



MRU Cardington Technical Note No. 15

A User Guide for Cardington Processing Software

by

shd, wph, mjb et Al

Last revised 10th July 1997

ORGS UKMO M

National Meteorological Library

FitzRoy Road, Exeter, Devon. EX1 3PB

Met. Office Research Unit
RAF Cardington
Shortstown
Beds, MK42 0TH

MRU CARDINGTON

Note

This paper has not been published and PMetO(Cardington) should be consulted before quoting from it.



MRU Cardington Technical Note No. 15

A User Guide for Cardington Processing Software

by

shd, wph, mjb et Al

Last revised 10th July 1997

Met. Office Research Unit
RAF Cardington
Shortstown
Beds, MK42 0TH

MRU CARDINGTON

Note

This paper has not been published and PMetO(Cardington) should be consulted before quoting from it.



MRU Cardington Technical Note No. 15

A User Guide for Cardington Processing Software

by

shd, wph, mjb, et Al

Last revised 10th July 1997

Met. Office Research Unit
RAF Cardington
Shortstown
Beds, MK42 0TH

Note

This paper has not been published and PMetO(Cardington) should be consulted before quoting from it.

A User Guide for Cardington Processing Software

shd, wph, mjb et Al

Contents

1	Introduction: the Multiple Turbulence Probe Processing Suite	3
2	Logbooks	4
3	Programs	4
4	Data structures, logging and display	5
5	Checks and procedures for processing	6
6	Logging and real-time display routines	10
7	Graphical output	11
8	Some VMS tips	12
9	Appendix 1: control format for probe-processing	13
9.1	Control variable	13
9.2	Further control lines	14
9.3	Probe lines	15
10	Appendix 2: Reference for probe-processing routines	16
11	Appendix 3: Calibration datasets	21
12	Appendix 4: Channel numbers	22
13	Appendix 4: Tapes	22
14	Appendix 5: Instrumental specification	23
15	Appendix 6: Scientific approaches and data analysis	23
16	Appendix 7: inertial subranges	32
17	Appendix 8: standard plots	34

18 Recent additions	34
18.1 INCORP	34
18.2 VAXMDIF_NEW	35
19 Programming standards	36

A User Guide for Cardington Processing Software

shd, wph, mjb et Al

Last revised 10th July 1997

Abstract

This note is intended as a guide to processing software, with references to further sources of information, hints for getting started and reminders of the checks which should be made when processing. This file, `$disk1:[dproc.tex]processing_guide.tex`, incorporates and supersedes earlier documentation in `[dproc.process]tm.txt` written by AJL around 1983. It also draws on the helpc library, which contains further information (printable versions can be found on `$disk0:[sysman.text]`). This guide can be accessed online by command `$EGD`.

Finally some methods of data analysis are outlined.

PLEASE UPDATE THIS FILE AND AMEND 'LAST REVISED' DATE

1 Introduction: the Multiple Turbulence Probe Processing Suite

The balloon-probe system, with attendant software, is the main unique facility at MRU Cardington. The use of balloon cables to mount turbulence instruments dates back to the 1950s at Cardington. Caughey (1977) gives a summary of the instrumentation and of the history. References in the open literature include Caughey and Wyngaard (1979), Caughey and Palmer (1979), Moores *et al.* (1979), Rayment and Readings (1974). Balloon measurements have advantages over the alternatives (tower or aircraft measurements) in minimizing flow distortion and in deployability. Aircraft are sometimes better for getting area-averages, but aircraft speed requires fast sensor response to resolve a given length-scale. Haugen *et al.* (1975) give a comparison between balloon and tower measurements. The catch of course is balloon and therefore cable motion, which makes horizontal wind components particularly unreliable unless explicit corrections are made.

An algorithm that corrects for cable motion was developed and tested by Lapworth and Mason (1988). This algorithm uses probe orientations as measured by inclinometers and magnetometers to reconstruct cable positions using a piecewise quadratic curve fit. (A time-filter is applied to the inclinometer data because at high frequency these are contaminated by accelerations, but the exact value of the filter period, typically 15 s, is not critical.) These positions are then time-differentiated to obtain cable motion which is then added to the measured wind to obtain the true wind. See CTN-12 by Alan Lapworth for further information. Note that some problems with the calculation in certain orientations have come to light since that note.

A specification for accuracy, copied from Lapworth and Mason, is given in Appendix 5. An error analysis by Alan Grant of the orientation algorithm is given in CTN-14; see also CTN-13 for a probe-sonic comparison of turbulence statistics.

Program	Input	Output	Remarks
TMST	DM*.TRB/.PRF	DZ*.OST	TRB level, PRF profile
TMCL	DZ*.OST	DZ*.OCL	
TMFO	.OCL	.OFO	
TMPW	.OFO	.OTM	
TMFD	.OTM	.OFD	
TMVG	.OFD	DM*.STA/SPC also .OTS	

Table 1: A standard processing sequence. Here * means ddmmmyy where ddmmyy is the date and a the sequence-letter for that dataset on the day in question. The file type .FOR is omitted here for brevity. Z denotes the probe number in hex (1, 2,..., 9, A, B,...,G).

2 Logbooks

Useful processing of data depends on info gathered at the time. Things like cloud observations and real-time display can make all the difference, so make notes of anything conceivably relevant.

However one should also note the difference between real-time checks and properly processed data. Also, beware that the units of some real-time displays are feet and knots.

3 Programs

Command \$CHKVS can be used to locate programs and their versions across all three nodes CABLE, STABLE, ABLE — useful also for maintenance.

The main functions of probe-processing are as follows:

- (i) Cut up long logged datasets into more manageable lengths, removing bad sections or alternatively merges datasets logged sequentially (TMOD). Often not needed.
- (ii) Sort out probes into separate datasets (TMST); the datasets remain separated until the final stage.
- (iii) Calibrate measurements (TMCL) and select sensors (slow/fast etc.)
- (iv) Time-filter the measured orientations (vector attitudes) and heights probe-by-probe (TMFO)
- (v) Use measured orientations from all probes to reconstruct cable position at each instant (TMPW — previously TMTM)
- (vi) Time-filter and differentiate cable position, hence reconstruct true wind components in NSEW axes (TMFD)
- (vii) Rotate to local mean-wind coordinates and construct datasets for graphical output (TMVG; TMAC; TMTS)

The inputs and outputs of the various TM programs are listed in Tables 1-3. These programs can be found on \$DISK1:[DPROC.PROCESS]. If you want to customize them to personal requirements, then take your own copies and modify, but in the long term it's undesirable (e.g. for maintenance) to have too many personal copies — better to talk to other users and agree.

The VMS SEARCH command can be used for an overview of program structure, e.g.:

```
$ SEARCH TMCL.FOR call,subroutine
```

```
$ SEARCH/OUT=TEMP.TMP TMCL.FOR call,subroutine
```


Program	Input	Output	Remarks
TMPROC	.OCL or subsequent		General plots
TMPRNT	.OST	.DAT	
TMPRNT_VAXG	.OST	" + .STA	an AG extension
TMTS	.OFD	.OTS	
TMAC	.OFD	.OAC	REGIS/LASER output
TMDT+TMVG	.OFD	.STA/.SPC	
PROBE_PLOTS	.OFD	.SIX	previously .PLT

Table 2: Alternative processing routines. TMDT and TMVG enable you to reconstruct stats and spectra from the OFD file and thus avoid complete reprocessing. TMDT recalculates the trends required by TMVG in TMCMD.DAT files. PROBE_PLOTS plots time-series

Program	Input	Output	Remarks
TMTM	.OFO	.OTM	superseded by TMPW
TMFR	?	?	filters selected input channels to required period
TMFQ	FOR005.DAT	FOR005.DAT	avges/interps data to fit req'd freq'cy

Table 3: Some oddments no longer used

4 Data structures, logging and display

For data-logging and display routines, see STABLE::[DPROC.LOGGING]. Note in particular PROBE_LOG.FOR.

During logging data are passed through the global common block /PROBE/ which enables multiple programs (e.g. display routines) to access the data. The main data is passed to display routines by the transfer array TRANS, currently dimensioned INTEGER*2 (17,16). In TRANS the first index is the channel number, the second the probe number. TRANS(17,*) also codes probe ID and averaging rate. At this stage the probe lines are not closely synchronized — (partial) synchronization is implemented during processing in TMST (?). Subroutine BAV of PROBE_LOG.FOR sorts and averages values. Program LOG_INPUT_DISPLAY.FOR can be used to give a hex display of numbers coming in.

Structure of logged dataset (usually DM*.TRB or .PRF):

```

HEADER
NRAV,END_TIME,IM,ID,IY,ITIM,IBLNK
TSEC,A,ERR (repeated for each line of data)

```

where A contains the block-averaged data in similar INTEGER*2 (17,16) format. TSEC is time from start in seconds and ERR the error count.

.OST datasets resemble the parent datasets but are sorted by probe. NB in TMST the array IN (=previous A), similarly dimensioned, codes probe ID and rate switch in IN(1,*). This means channel numbers in TMST or TMCL do not correspond to those in display routine DSP. Basically channel n in DSP corresponds to channel n in OCL and later datasets but to IN/A(n+1,*) in logged dataset and OST. This is currently rectified by TMCL (post OST) having a channel list displaced by 1 from that in DSP.

In general it is desirable to avoid changes of meaning in programming. The new routine TMSF (replaces TMCL, for surface site) preserves the DSP channel list by dimensioning TRANS as (0:NCHANSP,LFRAMEP), where NCHANSP=no. channels and LFRAMEP length of the

frame (here 16). (Actually TMSF also gives TRANS a third probe dimension enabling all probes to be stored at once, for other reasons).

Calibrated datasets (D*.OCL or later) contain for that dataset's probe:

```
FXD,NTITLE,DT
NRAV,TF,IM,ID,IY,LTIM,IJUNK,(TITLE(I),I=1,NTITLE)
(RECORD(I),I=1,NTITLE)      ! repeat for each line of data
```

where FXD is the (64-char) header, NTITLE the number of titles (and variables) passed to next stage of processing. NB RECORD is an array of REAL*4 data representing calibrated readings. TITLE is an array of 4-character strings specifying the meaning of each channel. Some programs, e.g. TMPROC, allow referencing of channels by their title. (SHD thinks this a good feature, which could perhaps be extended.)

The display programs include DSP, RTD, GRAPHICAL_RTD. DSP uses cunning 'QIO' calls to address the screen display, and if anyone is prepared to write a few lines on these here, then go ahead! Alan Grant wrote some notes on QIO calls for logging.

5 Checks and procedures for processing

You will need three things:

- (i) Datasets - Firstly. profile up to height e.g. DM161089A.PRF. Secondly, a level turbulence run e.g. DM161089B.TRB.
- (ii) Command files containing the processing programs to act upon each of the above datasets, see e.g. [hopwood.command]probe_profile_ph.com for profiles or [hopwood.command]probe_process_ph.com for level runs.
- (iii) Control files that contains information about the flight and how the programs within the command file should act upon the data, e.g. [hopwood.datq]DM161089.CMD which is equivalent to a TMCMD.DAT file.

