CSDL Informal Technical Note No. 5

RESTORATION OF DELAWARE RIVER AND BAY CIRCULATION SURVEY, CURRENT METER AND CTD OBSERVATIONS 1984-1985: COMPUTER PROGRAMS AND DOCUMENTATION

Silver Spring, Maryland
July 2006

U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration
National Ocean Service
Coast Survey Development Laboratory

CSDL Informal Technical Note No. 5

RESTORATION OF DELAWARE RIVER AND BAY CIRCULATION SURVEY, CURRENT METER AND CTD OBSERVATIONS 1984-1985: COMPUTER PROGRAMS AND DOCUMENTATION

Philip H. Richardson
Richard A. Schmalz, Jr.

July 2006

U.S. DEPARTMENT OF COMMERCE

National Oceanic and Atmospheric Administration

National Ocean Service
Coast Survey Development Laboratory

NOTICE

CSDL Informal Technical Notes present work in progress or summaries of results that are not appropriate to be published as either formal NOAA Office of Coast Survey Technical Reports or the less formal Technical Memoranda. Results are intended primarily for use within CSDL. Scientific review of the material is minimal, and CSDL makes no warranty as to its validity or completeness.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF COMPUTER PROGRAM LISTINGS

ABSTRACT

The purpose of this report is to document both the CTD and CT/Current analysis programs used to quality control and analyze the NOS Delaware River and Bay Circulation Survey data collected during 1984-1985. Two computer programs were developed to analyze CTD data. The first program was used to plot station locations, while the second program was used to plot vertical profiles from both AML and GRUNDY datasets. Based on a review of the plotted profiles, selected profiles were edited or discarded as described by Loeper (2006). Two additional computer programs were developed to analyze CT and current data. The first program was developed to plot salinity, temperature, current speed and direction at CT/Current moorings referenced in the NOS Delaware River and Bay Circulation Survey Report (1986). Temperature, salinity and current speed and direction spikes were filtered out of the record. Next current direction data were further clipped. A second program was developed to reformat current data and to determine the principal component direction using the Preisendorfer scheme after Zervas (2002). Control files and input data files necessary for use in the NOS 29-day harmonic analysis programs as described by Zervas (2002) were then produced. A script utilizing the 29 day harmonic analysis program was then executed to perform the harmonic analysis at all current meter locations with sufficient data length.

Herein each program is described and example input/output files are presented along with the shell script required to run the program. Sample plots are also presented to illustrate program function. Major new subroutines are listed and described in the report, while in Appendix A program location, data files, and control files are provided for more detailed study.

1. INTRODUCTION

From March 1984 through June 1985, the NOS conducted a circulation survey in Delaware River and Bay (1986). To support the Model Evaluation Environment Project within NOS's Coast Survey Development Laboratory and the future development of the NOS Delaware River and Bay Nowcast/Forecast System both the CTD and CT/Current data were quality controlled and analyzed.

Prior to the analysis, certain data irregularities were encountered. The *dos2unix* utility on CBBAY (Linux) was used to remove the DOS (^M) carriage control characters in the CTD datasets. In addition, it was necessary to develop a program called *oneline.f* using direct access reads to separate the current data into individual records.

This report describes each analysis program and provides example input/output files along with the shell script required to run each program. Sample plots are also presented. In Chapter 2, the two CTD programs developed to plot station locations and vertical profiles for both AML and GRUNDY datasets are described. In Chapter 3, two additional programs used to analyze CT and current data are presented. Salinity, temperature, current speed and direction data at CT/Current moorings are plotted. Temperature, salinity, and current speed and direction spikes were filtered out of the record. Next current direction data were further clipped. Current data were then reformatted and the principal component direction was determined using the Preisendorfer scheme after Zervas (2002). Control files and the input data file necessary for use in the NOS 29-day harmonic analysis programs as described by Zervas (2002) were then generated. In Chapter 4, the script executed to perform the harmonic analysis at all current meter locations with sufficient data length is presented. In Chapter 5, conclusions and recommendations for future work are advanced. Major new subroutines are listed and described in the report, while in Appendix A program location, data files, and control files are provided for further reference.

2. CTD ANALYSIS PROGRAMS

Two programs were used for the analysis and quality control of CTD data from the 1984-85 NOS circulation survey of Delaware River and Bay (Klavens et al., 1986). Program Gr_delctd.f was developed to plot CTD station locations. Program Del_ctd84.f was used to plot profiles of temperature, salinity, and sigma-t (density). This program will also quality control the data as discussed by Loeper (2006). Both programs are written in FORTRAN (lf95) and make use of NCAR graphic routines.

2.1 Program Gr_delctd.f

Program Gr_delctd will plot CTD station locations from Delaware Bay, River, and beyond the Bay on the continental shelf. Program Gr_delctd.f reads the following information from the control file : *idebug* is the debug switch, and *fname* is the file containing the coastline data for Delaware Bay. The program reads *p1, p2, p3, and p4*, where *p1* and *p3* are the southern and northern limits of latitude, while *p2* and *p4* are the eastern and western limits for longitude, respectively. Other variables which are read include *nlt* and *nln*; the number of latitude markers (tick marks) and the number of longitude markers, respectively. *Blat* is an array which holds the latitudes of the latitude markers, while *blon* holds the longitudes of the longitude markers. *Flatlon* is the filename of the data file containing the CTD profiles.

Program Gr_delctd first opens file *flatlon*. The program then reads tape information. *Nofiles* is the number of physical files, *tapenum* is the tape number, *tinstra* is the instrument type. These variables are not critical to our evaluation.

