon*!DTOR)*sin(lambda*!DTOR))*!RADEG ;print, 'jump'

ha2 = LST - alpha
ha = ha2 mod 360.0

Al  =  sin(delta*!DTOR)  *  sin(lattd*!DTOR)  +  cos(delta*!DTOR)  *  cos(lattd*!DTOR)  *
cos(ha*!DTOR)

altitude = asin(Al)*!RADEG

;print, 'alt2: altitude ', altitude

return, altitude

end
```

Atm_model.pro

```
; Calculates synthetic spectra of the earth's atmosphere
; must open the following programs:
; humlik.fcn
; woplot.pro
; layer.pro


PRO reaprf, gas, file, prof
;
; 28-NOV-00  AD
; IDL procedure for reading in profile from RFM .atm file
;
;   gas  (in)  string    name of species (eg 'CH4', 'HGT')
;   file (in)  string    name of .atm file (eg 'std.atm')
;   prof (out) real arr  profile
;
sgas = '*' + STRUPCASE(gas) + ' '
l = STRLEN(sgas)
OPENR, 1, file
```

```
header = ' '
READF, 1 ,header
IF header eq '' then READF, 1 ,header
header = '!'
nlev = 0
WHILE STRMID(header,0,1) EQ '!' DO READF, 1, header
READS, header, nlev
prof = FLTARR(nlev)
WHILE STRMID(header,0,1) NE sgas DO BEGIN
  READF, 1, header
  IF STRLEN(header) LT 1 THEN header = header + ' '
  header = STRUPCASE(header)
ENDWHILE
READF, 1, prof
CLOSE, 1
END



pro prs2, start, sizelr, pr5, tmp5, n2o5, ch45, h2o5, co25

atm = 'std.atm'

reaprf, 'hgt', atm, alt
reaprf, 'pre', atm, pres2
reaprf, 'tem', atm, temp
reaprf, 'n2o', atm, n2o
reaprf, 'ch4', atm, ch4
reaprf, 'h2o', atm, h2o
reaprf, 'co2', atm, co2
pres = pres2/pres2(0)
n2o2 = n2o/1e6
ch42 = ch4/1e6
h2o2 = h2o/1e6
co22 = co2/1e6

km2 = 120
height = findgen(km2*100.0)/100.0  ; *** height with 100 pts/km

tempc = spline(alt,temp,height,3)
presc = spline(alt,pres,height,3)

n2o3 = spline(alt,n2o2,height,3)
ch43 = spline(alt,ch42,height,3)

h2o3 = spline(alt,h2o2,height,3)
co23 = spline(alt,co22,height,3)

km = 30.0 ; *** 'km' top of Atmosphere

;bottom = floor(start*100)
topp = km+start

pindx = where(height ge start and height lt topp, count)
print, count
```

```
pathh = height(pindx) ; *** height with 100 pts/km plus starting height
tmp4 = tempc(pindx)
pr4 = presc(pindx)
n2o4 = n2o3(pindx)
ch44 = ch43(pindx)

h2o4 = h2o3(pindx)
co24 = co23(pindx)

szlyrs = sizelr/1.0e5 ; *** 5km is the size of each layer, so szlyrs = 5.0
szlyrs2 = szlyrs*100.0
lyrs = floor(km/szlyrs) ; *** number of layers

pr5=fltarr(lyrs)
tmp5=fltarr(lyrs)
n2o5=fltarr(lyrs)
ch45=fltarr(lyrs)

h2o5=fltarr(lyrs)
co25=fltarr(lyrs)


count = 0
for n=0,lyrs-1 do begin

 pr5(n)= mean(pr4(count:count+(szlyrs2-1)))
 tmp5(n)= mean(tmp4(count:count+(szlyrs2-1)))
 n2o5(n)= mean(n2o4(count:count+(szlyrs2-1)))
 ch45(n)= mean(ch44(count:count+(szlyrs2-1)))
 h2o5(n)= mean(h2o4(count:count+(szlyrs2-1)))
 co25(n)= mean(co24(count:count+(szlyrs2-1)))
 count = count + szlyrs2

endfor
; starting from the bottom of the atmosphere and going up,
; because it doesn't matter which direction you go

end


function instruf, h, smplrt

; h eq hwhm in wavenumbers
; smplrt eq to the number of data points per wavenumber
rangenum = floor(smplrt * 1.5)
fwhm=h*2.0
stddev=fwhm/2.35 ; sigma

v2=findgen(rangenum)/smplrt
vo= v2(floor(rangenum/2.0))
g=(1/(sqrt(2*!pi)*stddev))*exp(-0.5*((v2-vo)^2)/(stddev)^2)

mark = where(g gt 1e-6, count)
g2 = g(mark)

return, g2
```

```
end


function readkittpk

ppath=dialog_pickfile(path='John  Paul:RSI:IDL 5.3:Syn.  Spec.:working folder:',
get_path=path,$
title='Choose Hitran Excel txt file', filter = '*.txt')
;ppath=dialog_pickfile(path='Macintosh G3:RSI:IDL 5.1:Syn. Spec.:', get_path=path,$
;title='Choose Hitran Excel txt file', filter = '*.txt')
data2 = READ_ASCII(ppath)

wavnum = data2.field1(0:0,0:*)
sun = data2.field1(1:1,0:*)
earth = data2.field1(2:2,0:*)
combin = data2.field1(3:3,0:*)

wavnum = reform(wavnum)
sun = reform(sun)
earth = reform(earth)
combin = reform(combin)

n_ele=n_elements(wavnum)

all = fltarr(4,n_ele)

all(0,*) = wavnum
all(1,*) = sun
all(2,*) = earth
all(3,*) = combin

return, all

end


; **************  MAIN PROGRAM ********************

pro hager3


;Using the formula I=Iexp-(k(v)x) where K(v)=Sf(v) and x=PXl, we calculate spectra.
;S is line strength, P is pressure, X is mixing ratio, l is path length, f(v) shape
function

conv=2.4464e19 ;conversion constant
n2ovolmix=3.5e-7*1.0 ; volume mixing ratio for nitrous oxide
ch4volmix=1.6E-6 ;1.6e-6 ; volume mixing ratio for methane

start = 0.29 ; km altitude of suntracker at UT;
;start = 2.095 ; km altitude of suntracker at Kitt Peak

bo=0.08 ; broadening coefficient (cm-1) at 1 atm.
;bd = 0.003556 ; Doppler broadening coefficient of CO at 296K
vpath=5.0e5 ; 5.0e5 cm = 5 km
```

```
;hwhm= 0.033*3; 0.033 hwhm in wavenumbers of instrument kernal at UT for 1120 micron
slits.
hwhm= 0.061*1.4; 0.061*1.4 hwhm in wavenumbers of instrument kernal at UT for 1850 micron
slits.(*2.1 after 0404)
;hwhm= 0.005 ;hwhm in wavenumbers of instrument kernal at kitt peak.
shifty=0
cn=1.0 - 0.00 ;n2o concentrations
cc=1.0 + 0.25;ch4 concentrations
ch=1.0 + 0.0
co=1.0 + 0.0
brn= 1.0

dpkhitran = fltarr(2,40)
filename = ' '

ppath=dialog_pickfile(path='John  Paul:RSI:IDL 5.3:Syn.  Spec.:working folder:',
get_path=path,$
title='Choose If_.sav file', filter = 'I*.sav')
fname = strmid(ppath, strlen(path)) ; i.e. If0327na.sav
filename = 'Z' + strmid(fname,1,12)
print, fname
restore, ppath
;***** variables from restore ************
; pktable2= line # that you put in
; pktable3= peak x-axis wave# points
; newx= the new x-axis in wave#
; newy= the new y-axis in absorption units (neg. values)
; intg2= area of each line
; zenith= zenith angle of the sun of each 2nu1 line
; secant= the secant of zenith
; shour2= hour start time in 1/100
; shour= hour of start time (integer)
; smins= minutes of start time (integer)
; ehour2= hour end time in 1/100
; ehour= hour of end time (integer)
; emins= minutes of end time (integer)
; pky3= y value of peaks in absorption units (neg. values)
; pkhour2= hour of peak time (integer)
; pkhour= hour of peak time in 1/100
; pkmins= minutes of peak time (integer)
; normz= area of each line normalize to 1 air mass
; filename= file name of .sav file
; dpkhitran= 2-D array, dpkhitran(0,*)= trough values, 1 is in between p1 and p2
; 0 is on the high side of p1, dpkhitran(1,*)= trough wavenumber values
;*******************************

n_ele3 = n_elements(newx)
rng = newx(n_ele3-1)-newx(0)
smplrate = n_ele3/rng ;sampling rate
;smplrate = 108.0
print, 'sample rate: ', smplrate

;secth = mean(secant)
;secth=2.0 ; for kitt peak data
```

```
prs2, start, vpath, pressur, tmp, n2o6, ch46, h2o6, co26;pressure in atmospheres, 8.0 is
the approx. scale height for air,
; temperture in kelvin

n2o6=n2o6*cn
ch46=ch46*cc
h2o6=h2o6*ch
co26=co26*co
layers=n_elements(pressur) ; number of layers that you divide the atmosphere

tdepend = (296/tmp)^0.78; 0.78 is n in the HITRAN tables
bl=bo*pressur*tdepend  ; air-broadening coefficients (cm-1) at different pressures and
tempertures
;bd = 0.003556 ; Doppler broadening coefficient of CO at 296K

