Computing the Lights Majestic

Matthew Fetrow

May 24, 1988

English 201

ABSTRACT

This paper briefly describes some of the interferometer data reduction methods currently being used at Utah State University. Material relating to the Fourier transform of the data through the temperature fitting process is presented, along with an overview of the interferogram files themselves. No information about specific spectral regions or bands is given. This paper is presented in an attempt to familiarize a new user with the existing data reduction system.

Research work is often less than routine. Imagine your anxiety when your venerable employer asks you to spend two hours converting 4000 pages of data into five pages of more 'useful' data.

There is little need for concern, though. You find that the pages of data are stored digitally on magnetic tape, and there is a computer complete with seven programs to help produce the required five pages of fun. All that is required in the next two hours is that you make half a dozen subjective decisions and answer a hundred computer questions. Upon exploring your assigned job, however, you realize there are actually fifty of these magnetic tapes, all of which will eventually need to be reduced into there proper five pages of 'useful' data.

At this point, anxiety may overwhelm you, causing your mind to ramble wildly and start asking reasonable questions like, "What is all of this stuff on the tapes, and why does it need to need to be processed?" Other questions about how you might be able to avoid this new data burden may also roll through your head. Data reduction is a dirty job and management would rather have you do it. Since you're the chosen one, a moment should be spent on what the overall 'big picture' is.

The data on the tapes are output from an interferometer, an instrument to measure nighttime emissions of the atmosphere. The instrument is operated in various parts of the world, and the data are collected on tape to be processed later with the use of large computers. The processing itself takes the interferograms from the tape and laboriously converts them into a few data points relating to the temperature and intensity of the nighttime emissions. Physicists and other odd people will then later interpret the temperatures and intensities, and make wild speculations about chemical and electrical processes in the Earth's atmosphere.

The ultimate goal of the data reduction is to correctly relate temperatures and intensities to the original interferograms, in a timely fashion. A working knowledge of the data reduction system and its options could be helpful in alleviating some confusion. What follows, then, will be a description of the current method being used for interferometer data reduction at Utah State University.

There are three main operations used in the data reduction. First, a Fast Fourier Transform (FFT) is performed on the data to transform it from interferograms to wavenumber spectra. Next, a series of preparation steps must be executed to prepare inputs to the final step, the temperature fitter. The temperature fitter then combines the wavenumber spectra and the preparation files, and outputs the desired temperatures and intensities. Each of these topics, and a description of their associated programs will be included, as well as hints and suggestions that could make the entire process more pleasant and accurate. First, though, I'll take a moment to describe the format and content of the original interferogram files as found on tape.

Interferogram Files

The data to be reduced are interferograms, output from an interferometer. In order to have some depth of understanding of the data reduction process, it is important to know the format of these interferograms, as well as some of the processes from whence they came.

In this discussion, it is assumed that the reader has at least some knowledge of physics (about sophomore level) and a little experience with computers. This section of the paper will deal first with the physical description and meaning of the interferograms, and then will follow with the computer format and representation of the data.

The interferograms are digitized interference fringe patterns from a Michelson interferometer. When the optical path length of the split light beam (C) *(figure 1)* is different than that of (B), the recombined beams interfere, either constructively or destructively, according to the difference in optical path length between beams (C) and (B).

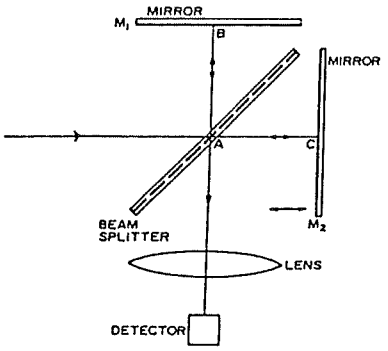


figure 1

As the length of (C) is varied throughout the physical scanning range of the instrument, the light interferes systematically at each difference in path length, producing light and dark areas on the detector. The detector outputs a voltage according to the intensity of light upon it, and as (C) scans, this voltage represents the interference intensity, producing an interferogram.

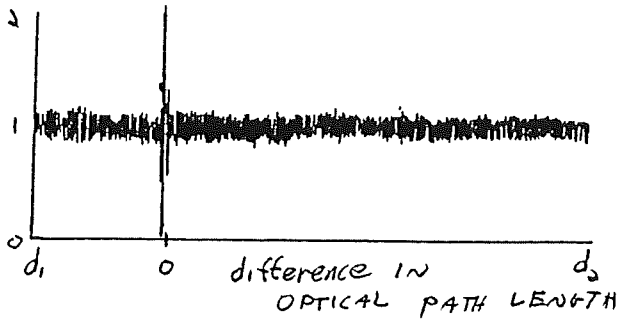


figure 2

It is the data reduction of these curves that will yield useful information about the incoming light that interfered to produce the curves.

Digitizing of the interferograms for computer use could have occurred in many ways using many digitizing systems. Because of this fact, it is assumed that the data has been digitized and translated into a standard format called the Fourier Transform System (FTS). The FTS format is the standard way in which the interferogram files are stored on the computer, and on the original digital tape. Many of the useful computer programs, including all with names starting with 'FTS', assume that the files are stored in this fashion. For this reason, it is important to describe the FTS format and some of the values that are stored within it.

On any given digital tape of data, there may be hundreds of interferogram data files. These files are unformatted to save space, and each consists of two parts: the header and the actual digitized interference fringes.