The 500 loop starts at nc=1, and runs through *ncast,* the number of CTD casts. The first information read from *flatlon* is *istanum*, the station number. Next read are the year, Julian day, hour, and minute. The latitude and longitude data are read in degrees, minutes, and seconds, all as integer values. *Deg_lat* and *deg_lon* are the latitude and longitude values converted to decimal form. These values are stored in arrays *alat(ncast)* and *alon(ncast)*. Finally, the CTD data are read. From each line of data the depth (pressure), temperature, conductivity, salinity, and density are read. The program picks up the null value, 9999, which indicates the end of that particular cast profile.

Program Gr_delctd.f calls Subroutine Precon to pre-condition the coastline data. The program does not call a separate subroutine to create the actual plot. All calls to NCAR graphics routines are made in the main program.

Subroutine Maptrn converts the plot limit coordinates (lat, lon) to Mercator projection coordinates. The call is then made to Subroutine Map, which actually plots the coastline data. The 400 loop iterates from l=1 through *ncast.* A call to NCAR graphics Subroutine PWRIT writes the station name, which in this case is a number, to the plot.

Input and job control files for Program Gr_delctd.f are given in Table 2.1.1 with a sample plot of the CTD station locations given in Figure 2.1.

Table 2.1.1 Input and Job Control Files for Program Gr_delctd.f

graph.1e.n *

```
0     idebug
/disks/NASUSER/philr/delaware/coastline/db.sl
38.0  -76.  40.0  -73.
    3    4
 38.0
 39.0
 40.0
 -73.0
 -74.0
 -75.0
 -76.0
/disks/NASWORK/drops/hydrodata/NOS/CTD/DECT01.r
```

graph.jcl

```
lf95 gr_delctd.f precon.f maptrn.f mapgr.f sub.f -o graph -
L/usr/lib/gcc-lib/i386-redhat-linux/3.2 -lg2c \
  -L/usr/local/ncarg/lib -lncarg -lncarg_gks -lncarg_c -
L/usr/X11R6/lib/ -lX11 \
  -I/usr/local/include -L/usr/local/lib -lnetcdf
  rm -f *.o

  ./graph < graph.1e.n > out.1e

  ctrans -d ps.mono gmeta > ps.1e

  rm fort.*
  rm graph
  rm gmeta
```

* In all, 11 CTD transects were performed. So the naming convention is graph.*e.n, where * is 1 – 11.

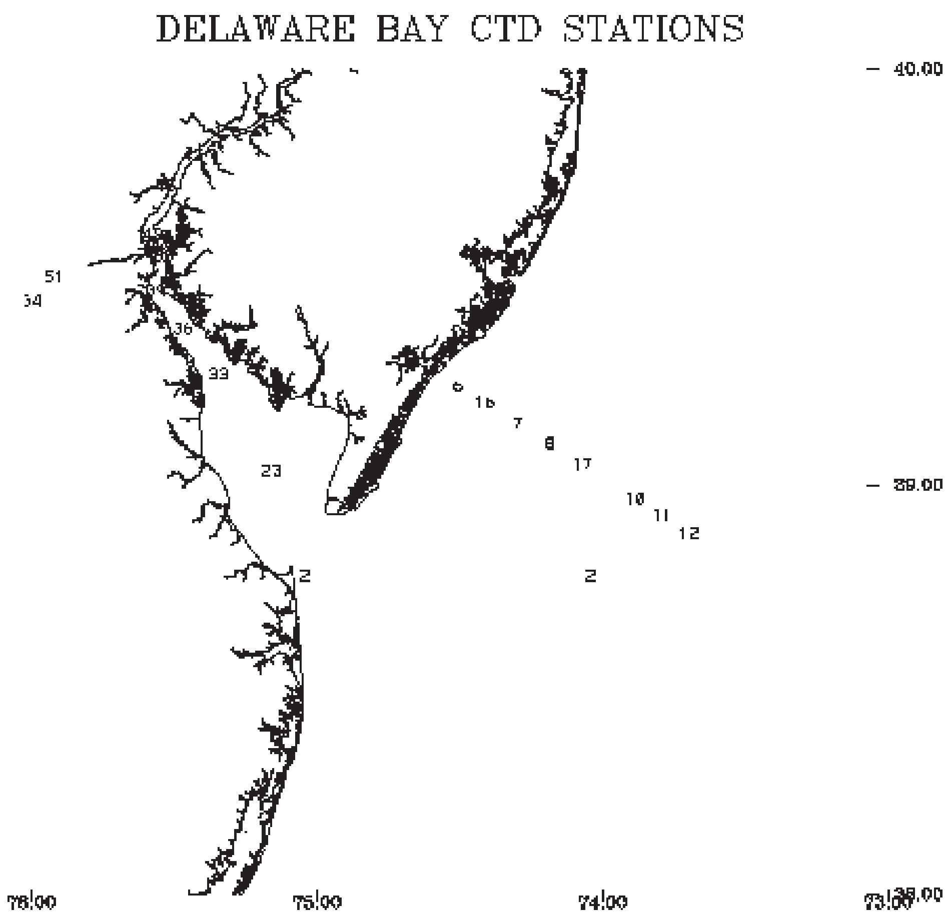

Figure 2.1 CTD Station Locations

2.2 Program Del_ctd84.f

This program plots a profile of temperature, salinity, and sigma-t from the given CTD cast. This program was originally written by John Cassidy for the Long Island Sound Study (Schmalz et al., 1994). The program was modified by Phil Richardson for use in assessing the vertical density structure in the Houston Ship Channel (Schmalz, 2000). Presently, the program is being used to plot CTD data profiles from the 1984-85 NOS circulation survey of Delaware River and Bay (Klavens et al., 1986). Input and job control files are given in Table 2.2.1.

