



MULTIPLE MIRROR TELESCOPE OBSERVATORY

Smithsonian Astrophysical Observatory and Steward Observatory, University of Arizona

Reply to: MMT Observatory
University of Arizona
Tucson, Arizona 85721
(602) 621-1558

MMT Technical Memorandum 87-1

From: Phil Massey
Re: Reducing MMT Spectrograph Data with IRAF
Date: February 2, 1987

Although many powerful tools exist within IRAF, packages such as ONEDSPEC are header driven and can be awkward to use on non-Kitt Peak data. Although this situation may be corrected in the near future, this guide will show you how to reduce MMT data relatively painlessly now, and may prove useful in reducing data taken with similar instruments. In particular I've noted what to do differently with Steward 90" data. This guide assumes that you have a passing acquaintance with IRAF (i.e., about what I had when I started these reductions) and that you are starting with a FITS tape written on the mountain. People trying to reduce 90" Steward data will need to have run their data through the Steward program IRS that will convert an internal Steward format to a FITS tape; you might as well do sky subtraction while you are at it. I did not actually flux-calibrate my MMT data, but I've given some notes as to how to do it.

(1) Get into **noao** and then load **dataio** and **onedspec**.

(2) Load your tape into IRAF by **allocate mta** (or **mtb** or whatever drive you happen to be using) and then do an **rfits mta 1-999 mmt.0 make+ long+ data=r > fitshead**. This will create file names of the proper form for ONEDSPEC to operate on; i.e., **mnt.0001**, **mnt.0002**, etc. The complete fits header will be written to the file **fitshead**. Substitute the final number for **999** in all of the following.

(3) Next we need to get rid of the garbage channels at the end. Use **splot** to examine a quartz exposure from the left channel. The cursor commands **a** (for autoexpand) and **c** (prints the value of the cursor position) will be useful; use **g** to get a quartz exposure from the right channel and finally **q** to quit. While looking at the data, decide what pixel range contains useful data (channels 1-3915 in 8k mode, say). To keep only the useful channels, do an **imcopy mmt.0*[1:3915] mmt.0***. This will rewrite only channels 1-3915 over the old files, so really know what channels you want to keep before doing this step.

(4) Next we are going to fix the headers so that ONEDSPEC can tell the left channel from the right; this will make everyone's life easier. Create two files containing the names of your spectra by typing **files mmt.0* > beam0** and **files mmt.0* > beam1**. Next **edit** these files so that only the left-beam file names are in **beam0** and only the right-beam file names are in **beam1**. This will likely be just the odd numbers and even numbers respectively. Add the beam-numbers to the header info by **hedit @beam0 BEAM-NUM 0 show+ ver+ add+ up+** and **hedit @beam1 BEAM-NUM 1 show+ ver+ add+ up+**.

(5) Similarly, we need to denote which files contain "object" and which files contain "sky." Create two files (**obj** and **sky**, say) which contain the appropriate file names as we did in (4). Next add this information to the headers by

```
hedit @obj OFLAG 1 show+ ver+ add+ up+ and
hedit @sky OFLAG 0 show+ ver+ add+ up+ .
```

If you are working with Steward 90" data that has already been sky-subtracted, then all you need to do is to create a single file containing all the names of the data files and do the first of these two commands.

(6) While you are at it would be a good time to fix any wrong labels. To change the title of an observation do **hedit mmt.0nnn title HD48099 up+ show+ ver+**. You should probably print out short versions of the header info at this point by doing **slist mmt 1-999 short+ > list** and **lprint list**.

(7) Now it's time to add up all the quartz exposures (INCANS, if you prefer). First you might want to convince yourself that the individual ones are OK by **sploting** each of them. For my data, fixed-pattern noise in the Reticon electronics grossly dominated over sensitivity variations on the image tube front (the former is the same regardless of the spectrograph configuration; the latter will vary each time you change gratings or tilts), so I added up all my quartz exposures regardless of grating or wavelength region. You should determine whether you want to do this or if you would be better off dealing with the quartz exposures night-by-night or grating-by-grating or somesuch. In any event you will want to then do something like a **flatfit mmt 11-16,201-220,351-360 flat order=1** in order to create two files **flat.0000** and **flat.0001** containing the quartz data (midpoint normalized to 1 to roughly preserve the number of counts in your data). See the parameter file in Fig. 1. Now divide this into all your data by **flatdiv mmt 1-999 mmtq 1 flat**. This will create new files called **mmtq.0nnn**, starting with **mmtq.0001**.