In your command file you may also wish to define the logical name CORRECTIONS to be a file containing user-corrections applied 'by hand' in TMCL over and above the calibrations. A typical simple format would be:

```
PROBE 6 SPRE OFFSET=0. SCALE=1.
```

where $\text{corrected} = \text{SCALE} \times \text{uncorrected} + \text{OFFSET}$, and the channel is in this case SPRE. Of course the above choice of numbers is a 'null correction'! See \$helpc for more details (under Datasets), with more optional parameters. NB by default the corrections are applied only to the basic measured parameters, but not to those derived from them e.g. using thermodynamic formulae. It would of course be highly undesirable to apply user-corrections to make results 'look nicer' without discussing the calibrations with MJB *et al.*

The format for the control file is given in Appendix 1.

Prior to processing the datasets can be examined using [DPROC.PROCESS]DRSREAD which gives an output to the screen of the raw channels. DRSREAD can also be used on the .OST files. They can also be examined using [DPROC.PROCESS]RVW which gives a DSP-type display using the logged dataset. After TMCL has been run in the processing suite, the datasets can be inspected using [DPROC.PROCESS]TMRD, which gives a calibrated output to the screen of selected important variables from the .OCL dataset, but not individual probe channels. It is convenient to define a symbol

TMRD:=RUN [DPROC.PROCESS]TMRD, and similarly for DRSREAD. It is also convenient to define a logical name (e.g. IN) for the input dataset name.

DRSREAD and TMRD are short programs and help to give insight into dataset structure. The basic logged datasets (.PRF or .TRB) start with a character-header line (as typed into the LR file before logging), then a general info line (date etc.) then a sequence of lines of the form TSEC,IN where the integer*2 array IN(17,16) contains 16 points with 17 channels. The first point IN(1,*) is $16 \times \text{ID} + \text{rate division}$. (So you can get probe number by an integer divide by 16). The following points are: pressure, Gills (top, port, starboard), inclinometer (pitch, roll), magnetometers (Y, Z, X), PRT, TD, TW/HCP, spare, TW/IRD, ref, battery. DRSREAD may appear to have been written for trainspotters, but understanding the format will help considerably in isolating any probe problems. A new program [DPROC.PROCESS]PNSEARCH (courtesy A.Lapworth, Jan 1995) can be run like DRSREAD on the basic datasets and will identify which probes are present and for how much of the time etc.

The program [DPROC.PROCESS]RVW can be run to review a .TRB or .PRF dataset (i.e. essentially rerun the DISP display) — the program was mod'ed recently to catch up with several years' processing changes. The program [DPROC.PROCESS]RVW_GRAPH (see also [DPROC.COMMAND]RVW_GRAPH.COM) can rerun PLOT in the same way.

A suggested processing procedure is as follows:

- (i) *Find probe heights* For correct processing of a level run the heights of the probes are an essential requirement. To get these we need to process the initial profile or .PRF run using TMST followed by TMPRNT. The control TMCMD.DAT file should contain the probe nos., initial height (often pole height $\simeq 3\text{m}$) and an averaging time ($\simeq 25\text{s}$ is ideal to give an output dataset that is not too big). The pressures and their corresponding heights, plus other variables, are output to the FOR005.DAT or FOR006.DAT files and the pressures, heights, etc. should be checked roughly against those noted down in the experimental log book. The probe heights found in this way are a rough estimate and can be used for further processing of the level runs. This is provided the initial conditions at the start of the profile are correct and there is little or no pressure tendency throughout the rest of the flight. To eliminate the above drawbacks you should use the heights and pressures output from the profile run to calculate pressure-height relationships for the probes. Then the heights can be deduced from the absolute pressure at the probes and the regular surface observations of pressure.
- (ii) *Check calibrations* by running TMPROC as follows:
\$RUN \$DISK1:[DPROC.PROCESS]TMPROC
REGIS OUT=REG.PLT
D8101091A.OCL
HT,PRT
GRAPH X=PRT Y=HT SYMBOL=1
XSCALE=12. 17. YSCALE=0. 600. AV=100.
END OPEN
READ
D2101091A.OCL
HT,PRT
GRAPH X=PRT Y=HT SYMBOL=2
XSCALE=12. 17. YSCALE=0. 600. AV=100.
END OPEN

STOP

This example plots profiles from probes 8 and 2, and can obviously be extended. The variables are referenced via the parameter TITLE strings, which can be found using TMRD. The plot with AV=100. contains a lot of information, and the clutter may hinder comparison in certain cases. Intermediate (i.e. 'symmetrical' up and down) profile runs are significantly better for recalibrations than initial or final runs because systematic changes with time can be largely removed. With an intermediate symmetric run, a useful alternative option is to take a large value of AV, equal to just under half the run length (in seconds). The probe plots are close to horizontal, around the average height. The midpoints should form a smooth curve and corrections can be readily. Define a corrections file and preprocess the level runs, checking that they improve both plots. For directions it is best to look at output from TMFO (which one normally doesn't run for profile runs). It's not possible to correct for cable motion, but for mean directions over a profile of over half an hour that's probably not critical. Significant direction errors (more than say 2°) should be reported to MJB — processing directions is much more complicated than processing temperatures and ad-hoc recalibration is dodgy.

TMPROC can also produce time-series using command PLOT.

- (iii) *Examine balloon motion spectra* After calculating the probe heights, we need to deduce a filter period that cleans out the balloon motions from the data. The method is to choose one probe approximately in the middle of the cable (say at around 300m), process the .TRB run with the whole processing suite for different filter periods (e.g. 5, 10, 15, 20, 25s). See [hopwood.command]probe_filtper_process.com which uses a set of [hopwood.datq]fp05sec.cmd, etc., control files. For each filter period we need to examine the balloon motion spectra and inclinometer cospectra. The balloon motions, VN and VW (N'y and W'y components) are always output into the .OFD files, but to get at the inclinometers we need to set the control variable equal to 2 in the TMCMD.DAT file so that IP and IP will be output to the T and Q channels of the .OFD file. To distinguish .OFD files of differing filter periods each is renamed by the convention DM161089B.F05, .F10, etc.. These can then be converted into spectra by running [DPROC.PROCESS]TMPROC_SPECTRA, see [hopwood.command]tmproc_spectra_filt.com. The output is a vaxgraph compatible .SPC file. Once the spectra have been plotted what we are looking for is the large peak at the high frequency end of the spectrum, due to the balloon motion. The best filter period is the one that reduces the size or spectral intensity of this peak to about 1/2 or less that of the turbulent peak

- (iv) *Full processing* using e.g. [hopwood.command]probe_process_ph.com with TMCMD.DAT using appropriate filter period and probe final heights for each probe.

- (v) *Examine time-series* with program PROBE.PLOTS. This may in fact be invoked from the previous file, but is described here separately for convenience. Example: \$SET DEF \$DISK2:[DERBYSHIRE.DATA]
\$ASSIGN D2080393E.OFD INPUT
\$RUN \$DISK1:[DPROC.PROCESS]PROBE.PLOTS

This will give you plots D2080393E1.SIX, D2080393E2.SIX, representing different pages of time-series (basic variables and fluxes). If you process a dataset with N segments you get N versions of both files. The plots are autoscaled. It should be noted that the flux plots have a hi-pass filter applied to them. Running PROBE.PLOTS is essential if you have a sensor malfunction or reception problems, and highly desirable anyway.

Remember the physical nature of the individual probes, e.g. 13 is not a normal probe [no Gills or PRT etc; but it does have slow dry TD, and IRT in the PRT channel]. Although these are taken into account automatically by the processing routines, you could still find yourself

doing silly things if you forget.

Read through the diagnostic printouts from the processing routines (DM*.DAT;*), which can help detect and correct all sorts of errors or problems. For instance if a data-block has a bad reference or TQ value then that block may be ignored and backed up instead by the previous block. If this happens a lot you'll want to know about it!

6 Logging and real-time display routines

Normally log on as DPROC.

\$COM : command-file subdirectory

\$LP : [dproc.logop] containing datasets etc

Many of the commands are best invoked via a MENU (type \$M).

Probes

\$LR (for logging ready) to edit log.com file

\$GO submits this

\$DGO allows display but doesn't log

\$DSP displays the numbers (check these!), calibrated in real time

(note the LIM variable on LHS: if > 0 indicates no. of sensors out of range — not working before 6/5/93!)

\$RTD real-time display (counts) — look out for ± 2047 (sensor not working)

\$GR ('graph ready') prepares for \$PLOT

\$PLOT plots pictures by invoking GRAPHICAL_RTD

\$LOGSTOP on ABLE stops probe logging (e.g. for profile run)

\$SETP edits file pt.dat to set surface pressure and temp for display routines height calculation

\$DRSREAD reads existing raw data (.TRB/.PRF/.OST) as hex and counts

See [dproc]login.com.

Building display routines:

\$FORN/NOCHECK prog

\$LINK/NOTRACE prog,GBL/OPTION

Installing logging routines:

for PROBE.LOG.FOR, use \$INSGO command.

for surface logging use \$INS.SURFACE_AUTOLOG.COM.

These require privileges

Surface

\$RHCALC invokes electronic humidity slide-rule

Sondes

\$SONDE logs sonde (wait for READY TO LAUNCH message!!!!) and creates e.g. RS*1.SND where * denotes date and l the id letter (A, B, C ...)

\$RSD displays sonde during flight (via program RS PLOT) and creates RS*1.PLT

To copy e.g. rs230991a.snd over to ABLE (from ABLE):

\$BAT COPY STABLE::DUA0:[DPROC.LOGOP]RS230991A.SND [DPROC.LOGOP]

(actually BAT, i.e. batch, isn't necessary for short files like sondes but more so for probe datasets).

There is also a program SONDE_PROC (due to Mr Hopwood) to produce vaxgraph-plottable files from .SND files. SHD has a version SONDE_PROC1 for the summer 1990 datasets which were taken using a slightly different system (Marwin) while the Digicora wasn't working. Consult B.Claxton, S.Derbyshire, P.Hopwood.

7 Graphical output

The normal means of obtaining graphical output for statistics is by using the Vaxgraph package, written by Alan Grant. This versatile program (invoked by \$vaxg) effectively separates the problems of (i) data storage and (ii) calculation/plotting. It does this by accepting data, e.g. turbulence statistics or spectra, in a labelled formatted file (often called *.STA or *.SPC) and then performing calculations as directed by a control file (usually *.ctl) with output to a REGIS or Tektronix file (usually *.plt).

To do this a cunning interpreter is incorporated which can evaluate standard FORTRAN expressions plus a few extras (e.g. standard boundary layer ϕ_m , ϕ_h functions). Before calculation a temporary direct-access dataset is constructed (*.DAC) from the input data, and this can be kept, speeding up any further plotting from the same dataset. NB .DAC files tend to be large and should be purged routinely and deleted when not used.

This format facilitates plotting of different data on the same graph, and has many nice features including optional logarithmic scales, 'binning' of data, curve-fitting or generation by hand, boundary-layer empirical functions and data-restriction (e.g. for quality-control). The program is fairly well documented on the CABLE help library (\$helpc oth vaxg). You are recommended to look at the SET FRAME options, especially if producing graphs for publication (XANN=INTEGER often looks tidier). Use CHANGE command to plot a different variable on the same graph (you need to fetch data again). Some options are not yet documented on helpc, e.g. NOTI on an axis-label line suppresses the title. A fairly recent addition is that device can be changed if you open a new plt file, e.g.:

```
NEWPLOT NEWFILE=LAS.PLT TEKTRONIX
```

Some useful formats:

```
GENE X=Y**2 FROM:0.0 TO:1.0 INCR=0.1
```

```
PLOT CURVE=DOTTED
```

generates and plots a curve.

```
SET RESTRICT:X=(0.0,1.0) or
```

```
SET RESTRICT:<A=HT>=(0.0,250.0)
```

before FETCHES allows you to select or quality-control data. This option is a bit fussy about formats — don't leave spaces except between SET and RESTRICT. Also I think the numbers have to be floating-point. Note the difference in format between specifying X or Y and specifying named 'physical' variables. Restriction can now be applied to 5 specified variables (A-E) as well as X and Y.