Program Del_ctd84.f first reads variables from the control file. *Idebug* is the debug switch, *file_in* is the CTD data file, and *file_out* is an output file. Next read is *des_valrange,* which is a one character string variable, y or n. If *des_valrange* is equal to "y", the program reads the plot axis limits for sigma-t, temperature, and salinity. Next read is *doff*, which is the top and bottom cast depth boundary offset. If *doff* = 0.25, any data point at a depth less than 0.25m, or within 0.25m of the cast depth, is omitted from the quality controlled data file the program generates. If *des_valrange* is equal to "n", the program will call Subroutine Hlrange to determine the scale ranges for the plot.

After the CTD data file is opened, information regarding the tape is read. *Nofiles* is the number of physical files on the tape. *Tapenum* is the tape number. *Projloc* is the project location, and *tinstr* is the instrument type. This information is not needed for the plotting of cast profiles.

Next read is *ncast*, which is the number of cast profiles in the data file. The 100 loop iterates from ns=1 through *ncast*. The first information from each cast is the station number, *istanum*. Next read are the year, Julian day, hour, and minute. The latitude and longitude data are read next. Values are read for the degrees, minutes, and seconds of latitude, then longitude. These are read as integer values. *Rlat* and *rlong* are arrays containing the decimal values for latitude and longitude.

Program Del_ctd84.f can read either GRUNDY or AML CTD data. In the case of AML data, the program reads depth (pressure), temperature, and conductivity. Subroutines Salin and Rho are called to calculate the salinity and density, respectively. Subroutine Salin has input arguments of pressure, temperature, and conductivity. Subroutine Rho has input arguments of the reference pressure, in situ pressure, temperature in degrees Celsius, and salinity. In the case of GRUNDY data, the program reads depth (pressure), temperature, conductivity, salinity, and sigma-t (density). Even though salinity and sigma-t are read from the data, the program calls Subroutines Salin and Rho to calculate salinity and density. These calculated values are then compared with the values read from the data. Values of depth, temperature, conductivity, salinity, and density are then stored in arrays.

The program creates an output file, unit 20, which contains the quality controlled profile data with format identical to the original CTD profile data. The difference between the original data file, and the quality controlled data file, is the omission of points at a depth either greater than the cast depth – *doff*, or at a depth less than *doff* from the surface.

All calls to NCAR graphics routines are made in the main program. The plotting of all three variables is done in a similar manner. First, there is a call to Subroutine Set. Then a call to Subroutine Curve, which plots the curve representing the data points. Calls to Subroutines Labmod and Gridal add the tick marks and the labeling to the plot.

A typical CTD cast profile is depicted in Figure 2.2. This cast occurred on April 29, 1985, at CTD Station 23. As can be seen in Figure 2.2, Station 23 lies approximately in the center of the Delaware Bay. This cast profile indicates a well mixed vertical structure. There is no evidence of significant stratification.

Table 2.2.1 Input and Job Control Files for Program Del_ctd84.f

cntrl.1 *

```
3    idebug
/disks/NASUSER/tloeper/delaware/ctd.1/Final/DECT01.rework_station
DECT01.final
 y
 -1.0 40.0    density min/max
 -1.0 40.0    temperature
 -1.0 40.0    salinity
  0.25         top and bottom cast depth boundary offset
```

delprof.jcl

```
  n=$1
  rm -f DECT0$n.final
  l='/disks/NASWORK/drops/programs/profile.f'

  lf95 del_ctd84.f $l/jdconv.f $l/prof.sub.f $l/hlrange.f $l/salin.f
$l/rho.f -o delprof \
  -L/usr/lib/gcc-lib/i386-redhat-linux/3.2 -lg2c \
  -L/usr/local/ncarg/lib -lncarg -lncarg_gks -lncarg_c -
L/usr/X11R6/lib/ -lX11 \
  -I/usr/local/include -L/usr/local/lib -lnetcdf

  rm *.o

  ./delprof < cntrl.$n > out.$n.cast

  mv fort.66 debug.$n
  mv fort.67 station.$n
# rm station.*
# rm out.*.cast

# ctrans -d ps.mono gmeta > gmeta.$n.ps
  rm gmeta
```

* Cntrl.1 is the control file for CTD data file number 1 of 7.