; *** number of data point need to match the sampling rate
rangenum = floor(smplrate * (rng + 2.0))
v=findgen(rangenum)/smplrate + (newx(0)-1.0) ; start 1 cm-1 before the first line center
print, 'Secant of peaks ', secant
coeff = linfit(pktable3, secant) ; *** use to compute changing path length

; **** Loop for reading in different HITRAN files that are at different Tempertures
*******

absorb2=fltarr(rangenum,layers)
for m=0,layers-1 do begin

layer,absorb,layers,newx,smplrate,secant,pktable3,vpath,start,coeff,rangenum,v,m,pressur,
tmp,$
n2o6,ch46,h2o6,co26,bl
absorb2(*,m) = absorb ; optical path for each layer
endfor

;*****************************************************************************

absb2 = total(absorb2,2) ; sum of the optical paths

absb=exp(absb2) ; calculate the transmission function

titl = 'H2_' + filename

fudge = 0.99 ;just gets rid of the negative values

;woplot, newx, (newy + fudge)
;woplot, v, absb, newx, (newy + fudge), title = titl

gauss2=-instruf(hwhm, smplrate)

intrum=convol(absb,gauss2,total(gauss2))

woplot, v, intrum, newx, (newy + fudge), title = titl

; **** refit observed data's base line to calculated *********
; *** takes the average of each trough and fits a line to it for both calc. and obs.
; *** subtracts them then fits a line to (obs. - calc.) then adds to obs. data.
; *** baselines should match up.
```

```
restore, 'dpkhitran.sav'  ; has the dpkhitran only, 2D array, p #'s and wave #'s
cnt = n_elements(pktable2)
pkt = pktable2[0:cnt-2] ; uses inside troughs, the two outside troughs are not used
nele6 = n_elements(pkt)
dpk1 = reform(dpkhitran[1,pkt]); wavenumber of each trough
nele2 =  n_elements(pkt)
high = dpk1 + 0.2 ; uses 0.4 cm-1 of each trough
low = dpk1 - 0.2

avgcalc = fltarr(nele2)
avgobs = fltarr(nele2)

newy = newy + fudge

for o=0, nele2-1 do begin

mark3 = where(newx gt low(o) and newx lt high(o),count)
;print, 'mark3 ', mark3
mark4 = where(v gt low(o) and v lt high(o),count)
;print, 'mark4 ', mark4
avgcalc(o) = mean(newy(mark3))
avgobs(o) = mean(intrum(mark4))

endfor

diff = avgobs - avgcalc ; subtracts obs. average troughs from calc. average troughs
;print, diff
;plot, dpk1, diff, color = 0, background=255
;plots, dpk1, diff, color = 0, psym = 4
; *** assumes the troughs on the outside are the matched
; *** so the outside troughs differences linearly goes to zero
; *** this calculate those ends
mk = where(newx ge dpk1(0) and newx le dpk1(nele2-1))
newx2 = newx(mk)
nele7 = n_elements(newx2)
lin2 = spline(dpk1, diff, newx2, 8.0)

nele4 = n_elements(newx2)
nele5 = n_elements(lin2)
; *** calculates begining data
xln1 = [newx(0), newx2(0)]
yln1 = [0.0, lin2(0)]
linstart1 = linfit(xln1, yln1)
mk2 = where(newx eq newx2(0))
newx3 = newx(0:mk2(0)-1)
linstart2 = linstart1(0) + linstart1(1)*newx3
; *** calculates ending data
xln2 = [newx2(nele4-1), newx(n_ele3-1)]
yln2 = [lin2(nele5-1), 0.0]
linend1 = linfit(xln2,yln2)
mk3 = where(newx eq newx2(nele4-1))
newx4 = newx(mk3(0)+1:n_ele3-1)
linend2 = linend1(0) + linend1(1)*newx4
; *** tacks them on
lin3 = [linstart2,lin2,linend2]
```

```
newy2 = newy + lin3

mk4 = where(newx gt dpk1(0) and newx lt dpk1(nele6-1))
mk5 = where(v gt dpk1(0) and v lt dpk1(nele6-1))
;shiff = floor(1.33*smplrate)
newx2 = newx(mk4)
newy3 = newy2(mk4)
v2 = v(mk5)
intrum2 = intrum(mk5)

; **** shows the non-linear newx2
;nelem= n_elements(newx2)
;nw2 = fltarr(nelem-1)
;vv2 = fltarr(nelem-1)
;for w=0, nelem-2 do begin
;nw2(w) = newx2(w+1)-newx2(w)
;print, nw2(w)
;vv2(w) = v2(w+1)-v2(w)
;print, '     ', vv2
;endfor
;plot, nw2 , /ynozero
; ********

;test = newy2(mk4-(shiff-2))

nele8=n_elements(v2)

;shifty=2
if shifty ge 0 then begin
newy3=newy3(shifty:(nele7-1))
newx2=newx2(0:(nele7-(shifty+1)))
endif else begin
shifty=abs(shifty)
intrum2=intrum2(shifty:nele8-1)
v2=v2(0:nele8-(shifty+1))
endelse
;print, (round(newx2(0:30)*100.0))/100.0
;print, (round(v2(0:30)*100.0))/100.0
;intrum3 = REBIN(intrum2,2*nele8)

nx3 = ((round(newx2*100.0))/100.0)
vv3 = ((round(v2*100.0))/100.0)
nx4 = uniq(nx3)
vv4 = uniq(vv3)

int3 = intrum2(vv4)
ny4 = newy3(nx4)

nx5 = nx3(nx4)
vv5 = vv3(vv4)

mk7=where(nx5 eq vv5, count)
woplot, v2, intrum2, newx2, newy3, title = titl
;woplot, newx2, intrum2, newx2, newy3, title = titl
```

```
plot, int3, color=0, background=255
oplot, ny4, color=0, linestyle=3
deltasqr = int3 - ny4

deltasqr = deltasqr/brn
;deltasqr = 1.0-intrum2/newy3
;dtest = intrum2 - test
;woplot, v2, deltasqr,newx2, deltasqr, title = 'delta_'+titl

woplot,nx5, deltasqr,vv5, deltasqr, title = 'delta_'+titl

;***** divides up spectra in to each line and integrates each line *********

plnnum = fltarr(nele6-1)
integ = fltarr(nele6-1)
secant2 = fltarr(nele6-1)

for q = 0, nele6-2 do begin ; -2 because of dpk1(q+1)
mk6 = where(newx gt dpk1(q) and newx lt dpk1(q+1))
xtmp = newx(mk6)
ytmp = newy2(mk6) - 1.0
plnnum(q) = pkt(q+1)
secant2(q) = secant(q+1)
integ(q) = int_tabulated(xtmp, ytmp)
endfor
print, ' '
normint = integ/secant2
;print, 'Integ ', integ, plnnum, normint

avgzenith = mean(zenith)

mn = mean(deltasqr)
st = stddev(deltasqr)

print, 'Mean   ', mn
print, 'Stddevs    ', st
print, ' '

;applescrpt3, filename, avgzenith, shour2, secant2, normint, plnnum

save, v2, absb, intrum2, newx2, newy3, pktable2, integ, secant2, normint, plnnum, $
 avgzenith, deltasqr, hwhm, cn, cc, mn, st, brn, int3,
ny4,nx5,vv5,pktable3,filename=filenam
end
```

humlik.pro

```
;******** Calculates the Voigt profile **************

function humlik, XX, YY

X=XX
Y=YY

;parameters
RRTPI = 1/SQRT(!PI)
Y0 = 1.5
Y0PY0 = 2*Y0
Y0Q = Y0^2

C = [1.0117281, -0.75197147, 0.012557727, 0.010022008, -0.00024206814, 0.00000050084806]
S = [1.393237, 0.23115241, -0.15535147, 0.0062183662, 0.000091908299, -0.00000062752596]
T = [0.31424038, 0.94778839, 1.5976826, 2.2795071, 3.0206370, 3.8897249]

N = n_elements(X)
K = DBLARR(N)
XP = DBLARR(6)
XM = DBLARR(6)
YP = DBLARR(6)
YM = DBLARR(6)
MQ = DBLARR(6)
PQ = DBLARR(6)
MF = DBLARR(6)
PF = DBLARR(6)


;******* START OF CODE **********

YQ = Y^2
YRRTPI = Y*RRTPI

IF (Y GE 70.55) THEN BEGIN

XQ = X^2
K = YRRTPI/(XQ+YQ)
RETURN, K

ENDIF

; ***** RG are flags that let you calculate const. only once
; to save processor time.  Had to move the first calc. of D and K
; into the RG loop because IDL's CASE statement isn't exactly like the
; FORTRAN ELSEIF statement.
```

```
RG1 = 1
RG2 = 1
RG3 = 1

XLIM0 = SQRT(15100.0 + Y*(40.0 - Y*3.6))
IF Y GE 8.425 THEN BEGIN
XLIM1 = 0.0
ENDIF ELSE BEGIN
XLIM1 = SQRT(164.0 - Y*(4.3 + Y*1.8))
ENDELSE