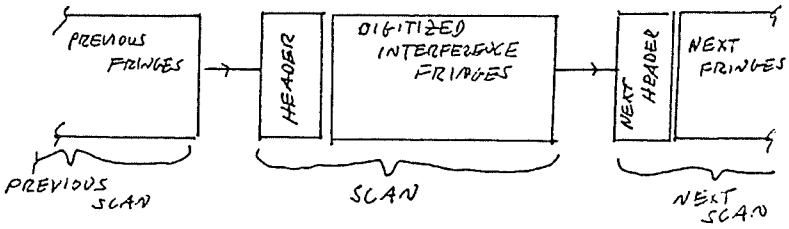


figure 3

The first part of each interferogram data file (scan), is a 512 byte segment called the 'header'. The header contains integer, real, and character data relating to when, where, and under what conditions the particular scan was taken. These values are entered automatically when the data is taken and digitized.

The program HDRPRNTC sequentially prints out some of the important header variables for each scan. Some of these header values will be described later in conjunction with other programs. Another program PRINTHDRL, prints out all of the header variables (if the program is current) and is occasionally used for more detailed information about each scan.

In general, the header variables are automatically calculated and arranged by the computer programs used in the upcoming data reduction process. In a sense, the header is just a place to store the list of 'houskeeping numbers' needed to describe different attributes of the actual interference fringes. These fringes are stored in the second half of each data scan.

The interference pattern of each scan is the interferometer's representation of what light was in the instrument during the scan. There are usually three source of light that any one interferogram may be representing, all of which will be discussed in detail later.

One of the sources (the interesting one) is the nighttime emissions of the atmosphere, acquired during the interferometer's operation at some remote site (Alaska, for instance). The other two sources of light to the interferometer are calibration sources taken periodically during the night in order to properly calibrate the instrument. All of these sources will of course have slightly different interference patterns, but in general, this is a good representation of what the interferograms look like.

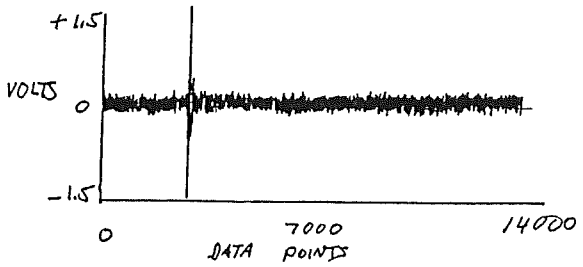


figure 4

The 'center' of the interferogram is the high point (and low point) on the graph and physically corresponds to the point of zero path difference between arms (A) and (B) of the interferometer. The peak voltage of the center (called imod), and the number of points to the left and right of the center are recorded in the header, and output by HDRPRNTC and PRINTHDRL. Because the interferometer scans in both directions, the number of points on the left side and on the right side of the center will shift in a symmetric fashion with every other scan.

The interferograms of the three different sources should all look about like the one above, with some changes in the number of points or in the relative relationship of the features. Occasionally, though, 'glitches' occur (a scientific term) in the acquisition and digitization of the interferograms, and while these errors may take on any form, there are a few common ones to watch out for.

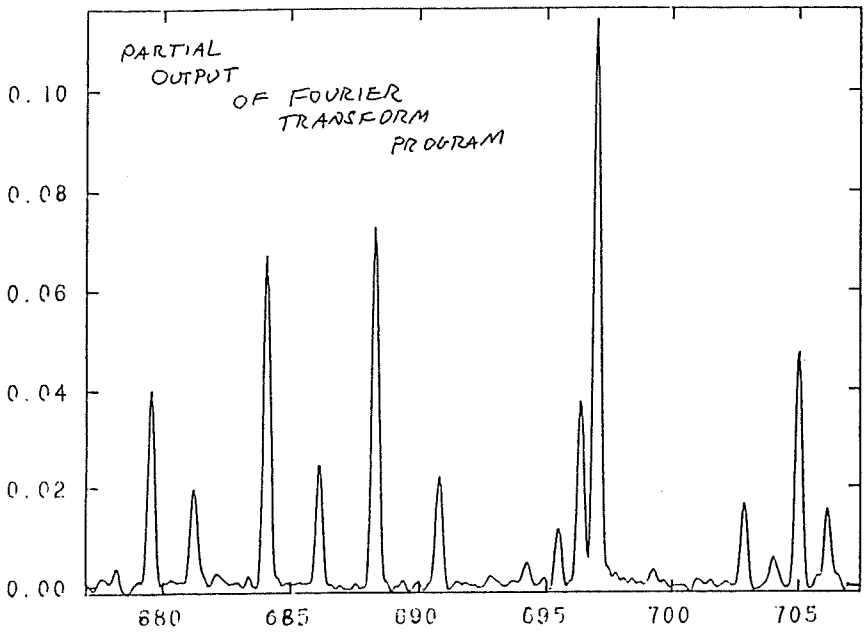
'Noise spikes', for instance, occasionally crop up in an interferogram, looking much like another center added on to the interferogram. A digitizing error, on the other hand, could lead to an interferogram with no apparent center at all. Luckily, though, it is generally not necessary to be to concerned about errors in the interferograms. Bad interferograms are rare and when they occur are usually 'kicked out', or automatically not used by other programs in the data reduction process.

While much more could be said about the storage and physical meaning of the interferogram files, it is probably time to take the initial data reduction step and transform the interferograms into a more meaningful form.