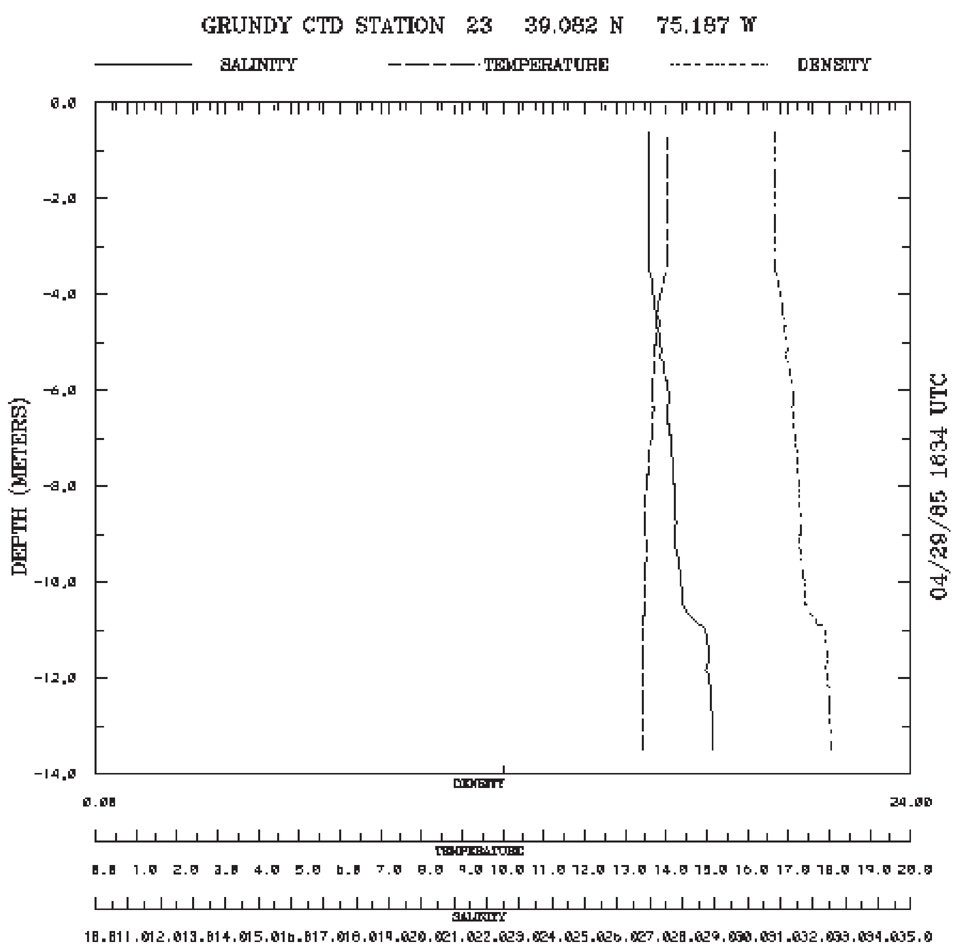

Figure 2.2 Salinity (ppt) , Temperature (^oC) , and Density (Sigma-t) Profiles

3. TIDAL CURRENT DATA ANALYSIS PROGRAMS

Two programs were used for the analysis of tidal current data from the 1984-85 NOS circulation survey of Delaware River and Bay (Klavens et al., 1986). Program Del_currnt.f was developed to plot time series data of salinity, temperature and current speed and direction. This program will also quality control the data. Program Hrm29.f will reformat the current data in preparation for harmonic analysis. It will also create the control files required by Program Harm29.f, the harmonic analysis program. Both programs are written in FORTRAN (lf95). Program Del_currnt.f makes use of NCAR graphics, while Program Hrm29.f has no graphing component.

3.1 Program Del_current.f

Program Del_currnt.f was created to plot time series data, two plots on a page, of current speed and direction (page 1) and salinity and temperature (page 2). Current speed is in cm/s, and direction is in degrees clockwise from true North. The salinity is plotted as parts per thousand, and the temperature is plotted in degrees Celsius. The current data, from the NOS 1984-85 survey of Delaware River and Bay is formatted using the CO-OPS historical circulation survey procedures (CO-OPS, 1999). Complete input data for Program Del_currnt.f are shown in Table 3.1.1 with the job control file given in Table 3.1.2. In addition to generating plots, Program Del_currnt.f will generate two output files, time.out and time.out2. Time.out includes information on each segment of data including station number, depth, start time, and stop time. Time.out2 (refer to Table 3.1.3 for a sample) includes the same information, but only for those segments of data greater than 29 days. A segment of current data must be at least 29 days in order to run the harmonic analysis program, Program Harm29.f.

Program Del_currnt.f calls Subroutine Filt, which filters the raw data (salinity, temperature, current speed, and current direction). Subroutine Filt (filt.f), as shown in Computer Listing 3.1.1, seeks to remove data outliers, by first splitting the 10 minute time series (x) into segments of 14.5 days. Within each segment, the mean (*xbar*) is first computed. Next, the data are normalized and the associated probability distribution is calculated. The *xmin* and *xmax* values are computed such that the probability of an x value less than *xmin* is less than 0.025 and the probability of x greater than *xmax* is 0.025. The standard deviation, *xstd*, is computed based on a normal distribution with *xstd=0.5(xl +xu)*, where *xl=(xbar-xmin)/1.96* and *xu=(xmax-xbar)/1.96*. Next, revised *xmin* and *xmax* values are computed such that *xmin=xbar -3*xstd* and *xmax=xbar+3*xstd*. A given point x is considered an outlier if it is less than *xmin* or greater than *xmax*. For each outlier point, the neighboring three points less than and three points greater than in time are considered. The first non-outlier is used to replace the outlier point in proceeding from the third point less to third point greater in time of the outlier. If none of the six neighboring points is a non-outlier, the outlier is not modified. While for the present time series data all outliers were removed, it may be necessary to slightly modify this procedure for other time series.

Clipping is performed for wind directions only. The variables read from input, for clipping to occur, include *nclip*, the number of stations, *stan_clip(nclip)*, station name, *dirl*, and *diru*. *Dirl* is the lower level of the clip, while *diru* is the upper level. Points which are outliers, that is either greater than *diru* or less than *dirl*, are set equal to the value of *diru* or *dirl*. Within the n=1, *nclip* loop are read *stan_clip(n)* and *ndepth(n)*. *Stanclp(n)* is the station name and *ndepth(n)* is the number of depths at that station. Within the k=1, *ndepth(n)* loop are read *stan_dep(k)*, *dirl(n,k)*, and *diru(n,k)*. *Stan_dep(n,k)* is the depth level in feet. *Dirl* and *diru* are the lower and upper clip levels, respectively.