XLIM2 = 6.8 - Y
XLIM3 = 2.4*Y
XLIM4 = 18.1*Y + 1.65

IF Y LE 0.000001 THEN BEGIN
XLIM1 = XLIM0
XLIM2 = XLIM0
ENDIF

; ******* BEGINNING OF BIG LOOP *******

FOR I=0,N-1 DO BEGIN

ABX = ABS(X(I))
XQ = ABX^2
CASE 1 OF
   (ABX GE XLIM0): K(I) = YRRTPI/(XQ + YQ)

   (ABX GE XLIM1) AND (RG1 NE 0): BEGIN
      RG1 = 0
      A0 = YQ + 0.5
      D0 = A0^2
      D2 = 2*YQ - 1.0

      D = RRTPI/(D0 + XQ*(D2 +XQ))
      K(I) = D*Y*(A0 + XQ)
   END

   (ABX GE XLIM1): BEGIN
      D = RRTPI/(D0 + XQ*(D2 +XQ))
      K(I) = D*Y*(A0 + XQ)
   END

   (ABX GT XLIM2) AND (RG2 NE 0): BEGIN
      RG2 = 0
      H0 = 0.5625 + YQ*(4.5 + YQ*(10.5 + YQ*(6.0+YQ)))
      H2 = -4.5 + YQ*(9.0 + YQ*(6.0 + YQ*4.0))
      H4 = 10.5 - YQ*(6.0 - YQ*6.0)
      H6 = -6.0 + YQ*4.0
      E0 = 1.875 + YQ*(8.25 + YQ*(5.5 + YQ))
      E2 = 5.25 + YQ*(1.0 + YQ*3.0)
      E4 = 0.75*H6

      D = RRTPI/(H0 + XQ*(H2 + XQ*(H4 + XQ*(H6 +XQ))))
   K(I) = D*Y*(E0 + XQ*(E2 + XQ*(E4 + XQ)))
END

(ABX GT XLIM2): BEGIN
   D = RRTPI/(H0 + XQ*(H2 + XQ*(H4 + XQ*(H6 +XQ))))
   K(I) = D*Y*(E0 + XQ*(E2 + XQ*(E4 + XQ)))
END

(ABX LT XLIM3) AND (RG3 NE 0): BEGIN
   RG3 = 0
   Z0 = 272.1014 + Y*(1280.829 + Y*(2802.870 + Y*(3764.966 $
   + Y*(3447.629 + Y*(2256.981 + Y*(1074.409 + Y*(369.1989 $
   + Y*(88.26741 + Y*(13.39880 + Y)))))))))
   Z2 = 211.678 + Y*(902.3066 + Y*(1758.336 + Y*(2037.310 $
   + Y*(1549.675 + Y*(793.4273 + Y*(266.2987  $
   + Y*(53.59518 + Y*5.0)))))))
   Z4 = 78.85685 + Y*(308.1852 + Y*(497.3014 + Y*(479.2567 $
   + Y*(269.2916 + Y*(80.39278 +Y*10.0)))))
   Z6 = 22.03523 + Y*(55.02933 + Y*(92.75679  + Y*(53.59518 $
   + Y*10.0)))
   Z8 = 1.496460 + Y*(13.39880 + Y*5.0)
   P0 = 153.5168 + Y*(549.3954 + Y*(919.4955 + Y*(946.8970 $
   + Y*(662.8097 + Y*(329.2151 + Y*(115.3771 + Y*(27.93941 $
   + Y*(4.264678 +Y*0.3183291))))))))
   P2 = -34.16955 + Y*(-1.322256 + Y*(124.5975 + Y*(189.7730 $
   + Y*(139.4665 + Y*(56.81652 +Y*(12.79458 $
   +Y*1.2733163))))))
   P4 = 2.584042 + Y*(10.46332 + Y*(24.0155 + Y*(29.81482 $
   + Y*(12.79568 + Y*1.9099744))))
   P6 = -0.07272979 + Y*(0.9377051 + Y*(4.266322 + Y*1.273316))
   P8 = 0.0005480304 + Y* 0.3183291

   D = 1.7724538/(Z0 + XQ*(Z2 + XQ*(Z4 +XQ*(Z6 + XQ*(Z8 + XQ)))))
   K(I) = D*(P0 + XQ*(P2 + XQ*(P4 + XQ*(P6 + XQ*P8))))
END

(ABX LT XLIM3): BEGIN
   D = 1.7724538/(Z0 + XQ*(Z2 + XQ*(Z4 +XQ*(Z6 + XQ*(Z8 + XQ)))))
   K(I) = D*(P0 + XQ*(P2 + XQ*(P4 + XQ*(P6 + XQ*P8))))
END

ELSE : BEGIN  ;executes this code if XLIM3 < ABX < XLIM2

   YPY0 = Y + Y0
   YPY0Q = YPY0^2
   K(I) = 0.0
   FOR J=0,5 DO BEGIN
      D = X(I) - T(J)
      MQ(J) = D^2
      MF(J) = 1.0/(MQ(J) + YPY0Q)
      XM(J) = MF(J)*D
      YM(J) = MF(J)*YPY0
      D = X(I) + T(J)
      PQ(J) = D^2
      PF(J) = 1.0/(PQ(J) + YPY0Q)
      XP(J) = PF(J)*D
```

```
         YP(J) = PF(J)*YPY0
      ENDFOR

      IF ABX LE XLIM4 THEN BEGIN

         FOR J = 0,5 DO BEGIN
            K(I) = K(I) + C(J)*(YM(J) + YP(J)) - S(J)*(XM(J) - XP(J))
         ENDFOR

      ENDIF ELSE BEGIN
         YF = Y + Y0PY0
         FOR J = 0,5 DO BEGIN
            K(I) = K(I) $
            + (C(J)* (MQ(J)*MF(J) - Y0*YM(J)) + S(J)*YF*XM(J))/(MQ(J)+Y0Q) $
            + (C(J)* (PQ(J)*PF(J) - Y0*YP(J)) - S(J)*YF*XP(J))/(PQ(J) + Y0Q)
         ENDFOR
         K(I) = Y*K(I) +EXP(-XQ)
      ENDELSE

   ENDELSE

ENDCASE

ENDFOR

RETURN, K

END
```

Layer.pro

```
; ****** calculates lines for one layer *********
pro layer,absorb,layers,newx,smplrate,secant,pktable3,vpath,
start,coeff,rangenum,v,m,pressur,tmp,n2o6,ch46, h2o6, co26,bl


conv=2.4464e19 ;conversion constant
bo=0.08 ; broadening coefficient (cm-1) at 1 atm.
n_ele3 = n_elements(newx)

;ppath=dialog_pickfile(path='John Paul:RSI:IDL 5.3:Syn. Spec.:', get_path=path,$
;title='Choose Hitran Excel txt file', filter = '*.txt')
case m of
0: data2 = READ_ASCII('John Paul:RSI:IDL 5.3:Syn. Spec.:working folder:ALLT270G.TXT')
1: data2 = READ_ASCII('John Paul:RSI:IDL 5.3:Syn. Spec.:working folder:ALLT238G.TXT')
2: data2 = READ_ASCII('John Paul:RSI:IDL 5.3:Syn. Spec.:working folder:ALLT217G.TXT')
3: data2 = READ_ASCII('John Paul:RSI:IDL 5.3:Syn. Spec.:working folder:ALLT217G.TXT')
```

```
4: data2 = READ_ASCII('John Paul:RSI:IDL 5.3:Syn. Spec.:working folder:ALLT217G.TXT')
5: data2 = READ_ASCII('John Paul:RSI:IDL 5.3:Syn. Spec.:working folder:ALLT225G.TXT')
endcase


molnum = data2.field1(0:0,0:*)
wavnum = data2.field1(1:1,0:*)
strn = data2.field1(2:2,0:*)

molnum = reform(molnum)
wavnum = reform(wavnum)
strn = reform(strn)

;********** molnum is molecule code used by Hitran
;********** wavnum is position of line center in cm-1
;********** strn is line strength in (cm-1)/(molecule/cm^2)

numl2 = n_elements(molnum)
print, numl2

; *** crops the hitran data to fit the range of the observed data
mk1 = where(wavnum gt (newx(0)-1.0) and wavnum lt (newx(n_ele3-1)+1.0))
molnum2 = molnum(mk1)
wavnum2 = wavnum(mk1)
strn2 = strn(mk1)

; *** picks only the strongest N2O and CH4 lines
strg = where((molnum2 ge 41 and molnum2 le 45) or (molnum2 ge 61 and molnum2 le 65) or
molnum2 eq 11 or molnum2 eq 21 )
molnum3 = molnum2(strg)
wavnum3 = wavnum2(strg)
strn3 = strn2(strg)

numl = n_elements(molnum3)

print, 'Number of lines in use in hitran file: ', numl

secth = coeff(0) + coeff(1)*wavnum3 ; varible secant at UT for each line
;secth = mean(secant)
;secth=2.0 ; constant secant at kitt peak
opath= vpath * secth  ; *** changing path length
strn4 = strn3 * conv * opath; *** changes line strengths in to (cm^-2*atm^-1) at 300K
sxl = fltarr(numl, layers)
bd2 = fltarr(numl)