(8) The longest (and to me, most frustrating) effort is in fitting all of the comparison (HeAr's to you) records. I strongly urge you to review the help pages on the commands **IDENTIFY** and **REIDENTIFY**. These routines are complex and occasionally subtle in what they do--usually something that you don't expect! My observing procedure was to obtain a long comparison each afternoon and then do a short comparison (CEX) at each new telescope position. Ideally, what you would like to do is cross-correlate the long exposure against the short exposure, letting the zeroth-order term and maybe a couple of the other terms flap around. You can't do this with IRAF. (You can cross-correlation the positions of "features" in the comparison, but this is not what we need.) What I found worked the best was to do a full-blown dispersion solution for each of the short exposures, using the long exposure as a guide; this worked OK, but you do have to be very careful about where **IDENTIFY** or **REIDENTIFY** are taking the centers of weak lines to be, as the attached figure (Fig. 2) demonstrates. However, with a little care I was able to keep average residuals to the fits down at the 0.1 A level (with the 832 grating); of course, I wasn't interested in radial velocities of high accuracy. I am sure you can do better than that if you use only the strongest lines; however, there are seldom strong lines at the ends and the distortion in the MMT spectrograph is sufficiently severe that

you can get into trouble at the ends. In order to understand the behavior of **IDENTIFY** and **REIDENTIFY**, keep in mind that it chooses a line center to be where the first inflection point is; i.e., if your line is jagged, the positive noise spike nearest the "expected" position of the line will be taken to be line center. You can always set the line position by eye using the cursor without letting it recenter by using the graphics command **t**.

(a) Make a file containing the wavelengths of good comparison lines, 1 entry per line. Keep in mind that there is lots of distortion in the system, and the more lines the better off you are. On the other hand, you probably don't want to include blends.

(b) Edit the parameter file for **IDENTIFY** (i.e., **epar identify**) so that it resembles the one shown in Fig. 3. The **threshold** parameter should be set to two-thirds of the weakest line you intend to use. Note that this will be different for your long exposure than for your short exposure. Make sure that the correct wavelength table has been entered.

(c) The comparison for each night (and each grating) will have to go into a new database. For the first (i.e., long) comparison do **ident mmtq.0999 database=832a**. Use the cursor keystroke **m** to mark and identify 8 or 9 comparison lines throughout the spectrum. Note that the program will grab the complete wavelength out of the tables, you need only identify a line to the nearest angstrom or so. Then make a preliminary fit using **f**. This will put you in the interactive curve fitting routine. Set the order to 4 (**:o 4**), and do a new fit using **f**. You can see the non-linear part of the fit by typing an **l** while in the interactive curve fitting routine (note that **l** does something quite different when you are back in the **IDENTIFY** portion. Delete any points that look wild, using a **d**. An **f** will make a new fit. Type a **q** to quit and go back to the **IDENTIFY** portion. Type an **l** and the program will find all the lines from your table based upon the preliminary fit. Check each one by first doing a **z** to expand around the first line, and a **+** to move on the next line. Bad lines can be removed by doing a **d**, or moved to a new position by doing a **t** and then a **c**. If this seems to do something bizarre, you probably tried to move a line that wasn't the "currently selected line" (which **+**, **-**, and **.** will all reset - have you read those help pages yet?). When you are done with this do a new fit **f**. Change the order to whatever you need; I found that a 10th order cheb worked fine on my MMT data: any lower and I could see systematic effects in the residuals (which can be shown using the **j** command when in the interactive curve fitting portion), and any higher did not make a significant improvements in the rms of the fit (Fig. 4). The Steward data seemed to work well with a 5th order cheb (Fig. 5). This would be a good time to set **niterate** to 2 (**:nit 2**) as this will prevent **REIDENTIFY** from getting too badly perturbed by wrong lines. **q** will return to **IDENTIFY** and another **q** will exit and save the solution. Answer "yes" when it asks you if you want to write to the database.

(d) Using this fit for channel 0 (**mmtq.0009**, say) we can easily fit the beam1 comparison that goes along with it. Do a **ident mmtq.0010**

database=832a. Then read in the old solution by doing a `:r mmtq.0009`. This will mark the features from the left beam on the spectrum for the right beam; of course, they will not match as yet. Do an `a` and then a `c` to recenter these features. This will work if the two beams are similar enough (try it and see what happens); if it didn't do so good, you might want to use `s` to shift the features - read that damn help page. Go through the lines one by one and fix up any mistakes (`z` and then `+`); then do a fit (`f`) of the same order as last time. Remember to set `:nit 2`).