Other variables read from input include *initplot* and *idebug*. The program normally will produce plots of the filtered (quality controlled) data as shown for Station 3 at 15 feet above the bottom in Figure 3.3 and 3.4 for salinity and temperature, repectively. If *initplot* = 1, plots of the unfiltered data will be generated as well as shown in Figures 3.1 and 3.2, for salinity and temperature, respectively at the same station depth. These figures are placed in the present order to enable comparison of raw versus filtered time series. One notes that the spikes just after Julian day 304 in 1984 have been removed by the filtering procedure. There are a number of debug options : *idebug* equals 1, station information; *idebug* equals 2, temperature and salinity data; *idebug* equals 3, speed, direction, and conductivity data; *idebug* equals 4, plot data; *idebug* equals 5, time information; *idebug* equals 6, echo data. *Fnamei* is the unfiltered current data file. *Fnameo* is the quality controlled output current data file.

After *fnamei* is opened, tape information is read. *Nofiles* is the number of physical files. *Tapenum* is the tape number. *Tinstr* is the instrument type. This information, with the exception of *tinstr*, is not necessary for the analysis of current data.

The program goes on to read station information including station number, and then reads the year, Julian start day, start hour, start minute, start second, and the time interval between data points in minutes. The latitude and longitude values are read for degrees, minutes, and seconds, as integer values.

The actual hydrographic data are then read. Values are read for temperature, conductivity, salinity, pressure, current direction, and current speed. After the necessary conversions are made, the values are stored in arrays *tdra3* (for time) and *xdra3* (for all physical quantities). *Xdra3* is a three-dimensional array of (*ns, lgr, 1-6*), where *ns* is the station number, *lgr* is the specific data point, and *1-6* is for data type.

Table 3.1.1 Input File for Program Del_currnt.f

currnt.n

```
0  0      initplot, idebug
/disks/NASUSER/philr/delaware/parallel/current/current14.r
/disks/NASUSER/philr/delaware/parallel/current/filt/current14.f
   33  nclip, number of stations
001    2
     5 130.0  0.0
    63 116.0  0.0
002    3
     5 130.
    22 144.
    47 130.
003    2
     5 144.
    15 144.
004    1
     6 130.
005    2
     5 130.
    15 116.
018    1
     6 122.
019    2
     5 144.
    26 144.
021    1
     5 130.
022    2
     5 137.
    20 137.
023    2
     8 144.
    26 158.
023X   2
     5 130.
    26 144.
024    1
     7 144.
025    1
     5 180. 208.
030    1
    27 144.
032    1
     5 144.
033    2
     5 116.
    36 130.
033X   1
    15 130.
041    1
    14 113.
042    1
    17 151.
043    1
    19  28. 230.
```

```
047    1
   28  58. 252.
049    1
   18 165. 190.
051    3
    5  86. 278.
    8  81. 280.
   28  86. 280.
052    1
   14  35. 216.
058    1
   16 132.
059    1
   35 158.
060    1
   30 158.
061    1
   13 144. 339.
134    1
   10  51. 237.
151    1
   25  51. 237.
152    1
   25  51. 237.
154    2
    5  72. 260.
   25  58. 274.
155    1
    7   0. 195.
```

Table 3.1.2 Job Control File for Program Del_currnt.f

current.jcl

```
  lf95 del_currnt.f plt_del.f jdconv.f filt.pp.f ncrght.f -o delcurrnt
-L/usr/lib/gcc-lib/i386-redhat-linux/3.2 -lg2c \
  -L/usr/local/ncarg/lib -lncarg -lncarg_gks -lncarg_c -
L/usr/X11R6/lib/ -lX11 \
  -I/usr/local/include -L/usr/local/lib -lnetcdf
  rm *.o
  delcurrnt < currnt.003n > out
# rm current14.f
  rm current14.0
# convert gmeta file to postscript
# ctrans -d ps.mono gmeta > gmeta.ps
  ictrans -d ps.mono -e "5 save 5frame" -e 'quit' gmeta
  ctrans -d ps.mono 5frame > gmeta5.ps
  rm delcurrnt
  rm time.out*
# rm gmeta
```

Table 3.1.3 Time.out2 : Output File (for records of current data greater than 29 days)

```
     1
tape number CURDAT
instrument type : GRUNDY
Station number 001 ,  Depth =  5ft
 record starts at time  94.986, and ends at time 127.521
                       4/ 3/84    to    5/ 6/84
Length of record is  32.535 days
Station number 001 ,  Depth = 63ft
 record starts at time  94.986, and ends at time 127.514
                       4/ 3/84    to    5/ 6/84
Length of record is  32.528 days
Station number 002 ,  Depth =  5ft
 record starts at time 115.862, and ends at time 173.702
                       4/24/84    to    6/21/84
Length of record is  57.840 days
Station number 002 ,  Depth =  5ft
 record starts at time 198.590, and ends at time 255.729
                       7/16/84    to    9/11/84
Length of record is  57.139 days
```