; *** takes the line strengths and times them by the mixing ratio and path length,
; keep only Methane and Nitrous Oxide
for i=0,numl-1 do begin

if molnum3(i) ge 41 and molnum3(i) le 45 then begin
   sxl(i,*) = strn4(i) * n2o6 *1.0
   bd2(i) = 3.5803e-7*wavnum3(i)/((44.0128)^0.5)
end
if molnum3(i) ge 61 and molnum3(i) le 65 then begin
   sxl(i,*) = strn4(i) * ch46 *1.0 ;* ch4volmix
   bd2(i) = 3.5803e-7*wavnum3(i)/((16.0465)^0.5)
```

```
end
if molnum3(i) ge 11 and molnum3(i) le 11 then begin
   sxl(i,*) = strn4(i) * h2o6 *1.0 ;*
   bd2(i) = 3.5803e-7*wavnum3(i)/((16.0465)^0.5)
end
if molnum3(i) ge 21 and molnum3(i) le 23 then begin
   sxl(i,*) = strn4(i) * co26 *1.0 ;*
   bd2(i) = 3.5803e-7*wavnum3(i)/((16.0465)^0.5)
end
endfor

;print, 'bd2 '
;print, bd2

absorb=fltarr(rangenum)

print, 'layer # ',m
bd = bd2*(tmp(m)^0.5)  ; checks out, size - numl
y = 0.8326*(bl(m)/bd)  ; size - numl
pp = (1/bd)*(0.8326/!pi)^0.5  ; size - numl

   for j=0,numl-1 do begin
      x = fltarr(rangenum) ; *** sets x back to zero for the next line ***
      x = 0.8326*((v - wavnum3(j))/bd(j))
      voith = fltarr(rangenum) ; *** sets voith back to zero for the next line ***
      voith = humlik(x,y(j))
      voit = voith*pp(j)
      ;print, int_tabulated(v,voit)
      ; *** add up all optical densities times absorption coefficient
      absorb= absorb + (-sxl(j,m)*voit*pressur(m))
      ;plot, v, absorb, color = 0, background=255
      ;stop
   endfor
;plot, v, absorb(*,m), color = 0, background=255
;stop

end
```

woplot.pro

```
; Plots graphs, interactively crops them and saves to
; a postscript file
; as of 11/6/01, has title keyword


PRO plotinteractive_doplot, pstate
```

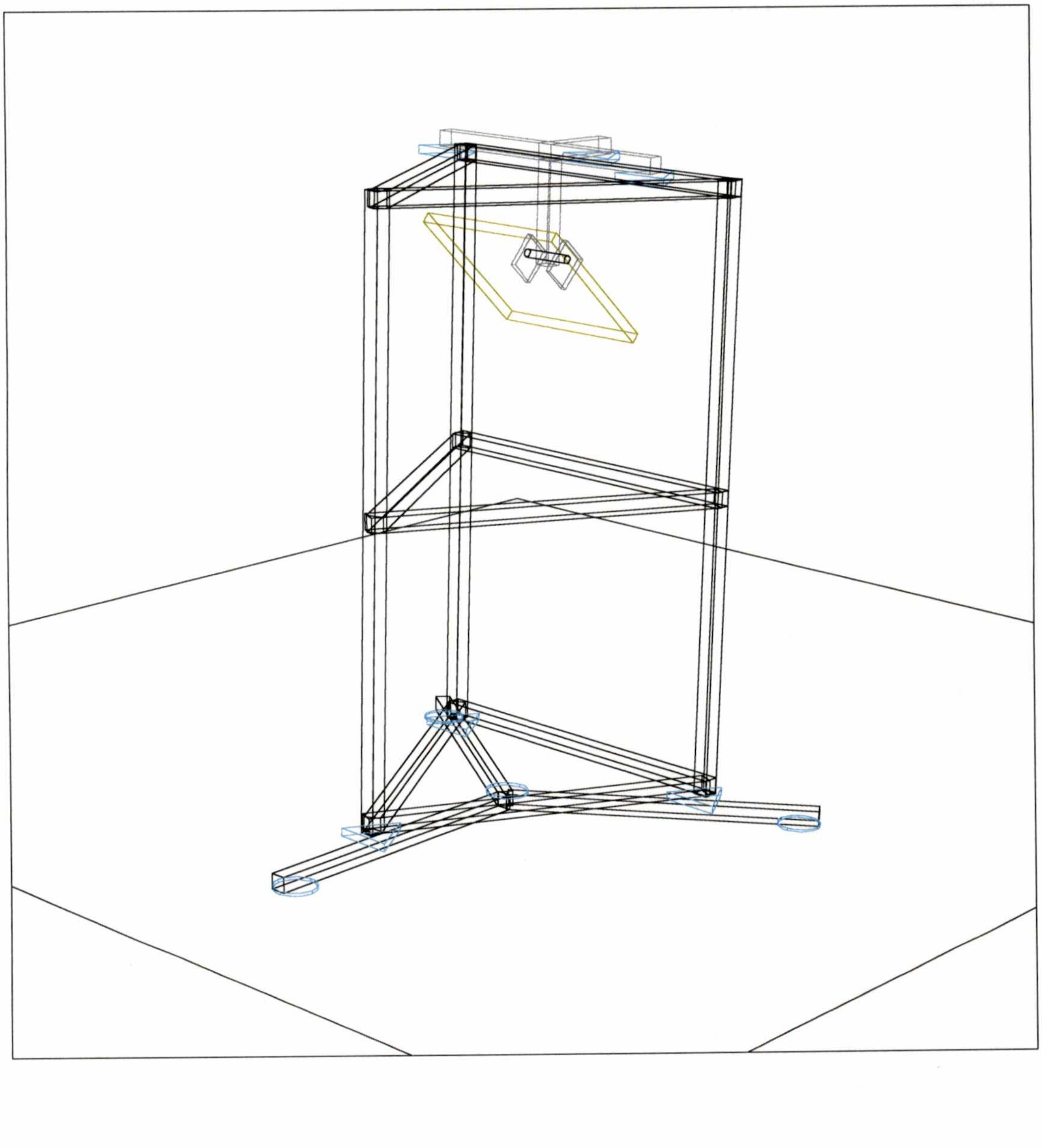
```
if (*pstate).wxmin eq 0 then begin
plot, (*pstate).x1, (*pstate).y1,  $
PSYM = (*pState).psym, color=0, background=255, title = (*pstate).title, XTITLE=
'Wavenumbers',$
YTITLE= 'Transmission'
oplot, (*pstate).x2, (*pstate).y2,  $
PSYM = (*pState).psym, color=0, linestyle=3
endif else begin
plot, (*pstate).x1((*pstate).wxmin:(*pstate).wxmax), $
(*pstate).y1((*pstate).wxmin:(*pstate).wxmax),  $
PSYM = (*pState).psym, color=0, background=255, title = (*pstate).title, XTITLE=
'Wavenumbers',$
YTITLE= 'Transmission'
oplot, (*pstate).x2, (*pstate).y2,  $
PSYM = (*pState).psym, color=0, linestyle=3
endelse

END

PRO plotinteractive_print, event
widget_control, event.top, get_uvalue = pstate
oktoprint = dialog_printersetup()
if oktoprint then begin
oldDev = !d.name
set_plot, 'PRINTER'
plotinteractive_doplot, pState
device, /close_document
set_plot, oldDev
wset, (*pState).winID
endif
END

PRO plotinteractive_postscript, event
widget_control, event.top, get_uvalue = pstate

oldDev = !d.name
set_plot, 'ps'
device, FILENAME = (*pState).title + 'PS', LANDSCAPE = 1, /color
plotinteractive_doplot, pState
device, /close
set_plot, oldDev
wset, (*pState).winID
print, 'postscript filename ', (*pState).title + 'PS'