To bin data,

```
SET BIN=X MIN=-0.05 WIDTH=0.1 MEAN
```

where the optional keyword MEAN gives error bars for the standard error of bin mean. If binning is used, isolated points are now bracketed on the plot, unless NOBRACK keyword appears on the SET BIN line.

Curve fitting:

```
FIT ORDER=3 DISPLAY
```

```
GENE LSQFIT FROM:-1.5 TO:1.0 INCR=0.1
```

```
PLOT CURVE=DASHED
```

If unspecified the order defaults to 1, i.e. linear fit. DISPLAY prints out coefficients.

To integrate Y over X, example:

```
INTEGRATE/RESET WRT:X SUM FROM:0.02 TO:0.1 SCALE=2.3
```

The command RESET is needed to set the running total to zero, rather than adding to what you had before. Note that this is NOT a true integral but rather $\sum Y\Delta(\text{SCALE} \times X)$. See

comments in Vaxgraph program. For a proper integral replace SUM by SIMP (for Simpson's rule integration) and don't use SCALE. I haven't properly tested this so you'd be wise to check this option against simple known results.

High graphical quality for output can be achieved by producing a REGIS plot, called e.g. REG.PLT, and then \$retos/reverse.video/device=ln03_plus REG.PLT produces the sixel file REG.SIX. Furthermore, if you doctor the sixel file by chopping off the final escape sequence and preferably also any trailing blanks, which are hyphens (-) in sixel, then you can incorporate the graph into a latex document, as shown in \$disk1:[dproc.tex]figure.tex. This enables you e.g. to put Latex symbols on a graph. If you forget to chop off the escape sequence it will spatter Latex over the page. Changes have now been made so that in REGIS mode dotted and dashed lines are drawn explicitly as segments which convert into sixel format automatically (whereas previously all inks converted into solid lines).

WPH has developed a way of producing axes along the 'opposite side' of the frame, i.e. top or right for X or Y respectively. This is invoked by adding the word SEC on the line defining axis type (LOG or LIN), normally in conjunction with NEWPLOT/NOFRAME (i.e. defining variables all over again, even if they're the same ones as for the left/bottom axes).

The main graphics routines are held in SYSLIB in the following modules: CABLEPLOT, GRAPHSUBS. VAXGRAPH is held on SYSUTIL.

8 Some VMS tips

There is some basic info in written documentation, albeit not very up-to-date. Quite a lot of the BLISS documentation applies to CABLE too.

Note that the commands \$EDIT/TPU and \$RUN can be abbreviated \$E and \$R respectively. Use \$E/READ as a readonly option when you don't want to change the file.

When editing, get to know the buffer commands. Change buffer with command [Do]BUF*FER; [Do]GET filename enables you to edit multiple files. On Vaxstations you can change buffer by clicking MB1 on the buffer line.

Command \$SEARCH is useful in program maintenance, searching through a list of files (e.g. specified using wildcard *) for a specified string.

Current workstations allow text display from one window to be captured by a file (FILE), even if the text format is different (e.g. COSMOS terminal emulator). Click and drag the left-hand mouse-button (MB1) to select text. Then click MB1 to identify the window and location where you want to copy. When ready, click MB2 to copy. Don't be afraid to copy more than you need — you can easily edit the file afterwards. This is quite useful in connection with the Notepad application.

Use of exclamation-mark comments as notes to yourself can be useful even in interactive sessions, especially if you're taking a break or may be interrupted. E.g.
CABLE> ! check TMVG correctly handling new sensors

Use command LINKT (defined in DPROC's login file) to place a test version of an image (EXE) file on directory TEST:, and thus avoid conflict with 'operational versions' until you are ready to accept the test version.

9 Appendix 1: control format for probe-processing

The control file TMCMD.DAT has the format:

Primary input dataset name (DSN; e.g. DM101091B.TRB)

Logging frequency in Hz (FREQ; e.g. 4.125)

Period in minutes (TSEG; e.g. 67.)

Number of segments (SEG; e.g. 1. or 2.)

Control variable(s) (see below; real format!)

Further control lines (2)

first probe lines (3)

second probe lines (3)

etc.

NB all numbers must be entered in REAL format! During processing means and trends (the latter in peculiar units) are written back out to end of TMCMD.DAT, and should be deleted subsequently.

9.1 Control variable

The (REAL!) control variable is defined as follows:

0. is standard
1. gives input arrays in TMCL
2. gives balloon motions VN and VW in T and Q
3. gives inclinometers ROLL and PITCH in T and Q
4. replaces PRT with TD in T and Q
5. replaces PRT or TD (whichever is selected by the further control lines described in the next section) with potential temp (calculated using smoothed pressure in T
6. removes balloon motion corrections (useful e.g. for mast probes)
7. gives filtered orientation functions
8. replaces TWF with TW in Q
9. gives magnetometers in T and Q
10. deactivates the Gill response correction
11. slow dry for humidity
12. adjusts PRT with constant offset so mean agrees with TD
14. uses inclinometer to help fix 'southern sector' magnetometer bug
15. chooses 'opposite branch' to standard, so correct only in S-sector
16. replaces PRT or TD (whichever is selected by the further control lines described in the next section) with equivalent potential temp (calculated using smoothed pressure) in T. Overridden by C.V. 5
17. replaces Q with relative humidity. Overridden by C.V.16 and other replacements.
18. replaces Q with dew point PRT with TD, and Spare with pressure. It also extends the run processed to the end by repeating the last block - useful for BALTHUM type profiles which can then be further processed with TMBT after the TMFD stage.
19. replaces SPare channel (SPA) with Pressure.
20. applies Alan L's latest inclinometer/accelerometer corrections to horizontal components
21. as for 20., plus corrections to *w*
22. applies Gill response correction implemented July 1996. *See CTN no 34 on second order Gill response correction by B.Claxton*
23. creates JDP output list in .OFD files. This list (as of Oct 1996) is:
U,V,W,PRT,RH,TD,TM,HT,P,VW,VZ,SPA,SPB,SPC.

Not compatible with any other output-list options.

Note that option 12. is actually implemented in TMFO from means calculated in TMCL. If using options 20.,21. check on the hardware position re inclinometer electronic filter. This was overdamped for datasets up to around 1993 and so the correction will not be effective for the higher frequencies for older datasets. After 1993, all probes had an increased cutoff frequency on the hi-pass electronic filter.

NB software changes were made to subroutine GILL of TMCL Jan. 1996 after discovering that the response-correction could go wild and cause spikes if the wind on any propeller fell very light.

Several different control variables can be used, in which case they must all be entered on the same line and separated by commas. MJB uses DISC,IC on this card when using Tmprnt for intercomparison flights.

9.2 Further control lines

The next two cards are used to input further processing controls and to define the final disposition of the datasets. A 'card' is a line, for pre-historical reasons — when programmers were men of cardboard (WPH/MJB).

- (i) The first card is used to control the TMCL stage as follows:

The keyword POLYPROPYLENE on this line will result in length constants for polypropylene Gills (3 metres) being used for all the probes. If this keyword is followed by numbers in brackets e.g. POLYPROPYLENE (7,8,9,10) then only those probes given in brackets will have the polpropylene length constants. The default if the keyword is missing is POLYSTYRENE.

A second optional keyword gives the rate of surface pressure change e.g. DPDT=-2.0 where the figure is in mb/hour. If this is used, the heights will be adjusted accordingly in a profile run (what about a level run??). Note that 1mb corresponds to about 8m in height — not a negligible error, especially low down. It also corresponds to about 0.1K on potential temperature.

The keywords TD,TW,PRT,TWF can be used to override the default wet and dry sensors set by control variables 4 and 8. (Note added 3/11/95 This does not appear to be true as far as output goes - i.e. TD does not equate to control variable 4) TD refers to the slow dry thermistor, TW to the slow wet thermistor, PRT to the platinum resistance thermometer and TWF to the fast wet thermistor. These keywords can be used in conjunction with integers in brackets after the keyword to override the default sensors for particular probes as with the POLYPROPYLENE keyword. E.g., if the control variable 4 is not set and the keyword TD (2,3) is used the platinum resistance thermometer is used to compute the temperature variable for all probes except probes 2 and 3 which use the slow dry thermistors.

Other keywords: 'PAB=' allows an outside pressure reading in mb (that's hPa to the more politically correct) to be entered, particularly useful for surface instrument and probe 13 processing. 'DESPIKE' applies an additional broad-band, two-tier ($3\sigma+2\sigma$) despiking routine to the T and Q channels. 'DESPIKE_ALL' applies it to all channels (in case of dire need). Of course it's important not to be complacent about despiking and noise — really noisy data should be rejected altogether. Despiking is a legitimate procedure only for isolated (and usually gross) errors. Normal processing involves a 2-point despiker automatically, but the routine invoked by DESPIKE is an extra despiker for noisy channels.

Further keywords used on this line can be used to control TMAC, and are described with this program below.

- (ii) The second card determines the disposition of the datasets at the end of each stage of the processing. The form is xxx=KEEP or yyy=DEL, where xxx and yyy are one of TRB, PRF, OST, OCL, OFO, OTM, OFD, OTS, SPC. If a dataset is not included in the list the default is KEEP.

9.3 Probe lines

The following three lines are required for each probe:

Probe number (real format!)

Height

Period of filter (half 3Db period)

Various keywords can be entered on the probe number line to control the processing of individual probes - this is in most cases an alternative to using the first control line as described above. The keywords are as follows:

- (i) HT(PRT) - height is calculated using the PRT rather than TD
- (ii) Q(PRT,TWF) - Q from wet bulbs is calculated using PRT and TWF rather than TD and TW (N.B. any combination from TD, PRT and TW, TWF can be chosen)
- (iii) THTA(PRT) - THTA is calculated using PRT rather than TD
- (iv) QH(TD) - Q from humicaps is calculated using TD rather than PRT
- (v) POLYPR - polpropylene propellers
- (vi) DESPIKE - broad band despiking on T and Q channels
- (vii) DESPIKE.ALL - broad band despiking on all channels
- (viii) LYMAN - lyman alpha on spare channel
- (ix) RADIO - radiometer on spare channel
- (x) HUMICAP - humicap on spare channel
- (xi) WATER - ruskin meter on spare channel
- (xii) NOCONTOUR - suppresses data from being used for TMAC contours
- (xiii) NOARROW - suppresses data from being used for TMAC arrows
- (xiv) SPB - will enable the spare B channel (old TWF) even if it is not a special instrument

10 Appendix 2: Reference for probe-processing routines

The individual programs in the processing suite will now be described.

TMST

This routine takes in the logged output, sorts it into buffers according to probe number and outputs it into individual datasets. It notes gaps in timing and fills them with repeated blocks. This also happens if the reference falls outside limits. Error messages due to such data problems are printed. In addition, where individual probes have been switched on after the start of the dataset, the individual probe output datasets are backfilled so that they start from the same initial time.