Program Listing 3.1.1 Subroutine Filt

```
      subroutine filt(numt,itype)

******************************************************

      parameter (nprob=1001,pout=.025,fstd=3.0,ndays=14.5)
      parameter (numd=144*ndays,nint=3,ntot=25000)

      dimension ifilt(numd),pdf(numd),xn(numd)

      common/filter/x(ntot),y(ntot)
      common/debug/idbug

******************************************************

      nseg=numt/numd + 1
      num=numd
      numf=numt-(nseg-1)*numd

      if(idbug.eq.7)then
         write(6,*)'itype numt nseg num numf: ',
     *               itype,numt,nseg,num,numf
      endif

      do nn=1,nseg
         if(nn.eq.nseg)num=numf
         is=(nn-1)*numd

         sum=0.
      do i=1,num
        sum=sum+x(is+i)
  enddo
  xbar=sum/float(num)
  sum=0.
         do i=1,num
            sum=sum+(x(is+i)-xbar)*(x(is+i)-xbar)
         enddo
         xstdo=sqrt(sum/float(num-1))
         if(idbug.eq.7)then
            write(6,*)'itype: ',itype
            write(6,*)'xbar xstdo: ',xbar,xstdo
         endif
         do n=1,nprob
            pdf(n)=0.
         enddo
  xnorm=0.
  do i=1,num
  xnorm=amax1(x(is+i),xnorm)
  enddo
      if(idbug.eq.7)write(6,*)'xnorm: ',xnorm
  do i=1,num
  xn(i)=x(is+i)/xnorm
  ibin=xn(i)*1000. + 1
  pdf(ibin)=pdf(ibin) + 1
```

```
      enddo
      denom=float(num)
      sum=0.
          do i=1,nprob
             pdf(i)=pdf(i)/denom
             sum=sum+pdf(i)
          enddo
      sum=0.
      do n=1,nprob
      sum=sum + pdf(n)
      if(sum .ge. pout)then
      nmin=n
      go to 1
      endif
      enddo
    1 xmin=float(nmin-1)/1000.*xnorm
      sum=0.
      do n=1,nprob
      sum=sum + pdf(n)
      if(sum .ge. 1.-pout)then
      nmax=n
      go to 2
      endif
      enddo
    2 xmax=float(nmax-1)/1000.*xnorm
          if(idbug.eq.7)then
             write(6,*)'nmin nmax: ',nmin,nmax
             write(6,*)'xmin xmax: ',xmin,xmax
          endif
      xstdl=(xbar-xmin)/1.96
      xstdu=(xmax-xbar)/1.96
      xstd=0.5*(xstdl+xstdu)

      xmin=xbar-fstd*xstd
      xmax=xbar+fstd*xstd
      if(itype.eq.4)then
      xmin=amax1(0.,xmin)
      xmax=amin1(360.,xmax)
          endif
          if(idbug.eq.7)then
             write(6,*)'xmin xmax: ',xmin,xmax
          endif

          do i=1,num
             ifilt(i)=0
             if(x(is+i) .gt. xmax .or. x(is+i) .lt. xmin)ifilt(i)=1
             if(ifilt(i).eq.0)y(is+i)=x(is+i)
          enddo

      do 3 i=1,num
      if(ifilt(i).eq.1)then
      do ii=i-nint,i+nint
      if=min(num,ii)
      if=max(1,if)
      if(ifilt(if).eq.0)then
      y(is+i)=x(is+if)
          if(idbug.eq.7)then
```

```
          write(6,*)'ifilt x y: ',ifilt(i),x(is+i),y(is+i)
        endif
   x(is+i)=x(is+if)
   ifilt(i)=0
   go to 3
   endif
   enddo
   endif
3  continue
        do i=1,num
          if(ifilt(i).eq.1)y(is+i)=x(is+i)
        enddo

        if(idbug.eq.7)write(6,*)'is + num: ',is+num
        enddo
        return
        end
```