;devkeywords = ps_form(CANCEL = cancel, GROUP_LEADER = event.top)
;if (not cancel) then begin
;oldDev = !d.name
;set_plot, 'ps'
;device, _extra = devkeywords
;plotinteractive_doplot, pState
;device, /close
;set_plot, oldDev
;wset, (*pState).winID
;endif
END
```

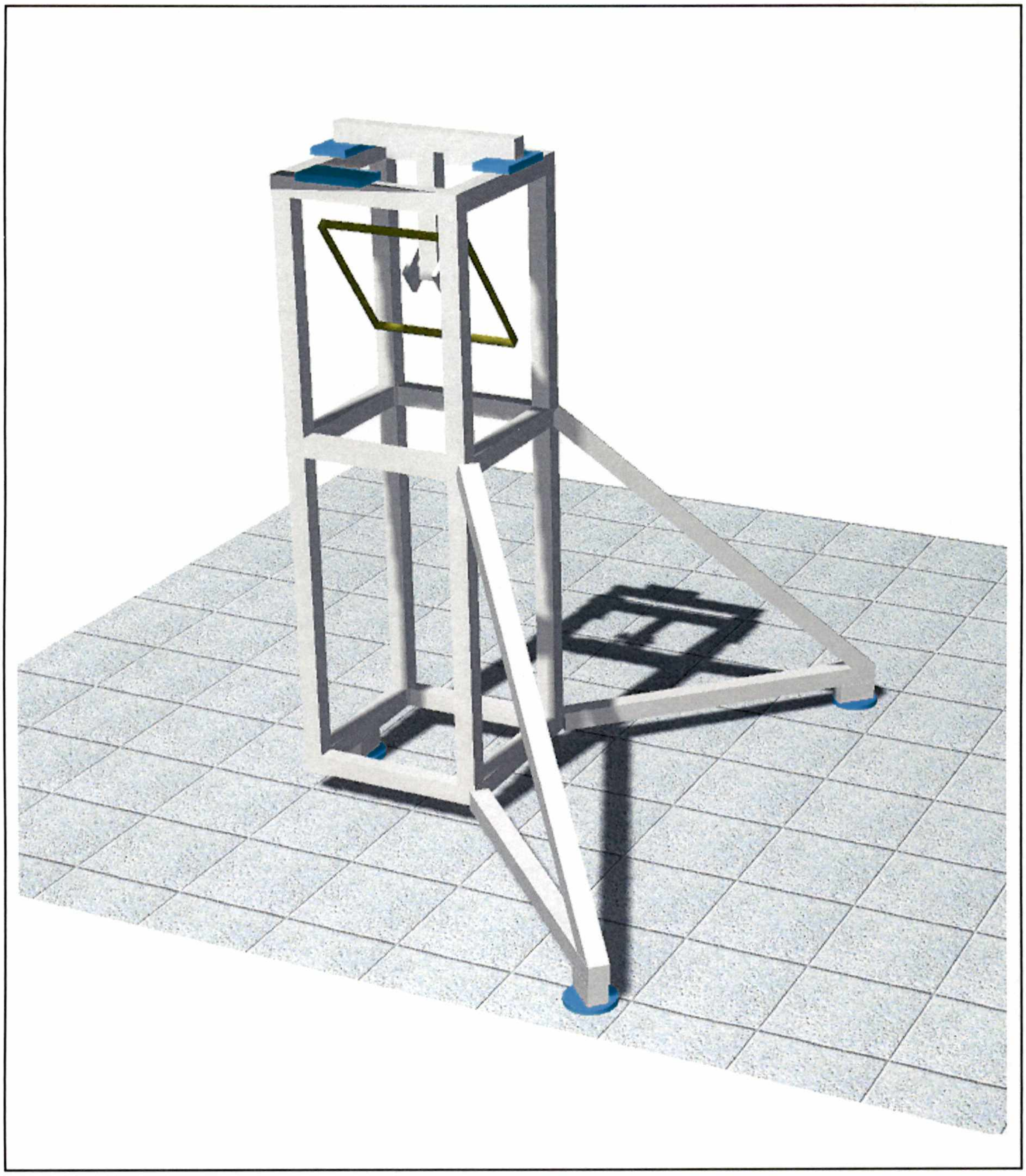
```
PRO plotinteractive_resize, event
widget_control, event.top, get_uvalue = pstate

; to propertly resize a draw widget, the geometry of the bases
; surrounding the draw must be determined and subtracted.
statusg = widget_info((*pState).status, /GEOMETRY)
controlg = widget_info((*pState).controlbase, /GEOMETRY)
tlbg = widget_info(event.top, /GEOMETRY)

; subtract out the xpadding of the tlb for the xsize
newx = event.x - (2*tlbg.xpad)

; subtract the size and padding of the bases
newy = event.y - statusg.scr_ysize - (2*statusg.ypad) - $
            controlg.scr_ysize - (2*controlg.ypad) - $
            (2*tlbg.ypad)

; resize the draw widget
widget_control, (*pState).draw, XSIZE = newx, YSIZE = newy

; re-plot
plotinteractive_doplot, pState

END

PRO plotinteractive_sym, event

; get the pointer to the data structure from the UVALUE
; (data slot) of the top-level base
widget_control, event.top, get_uvalue = pstate

; store the new linestyle
(*pstate).psym = event.index

; replot
plotinteractive_doplot, pstate

END

PRO plotinteractive_connect, event

widget_control, event.top, get_uvalue = pstate

; button on - flip pos sym to neg
if (event.select) then begin
if ((*pState).psym ge 0) then (*pState).psym = (*pState).psym * (-1)

; button off - flop neg sym to pos
endif else begin
if ((*pState).psym lt 0) then (*pState).psym = (*pState).psym * (-1)
endelse

plotinteractive_doplot, pstate

END
```

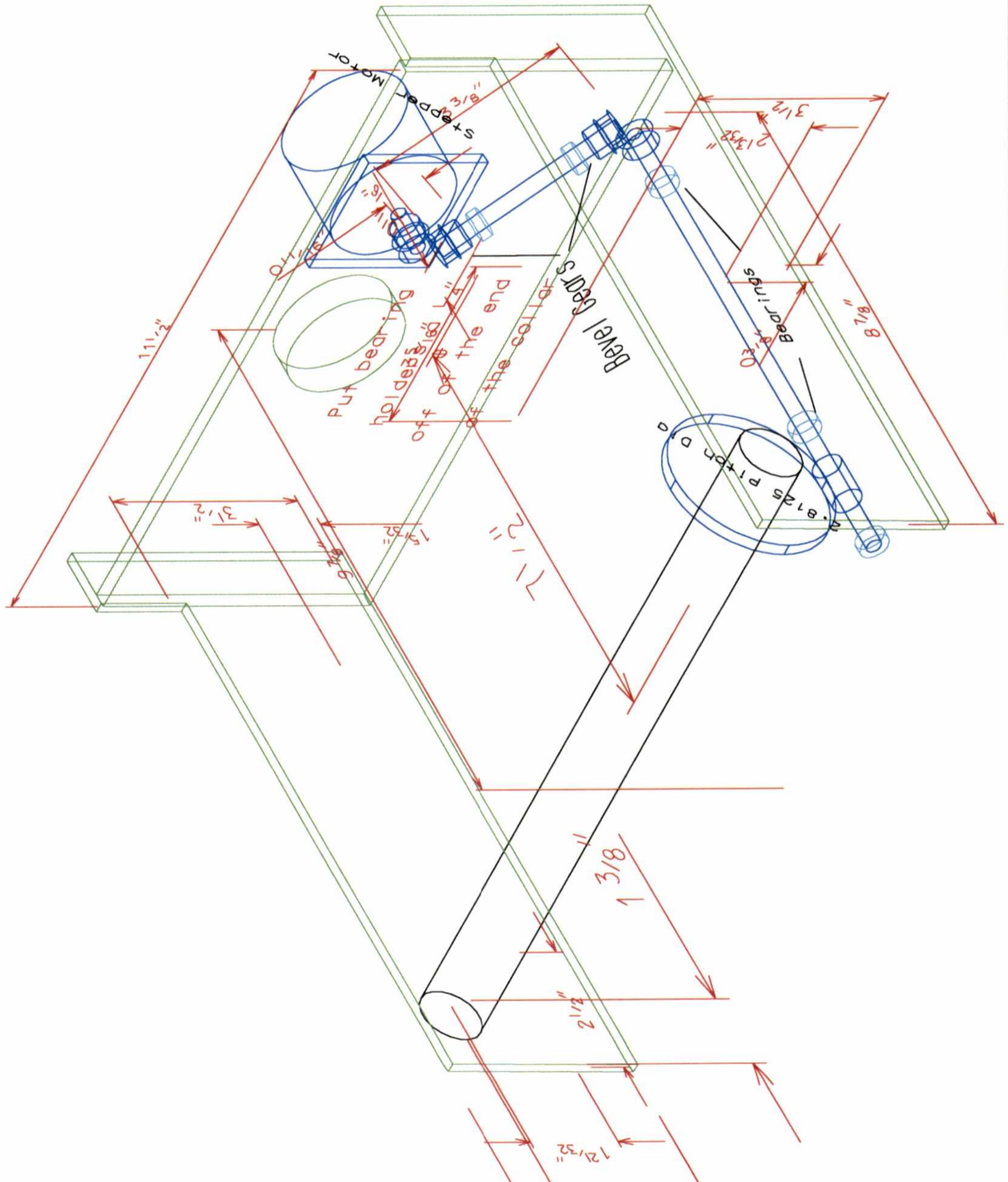
```
PRO plotinteractive_draw, event
widget_control, event.top, GET_UVALUE = pState
m=(*pstate).m

; check for motion events
if (event.type eq 2) then begin
; convert the coordinates from device to data
datac = convert_coord(event.x, event.y, /DEVICE, /TO_DATA)
str = strtrim(datac[0],2)+','+strtrim(datac[1],2)
widget_control, (*pstate).status, SET_VALUE = str
endif

if (event.type eq 0) then begin
; convert the coordinates from device to data
datac = convert_coord(event.x, event.y, /DEVICE, /TO_DATA)
print, datac[0],datac[1],!y.crange
dblx=[datac[0],datac[0]]
oplot, dblx,!y.crange, color=0;, /data
;plots, datac[0],!y.crange[1], color=0, /data
(*pstate).posx(m)=datac[0]
(*pstate).posy(m)=datac[1]
;*** m can go to 15 or 15 line fitting points ***
m=m+1
(*pstate).m=m