(e) You now should have two entries in the database `832a` (or whatever you chose to call it); if you do a `dir 832a` you should see something analogous to `idmmtq.0009` and `idmmtq.0010`. You should now be able to use `REIDENTIFY` to fix up the rest of your arcs for that night. First edit the parameter file `REIDENTIFY` (i.e., `epar reidentify`) to have the appropriate database and a threshold appropriate to your short exposure. Let `nlost` equal 5 or so (see Fig. 6). Then do a `reidentify mmtq.0009 mmtq.0nnn` for each left-beam arc for that night `0nnn`, and a `reidentify mmtq.0010 mmtq.0mmm` for each right-beam arc `0mmm`. After you have done that there should be an entry for each comparison record in the database. You should check each fit by running `ident` on each record and doing a fit; fix up any bad points. Go ahead and do all the other nights, keeping each new night in a new database. When you do the first exposure from the second night you will want to a solution from the previous night as a guide; the easiest way to do this is `ident mmtq.0337 database=oldname`, and then in cursor mode read in the old solution `:r mmtq.0009` and then switch databases so that the new solution will be written there by doing a `:database newname`.

Note that if you are reducing Steward 90" data you are going to have to do things just a little bit differently at this point. Since there are no times in your headers, you are better off if you simply write all your solutions into the same database at this point, and then, when you are done, copy each pair of them into a new database so that you wind up with a new database for each arc exposure. You can do this relatively painlessly by first creating the new database using `mkdir 832e` and then renaming each solution by `rename 832a/idmmtq.0015 832e/idmmtq.0015`, but remember to keep the pairs together in the same database.

(9) Before putting your spectra on a linear wavelength scale, it's time to put the time into the headers. `ONEDSPEC` expects the UT time, so we will lie to it and insert the MST available via `DATE-OBS` as the UT. Not to worry. To do all this, though, you will have to type in an IRAF script ("program" to you) that will fix up the way the MMT writes their FITS tape. The one shown in Fig. 7 was written by Jeannette Barnes and subsequently modified by yours truly to include the integration time as well. To get this program running you need to edit a new file called `mmttime.cl` and type in the script shown in Fig. 7. Unfortunately, the exact form of this script depends a little on whether you are using the "old" IRAF (version 2.3 or earlier) or the "new" IRAF (version 2.5); ask someone or read the logon message carefully when you start up and make sure you type in the appropriate version of `mmttime.cl`. You will then need to edit another new

file called `mmtime.par` which will contain the parameter file info which is also provided on the next page. Don't lose any commas!

To run the program, first create a file containing all the names of all your quartz-divided spectra (whatever you decided to call them in 7 above, `mmtq.0nnn` for instance), i.e., files `mmtq.0* > stuff`. Then you need to tell IRAF that `mmtime` is really an IRAF thing: `task mmtime=mmtime.cl`. Ok, ready? `mmtime stuff sh+ ver-`. If you were using the Steward 90" you can just forget all this stuff.

(10) It's time to dispersion-correct (linearize) your data! Read the help page on DISPCOR. Did you always take one HeAr before each exposure? If so, you want to set `guide=precede`. Did you instead do one following each exposure? If so, `guide=follow`. Did you do both? Then `guide=linear`. Random? Then you might try `guide=nearest`. If you skipped step (9) above and there are no times in your header, none of this matters, as you are just going to have to specify which database goes with which files. Attached (Fig. 8) is a sample parameter file. Note that `time_wrap=12` simply means that a new observing day starts noon, since you are using MST.

In using DISPCOR you have the option of setting the starting wavelength (`wstart`) and wavelength per pixel (`wpc`) or of accepting the defaults, which will be computed fresh from each database. All the data from a single night (i.e., a single database) will have the exact same starting wavelength and wavelength per channel, but data taken the next night, even with an identical setup, will have slightly different default values owing to the slightly different fits in the second database. If you might want to add spectra taken on several different nights, the easiest thing to do is to make sure that the values of these two parameters are the same from each night; i.e., accept the default the first time but then enter the values explicitly for the second night. To run dispersion, correct your data from the first night do a

`dispc 832a mmtq 1-nnn`. It will query you for the name of the dispersion corrected files (we will assume `mmtqd` in what follows) and the starting record number (1). If you are reducing Steward 90" data instead, you would do the same except that you must execute `dispc` once for each different comparison solution (i.e., for each database).