Fourier Transform

The first step in the data reduction process is to transform the interferograms into wavenumber spectra. While the interferograms give a measure of intensity as a function of difference in path length, this tells little about the original light source being measured. Knowing the exact position of the moving interferometer arm (slide), though, it is possible to calculate how the interferograms relate to the relative intensity and relative wavelength of the original light. This calculation process is known as taking the 'Fourier Transform' of the data.

The output of the transform is intensity versus wavenumber where a wavenumber is simply the reciprocal of the wavelength, and is usually used in units of 1/centimeters. This is a small sample of what an interferogram could look like after being transformed into wavenumber spectra. Plotted is relatively how bright the incoming light was at each relative wavenumber (or 1 over wavelength).



PARTIAL NIGHT SKY SPECTRUM

figure 5

The Fourier Transform of a large interferogram takes many computer calculations. One type of computer program that vastly reduced the number of steps it took to do a transform was the Fast Fourier Transform (FFT) algorithm by Cooley and Tukey. Even with the FFT programs, it still takes a healthy computer to transform the interferograms in a reasonable amount of time.

The Fourier transform program currently being used is TRNSFORM (no 'A') by Chuck Harris. This program has many options, which will be discussed, but in general it takes an interferogram (in the FTS format) and transforms it into wavenumber spectra (still in FTS format).

Discussion of the program TRNSFORM is probably best handled by traversing through the questions the program asks, and discussing the possibilities and implications of the answers.

—Input to the program are the interferograms themselves in FTS format. These are the digital data that have been described previously. The output of the program will be the wavenumber spectra, still in the unformatted FTS system.

—The average value of the interferogram is subtracted off, and the 'Delx' sample unit is usually chosen. Details on why these two options should be chosen are beyond the scope of the writer.

—The program asks for the number of points and the weighting to use in the phase correction. The program will use the specified number of points left of the center and attempt to correct any variations in phase throughout the scan. The number of points used in the phase correction must be less than the number of points left of center in the interferograms, and 128 points are usually used. The number given must be a power of two. A ramp weighting is usually chosen. Errors regarding the phase correction will be discussed at the end of this section.

—At this point, the program asks for an apodization type, with obvious choices of zero through five, and a moment will be spent on this input.

The transform of the interferograms yields the interferometer's representation of the incoming light. This representation, however, is far from perfect. Because the instrument is limited in the actual distances it scans the resolution (how well the instrument resolves two close lines) is limited. Furthermore, the very act of stopping the slide causes imperfections in the transformed spectral lines. 'Sidelobes' exist as miniature bumps on both sides of the lines. These sidelobes damp out as they are farther away from the line, but being non-physical, it would be better not to have them at all.

In an effort to minimize the size of the sidelobes, the interferograms are weighted with an apodizing function. In the world we live in though, you don't get something for nothing. The reduction of the sidelobes decreases the effective resolution of the instrument. So if we get rid of the sidelobes altogether, the remaining real lines will become

broad or 'mushy' (another great physics term).

The compromise is a 'little apodization', the sidelobes are greatly diminished, but the resolution is still reasonable. This apodization corresponds to a TRNSFORM input of 1. The programs choices of zero to five are in increasing apodization with zero being totally unapodized.

Another factor inhibiting the resolution of the instrument is the actual number of points used in the transform. For obvious physical reasons, it is impossible to use an infinite number of points in the transform. That is, however, the number of points that would be required to recover the actual width of the incoming wavelengths of light. The fewer the number of points used, the worse the resolution is, and the broader the lines become.

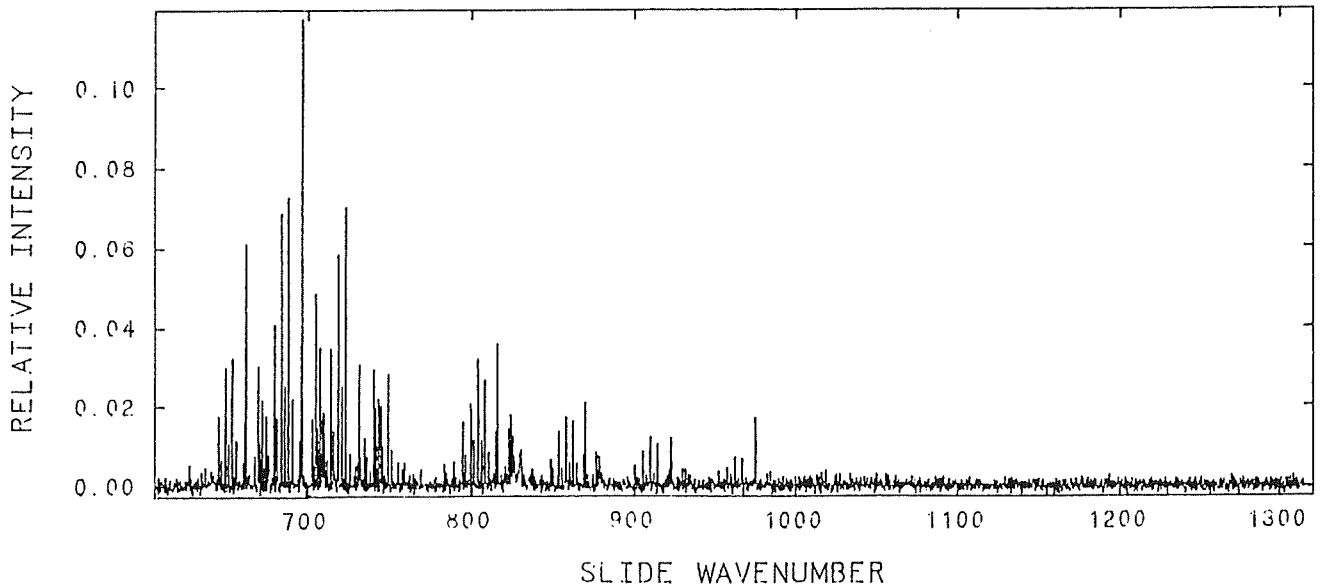
In general, then, it would be nice to use all of the points in the interferogram for the transform. In reality, though, this practice does not yield the desired effect. The program can only use what points are in the interferogram. Therefore if you suggest it use 3000 points to the left of center, and the interferogram in question doesn't have 3000 points left of center, the TRNSFORM program will simply move on to the next scan without computing anything. Also, in an effort to obtain 'standardized' spectra, it is desirable to have the same resolution, or use the same number of points in all the three types of interferograms.