;widget_control, event.top, SET_UVALUE = pState
endif

END

pro plotinteractive_redraw, event
widget_control, event.top, GET_UVALUE = pState

xrmin = (*pstate).posx(0)
xrmax = (*pstate).posx(1)
x1 = (*pstate).x1
y1 = (*pstate).y1

strg = where(x1 gt xrmin and x1 lt xrmax, count)
wxmax=max(strg, min=wxmin)
(*pstate).wxmin = wxmin
(*pstate).wxmax = wxmax

plotinteractive_doplot, pstate

end


pro plotinteractive_exit, event

widget_control, event.top, /destroy
return

end
```

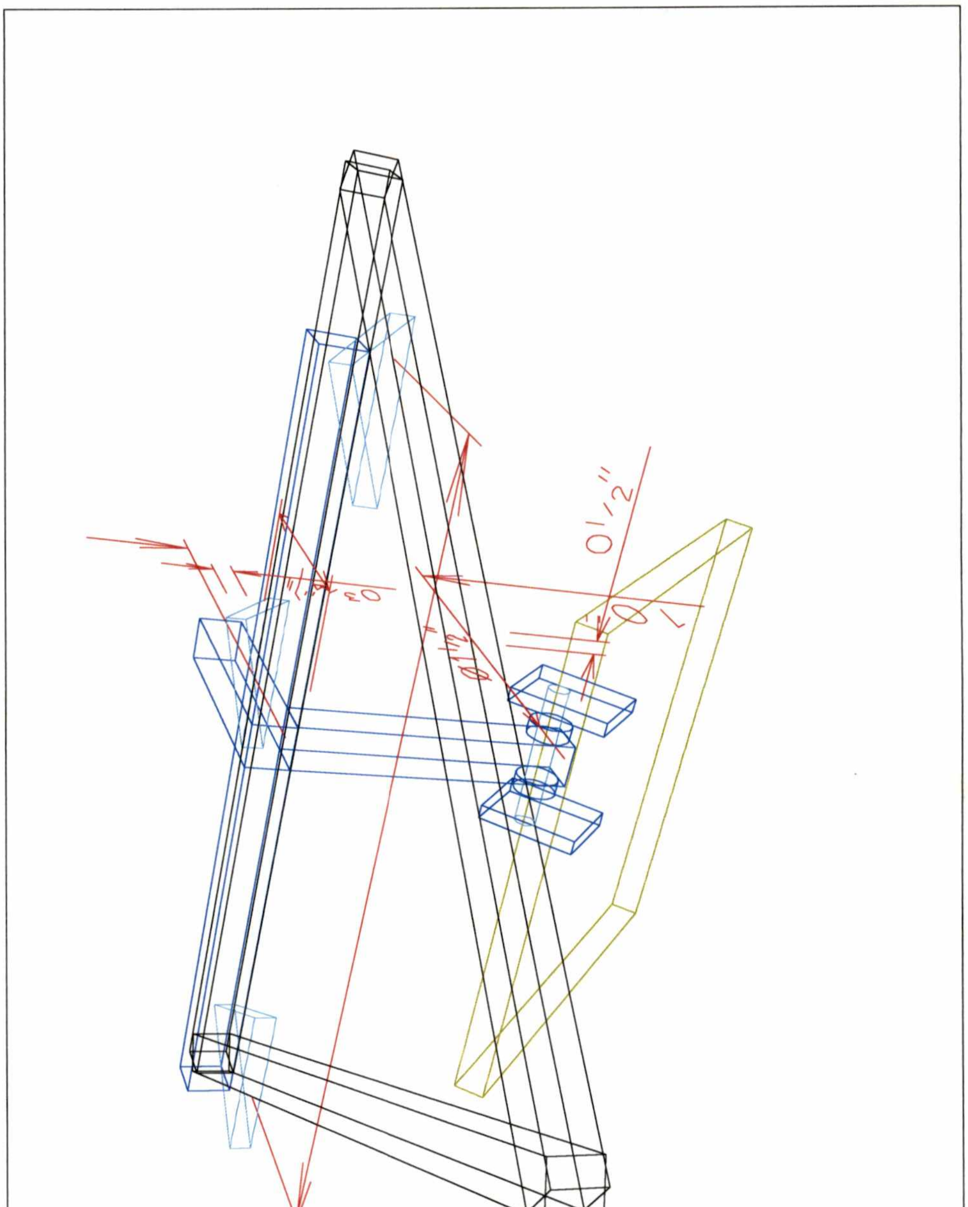
```
pro woplot, x, y, xa, ya, title = title

;******************************************************
;************** Widget program ***************************

; if an error occurs, return to the main level
on_error, 2

; check the parameters
case n_params() of
0: message, 'ERROR: 1 or 2 parameters required'
1: begin
x1 = findgen(n_elements(x))
y1 = x
y2 = findgen(n_elements(x))
x2 = findgen(n_elements(x))
   end
2: begin
x1 = x
y1 = y
y2 = findgen(n_elements(x))
x2 = findgen(n_elements(x))
   end
3: begin
x1 = x
y1 = y
y2 = ya
x2 = findgen(n_elements(x))
   end
4: begin
x1 = x
y1 = y
y2 = ya
x2 = xa
   end
5: begin
x1 = x
y1 = y
y2 = ya
x2 = xa
title = title
   end
endcase

tlb = widget_base(TITLE = 'Kitt peak data', /COLUMN, $
/TLB_SIZE_EVENTS, MBAR = menubase)

fileMenu = widget_button(menubase, VALUe = 'File', /MENU)
printButton = widget_button(fileMenu, VALUE = 'Print', $
EVENT_PRO = 'plotinteractive_print')
psbutton = widget_button(fileMenu, VALUE = 'PostScript', $
EVENT_PRO = 'plotinteractive_postscript')

; main draw widget
status = widget_label(tlb, VALUE = '', /DYNAMIC_RESIZE)
draw = widget_draw(tlb, XSIZE = 700, YSIZE = 400, $
/BUTTON_EVENTS, /MOTION_EVENTS, EVENT_PRO = 'plotinteractive_draw')

; create a row base at the bottom of the display to hold
; the widgets to control the plot
controlbase = widget_base(tlb, /ROW)


usersym, [1, 0.5, -0.5, -0.5, 0.5, 1], [0,1,1,0,-1,-1]
symbols = ['None','Plus sign (+)','Asterisk (*)','Period (.)','Diamond', $
           'Triangle','Square','X','Hex']
symboldrop = widget_droplist(controlbase, VALUE = symbols, $
TITLE = 'Symbol', EVENT_PRO = 'plotinteractive_sym')

redrawbutton = widget_button(controlbase, VALUe = 'Redraw', $
EVENT_PRO = 'plotinteractive_redraw')

exitbutton2 = widget_button(controlbase, VALUe = 'EXIT', $
EVENT_PRO = 'plotinteractive_exit')

widget_control, tlb, /REALIZE

; get the window ID of the draw widget
widget_control, draw, GET_VALUE = winID

; set the draw widget to be the current window and draw the plot
; with all the default settings
wset, winID

posx=fltarr(15)
posy=fltarr(15)
n_ele = n_elements(x1)
; create a structure of data for the application
state = {x1:x1, y1:y1, y2:y2, x2:x2, winID:winID, psym:0, m:0, $
status:status, controlbase:controlbase, draw:draw, $
posx:posx,posy:posy,wxmin:0,wxmax:0,title:title}


; create a pointer to the state structure and put that pointer
; into the UVALUE (data slot) of the top-level base
pstate = ptr_new(state, /no_copy)

widget_control, tlb, SET_UVALUE = pState

plotinteractive_doplot, pstate

; call xmanager to start the event handling procedure
Xmanager, 'woplot', tlb, EVENT_HANDLER = 'plotinteractive_resize'