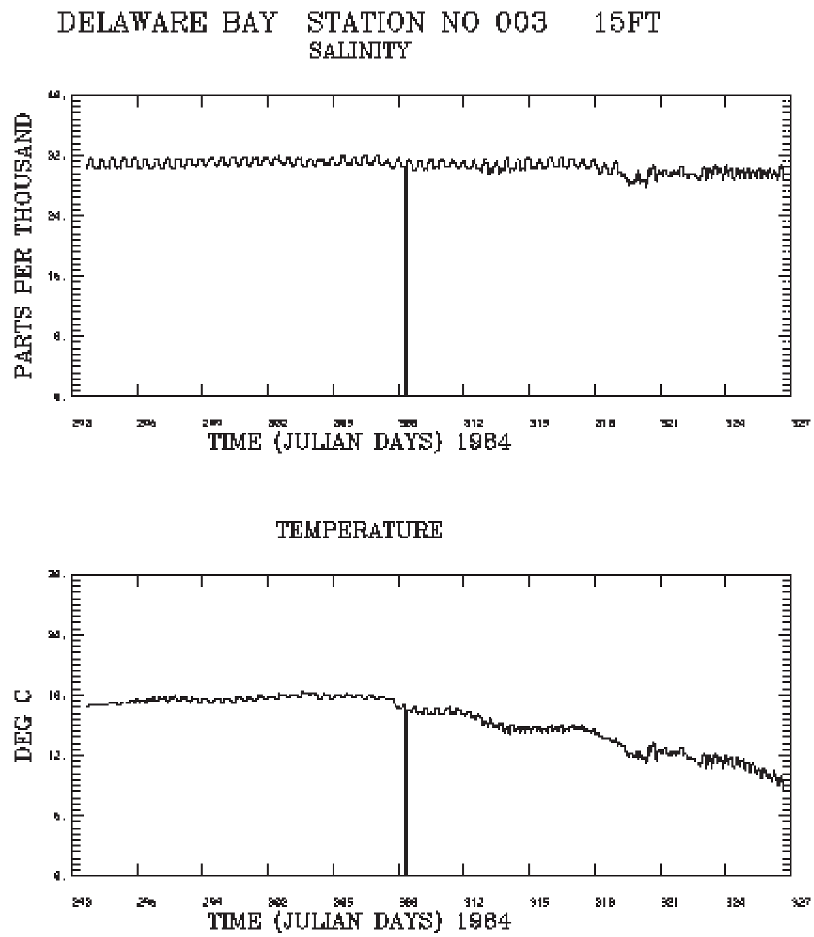

Figure 3.1 Observed (raw) Salinity and Temperature

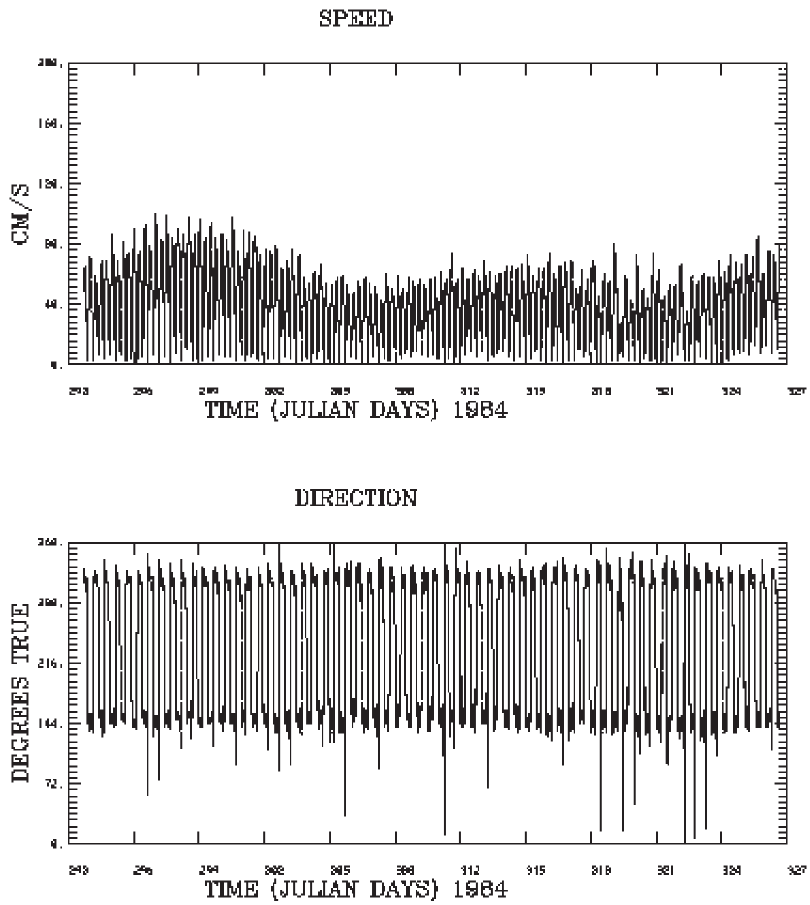

Figure 3.2 Observed (raw) Current Speed and Direction

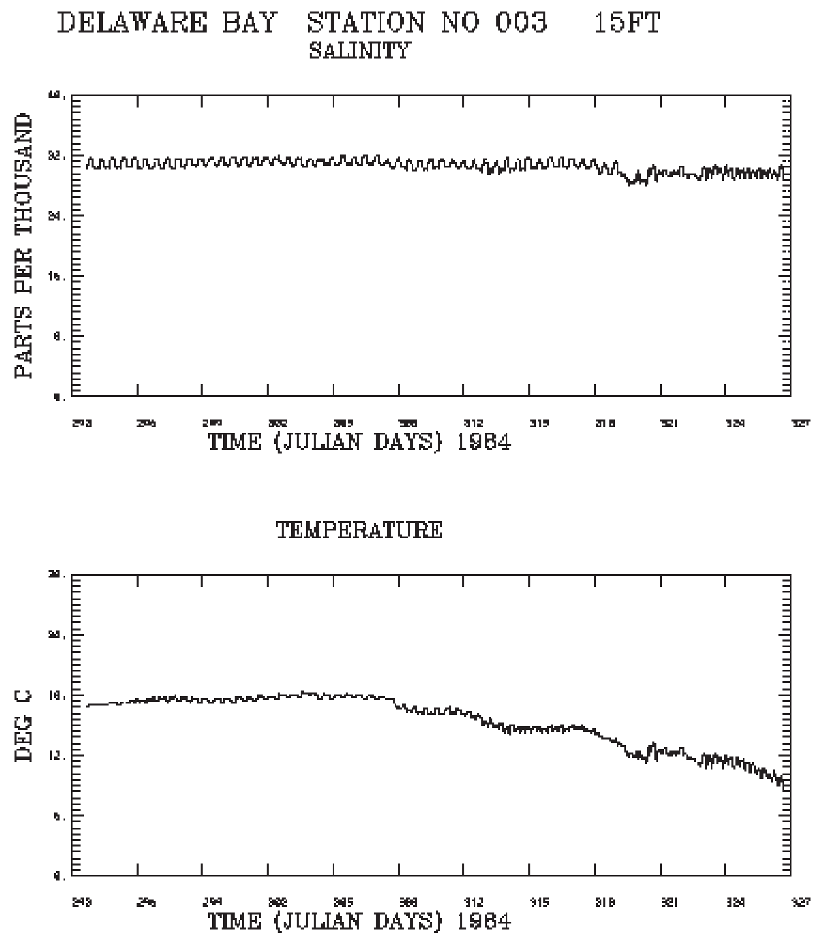

Figure 3.3 Filtered Salinity and Temperature

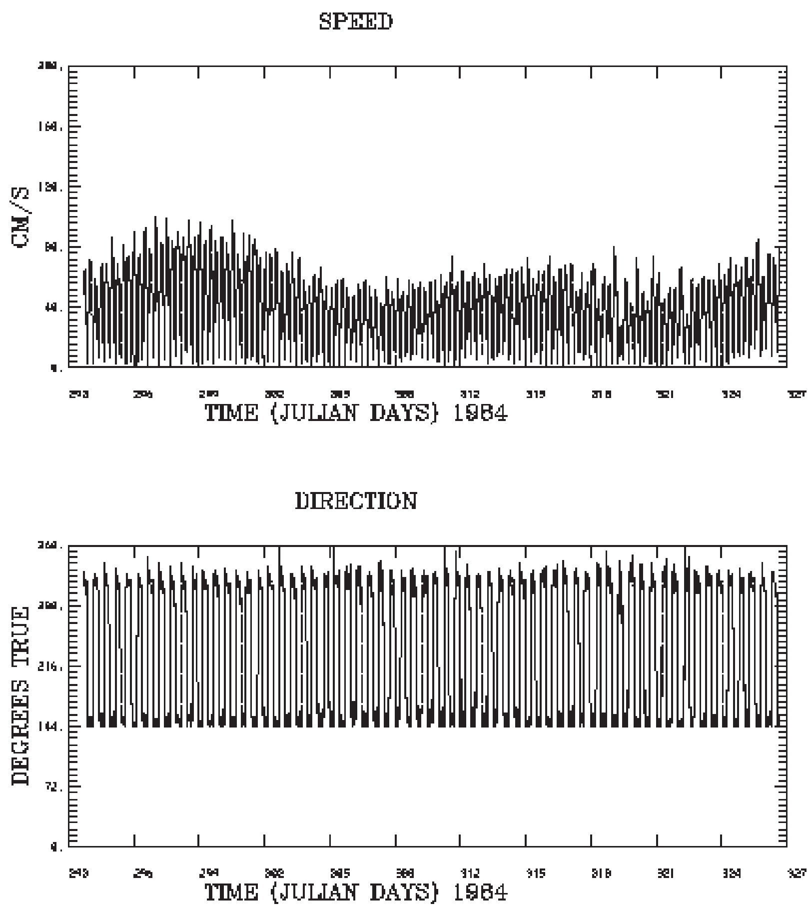

Figure 3.4 Filtered Current Speed and Direction

3.2 Program Hrm29.f

Program Hrm29.f will reformat tidal current data, either the raw current data, or the filtered output from Program Del_currnt.f, in preparation for the harmonic analysis program, Program Harm29.f. Input and job control are given in the first two sections of Table 3.2.1. A separate data file is created for each depth at each station. The data output filenames are of the form data.(station name).(depth).(Julian start day).out. The format of the output data is 13x, f6.3, f8.3, where the two fields of real values are filled with current speed and direction as shown in Table 3.2.2. Program Hrm29.f also creates control files used by Program Harm29.f; a sample control file is shown in the third section of Table 3.2.1. The naming convention for the control files is the same as that of the data files except that the word control is substituted for the word data. See section three of Table 3.2.1 for a sample control file for Station 33X, 15 feet above the bottom, starting at Julian Day 71, which is next discussed in detail.

The control file for Program Harm29.f includes, as documented by Zervas (1999), on page 30 for Program Harm29.f documentation, on line one, the data filename of the reformatted current data. Program Harm29.f is run separately for each individual station and each individual segment of data. Hence, there will be numerous data files created, and an equal number of control files. Line two includes values for the Program Harm29.f variables *nj, iand, nsph, iref*, and *iit*. *Nj* is the number of harmonic analyses to perform, in this case one. If *iand* is set to 0, *fname* is an ASCII file. *Nsph* is the number of samples per hour, in this case six (10 minute data). When *iref* is set to 0, Program Harm29.f will determine inferred amplitude and phase from ratios of amplitudes at known stations. *Iit* equals 0 indicates the number of iterations to carry out to eliminate perturbations between nearby constituents.