TMFQ

This routine averages or interpolates by linear interpolation to meet a given frequency specification. Hardly ever used

TMCL

This routine calibrates data (sr GET_CALIB) using the calibration dataset (Appendix 3), determines UP, VP, WP relative to probe using cable axis (GILL, GILCOR), calculates time, thermodynamic variables and height (CALC), calculates diagnostics (CHECK) and then determines cable angle functions for filtering (CABLE; GROUND). It applies the Gill frequency response correction. The improved version 'date-traps' all important instrument changes and probes 15 and 16 (WPH). The following variable are output: UP, VP, WP, T, Q, THV, THTA, HT, TM, PN, PE, FMX, FMY, FMZ, SP. PE means 'position East' and PN 'position North'.

TMPRNT

This routine is used after TMST instead of TMCL and gives additional parameters and averaged output listings plus intercomparisons.

TMFO

This routine filters the variables FN, FE, Z and calculates U, V, W relative to the moving probe oriented along North, West, Up. It outputs the variables U, V, W, T, Q, THV, THTA, Z, TM, DXDZ, DYDZ, POSZ. AJL will be happy to explain the rationale for this slightly strange but admittedly right-handed co-ordinate system.

TMPW or TMTM

These routines takes in datasets from all probes simultaneously. It uses DXDZ, DYDZ, POSZ to determine the cable gradient and azimuth at a given height and the probe position POSN, POSW, POSZ is calculated and output after U, V, W, T, Q, THV, THTA, HT, TM, POSN, POSW, POSZ.

TMPW works using a piecewise quadratic fit to the cable and is now the recommended routine to use. TMTM uses a polynomial fit to the cable and has been found to be ill-conditioned

for more than 3 or 4 probes. The degree of the polynomial used is one less than the number of probes. Other routines such as TMLS have been tried which use least squares fits to low order polynomials but they are more complex than TMPW and produce results that are no better.

TMFD

This routine differentiates the probe positions POSN, POSW, POSZ (with a small phase lag) to determine probe velocities, which are filtered using a similar filter to that in TMFO, and added to U, V, W. The mean and trends of these final U, V, W, Q, T, THTA values are determined (AVDT) and output on the end of TMCMD.DAT. together with probe identity. The output is U, V, W, T, Q, THTA, HT, TM, VNN, VWN, VZN where the later variables are balloon motions. Datasets created before November 1988 do not have height information, but subsequent processing routines have been modified to work on both types of OFD output. The new version copes with surface instruments on probes 15 and 16.

TMTS

This routine removes trends (DT) using means from the previous program, and rotates the data into the frame of reference of the mean wind (ROTATE). The turbulence statistics are determined (AV) and the program SPEC is called for each segment, and finally, to calculate the spectra. The output dataset contains U, V, W, T, Q, HT, TM.

TMVG

This is an alternative to TMTS and outputs data in VAXGRAPH format. Spectra are output in a .SPC dataset and statistics in a .STA dataset. The names of statistics should be self-explanatory. The spectra are converted to wavenumber using G.I.Taylor's frozen-field approximation (i.e. fields just advected at speed U). Note that K denotes cyclic wavenumber, i.e. cycles/m, *not* radians/m (as per standard fluid-dynamics wavenumber k). The spectra SUU etc. refer to $nS(n)$ spectra, the usual convention for meteorological spectra. They should vary as $k^{-2/3}$ in a Kolmogorov inertial subrange. This means that integrated over $\ln K$ (or indeed $\ln k$ or $\ln n$) the spectra should come to the total variance. In fact the averaging is rather cunning (courtesy of John King) so that to integrate the spectra just add up the numbers and multiply by $\ln 10/10$ (because the bandwidth is a tenth of a decade). The last point on the spectra is often low for reasons connected with the banding (not necessarily anything to do with instrument response).

Most of those pearls of wisdom came courtesy of Alan Grant. See Appendix 7 for more detail.

TMAC

This routine uses the .OFD (or optionally .OTS, see INDS keyword below) output files to produce a contour plot of potential temperatures for the whole (or part) of a run using all (or some) of the probes. In addition arrows are plotted at each data point with components (U-Uav) and W where U is the alongwind horizontal component of wind, Uav is the mean value of U for ALL probes over the WHOLE run and W is the updraught. U and W are measured relative to the mean wind for the particular probe for which they are given. The output file is DM*.OAC. If you use .OFD as input you may want to use the keyword THTA described below.

By default a plot is produced using all the probes for the whole run, and preset values are

used for the scaling and other parameters. However the plot can be magnified and the scales etc altered by using control keywords in the sixth line of TMCMD.DAT, which is the line used to assign datasets to different discs. The keywords must all be used on this line. The keyword is associated with the first number found on the line after the keyword. It is only necessary to enter the keywords of parameters which are to be altered from their defaults. The keywords are as follows: (notation NOAR*ROW means can abbreviate to part before *).

- HMIN Minimum height required (in metres) - default the minimum height taken over the whole run.
- HMAX Maximum height required (in metres) - default the maximum height taken over the whole run.
- TMIN Minimum time required (in seconds after the start of the the run)- default 0. seconds
- TMAX Maximum time required (in seconds after the start of the the run)- default the run time up to the end of the .OFD datasets
- DELT Time over which each datapoint is averaged - default (TMAX-TMIN)/100.
- USCL Factor used to multiply the horizontal velocity scale of 1 cm per metre/sec - default 1.
- WSCL Factor used to multiply the vertical velocity scale of 1 cm per metre/sec - default 1.
- CSPC Contour spacing - there are sensible defaults for the different variables if this is not selected
- CONT The contours can be of U, V, W, T, or Q (specified by e.g. CONT=U). Negative values are dotted. Default is T. Since 1996, CONT values can also be UW, WT, UT, WQ, UQ, UU, QQ, TT, EPS (energy dissipation), SP (energy production), BP (buoyancy production), RG and RF (gradient and flux Richardson numbers)
- LCONT Same as CONT but a line is drawn at colour boundaries - this usually looks better unless the data is noisy.
- LINE This is essentially the same as CONT but produces a line contour only. In gives the possibility of doing line contours as well as colour contours
- CSPL Contour spacing for LINE - there are sensible defaults for the different variables if this is not selected
- THTA The horizontal component for the arrows can be specified by THTA, where THTA is the direction in degrees clockwise from North that the component is to go to - default the mean wind direction.
(errm... one feels that angles in meteorology should be called something else)
- NOTI The instrument vertical will be used, not tilted to make mean w equal to zero.
- INDS If this keyword is followed by OTS then the input files are the .OTS datasets. This has the disadvantage that the mean horizontal wind profile does not appear in the plot
- LOCA The local mean wind (not average over all probes) is subtracted, i.e. you're plotting u' in fluid dynamics notation (quite legitimate)
- NOAV The mean U is NOT subtracted - this also overrides SURF and LOCA

NOAR The arrow plot is suppressed

NOCO The contour plot is suppressed

NOFL The arrow head isn't filled in - useful for testing (plots quicker!)

DTAV The means for rotating the gust vectors and for subtracting from the horizontal component are taken over the display period rather than over the whole flight.

NOHT The height of each probe is taken as a constant equal to the value given in TMCMD.DAT Default is the height from the input dataset unless the dataset has no height information - i.e. was created before November 1988 - in which case the height taken from TMCMD.DAT will automatically be used.
N.B. This option MUST be used in baroclinic and other two-dimensional situations as the height measured by digiquartz and thermistor will vary in a spurious way which may exactly cancel out the variations in the quantity you are trying to measure!

SURF The average wind subtracted from the horizontal component is only taken from the probe nearest the surface rather than a mean over all probes. Default is over all probes.

UFRN Defines a fixed velocity subtracted from all horizontal velocities - useful for fronts.

LA75,TEKT*RONIX,REGIS,INKJ*ET
Keywords denoting the output device - the output is designed for the LA75. TEKTRONIX would be needed for the laserprinter. REGIS is definitely NOT recommended - definition is so poor that the result is usually a mess and it takes a long time to output on the VDU. The arrowheads are not filled for a REGIS plot. INKJET gives a REGIS output with appropriately coloured inks on a white background. It should then be processed with the RETOS software: "BAT RETOS/NOREVERSE dsn.OAC" followed by "IPRINT dsn.SIX" on BLISS

VSIZ The vertical size of the plotting frame in cm. The default value is 10.0cm. Setting this to a number nearer to 17.0 will use up more of the paper, but also increase both the rate of use of ink and the time required to print a picture.

HSIZ The horizontal size of the plotting frame in cm. The default value is 16.8cm. This is about the widest that can be achieved on the inkjet printer.

PLOT Two one dimensional plots are made of the variables selected by CONT and LINE in the vertical, with horizontal averaging over intervals DELT between times selected by TMIN and TMAX, excepting first and last columns, if PLOT=0. Otherwise only column N selected by PLOT=N is used. The output file is in GRAPH.DAT

LPLO A one dimensional plot is made of the variable selected by LINE in the vertical, with horizontal averaging over intervals DELT between times selected by TMIN and TMAX, excepting first and last columns, if LPLO=0. Otherwise only column N selected by LPLO=N is used. The

output file is in GRAPH.DAT. (keyword made obsolete by revised PLOT keyword)

DPLO A one dimensional plot is made of the two variables selected by CONT and LINE in the vertical, with horizontal averaging over intervals DELT between times selected by TMIN and TMAX, excepting first and last columns, if DPLO=0. Otherwise only column N selected by DPLO=N is used. The output file is in GRAPH.DAT

HPLO Two one dimensional plots are made of the variables selected by CONT and LINE in the horizontal, between times TMIN and TMAX for the probe identified by IZ= . The IZ number is 1 for the bottom probe and then sequentially upwards. If IZ is 0, an average is made of all levels between HMIN and HMAX. The output file is in GRAPH.DAT

DHPL A one dimensional plot is made of two variables selected by CONT and LINE in the horizontal, between times TMIN and TMAX for the probe identified by IZ= . The IZ number is 1 for the bottom probe and then sequentially upwards. If IZ is 0, an average is made of all levels between HMIN and HMAX. The output file is in GRAPH.DAT

XMIN, These define the maximum and minimum of the X scales in the
XMAX one dimensional plots generated by PLOT, LPLO, DPLO, HPLO, which are otherwise autoscaled.

YMIN, These define the maximum and minimum of the Y scales in the
YMAX one dimensional plots generated by PLOT, LPLO, DPLO, HPLO, which are otherwise autoscaled.

Notes on extended version (originally TMAP) available since autumn 1995

This version plots contours as shaded bands of 14 colours if INKJ*ET is specified. The OAC file will then have to be RETOS'ed and the resultant SIX file IPRINT'ed. The colours are cycled over again if the contour range is greater than 14*CSPC. The first colour used will be the first in the standard range used by the program, and if the STCC (set colour cycle) value is not defined it will correspond to whatever the minimum value of the variable to be contoured is within the defined range of H and T. If STCC is defined, then its value will replace the minimum value.

Note the decimal point after the STCC number. This is useful either for shifting the colour cycle or for ensuring that two or more pictures from different datasets have the same defined colour index.

This version can also produce trajectories that will replace arrows if PATT*ERN is specified. The trajectories can be increased in length by setting RLEN= the length in cms. (default is 3. I think.)