;************ end of Widget program ************

end
```

Microstation plots

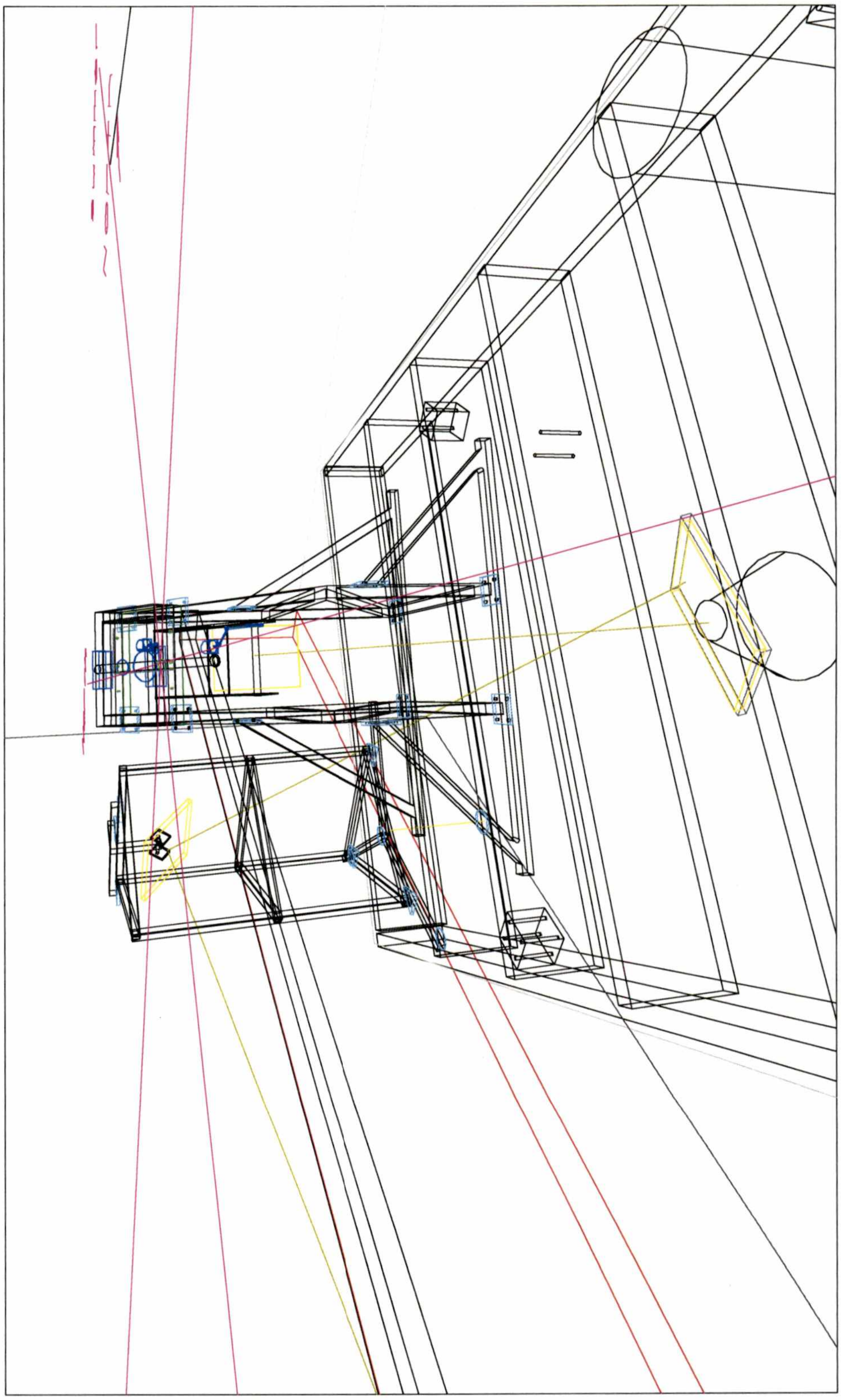


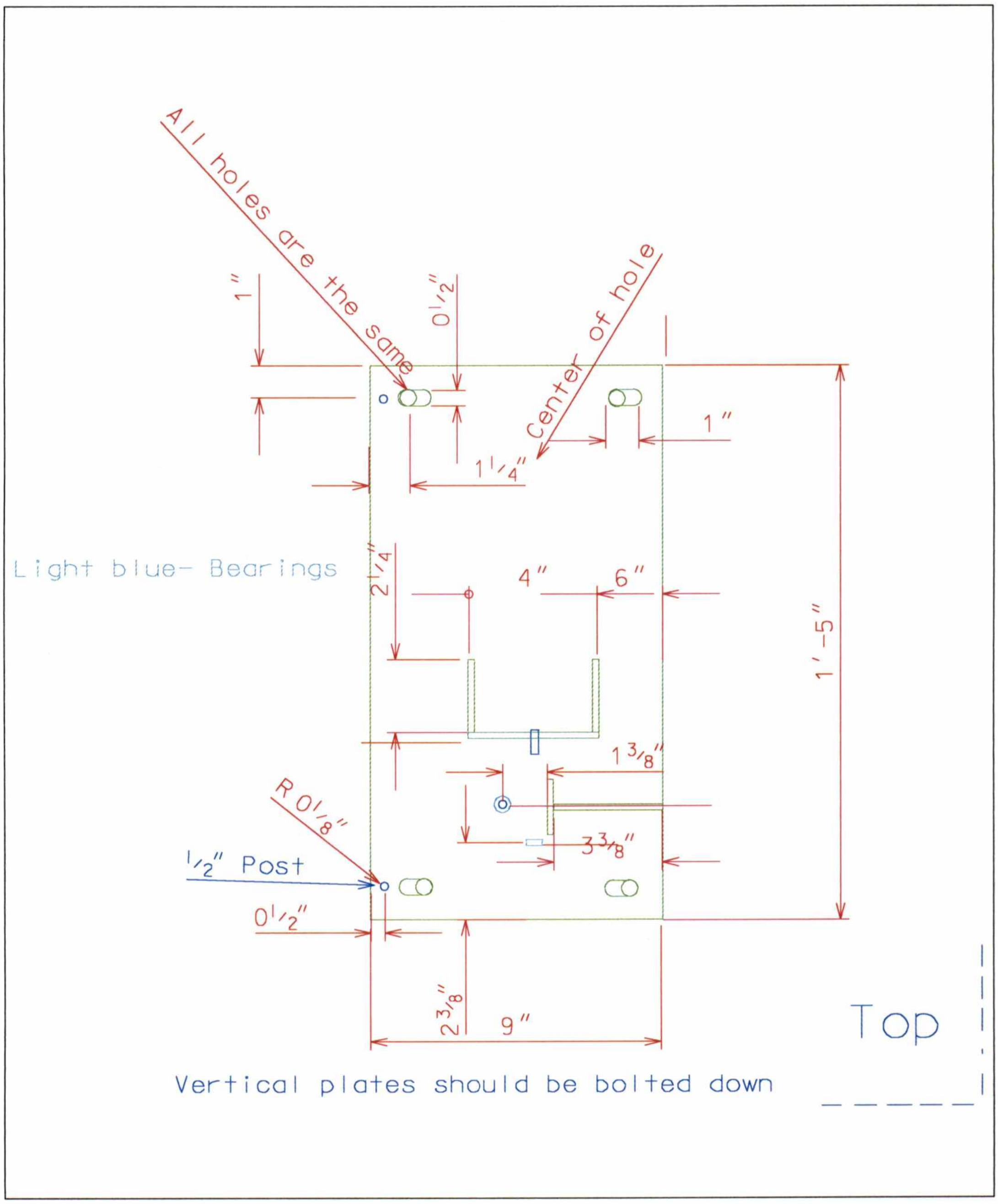



Stepper Motor
3/8"
Ø1.10
11 1/2"
Put bearing holders 1/4" off of the end of the collar
Bevel Gears
Bearings
3 1/2
2 13/32"
8 7/8"
Ø3/4"
Ø.8125 pitch Dia
7 1/2"
1 5/32"
3 1/2"
9 7/8"
1 3/8"
2 1/2"
1 21/32"
2 3/4
3 1/2"



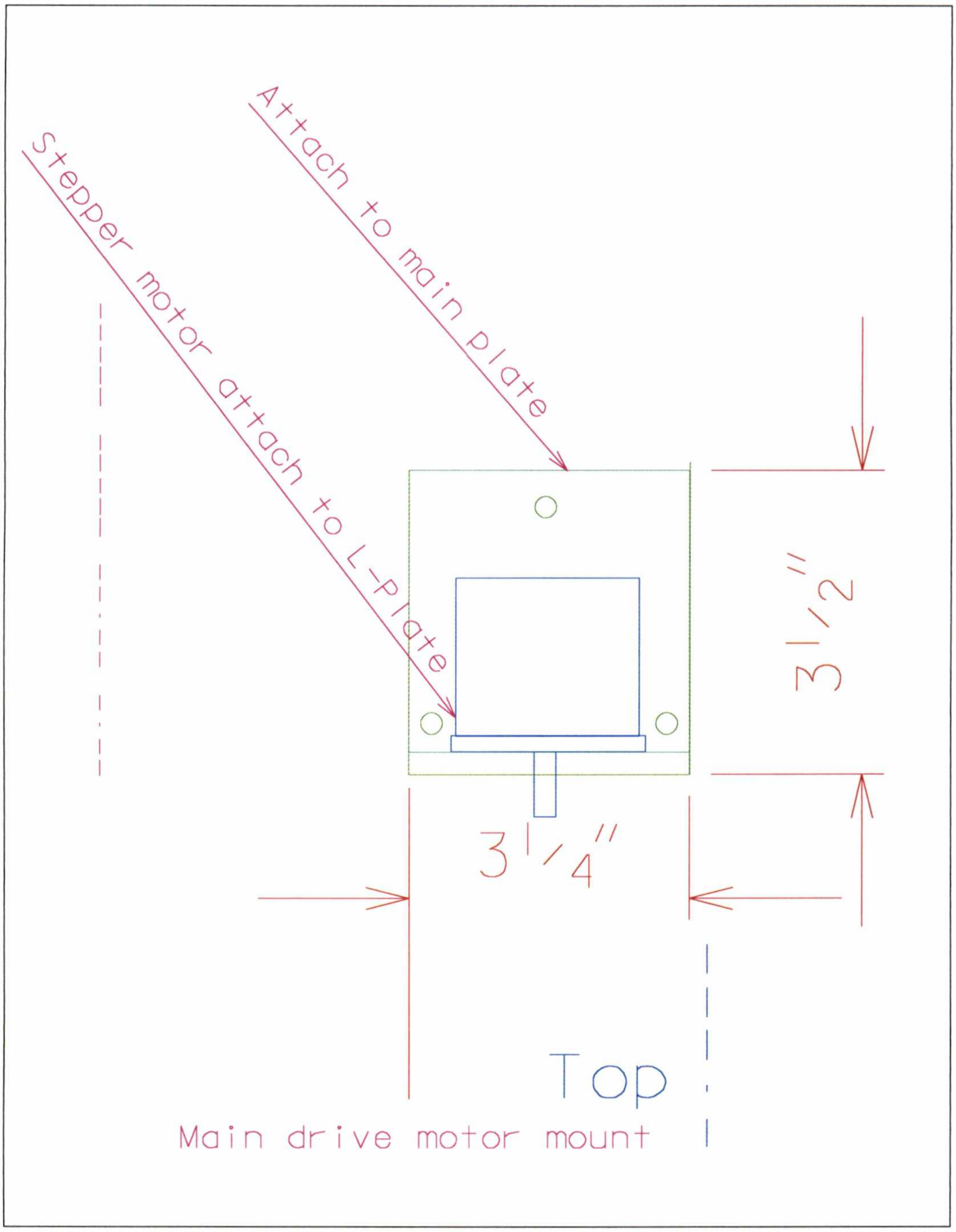
0'-1/2"
0'-3/4"
0'-3/4"
1'-0"
2'-2"



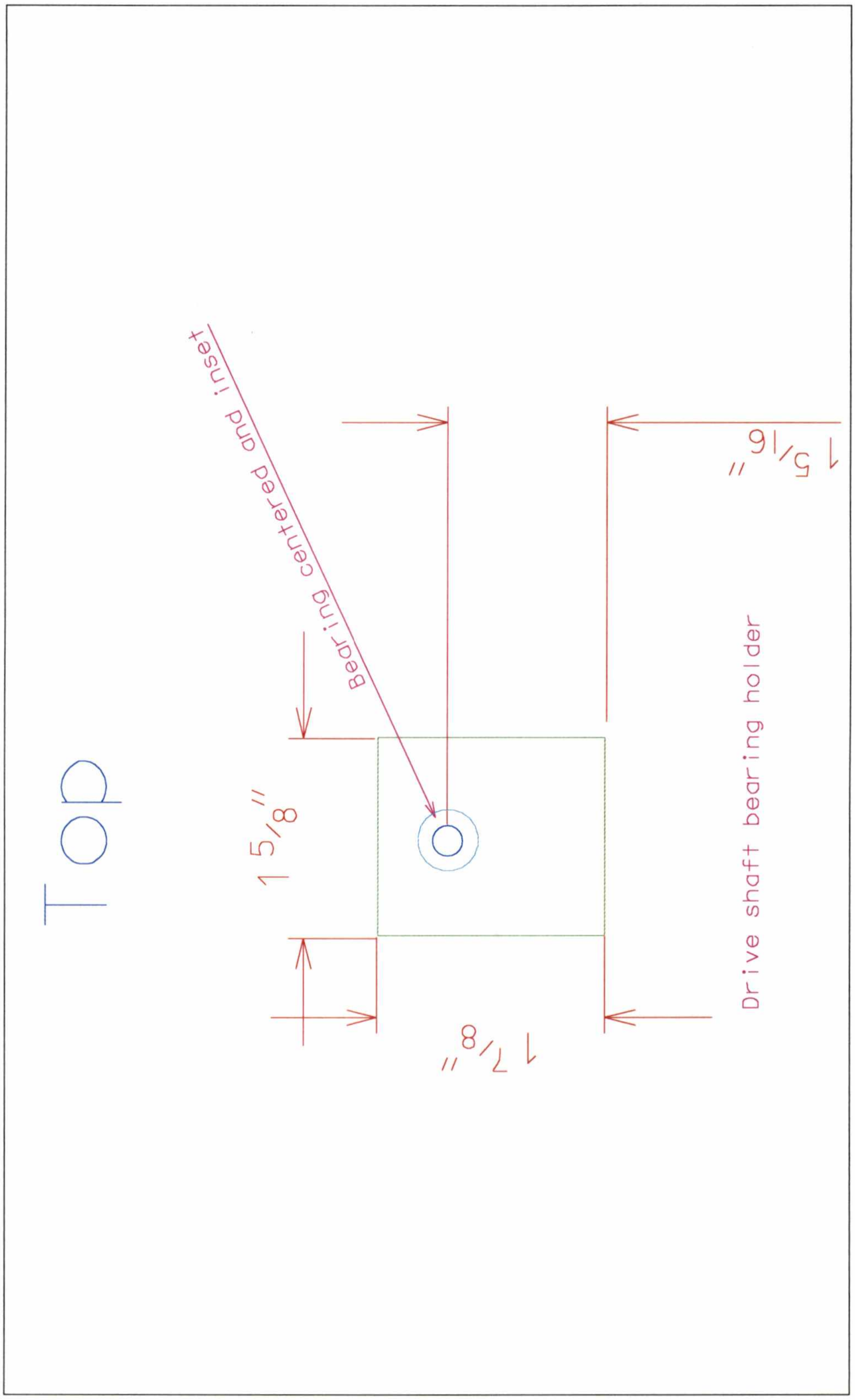


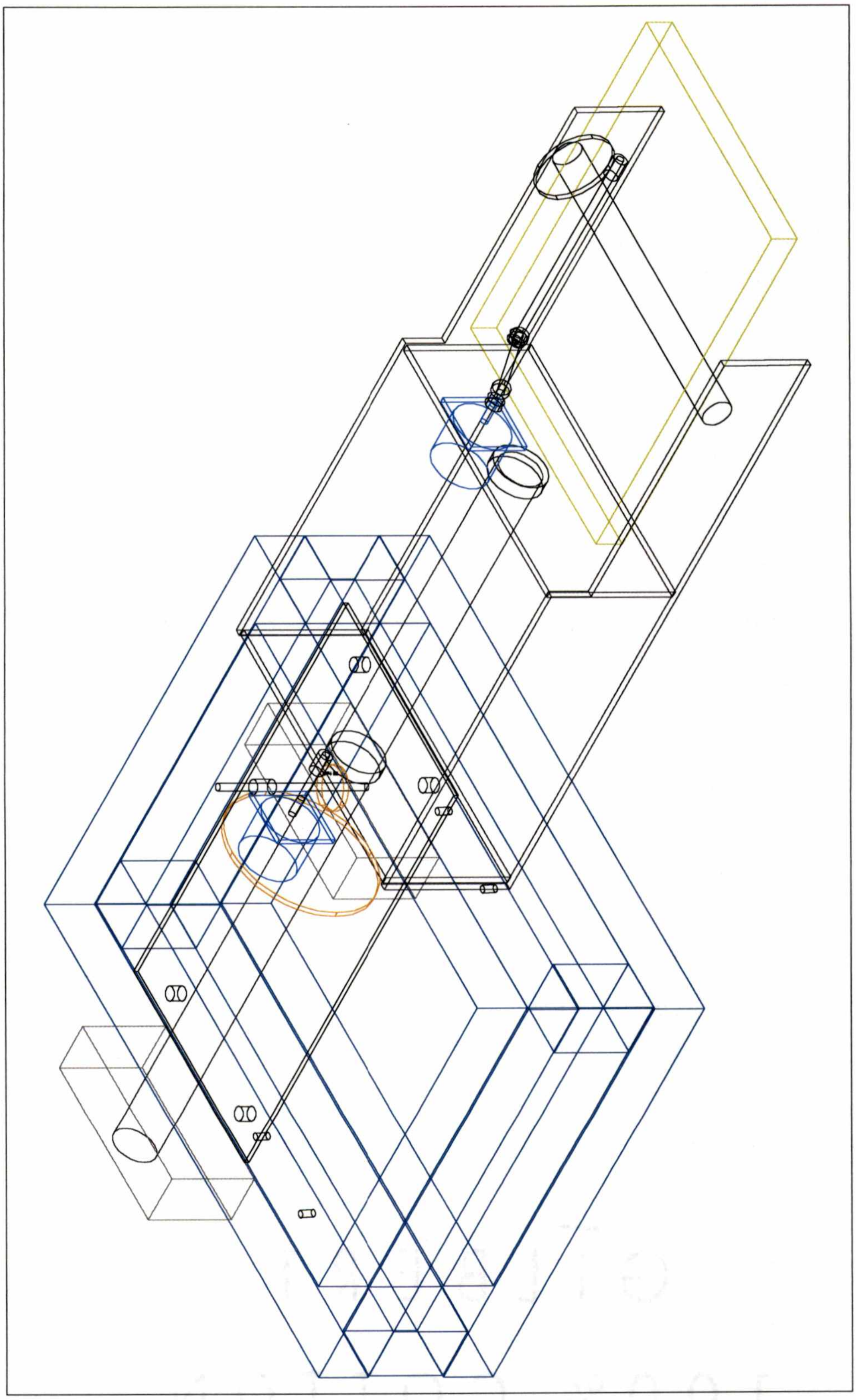
Light blue- Bearings

Vertical plates should be bolted down



Attach to main plate
Stepper motor attach to L-Plate
3 1/2"
3 1/4"
Top
Main drive motor mount


Top
Bearing centered and inset
1 5/16"
1 5/8"
1 7/8"
Drive shaft bearing holder



Vita

John Stewart Hager Jr. was born in Owensboro, Kentucky on July 6, 1959. He graduated from Pine Crest Preparatory School in Ft. Lauderdale, Florida. He attended the University of Alabama and Embry-Riddle Aeronautical University before finishing his Bachelor of Science and Masters Degrees in Physics at the University of Tennessee. He is a member of The Society of Physics Students and is in their Honor Society.

In December of 1992, he married Yolla Rita Hage and has four beautiful children, John Paul, Alex, Lauren and Sydney.