The TRNSFORM program is also rather stubborn about powers of two. While you must specify fewer points on each side of center than are actually in the interferograms, the program will pad the interferograms with zeros up to the next power of two. Thus if you specify 8000 points right of center because all of the interferograms have fewer right side points than this, the program will still use 8192 (next power of two) points in the transform. The number of points specified on either side of the center can never be greater than 16384 because of the array sizes of the programs.

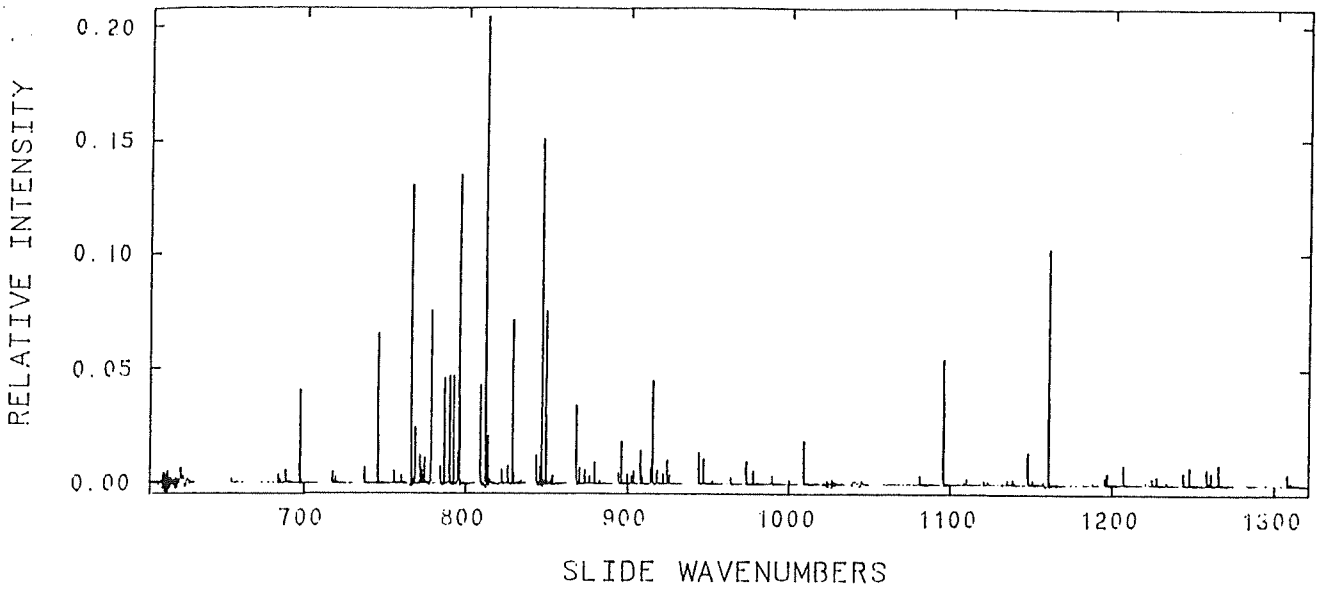
The different types of interferograms (night sky, black body, and neon) may have different numbers of points in them, and may need to be transformed differently (separately). The black body scans in particular are usually transformed short (symmetrically, about 512 by 512 points) because little resolution is required of these scans.

—All that is left then is to tell the program which scans to transform under all of the input conditions. If it is found from the headers (HDRPRNTC) that the interferograms 2-30 and 50-80 are night sky data, the correct inputs are first entered and the program is lastly told to transform scans 2-30 and 50-80.