Keyword CLOUD=xx (e.g. xx=100.) invokes relative humidity line

contouring. Values above the threshold `xx` are solid lines, below (by up to 6%) dashed. If control variable 5 is invoked the program makes a crude approximation to convert potential to actual temperature, assuming surface pressure of 1000mb. If surface pressure minus 1000mb = `dp`, then `xx` should be increased by roughly 0.5`dp` if option 5 is invoked. CLOUD is not recommended in conjunction with PATT*ERN.

Frontogenesis and other specialised terms are also now calculated, and invoked by `CONT=11`, `CONT=52` etc. The actual terms plotted can be found by looking in the program - which has been amended throughout 1996. In addition a frontal velocity needs to be subtracted to calculate these terms and this is provided by `UFRN=` which is subtracted from ALL horizontal velocities.

In addition the keywords `NOCO*NTOUR` and `NOAR*ROW` can be used opposite the numbers of individual probes (as with `DESPIKE` etc). These two keywords suppress the data from the relevant probe so that it has no effect on the contour or arrow fields respectively and can be used when a faulty sensor on a particular probe would otherwise give a corrupt field.

Some mods have been made recently. Note `LOCA`, `NOFL` above.

Output routines

There are four output routines. These are `DRSREAD` which reads output from the logging routines, and also from `TMST`, `TMFQ`. Particularly useful is `TMRD`, which reads output from `TMCL`, `TMFO`, `TMTM`, `TMFD`, `TMTS`. `TMGR` plots output from these last five programs using a control dataset `TMGRCMD.DAT` or it can be used interactively. A useful display program is `PROBE.PLOTS`, run using a simple com-file without reference to `TMCMD` etc., where you need simply set the default directory to `taste` and `ASSIGN DZ*.OFD INPUT`, where `Z*` identifies probe-date-run as per Table 1. This then produces regis-able `DZ*1,2.PLT` where the 1 and 2 denote separate pages. If the run has more than 1 segment the different segments will appear as separate version numbers. `TMDS` operates on the output datasets `OCL`, `OFD` etc to remove spikes that have evaded the processing. It allows you to define minimum and maximum values for a particular variable and will backup variables outside this range. It should be used in conjunction with `TMGR`.

11 Appendix 3: Calibration datasets

After probes are sorted by `TMST`, they are then each calibrated by `TMCL` using a calibration dataset `CALIByy.DAT` (where `yy` refers to the last two digits of the year) which has the following format:

```
Probe number
date and letter of use
Instrument abbreviation
Parabolic coefficient(often 0.)
Linear coefficient
Constant coefficient
```


for any of the following instruments:

- Gill Top
- Gill Port
- Gill Starboard
- Thermistor Dry
- Thermistor Wet
- Thermistor Wet Fast
- Platinum Resistance Thermometer (only two coeffs)
- Magnetometer X
- Magnetometer Y
- Magnetometer Z
- Inclinometer Pitch
- Inclinometer Roll

All coefficients given in terms of a range -2048 to +2048??

12 Appendix 4: Channel numbers

Each probe (except 15 and 16, which are not really probes but pseudo-probes incorporating the surface site), uses the following channels:

CHANNEL NUMBER	QUANTITY
1	Pressure
2	Gill Top
3	Gill Port
4	Gill Starboard
5	Inclinometer Pitch
6	Inclinometer Roll
7	Magnetometer -Y
8	Magnetometer Z
9	Magnetometer X
10	PRT
11	Thermistor Dry
12	TW (actually humicap now)
13	Spare
14	TWF (actually spare now)
15	Reference
16	Battery volts
17	Rate switch settings(-1) in middle nibble
18	Zeroes (frame reference)

13 Appendix 4: Tapes

To read from tape use `$(dproc.command)batch_tape_read.com`, whilst `batch.tlist.com` on the same directory will list contents in terms of savesets. Log book should say what's on which tape.

Physical usage:

You MUST ensure tape write-locked (red showing) before trying to read, else you'll wipe the

tape.

To insert: push in, pull flap back gently, wait for green light then press red.

To extract: press red, wait for green on and red off, reverse flap.

Leave scratch tape (i.e. not a data-tape) in the drive.

14 Appendix 5: Instrumental specification

Lapworth and Mason (1988) give the following specification for a probe.

Weight of package (including batteries): 10 kg.

Total power consumption: 2 watts.

Absolute accuracies specified below assume regular calibration. Both absolute and relative accuracies are expressed as standard deviations.

Instrument	Absolute accuracy	Relative accuracy	Time constant
Magnetometer (probe upright)	0.5°	0.1°	0.01s
Inclinometer (probe upright)	0.5°	0.05°	0.5s
Gill (at 10 ms ⁻¹)	0.1 ms ⁻¹	0.01 ms ⁻¹	0.03s ??
u-component (at 10 ms ⁻¹)	0.2 ms ⁻¹	0.02 ms ⁻¹	0.03s ??
v-component (at 10 ms ⁻¹)	0.4 ms ⁻¹	0.03 ms ⁻¹	0.03s ??
w-component (at 10 ms ⁻¹)	0.4 ms ⁻¹	0.04 ms ⁻¹	0.03s ??
temperature slow	0.2K	0.02K	7.0 s
temperature fast	0.5K	0.02K	0.01s
humidity slow (at 10g/kg)	0.1g/kg	0.02 g/kg	13.0s
humidity fast	0.1g/kg	0.02 g/kg	7.0s
pressure	0.5mb	0.2mb	2.5s

Please comment on these figures in the light of present experience! The time constant quoted for the Gills seems optimistic, as does perhaps the absolute accuracy for humidity (of course now we have humicaps). I am not sure why the quoted accuracies for humidity are identical for slow and fast.

I am told that the slow temperature accuracy is probably pessimistic (0.1K absolute accuracy these days?), and the time constant more like 4s.

See MJB, CTN-16 for more up-to-date info.

15 Appendix 6: Scientific approaches and data analysis

The number of ways of analyzing data is virtually unlimited and what follows is just a set of suggestions, based on SHD's experience.

Recognize *surprising results*, which may be telling you either that your instruments aren't working or that your theories are wrong. Have a stab at predicting *results* before the experiment (if only to consider 'signal-to-noise ratio')

Don't despise the simple profile or sequence of profiles, whether of wind, temperature or fancy higher moments. Yes, it's often good to plot things in an appropriately non-dimensionalized way for general application. But there are many things about boundary layers we don't understand very well, and particularly over heterogeneous or complex terrain, or in evolving boundary layers, classical scalings (i.e. non-dimensionalizations) cannot be taken for granted. Plotting

'naive' dimensional quantities against e.g. height can reveal things both about turbulence and about possible measurement problems which non-dimensionalizations hide.

Use model comparisons to enhance the value of measurements (even simple measurements). Direct comparisons with NWP are important — the parametrization people need clear evidence of problems before they will rush to make changes.

Data analysis

Understand the sources of error. A good experimentalist — any good scientist — realizes what can go wrong, and documents relevant aspects so that readers can assess the quality of the work. Wieringa (1993) comments on how much information about both terrain and experimental procedures is needed for individual measurements of surface roughness to be generally useful.

All data of course have some scatter, but how scattered and why? Which scatter is (I) instrumental and which (II) real? With a good instrument, designed and tested to a specification, the distinction should be fairly clear, though with a crude instrument e.g. a windsock, it is blurred. Even using the famed Cardington turbulence probe the distinction becomes difficult in practice when performing under unfavourable conditions of orientation, frequency response etc., but it remains conceptually useful. In an experiment with a large sample and without systematic instrumental drift, the main role of re-calibrations is to maintain the distinction between (I) and (II), i.e. to tell us whether the scatter is instrumental or real. If the scatter is large this will have a significant bearing on how we analyze the numbers.

(I) First consider possible instrument malfunctions, drift or response changes, intermittent problems (usually the most frustrating sort), spikes due to reception problems or RF interference ... Part of instrumental technique is of course to minimize the impact by optimal deployment, timely remedial action and noting any problems in the log-book. In the final analysis, you can't tolerate a curate's egg experiment ('good in parts'): the bad bits must be eliminated, either before or after, even at the risk of discarding some good bits.

(II) Secondly, consider *representativeness* errors. Scientific results need to be reproducible or at least placed in context. Purely anecdotal observations are of limited value, unless both exceptionally interesting (i.e. surprising) *and* well-validated (think of crop circles!). Research is usually about trying to make general statements, testing them properly against data and assessing them theoretically. For instance, the statement 'on 26/02/93 at Cardington at 12Z the screen temperature was 5.3°C' *on its own* has very little value, and in research virtually nil. However *taken together* with similar measurements at different places and/or different times, or the same place and time with different instruments, or with other predictions or predicted relations between measurable quantities, it starts to become useful.

So context and comparison is vital: a number is no use unless we know what it means. Thus in combining results from different occasions, as one has to, some scatter, and hence errors, arise because the situations are slightly different. This is true even after allowing for 'standard effects', e.g. stability, since the atmosphere is in reality much more complicated than any of our scalings or models. Similar representativeness-errors arise if the averaging time is not perfect (and there may not be any perfect averaging time if there is no spectral gap). Such errors are difficult to define for single measurements: they depend 'subjectively' on what the data are supposed to represent. They become objective errors however when combined to form a single dataset of *nominally similar* measurements, from which averages or 'best fits' are taken. Choosing a 'good site' is also about 'representativeness' in a number of ways. A site which is a 'complete mess' might be fine if you had a good theory or model for the complete mess (although that's unlikely).

The classic problems of shielding and exposure, as discussed in the Handbook of Meteorological Instrumentation, may also be regarded as representativeness issues, in a slightly different sense.

Quality control (QC) is necessary even with a large sample, because in general averaging a lot of bad data doesn't give a good dataset. That's partly why so much uncertainty remains after decades of boundary-layer measurement! Data will be reliably cleaned-up by averaging only if the errors (with standard deviation σ) are (i) unsystematic (gross instrumental errors are often one-sided) and (ii) small enough for nonlinearities in manipulation to be neglected. If for instance the harmonic or geometric means differ from the arithmetic mean, as they will at order $O(\sigma^2)$, this may indicate a residual uncertainty of that order even after averaging, unless there are strong particular reasons why one such mean is correct as measured and averaged. (One might argue for instance that the arithmetic mean of Reynolds stress corresponds to the correct average for dynamical purposes). Some other second-order errors, inducing spurious correlations, are discussed by Hicks (1978, 1981). It is a useful exercise to generate a 'noise' dataset and analyze it as if it were measured: sometimes the results can be alarmingly plausible, and if so then some hard thinking is called for.

In fact the harmonic mean HM, which is based on averaging reciprocals, will be lower than the arithmetic mean AM by $\sigma^2/\mu + O(\sigma^4/\mu^3)$, where the mean $\mu \gg \sigma$, provided that points close to 0 are excluded. If $\sigma \ll |\mu|$ these probabilities are super-exponentially small for Gaussian pdf's, but if they are not excluded the probability integral formally doesn't converge. An alternative procedure, somewhat similar in effect, is to redefine the function by 'capping' high values.

In general, an averaging procedure AP based on averaging a function f will imply

$$AP - AM \sim \frac{1}{2} \sigma^2 f''/f$$

if scatter is small and $f' \neq 0$. In particular, if f is a simple n th power, then

$$AP - AM \sim \frac{\sigma^2}{2\mu} (n - 1)$$

In Vaxgraph, quality control before averaging may be handled using SET RESTRICT command (up to 3 criteria presently allowed). This is best done using instrumental criteria, e.g. whether a temperature difference is large enough to be accurately measurable, or some reason to suspect a particular probe. Expect to lose a significant fraction of data points through objective quality control. The value of collecting a lot of data is partly that one afford to be choosy!