(11) If you are planning to flux-calibrate, this is the time to go to work on your headers adding the airmass to each record. Otherwise proceed to step (12). This step can be very painful, but we will use a trick to make it a little more tolerable: first we will lie to it and tell it that all the airmasses are 1.0 by doing a `hedit mmtqd.0* AIRMASS 1.0 ver- add+`. If you are using an "old" version of IRAF, this will have put a nonsense number in for the airmass, in any event, due to a "well-known" bug, but it doesn't matter at this point. Next do a `hedit mmtqd.0* AIRMASS add- up+`. It will query you for the correct airmass value for each file. If you are reducing Steward data, the airmass information is buried in the image header as a "comment": you can view this by `inhead mmtq.0nnn long+` but you can't extract it, as IRAF is not yet smart enough to be able to distinguish one COMMENT line from any other COMMENT line. If you are reducing MMT data, hopefully you were fast enough to write the airmass information down on the log sheets. Otherwise, you need to get the sidereal time, the RA, and the Dec out of the header comments (`inhead mmtq.0* long+ > filestuff` and `lprint`

filestuff will do this) and then type them back in using the magic keywords **ST**, **RA**, and **DEC** using **hedit**, (i.e., **hedit mmtqd.0nnn ST 08:32:12**). This can be pretty painful, I know, but that's how life is sometimes. (If you are fanatically compulsive, you're welcome to precess the 1950 coordinates to that of the current epoch so that the airmass will be strictly correct, but then why are you trying to flux data through such tiny holes?)

If you are reducing Steward data and plan to flux-calibrate, unfortunately you are going to have to add the exposure times by hand as well. The format is **hedit mmtqd.0nnn EXPOSURE 960**, or, if you want it to query you for each one, do a **hedit mmtqd.0* EXPOSURE 1 ver-**, followed by a **hedit mmtqd.0* EXPOSURE add- up+**.

The next step, if you are going to flux-calibrate your data, is to construct a table containing the standard star magnitudes, unless you used flux standards found the IIDS Standard Star Book. If you need to enter one of those Fillipenko stars, then use the format shown on the attached sheet (Fig. 9) - the first number is the number of flux points, the next group is the wavelengths of the flux points, followed by the bandpass widths at these wavelengths and the extinction (magnitudes per airmass) at these wavelengths, followed by the standard star data itself. For each set of standard star observations you need to run **standard inter+ beam+**, having first done an **epar** on **standard** to tell it what table you are going to use for the standard star flux data (the IIDS standards are in **onedspec\$iidscal.dat**). For sky-subtracted Steward data, you need to run **STANDARD** with **beam-switch** off, i.e., **standard inter+ beam-**. Give **standard** the exact same name as the name listed in the standard table. When you are done doing **standard** you must run **sensfunc** to actually fit a sensitivity function for the two beams. I strongly recommend you read the relevant HELP pages on these two routines and maybe glance through the IRS/IIDS IRAF Cookbook.

If you are reducing Steward data that has already been sky subtracted, you will be skipping the next step (**BATCHRED**) and thus will have to **extinction** correct your spectra now using **extinct mmtqd 1-999 red start=1** to generate new, "reduced" spectra.

(12) You are now ready to do sky subtraction. The assumption at this point is that for each object, you have observed an equal number of skys and objects in each of the two channels; the order is not important except that **BSWITCH** will require an equal number of skys and objects within each group of four records that it tries to process. You should have no trouble with this, though, as a group of four records does not have to consist of consecutive numbers; i.e., if your records 1-8 were object, sky, object, sky, sky, object, sky, object you could specify 1-2; 5-6; 3-4; 7-8. Edit the parameter set for **BSWITCH** (i.e., **epar bswitch**) so that **ids_mode=yes** and **weighting=no**. If you are not going to extinction correct (see above) then set **extinct=no**; otherwise **extinct=yes** and you must set **latitude=32.0** if **extinct** is going to have to figure out the airmass from other information in the header (e.g., the **ST**, **RA**, and **DEC**). Next we want to run **BATCHRED** (which will call **bswitch**; that's why we had to **epar** it). The help page on **BATCHRED** isn't as helpful as the IRS/IIDS Cookbook description, but in fact there's nothing much to it. Just follow the example given here:

```
on>batchred
```

A batch file already exists -