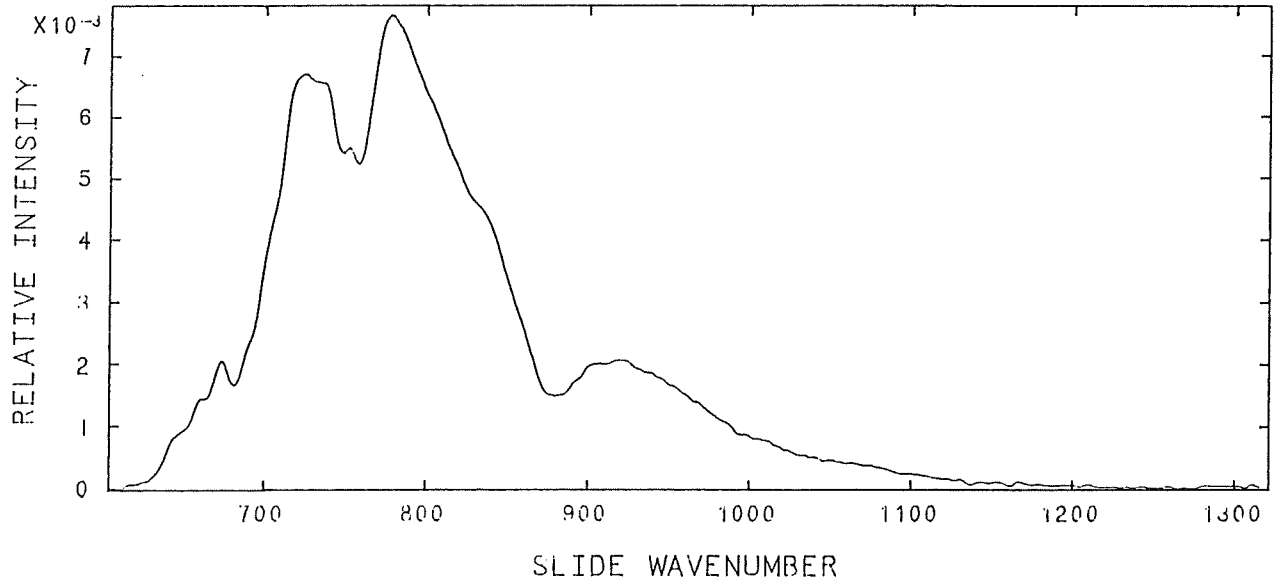
If all goes well with the TRNSFORM program, and if the interferometer was behaving nicely, this is a good representation of what each type of spectra should look like.



SAMPLE NIGHT SKY SPECTRUM



SAMPLE NEON CALIBRATION SPECTRUM



SAMPLE BLACK BODY CALIBRATION SPECRTRUM

The night sky spectra may have other noticeable features due to changes in the atmosphere. The black body and neon scans will be about this shape, although absolute intensity on the Y-axis is almost meaningless.

When errors occur in using TRNSFORM, the program relays a wealth of unreadable information on what the error was. The most common error that the program finds is that the scan in question (number 127, for instance) had fewer than the specified number of points and thus couldn't be transformed. While the number of points left and right of center should be enough to encompass most of the scans, 'short scans' occasionally occur in which one or the other side of center has a fewer

number of points than usual. The TRNSFORM program doesn't transform these scans which is just as well, as short scans are rarely very worthwhile.

Other types of errors may crop up in the transformed data, and no record of that error will be made. If the interferogram has a noise spike, and the program thinks that the noise spike is actually the center, then the transformed spectra will be worthless. It is possible to correct this error by researching the center (having the computer find the real center), but it is generally not worth the time. Occasionally, because of large changes in the intensity of the incoming light during a scan, the spectra will have large 'undershoots' on each line resulting in large negative portions of spectra. As with the noise spikes, it is sometimes possible to correct this 'phase error', but in general it is not worthwhile due to the rarity of the error.

Although at this point the transforming process seems cumbersome, a standardized set of inputs can usually be found for each set of data, making the transforming of the interferograms reasonably easy. Complicating matters slightly will be the assortment of other auxiliary or calibration files that must be created in order to run the temperature-intensity program, which is the final goal. Next then, will be the disussion of the needed auxiliary files, beginning with a short summary of what these files are, and moving on to the programs and inputs to create them.

Within the night sky spectra, there are many different groups of lines or bands, relating to different emissions in the atmosphere. Assigning a temperature and intensity to one or more of these bands is the goal of this data reduction process. In general, each band has different characteristics, and the 'fitting' of the bands to a temperature and intensity is specific and even rather subjective. One of the aspects of the data reduction that is pertinent to any of the features of the night sky, however, is the need for proper characterization of the interferometer.

The interferometer is a dynamic instrument. Along with the deliberate changes in scan time and path length difference, come many less desired changes, like changes in the alignment and wavelength calibration.

The alignment of the instrument is a generic term for how well the instrument can 'see' the incoming light. A good alignment means that for a standard input of light, the detector is outputting a high voltage. Because the instrument response (sensitivity) to light is changing, it is necessary to occasionally look at a standard light source, in order to access how sensitive the instrument is at any time. The previously transformed black body spectra are the relative calibration of the instruments sensitivity, and will be used to find the absolute response of the interferometer.

Another characteristic of the interferometer is that the 'wavenumber' scale on the spectra is also changing & not easily related to any known quantity. The wavenumbers shown are a result of the optics of the instrument and they don't have any physical meaning with respect to the incoming light. In order to fit a temperature and intensity to these night sky lines, it is mandatory that there true position (relative to some known quantity like the speed of light) be determined. Knowing something about the relative wavenumber, or *slide* wavenumber that the night sky lines usually appear near, it is possible to use tabulated values of the true position of the lines and obtain a relationship between the slide wavenumbers shown, and the true wavenumber position of the lines. This process, known as the wavelength calibration, is necessary for proper determination of temperature and intensity, and the programs to achieve the wavelength calibration are the first of the calibrations programs to be considered.

Wavelength Calibration

The wavelength calibration of the instrument is a set of four numbers that relate the slide wavenumber as shown in the original spectra to the true, real wavenumbers. To create these four numbers, the slide wavenumber positions of large, well defined lines are compared with where those lines should appear on a real wavenumber scale. The four parameters of the wavelength calibration can be thought of as the 'conversion' from slide wavenumbers to real wavenumbers.