The treatment of outliers is a delicate subject. Isolated points which are obviously outside the normal scatter should be removed from any *averaging* if you are sure they are not 'real' — a kind of despiking if you like. Since the effect of such outliers is often heavily dependent on the type of averaging, and the latter is usually somewhat arbitrary, it is unlikely that they will significantly improve your dataset unless it's beyond hope anyway! Faced with conflicting results, sometimes you just have to choose which you believe; if you think both are real then sampling issues need to be addressed.

Removal of outliers tends to bring means towards the *median*, which is less sensitive to extreme values, and thus can be justified in terms of pure data-analysis if you are sampling an essentially homogeneous (e.g. Gaussian) population. However it is better to remove them for solid instrumental reasons (e.g. differences too small to measure). Sometimes, e.g. in stable boundary layers, turbulence may be naturally intermittent and strongly non-Gaussian, possibly even with a bi-modal probability distribution. In such cases representativeness errors can be severe, and it's hard, perhaps impossible, to apply quality control without biasing results.

For many datasets it's best to have two plots of each derived quantity:

- (i) with *all* points and minimal quality-control, removing only those points which are suspect for explicit instrumental reasons
- (ii) with binning, after removing gross outliers even if the instrumental reasons are unclear, and the standard error of the bin mean plotted. If such removal of outliers has significant impact then it must be discussed in the accompanying text.

Model comparisons

Project RP15 is currently entitled 'Observational verification of boundary layer models'. The word 'models' covers at least two rather different species:

- (i) Forecast models, which know about weather systems, and something about topographic or land-use variation on large scales. In representing physical processes they tend necessarily to be 'jacks of all trades'. For instance they do not really include gusts in the main model; any gust prediction is by subsequent diagnostic procedures.
- (ii) Process research models, which know quite a lot about turbulence but usually simplify topography, land-use, physics ... Even here there is a distinction between models for uniform terrain, such as the Large Eddy Model, and models for heterogeneous or complex terrain. The penalty for a more complicated surface tends to be loss of turbulence resolution, so that e.g. a flow-over-hills model is typically something between a Large-Eddy model and a forecast model. Generally process models make choices about processes and scales which require verification.

There are various reasons for comparing with models. One reason is 'representativeness'. In case of doubt, models often provide the best guide to what data 'should' represent, i.e. the *definitions* of certain variables like z_0 should as far as possible be consistent between models and experiments. Indeed certain quantities, perhaps including z_{0t} , cannot really be defined without thinking about models. Normally nowadays the aim of a good atmospheric experiment is to match the definitions sufficiently well, and keep experimental errors sufficiently small, to be able to make clear recommendations to modellers.

Models are best thought of perhaps via *prognostic* equations of the form

$$\partial U / \partial t = -\partial \overline{u'w'} / \partial z + \dots \quad (1)$$

combined with *diagnostic* equations

$$\overline{u'w'} = \text{function}(\partial U / \partial z, \partial \theta / \partial z, \dots) \quad (2)$$

Actually, models of course don't store derivatives, but discrete gridpoint values. However discretization methods are a highly technical area, not really relevant to us. Like e.g. electronic circuits, models can suffer from instabilities and other problems which only the experts really understand. One should however note the *scales*. Even mesoscale models hold wind speeds, temperatures etc. notionally averaged over at least 10 km horizontally, which may roughly correspond to a half-hour or one-hour level run.

The most analytical way of testing models against data is to evaluate the terms in these equations! Suitably averaging the trends and stress divergence, one should be able to infer additional terms (e.g. advection) as a residual. For temperature the corresponding residual would include radiation terms (which we can to some extent measure). Not easy but worth a go I think. We do now output DUDT etc. from TMVG, though the stress divergences would require some kind of differentiation.

The second way is to compare the measured stresses with those predicted by model parametrizations (or DIY parametrizations?) from the measured gradients. Of course this is not easy; again it requires some kind of differentiation, by hand or otherwise.

There are other and essentially equivalent ways of doing these, but the above comments show what you're up against.

Evaluating the prognostic equation is perhaps the most difficult. There are two main ways to avoid it:

- (i) Assume a horizontally homogeneous steady state and apply 'universal' scalings
- (ii) Do a climatological comparison with a model

We wouldn't want to duplicate the routine model-verification by operational observations, except where we have better instruments or location. Boundary-layer verification is always a weakness though, especially in complicated terrain. It's much easier to verify NWP models in their pressure-prediction (which is indirectly important) than in boundary-layer winds, temperatures, clouds etc. or indeed weather. But this needs to be done, otherwise we get a rosy picture.

Reprise on models and obs

One shouldn't get an inferiority complex towards models. It can even be argued that models in themselves aren't science. It is surely the *comparison* of independent evidence, e.g. good models and good observations, which distinguishes science from speculation, e.g. because one can then start to suggest confidence limits based on the discrepancy between models and observations. The more comparisons the better.

The influential philosopher of science Karl Popper (see Weather, Sep 1992) has argued that falsification or attempted falsification of theories by stringent tests against independent evidence is the basic method of scientific advance. This view implies that experiments cannot prove a theory, merely disprove one, and that ultimately theories come out of our heads, not in some mysterious way out of 'data' conceived as a remorseless accumulation of facts. Normally a range of curves is consistent with data, and to show that a sample is representative requires at least a conceptual model. Hence the derivation of theories from data is partly 'creative', but testing is more objective. Whether or not you entirely agree with Popper, try to test theories — if you don't like the UM schemes, *show* that they're wrong.

All this philosophy does have implications for the *design* (and interpretation) of experiments. Models systematically over-idealize out of necessity, so experiments over 'ideal' terrain are generally more comparable. It is important to get a feel for extraneous effects in terms of magnitudes and signs. It's also highly desirable to ensure that experimental results are analyzed in terms which 'make sense' theoretically, particularly in rare limiting cases such as winds tending to zero. Such cases may occur rarely in reality (in which case 'experimental evidence' is surely limited) but cause trouble in models. An example is the Businger-Dyer scaling in the free-convection limit, which causes modellers grief by giving nonsense in that limit. See also Kaimal (1973) who chooses to scale certain things on R_i in a way that (to SHD) doesn't really make sense, nor even agree with his own data, at either end and consequently has to restrict the range of validity of R_i . Extrapolation from observations, and perhaps even interpolation, must be guided by theory. It's vital to have a feel for the strength or weakness of theoretical arguments because at some point, e.g. if your measurements appear to show that $2+2=5$, a strong theory may have to override measurement. Disagreements with theory should be conscious and considered, not 'accidental'.

Surface transfer

Quantity	comments
c_d, c_E	for various quantities
$c_d^{(n)}$	incorporating stability correction
z_0	obtained by plotting c_D vs. height or kU/u_* vs. $\log z$
$\ln(z_0/z_{0t}) = k/B$	see Beljaars and Holtslag, eq. (15) (requires e.g. radiative surface T)
$Re_* = u_* z_0/\nu$	roughness Reynolds number (influences z_0/z_{0t})

Plots of the surface energy balance components (sensible, latent heat, radiation ...) against time are useful. We should be measuring everything bar the soil heat-flux which can in principle be inferred as a residual.

General boundary layers

Profiles U, θ and flux-profiles are standard but essential. Variances σ_w^2 etc. are also useful. σ_w^2/u_*^2 is useful not least as a measurement check, because in neutral and stable conditions it's usually close to 2. The skewness $\overline{w^3}/\sigma_w^3$ should be small except in convective boundary layers (near surface behaviour in canopies?). The kurtosis or flatness $\overline{w^4}/\sigma_w^4$ is useful because 'normal' homogeneous turbulence is close to Gaussian (kurtosis=3) in the velocity statistics. (See Met. Glossary). In the lower convective boundary layer the kurtosis can be somewhat higher. Also in any situation where turbulence is intermittent the kurtosis is likely to be substantially raised. Higher order moments like $\overline{w^4}$ have a reputation as difficult to measure (requiring long averaging times), but combination with σ_w to form kurtosis tends to minimize this problem. But check the time-series for spikes.

Dissipation ϵ can be estimated from spectra. This is somewhat time consuming but useful, as the results seem reasonably accurate. SHD has some pre-prepared CTL files which help. ϵ_T can also be estimated, though the accuracy is much less clear owing to noise at high frequency on PRTs. I have tried removing this empirically by subtracting a white noise spectrum $STT \propto K$; with suitable constants the spectrum can be much improved but although the approach is basically rational I haven't got total confidence in the remaining numbers!

An important step forward comes if you can calculate gradients of wind and temperature. (SHD has a package for doing this intended basically for model initialization; at present it's a bit cumbersome but may be adapted.) This enables gradient and flux Richardson numbers R_i, R_f to be calculated, together with a wealth of local-scaling diagnostics.

Pictures: Use of TMAC to produce pictures of the flow is also useful, to reveal e.g. convective boundary layer structures or just for reference (use of the new control variable option 12 helps combine the stability of the slow thermistor with the response of the PRT). But try to use proprietary software such as PV-Wave — MRU hasn't the time to be a graphics programming team.

Heterogeneous terrain

Surface transfers as above. Look at stress and flux divergence (implied advections). Tendency terms DUDT etc. are calculated (*correctly?*) but probably relatively small in surface layer. Identify height scales and try to relate them to horizontal scales e.g. with an aspect ratio $c_d^{1/2}$ (cf. Grant 1991).

Is the fall-off of surface stress over a (relatively smooth) local surface consistent with the estimated ratios of z_0 at that surface and upwind to the blending height l_b ? According to Mason's heuristic model, in neutral conditions the drag should be partitioned across surfaces of different roughness with local stresses in proportion to $\ln(l_b/z_0)^{-2}$. E.g. if $l_b=10\text{m}$ and $z_0 = 0.01\text{m}$ locally as against an area-average of 0.1m , then the local stress should be $4/9$ of the area-average value (the latter can be estimated from extrapolating the stress profile above the blending height down to the surface).

Humidity

Some ideas on humidity analysis are given in the detachment document for Charney Bassett. One of the most basic quantities is the Bowen ratio $H/\lambda E$, i.e. ratio of sensible to latent heat. Various resistance formulae are also useful. However you'll need to evaluate an appropriate (local or average) surface temperature to obtain saturation humidity.

Calculation routines for various thermodynamic quantities may be found in [dproc.process]hmr.for. Here their significance and behaviour will be briefly discussed. Measured in absolute temperature scale, virtual temperature T_v is greater than temperature by a factor $(1 + 0.61q)$ where q is in kg/kg. This is often relatively unimportant, since 1 g/kg corresponds to only 0.2 K in virtual temperature. Similarly, virtual potential temperature $\theta_v = (1 + 0.61q)\theta$. Both θ_v and T_v are often expressed in Celsius, as per θ, T but of course the humidity factor has to be applied to the absolute temperature. We can define buoyancy $b = g\theta_v/\theta_{\text{ref}}$, and in an *unsaturated* atmosphere the static stability $N^2 = g\partial \ln \theta_v / \partial z$.

By contrast equivalent potential temperature

$$\theta_e \simeq \theta \exp(\lambda q / c_p T) \quad (3)$$

(the approximation assumes moisture can be condensed with small fractional variation in absolute temperature). Note that θ_e is a function of θ_w alone, i.e. it represents just a different (and analytically more convenient) way of labelling θ_w lines. In fact θ_e can be evaluated on a tephigram by following a θ_w line (moist adiabat) up to low pressure where it becomes parallel to a θ -line, which gives the value in question (see Forecasters' Reference Book 1993, p.61, for graphical construction). Roughly, the difference between θ_e and θ_w corresponds to the total latent heat of saturation at θ_w at 1000mb . Graphically to obtain θ_e from θ_w , follow the saturated θ_w line to 1000mb , where $\theta = T = T_w = \theta_w$, and substitute $T = \theta = \theta_w$ and $q = q_s(T = \theta_w, p = 1000\text{mb})$ into (3). In other words $\theta \rightarrow \theta_w$ at the bottom of a saturated adiabat but $\rightarrow \theta_e$ at the top.

One reason for preferring equivalent to wet-bulb temperatures is that the 'practical' definition of T_w is slightly vague. It doesn't correspond to a thermodynamic equilibrium because the wet-bulb sensor is at a different temperature from the environment. The conventional definition as 'the temperature to which the air can be cooled by evaporation' is therefore ambiguous as to the temperature, and therefore the sensible heat content, of the evaporating water. This ambiguity is fortunately negligible for practical purposes.

For reference, we now briefly summarize the relation between wet-bulb temperature T_w and dew-point T_d . Each is defined as the final temperature of a cooling-procedure at constant pressure:

- (i) T_w : cool a parcel to saturation through the *latent heat effect of adding moisture*:

$$dT = -(\lambda/c_p)dq$$

where as noted above we neglect any sensible heat added through moisture. Hence if q_s is

the saturation specific humidity, dependent only on T, p , then

$$d(q - q_s) = dq - dq_s = dq - (\partial q / \partial T)_p dT$$

whence by setting $T = T_w$ at the endpoint $q = q_s$, and translating the differentials d into finite changes (justified if the dewpoint depressions are small cf. 15 K),

$$q - q_s \sim (T - T_w)[c_p/\lambda + (\partial q_s / \partial T)_p]$$

(ii) T_d : cool a parcel *purely by thermal contact*, e.g. with a cold dry body. Here $dq = 0$, so the equations are easier, giving

$$q - q_s \sim (T - T_d)(\partial q_s / \partial T)_p$$

T_w is the temperature to which a wet body, e.g. a screen wet-bulb or a cloud droplet, tends to cool by evaporation. Dew-point T_d is the temperature at which a dry body, e.g. a mirror, first suffers condensation. Clearly the ratio of wet-bulb depression to dew-point depression is governed by the ratio of terms in square brackets in (1). Since the second term $(\partial q_s / \partial T)_p$, like q_s itself, rises approximately exponentially with temperature, it predominates at higher temperatures and T_w becomes closer to T_d .

The fractional effect per unit specific humidity q on total (sens.+latent) heat content, as translated into θ_e , is $\lambda/c_p \simeq 2500K$ whereas the corresponding effect on buoyancy, translated into virtual temperature, is $\theta_0 0.61 \simeq 200K$, where the constant 0.61 in fact represents the fractional difference in molecular weights ($\simeq 29/18 - 1$). Both these numbers seem large, but for typical values $q \simeq 0.01$ the total effects are of order 25K, 2K respectively. The latent heat effect is genuinely 'large' for most thermodynamic purposes, in that $\lambda/c_p T \simeq 8$ for typical absolute temperatures T , whereas the scale for virtual temperature effects is $O(1)$ when non-dimensionalized in the same way. The approximation (3) is also effectively based on $\lambda/c_p T \gg 1$.

The saturation vapour pressure e_s can be calculated to sufficient accuracy for most purposes by

$$\ln e_s / 6.11 \text{mb} \sim (\lambda/R_v)(1/273 - 1/T) \quad (4)$$

This dependence arises from Maxwell's relation

$$\left(\frac{\partial p}{\partial T}\right)_v = \left(\frac{\partial S}{\partial V}\right)_T \quad (5)$$

applied to a vapour-liquid phase mixture in thermodynamic equilibrium. Here S is the total entropy, with the change between phases given by $dS = d\text{heat}/T$ for a reversible change such as isothermal expansion of a chamber with conducting walls by varying the applied pressure. Using this, the right hand side of (5) can be evaluated as $\lambda/T\Delta\alpha$, where $\Delta\alpha$ is the change in specific volume between phases, which is approximately the specific volume of the vapour and can therefore be determined from gas laws. For thermodynamic equilibrium of such a system, the pressure p must of course be the saturation vapour pressure e_s , whence combining all this we obtain the Clausius-Clapeyron relation (approximated)

$$\frac{de_s}{dT} \sim e_s \frac{\lambda}{R_v T^2} \quad (6)$$

of which (4) is simply an integrated form.

Equation (4) means that e_s (or q_s at a given pressure) is a quasi-exponential function of absolute temperature T , e -folding on a temperature scale

$$R_v T^2 / \lambda \simeq 15K.$$

or doubling every 10 degrees as a rule of thumb. This is much more nonlinear over typical surface temperature range than other thermodynamic equations, and partly explains why tropical meteorology is so dominated by latent heat release! The conclusion is that the effect of moisture on latent heat is much more important than its effect on buoyancy (or θ_v). Remember that 'latent heat' only operates physically during evaporation or condensation, so that otherwise it's just a 'book-keeping' item.

It may be worth considering the relation of cloud-base to dewpoint depression. Handbook of Wx'ing gives cloudbase/depression around 130m/K for a well-mixed boundary layer. This is partially related to the dry adiabatic lapse-rate $g/c_p \simeq 1\text{K}/100\text{m}$. The cloudbase formula is not totally reliable in practice since boundary layers only approximately well-mixed, but gives a ballpark figure. Typically 1g/kg of specific humidity means about 1K dewpoint depression at 8°C, but this actually depends quite strongly on temperature.

The relation between humidity and rainfall is also worth bearing in mind. Suppose that n g/kg of q are precipitated out over a height (pressure) scale of $m \times 100\text{mb}$. This corresponds to about mn mm of rain. Typically if all the water *vapour* of the daytime boundary-layer precipitated you'd get about 1cm rain. This is not enormous but of course we're ignoring liquid water in clouds.

16 Appendix 7: inertial subranges

The estimation of ϵ proceeded as follows. We assume Kolmogorov (1941) scaling in an inertial subrange, and also Taylor's frozen-field approximation. Following the notation of Monin and Yaglom, let $E_i(k)$ ($i = 1, 2, 3$) be the 1-D spectra such that

$$\int_0^\infty E_i(k) dk = \langle u_i^2 \rangle$$

and $E(k)$ the conventional 3-D energy spectrum such that

$$\int_0^\infty E(k) dk = \frac{1}{2} \langle u^2 \rangle$$

We can measure the E_i spectra but not $E(k)$.

Then from Monin and Yaglom, in the inertial subrange

$$E(k) = \frac{55}{18} E_1(k) = \frac{55}{24} E_2(k) = \frac{55}{24} E_3(k)$$

Kolmogorov (1941) scaling implies

$$E(k) = \alpha \epsilon^{2/3} k^{-5/3}$$

$$E_1(k) = \alpha_1 \epsilon^{2/3} k^{-5/3}$$

where $\alpha \simeq 1.5$ is the Kolmogorov constant and $\alpha_1 = 18\alpha/55$. Similarly we may define $\alpha_2 = \frac{4}{3}\alpha_1$, for which theory shows that

$$E_2(k) = \alpha_2 \epsilon^{2/3} k^{-5/3}$$

So

$$\epsilon = k[kE_1(k)/\alpha_1]^{3/2} = k[kE_2(k)/\alpha_2]^{3/2} \quad (7)$$

Note also that

$$E(k) = \frac{5}{6} (E_1(k) + E_2(k) + E_3(k))$$

After some experimentation and consultation I have come to the conclusion that SUU in the SPC files produced by TMVG means $kE_1 (= nS_u(n))$ and similarly for SVV and SWW. However K means reciprocal of wavelength, i.e. wavenumber divided by 2π . So ϵ is estimated as

$$6.2831852 * K * (((55./18.) * SUU / 1.5)) ** 1.5$$

or

$$6.2831852 * K * (((55./24.) * SVV / 1.5)) ** 1.5$$

or

$$6.2831852 * K * (((55./24.) * SWW / 1.5)) ** 1.5$$

or

$$6.2831852 * K * (((5./6.) * (SUU + SVV + SWW) / 1.5)) ** 1.5$$

For statistical stability these estimates are averaged into wavenumber bands, about 4 per decade. The estimates based on total energy spectrum were not surprisingly rather more stable. An inertial subrange corresponds to a wavenumber region where ϵ -estimates are constant. The subjective procedure of estimating ϵ may be loosely defined as (i) fitting a smooth curve by eye; (ii) reading off the approximate maximum value of that curve (since the deviations at either end of the spectrum from inertial-range scaling tend to produce lower values).

The spectral components measured by our probes do not exactly agree with the predicted ratios. The theory is unusually rigorous and well-confirmed (by the standards of turbulence research!) and the problem is probably in the measurements. The streamwise component of the 1-D spectra seems to be on average about 60% rather than 75% of the transverse spectra. I understand this is a known problem. If all components are measured accurately then an even average over all three would tend to minimize scatter. However in practice some components are better than others!

Experience and instrumental comparisons suggest that probe-measured lateral components VV and SVV show excess energy. In a spectral range where the estimate for ϵ differs systematically between e.g. SUU and SVV it is probably better to go for the lower value because most sources of measurement error would give spurious additional noise.

Basically SUU is probably best, then SWW, with SVV worst. This is helped by the fact that the inertial subrange is slightly longer with SUU, and at lower frequencies.

17 Appendix 8: standard plots

Any other standard plots, Hojstrup spectra etc. etc. please list here (verbatim) plus references.

18 Recent additions

18.1 INCORP

SHD has developed a text-compiler called INCORP (now on [DPROC.PROCESS]). The basic action is to include a section of a file via command

-inc filename,sectionname

where the current file can be referenced by a dot (.). This text-compiler can preprocess files for a wide range of other programs. The command *-incdef symbolname* creates a symbol which can then be invoked via the string

|||symbolname|||

A VMS symbol can be invoked via

|||+symbolname|||

There is also provision for lists (a sort of DO-loop via command *-inclist*). Each section finishes with *-stop*. A line starting *-noinc* will be ignored (i.e. can be used as a comment).

Example:

```
-incdef cent:=19          !! century
-incdef period:=choose    !! morning, allday, choose
-incdef vble:=wind        !! specify plotting variable
-incnoprint               !! switches off certain diagnostic output
REGIS OUT=REG.PLT
-inc .,plot1
NEWPLOT NEWFILE=LAS.PLT TEKTRONIX
-inc .,plot1
-stop-----
-sec plot1
|||+DDATE||| |||+FXC||| FX from LF R&D
-inc .,|||vble||| !! options wind,temp,q,dirn
HTFT
LIN

O. 5000.
SET FRAME DEFAULT
SET FRAME YTICK=10 YANN=INTEGER
SET RESTRICT:Y=(0.0,5000.0)
-inclist .,time=|||period|||
FETCH LABEL=|||+FXC|||||cent|||||+revdate|||||time|||
PLOT CURVE=FULL SYMBOL=|||time_ctr||| INDEX=<VT=|||time|||Z>
-incendlist
-inc .,|||markx|||
```



```

-stop-----
-list choose !! choose list of times
0030
0330
0830
1030
1330
1730
-stop.....
-sec wind
WIND
LIN
wind /(m/s)
0. 25.
-incdef markx:==markunits
-stop.....
-sec markunits
MARK X=0.0 CURVE=DOTTED
MARK X=1.0 CURVE=DOTTED
MARK X=2.0 CURVE=DOTTED
MARK X=3.0 CURVE=DOTTED !! etc. - shortened for display here

```

Comments: the above is a simplified form of the file LF:LF.ICTL used to do the daily forecast plots. I give such files the type .ICTL (I for incorp). The body of the plotting sequence (plot1) is invoked twice, once for REGIS and once for TEKTRONIX. The symbol DDATE (defined previously at VMS level) is invoked early in plot1. The variable vble is defined early in the file and used to switch the section specifying (here) wind. The list facility is used to loop over various forecast times. The loop counter is called *time_ctr* (automatically, because the loop variable is *time*).