The first step of this process is to obtain a very good 'calibration spectra'. The calibration spectra is usually a collection of night sky or neon scans all added together. The programs can then compare the slide wavenumber (swn) positions of the lines in the summed night sky or neon, to a table of real wavenumber (rwn) positions. While either the night sky or the neon spectra can be used for this process, this discussion will focus on the night sky wavelength calibration.

—All of the spectra have 'noise', or small random fluxuations in them, due to imperfections in the instrument and the recording system. This noise is unphysical and unwanted, and could hamper the wavelength calibration process. The program FTSCADDQ is used to add together several (around one hundred) night sky spectra. Because the noise on the scans is random, adding many scans together cancels the noise out by a factor of the square root of the number added together. Adding 100 scans together, then, reduces the noise on the resutling added scan by a factor of 10. The coadding program needs to know what output type to use, and '0', for low precision is usually entered. The night sky scans to be added together should all have the same resolution. The result of this first step is a good, relatively noiseless calibration scan to compare with the tabulated positions of the lines.

—A table of line positions is used by the program PEAKFND to compare the swn positions of the lines in the calibration scan to their real wavenumber positions. The table consists of 'best guesses' of where the line usually appears in the data, coupled with the real position that line should be in. The program PEAKFND looks at a table (AKOH.LNP for example), reads the estimated position of a line, and looks at that swn position in the calibration scan, trying to find the exact (albeit unphysical) slide wavenumber positon of the line. The output of PEAKFND is a table of real wavenumber positons and exact slide wavenumber positions of many of these lines.

—The output of PEAKFND needs to be passed to the program WAVEFIT which takes the exact slide wavenumber and corresponding real wavenumber positions and calculates the four parameters that relate the two scales. These four parameters can now be used to convert any slide wavenumber to a true wavenumber, and they are considered the wavelength calibration for the instrument around the time the night sky scans used in the calibration spectra were taken.

The wavelength calibration is something that needs only to be 'close enough' to work properly. There is a well defined line between a good set of four numbers and set that won't work for the data that needs to be reduced. Most of the time, using the FTSCOADDQ, PEAKFND, and WAVEFIT programs with a standard table of line positions will place the wavelength calibration well into the satisfactory region. There are times, however, when the process needs a bit of 'tweeking' to yield the correct conversion, and it is worthwhile to look into the errors that might occur.

PEAKFND and WAVEFIT output many parameters to the screen. One of these numbers, the 'RMS error of the fit', placed on the screen by WAVEFIT, is a measure of how well the program could fit the exact swn , rwn position into the four parameters. This number needs to be *less than .02* for a satisfactory wavelength calibration. Checking the RMS error is the first and possibly only indications of whether the calibration is good enough. If the RMS error is not less than .02, probably one of two errors have occurred. First, some of the swn 'guesses' in the line position file given to PEAKFND, are not close enough to the actual swn positions of the lines in the calibration scan, and second, it is simply not possible to find a good set of four parameters that fit the entire scan.

The screen output of WAVEFIT includes a table of the difference between what the program calculated the swn of each line would be and what the exact slide wavenumber of that line actually was. If this number is large, either positive or negative, there is a good chance that the initial line table guess about the swn position of the line is wrong. The initial guess must be within .4 swn of the exact swn position of the line in the calibration spectra in order for PEAKFND to find the line. Drastic changes in the instrument (generally between trips to remote sites) can occasionally put the swn position of a line or group of lines to far from the initial guess, and the line table of guesses, rwn positions must be changed.

If it is suspected that a line is being 'missed' by PEAKFND, a good approach is to plot the line on the screen and make an estimate of what swn

the line peaks at. It is usually possible with the right scaling on the plot to make a guess within the .4 swn limit that the PEAKFND program needs. The line position table can then be updated with the new guess, which should be close enough to cure the problem.

Sometimes it seems as if all of the lines in the calibration spectra are being found correctly, but the RMS error of the wavelength calibration fit is still too high. In this instance, plotting out the calibration spectra, checking the resolution of the scans (HDRPRNTC) used in the coadd, or even switching to a neon calibration coadd and line position file may be necessary to cure the problem.

It is important to note that the wavelength calibration works well only over the range for which it is fit. For instance if the line position file does not contain swn guess, rwn position pairs for wavenumbers covering the whole spectrum, it may not be justified to use that calibration outside of the wavenumbers that were in the table. Shorter tables, with only a few lines in them work fine as long as the range your fitting temperatures and intensities to does not go too far outside the wavenumbers in that table.

The wavelength calibration works well over a wide variety of conditions. While the procedure for obtaining the calibration may be more rigorous under special conditions, for most applications a process as described above using the night sky or neon spectra proves satisfactory. Now that a relationship between the swn and the rwn of the instrument has been found, it is important to find out how sensitive the instrument was when the night sky data was taken. This leads to a discussion of the response calibration.

Response Calibration

The 'alignment' of the interferometer describes how well the light from the instrument is focused on the detector. This focussing then effects the output of the instrument for any given light input. The alignment of the instrument can change due to temperature variation, vibration, or other factors, so some measure of the interferometer's ability to detect light is needed at various times throughout an observing period. Periodically throughout an observing night, a known, constant source is observed through the interferometer. These black body scans show the relative sensitivity of the instrument at each wavelength. As the alignment of the instrument changes, successive black body calibrations will be different, most likely showing a decrease in sensitivity of the instrument.

The source used as a constant light (the black body source) is a 'light bulb' inside the interferometer. While periodic scans of this source will tell how the instrument alignment has changed, the source itself is relative in that its light output isn't known in measurable units. What is done is a 'calibration' of the black body calibration source. Before each trip to a remote site, the brightness of the black body source is compared to the brightness of a known, calibrated, 'offical' source in the laboratory. A standard ratio between the real, calibrated source and the internal black body source is calculated and can thereafter be used to 'correct' the internal black body so that the *absolute* response of the instrument can be found. It is this absolute response of the instrument that the temperature fitting program needs to compute temperature and intensities.

—After the black body scans have been transformed, the scans corresponding to each calibration period are coadded using FTSCOADDQ to acheive the relative response of the instrument at that time.

To find the absolute response of the instrument, the coadded black body scan is corrected by the appropriate 'standard' found in the lab comparison to a known source. The program IBBRSPMKR takes the coadded black body scan, and outputs a response curve, still in FTS format that is the absolute response of the instrument at the time the black bodies were taken.

—One input to IBBRSPMKR is the coadded black body scan. The slide wavenunber range is not reset, so the program caluclates over the whole

spectra. IBBRSPMKR also asks for a black body 'standard'. This is the lab correction to the internal black body (NSF85IBB.STD is an example), and each mission or trip to a remote site has a different standard. The output of this program is the absolute response file for the instrument at the time the black bodies were taken.

While in principle this process will yield a very good absolute response estimate of the instrument, a problem occurs in doing the lab work and calculating the standards. The instrument when used in the field is rarely in the same configuration as when calibrated in the lab. These slight differences mean that the absolute calibration of the instrument is in itself a bit relative, and that the intensities calculated with the temperature fitter may have absoute units that are slightly in error.

Both the wavelength calibration and response calibration need to be kept current. When working with data from the late part of an observing night, the calibrations used with that data should come from the same part of the night. Since night sky, black body, and neon scans are all taken frequently throughout the night, this usually isn't a problem.

Occasionally the instrument is realigned. The sensitivity of the instrument drops so low the operator stops taking data and adjusts the optics to bring the interferometer back into alignment. Black bodies scans taken just before the realignment will obviously be much different than the ones taken after the sensitivity was deliberately changed. Care should be used to keep separate the data and calibration scans taken before and after the realignment. At other times, the instrument will not change in sensitivity for long periods of time (a whole night is considered a long time), and it will be found that all the black bodies for the entire night will look about the same. Under these circumstances it may be possible to coadd the black bodies from several *different* calibration sequences and make one response file for characterizing the entire night's instrument response.

After working through the transforms and calibration programs, it may seem that you are no closer to reaching the imposed goal of temperature and intensities. In reality, though, only one lengthy program separates you from that goal. The last program, FTSTMPFIT, takes the night sky spectra, wavelength calibration, response calibration, and a little information about quantum mechanics, and outputs the desired results (maybe). There are several versions of this program for special applications, and

the emphasis of the rest of this paper will be on FTSTMPFIT.

Temperature Fitter

The program FTSTMPFIT was written by Patrick Espy to fit a temperature and intensity to a night sky scan. The internal workings of the program are quite interesting but not terribly relevant to this paper. FTSTMPFIT uses the night sky and calibration data, and tries to 'best fit' quantum mechanics constants to the band in question. The temperature and intensity of this 'best fit' are then output.

Many of the inputs to this program are quite dependent on which band, or group of lines, that the user would like to know the temperature and intensity of. There are dozens of these bands, and it would be unreasonable to try and discuss them all. A better approach may be to run through the various possibilities of inputs of the program. Then, if the band to be fit and some other information would be provided (like which swn range the band lies in), the user would be in a position to use FTSTMPFIT effectively.

—The temperature fitter types out a string of useful information relating to how 'easily' the data is being fit. The first input asks the user to specify a file for this status information. Generally, it is good to see this information as soon as possible, and entering 'TT' will print the status report to the screen as the program runs.

—A choice of filtering the data is given next. Unwanted scattered light sometimes enters the interferometer as it is taking data, and this light will show up as a broad rounded baseline that all of the

spectral lines seem to be riding on. Because this light has different characteristics than the light of the nighttime emissions of the atmosphere, it is possible to filter out the scattered light portion, and doing so can give a more accurate result.

If it is felt that filtering is needed, the time constant and order of the filter will have to be specified. The time constant is really a measure of slide wavenumbers, and as an example, if 100 were entered, the program would try to filter out all data in the night sky spectra that was broader than 100 swn. It is important to ensure that the time constant specified is larger than the breadth of the band being fit, otherwise it might be possible to filter out needed information. The 'order' of the filter relates to how 'strongly' the filter is applied. With the way the program is set up the order is usually set to four or eight, the latter being used for 'stronger' filtering.

—FTSTMPFIT assumes several 'default settings' which rarely have to be changed. These settings include exciting values like 'parameter fractional error', 'nunber of sidelobes', and 'initial starting temperature.' Only under special conditions will these values need to be altered, and it would probably be best to move on to the next input, the actual night sky data.

—The filename of the night sky data that you would like temperatures and intensities of is next on the agenda. This file is the FTS formatted spectra that has been discussed at some length previously. If the program cannot find the data file, the program stops.

—'Enter output filename' are the computer codewords for 'what would you like the output file of temperatures and intensities to be called'. The best bet is to answer with whatever you want the file of temperatures and intensities to be called (this is an easy one). If the filename you enter is not a valid computer filename, or if you have no room on you account to open the file, the program will not continue.