18.2 VAXMDIF_NEW

For direct comparison with NWP we may wish to differentiate profiles of Vaxgraph level runs. SHD has developed a semi-objective method, based on estimating relative errors in direct measurement compared to interpolation. Differentiation requires smoothing, or at least consideration of smoothing.

Currently the relevant programs are on a directory called CRONED: (meaning \$DISK2:[DERBYSHIRE.CRAYTEMP.ONED]). The command file CRONED:DIFNEW.COM invokes CRONED:DIFMERPLT_DEFS.COM and CRONED:DIFMERPLT_NEW.COM, and the latter runs \$DISK2:[DERBYSHIRE.DATA.TEMP]VAXMDIF_NEW.FOR which differentiates and merges back variables like RI. This program was the basis of results in Derbyshire (1995), q.v. for further discussion.

A program STASORT is used to tidy up the STA datasets afterwards.

When using VAXMDIF_NEW, you should check the smoothed profiles THS (smoothed THTA) etc.

One of the advantages of the automated part of VAXMDIF_NEW is that it can easily be rerun if the main dataset is reprocessed.

19 Programming standards

In computing circles much thought is given to the 'software crisis'. Basically hardware gets steadily better, and occasionally cheaper, whilst the task of developing, verifying and maintaining software becomes more and more expensive, chiefly in manpower. See the article by Thorne in 'Newer Uses of Mathematics' (ed. C.Bondi, Penguin 1991).

For our purposes we do not really have the option of adopting fancy 'verification' languages, and even if we did it might not help much (no such language will spot that a probe's wiring is inconsistent with TMCL). The best we can do is to ensure that programs are well-structured, well-documented and well-tested, with useful diagnostic output. We should try not to work in a hand-to-mouth way, in which thinking is driven purely by perceived bugs.

The following is SHD's recommendations (see also FORTRAN 77 standard for the Met. Office, copy in MRU library):

- (i) modularization where possible — break up a complex task into subtasks so that if something goes wrong you can isolate the problem. (Actually this is a plus point of the current processing suite and the separation of graphics from calculation).
- (ii) layout does matter, e.g. indentation (preferably 2 spaces) for DO-loops and IF blocks.
- (iii) keep routines, including main routines, reasonably short (ideally 1 or 2 pages of listing so can see almost at a glance)
- (iv) use numbered sections for long-ish routines
- (v) use plentiful comments explaining rationale for mods etc. (exclamation-mark comments on the same line are useful)
- (vi) yes, this includes stating the obvious — consider the uninitiated! for instance, avoid abbreviations in documentation.
- (vii) explain clearly the flow of control: e.g. at end of a long loop over probes
30 CONTINUE ! END OF LOOP OVER PROBES
- (viii) don't be too clever unless *really* necessary — and then explain. Many 'clever' programmers have been cursed by their successors.
- (ix) avoid changing the meaning of variables through the code unless really necessary — it's seriously confusing
- (x) updates should be commented with name of programmer and date (usually best to comment out old bits, not deleting unless sure they're obsolete or wrong)
- (xi) purpose of a program or routine plus other calls, datasets, etc., within reason should be commented at beginning
- (xii) common blocks are best dimensioned within the common statement, and then copied identically to each relevant routine, otherwise you risk getting things scrambled if you forget to dimension an array
- (xiii) beware lines approaching 72 columns and stay well short of that if possible, particularly if you have a following continuation line (errors often occur via characters being chopped off; if a comma is chopped off variable names may be run into those on the continuation line *and compiler often doesn't spot the error*)
- (xiv) archive working versions periodically (avoid using command PUN on directories with large programs — PUN/SINCE=YESTERDAY is safer; I abbreviate this to symbol PUNSY; if you purge anything older than a week or two you need to be sure what it is)
- (xv) try to write software so that datasets are well labelled internally (so you don't end up wondering where things came from)

(xvi) print out or display suitable diagnostics but emphasizing the important ones (don't hide them amongst routine bits)

Think symbolically, using variables or parameters rather than constants as far as possible. Arrays cannot be dimensioned using variables (except for variables passed as subroutine arguments, but this is fiddly and requires sufficient storage allocation in the calling routine). Use instead PARAMETERS (see FORTRAN manual if in doubt). This is extremely useful, and I can think of several programs which would be easier to understand and modify if they had been written using parameters from the start (20/20 hindsight, of course). I tend to use the convention that parameters end in P. You can also use real parameters but they're much the same as real constants in practice. A loop 1,16 — is 16 here the number or probes or some other 16? What changes should we make if we had more probes? Clearly the code would be more self-explanatory if 16 were replaced by some symbol (variable or parameter) like NUMBER_OF_PROBES, or perhaps a slightly shorter version! Also don't forget that WRITE or OPEN statements allow the unit-number to be a variable, and READ(LUNCALIB,...) is more helpful to the casual reader than READ(4,...) — I always have to check my 3's, 4's and 5's.

Remember that FORTRAN tends to ignore spaces.

Optimally efficient user-control of programs or procedures can be viewed as a problem in information theory: how to convey the maximum information for a given number of characters (or words), or conversely to minimize the number of characters for a given piece of info. (There are other arguments against duplication of info — it can lead to inconsistency.) Formal information theory says that information content varies like $-\log(p)$ where p is the probability of the message. Overall information content is maximized by making the number of characters for a given 'instruction' reflect proportionately the information content, although we might also have other considerations like memorability. Normally in DCL abbreviations it is hard to justify more than three letters except for mnemonic reasons, since $26^3 > 17000$. Two-letter words are also significantly easier to type. Use abbreviated symbols and logical names as much as possible in DCL, for commands you use more than about once! There's a lot to be said for define logical names for directories, e.g. `$DEFINE H $DISK1:[DPROC]`, so you can then edit `H:LOGIN.COM` without changing default directories. In VMS there is no conflict between using H as a *symbol* meaning `SET DEF $DISK1:[DPROC]` and the above logical name definition.

Get to know the .LIS files. They are an important debugging tool, particularly the variable lists, which can show up mistyped variable names. (On the subject of mistyping, it is worth avoiding Os and 0s in variable names unless it's exceptionally obviously which is which).

One final plea. In typing symbolic instructions, whether FORTRAN or DCL, use abbreviations as above but *don't try to type as fast as your fingers will let you*. You may be able to type your Report of Work at 100 words/min, and the odd spelling mistake wouldn't bring the house down. But unless your mind is much quicker or your fingers much slower than mine, the optimum *programming* speed will be determined by your speed of thought. In modifying a large program, I would seriously question whether you can add more than about 20 lines a day of completely correct FORTRAN (an estimate I made based on the time to write and debug the Large Eddy Model). If you're adding a lot more than that, make sure you are carrying out all reasonable tests. *A large complex program known to be correct is like gold.*

References

- Beljaars, A.C.M. and Holtslag, A.A.M. (1991): Flux parametrization and land surfaces in atmospheric models. *J. Appl. Met.* **30**, 327-341.
- Caughey, S.J. (1977): The Cardington turbulence instrumentation. Met. Office unpublished technical note (TDN 82).
- Caughey, S.J. and Palmer, S.G. (1979): Some aspects of turbulence structure through the depth of the convective boundary layer. *Quart. J. Roy. Meteor. Soc.*, **105**, 811-827.
- Caughey, S.J. and Wyngaard, J.C. (1979): The turbulence kinetic energy budget in convective conditions. *Quart. J. Roy. Meteor. Soc.*, **105**, 231-240.
- Caughey, S.J., Wyngaard, J.C. and Kaimal, J.C. (1979): Turbulence in the evolving stable boundary layer. *J. Atm. Sci.* **6**, 1041-1052.
- Derbyshire, S.H. (1995): Stable boundary layers: Observations, models and variability Part II: data analysis and averaging effects. *Bound. Layer Meteor.*, **75**, 1-24.
- Grant, A.L.M. (1992): A comparison between the Cardington turbulence probe and a sonic anemometer. CTN-13.
- Grant, A.L.M. (1992): An error analysis of the Cardington turbulence probe orientation algorithm. CTN-14.
- Grant, A.L.M. and Watkins, R.D. (1992): Errors in turbulence measurements with a sonic anemometer. *Bound. Layer Meteor.*, **46**, 181-194.
- Haugen, D.A., Kaimal, J.C., Readings, C.J. and Rayment, R. (1975): A comparison of balloon-borne and tower-mounted instrumentation for probing the atmospheric boundary layer. *J. Appl. Met.* **14**, 540-5.
- Hicks, B.B. (1978): Some limitations of dimensional analysis and power laws. *Bound. Layer Meteor.*, **14**, 567-569.
- Hicks, B.B. (1981): An examination of turbulence statistics in the surface boundary layer. *Bound. Layer Meteor.*, **21**, 389-402.
- Hignett, P. (1991): The performance of an OPHIR IR-2000 Infra-Red Hygrometer. MetO(P) Internal Technical Memorandum NO. 18.
- Jones, J. (1991): An investigation into the validity of the temperature fluctuation and heat flux relations for several stability conditions with a sonic anemometer. CTN-?
- Lapworth, A.J. and Mason, P.J. (1988): The new Cardington balloon-borne turbulence probe system. *J. Atmos. Oceanic Technol.* **5**, 699-714.
- Mahrt, L. (1979): Penetrative convection at the top of a growing boundary layer. *Quart. J. Roy. Meteor. Soc.*, **105**, 469-486.
- Monin, A.S. and Yaglom, A.M. (1971): Statistical Fluid mechanics. Vol. I, MIT Press, Cambridge, Mass.
- Monin, A.S. and Yaglom, A.M. (1975): Statistical Fluid mechanics. Vol. II, MIT Press, Cambridge, Mass.
- Moores, W.H., Caughey, S.J., Readings, C.J., Milford, J.R., Mansfield, D.A., Abdulla, S., Guymer, T.H. and Johnston, W.B. (1979): Measurements of boundary layer structure and development over SE England using aircraft and tethered balloon instrumentation. *Quart. J. Roy. Meteor. Soc.*, **105**, 397-422.
- Moores, W.H. (1977): Direct measurements of boundary layer radiative-flux divergences. Met. Office unpublished technical note (TDN 85).

- Rayment, R. and Readings, C.J. (1974): A case study of the structure and energetics of an inversion. *Quart. J. Roy. Meteor. Soc.*, **100**, 221-233.
- Wieringa, J. (1993): Representative roughness parameters for homogeneous terrain. *Bound. Layer Meteor.*, **63**, 323-363.