—At this point, the FTSTMPFIT program wants to know what slide wavenumber range to use in the fit. This range should be the range for which the band in question falls. Usually entered are the swn where the first line of this band appear, and the swn for where the last line appears. Under special conditions, a swn range larger or smaller than this could be specified, but if a good deal of the band is not in this range, the program won't go look for the rest, and the temperatures and intensities may suffer.

—The temperature fitter needs to know which of the night sky scans of the input file that you would like output for. The answers here are the beginning and ending input file scan numbers for the night sky spectra of *this* calibration period. The wavelength calibration and response files will be entered later, and the night sky scans specified here must correspond to those calibration files. It will be possible to enter the scan numbers for the next calibration period later, but this input is only for the first calibration period.

—Now is the time to enter the filename of the wavelength calibration that was calculate previously. This calibration will be used on the scan numbers entered above. If the program cannot find the filename specified, *no error message is given.* The four parameters of the wavelength calibration are all assumed to be zero. While this will obviously cause problems in the actual temperature fitting process, no error message is given at this time.

—Quantum mechanics plays the deciding role in this temperature fitting process, and it is at this input that the 'rotational file' is entered. The rotational file contains numbers that the the program will try to fit to the data. These constants are extremely specific, and change with each and every band. The correct rotational constants must be entered at this point, and if the file is not found, the program will not only stop, but it crashes to an ugly halt. If the rotational file used does not correspond to the swn range entered previously, problems are about to occur, but the program will make no mention of it at this time. The rotation files can be generated using another set of programs, and an extensive selection of the most 'popular' files can be found all having the filename extension '.ROT' for 'rotational'. Which rotational file to use will need to be specified, probably by the person who asked you to do this in the first place.

—The initial and final state weightings are options that tell the program how to internally

compute some of the fitting parameters. The choice of initial of final state weighting depends on which band you are fitting. Currently, the initial state (I) is being used for all bands *except* for the Meinel bands of the molecular nitrogen system. Trust me on this one.

—The last of the calibration files, the response file, is now input. Like the wavelength calibration file, this calibration needs to correspond to the scans to be fit as specified at the beginning of the program. Once again, if the program can't find the file, no immediate problem is seen.

—Some of the bands to be fit contain spectral lines from other emissions. These extra lines could confuse the program, which only knows about the lines that were in the specific rotational file. While sometimes it is possible to choose a swn range so that the offending lines are not part of the fit, it is often necessary to 'mask out' the unwanted lines. A masking file, if needed, contains beginning and ending slide wavenumbers of small spectral regions that the program should not use in the fit. These swn, arranged in simple (x,y) pairs will be the places were the unwanted lines occur, and can usually be found closely by plotting to the screen the unwanted lines and picking off the correct swn. A masking file of the (x,y) pairs will then need to be made from scratch. Often a masking file is not used, and if the program can't find your masking file, it assumes you're not using one.

After entering the masking file name, the program sets about fitting the night sky data with a temperature and intensity. The status file is started first and will print to the screen if so specified. The status file contains information about the parameters input to the program, and the report on how long each scan took to fit. Many of the errors that occur with FTSTMPFIT can be detected in the output of the status file.

The first part of the status file consists of a list of all the inputs to the program. If, for instance, the program does not list a response file, then the program could not find the response file you entered and won't be using one in the fitting (yes, this is an error).

After showing you the parameters about to be used in the fitting process, FTSTMPFIT *actually* begins to fit data. A 'scratch file' is opened (something personal within the program) and the program fits each data scan, printing out a status report message for each scan. This message tells how many 'iterations' or loops the program went through to fit the particular scan, or 'explains' an error on why the program wasn't able to complete the fit.

When the fitting of all of the specified scans is complete, the program returns with the question of which scans to do next. If more scan numbers are entered, the program asks for wavelength and response calibration filenames in case the night sky scans are from a different calibration sequence. The program continues this process of fitting data and asking for more scan numbers and calibration files until the scan numbers '0,0' are entered, or the last night sky scan is reached. At this point, the fitting is done and the output file you specified earlier contains the band temperature and intensity for each scan in the night sky data file.

As with all of the data reduction programs, FTSTMPFIT is full of places for an error to occur. The most common error is that the temperature and intensities output have large error estimates, or may be *totally* unphysical (negative intensity for instance). FTSTMPFIT is a least squares fitting program and seems almost to have a mind if its own. As long as the files input to the program are correct, and the program was able to find those files, deducing the reasons for bad output can be quite hard. The program needs to be 'lived with' for a while to learn all of its personal preferences. If an error occurs, and all of the inputs to the program seem appropriate, it may be necessary at first to seek help from someone that you perceive to understand this stuff.

In most cases, FTSTMPFIT and the whole data reduction process work quite well. After a while, the system may even become very routine. This paper is in no way the total amount of information needed use the data reduction process under all conditions. It is hoped, however that the provided information will be useful as an initial guide and a reference, and will clear up some of the questions that could take years to infer under regular work conditions.

The interferometer and data reduction system have taken over a decade to develop. If you don't feel you quite understand the process and all of its 'personal' intricacies, find comfort in the fact that no single person understands it all. So at first, if your employer asks you to spend two hours converting 4000 pages of data into more 'useful' data, find out if *he* could execute this data reduction process.

Cooley J. and Tukey J. *Maths Comput.* 19 (1965), 297.

Chuck Harris and Patrick Espy of the Center for Space Engineering, Utah State University, Logan, Utah.

This paper was written in Word Perfect and then transferred to the USU main computer, a VAX 8650. The paper was typeset and printed using the public domain program TEX.