Line 3 of the control file created will include *azi, tm, gonl, cvar*, and *iel*. *Azi* is the azimuth of the major axis, flood direction, for current data. *Tm* is the time meridian, in this case 75.0, for Eastern Standard Time. *Gonl* is the longitude of the station in decimal degrees. For Station 33X, *gonl* is equal to 75.31555. *Cvar* is the direction of magnetic north, 0.0. Iel equal to 0 indicates that the data to be read in is speed and direction.

Line 4 of the control file provides the data format of the reformatted data file (for a sample file see Table 3.3.2). Included in line 5 are the year, start month, start day, and *STT* (start hour of first data point). *STTM* is the minute of the first data point, and *iskip9* is the number of data points to skip at the start of the file.

Program Hrm29.f reads from input *idebug*, fname, and tm. *Idebug* controls the debug option. There are a number of debug options including :

idebug = 1, write U and V values
idebug = 2, output from Subroutine principal
idebug = 3, output from Subroutine distr
idebug = 4, time information
idebug = 5, echo all data read.

Fname is the filename of the tidal current data. *Tm* is the time meridian, needed for harmonic analysis.

Similar to Program Del_currnt.f, Program Hrm29.f first opens the current data file, then reads tape information. *Nofiles* is the number of files, *tapenum* is the tape number, and *tinstr* is the instrument type. These variables are not critical to our analysis. Next read is station information, beginning with the station name as a character string of four. The program then reads the year, Julian start day, start hour, start minute, start second, and the time interval in minutes. Since the data is in GMT, and our analysis is performed in local time, the start hour must be adjusted by subtracting five hours. Next read is the latitude and longitude data in degrees, minutes, and seconds. These are read as integer values. Program Hrm29.f then reads the actual data. The quantities read include temperature, conductivity, salinity, pressure, direction, and current speed. For harmonic analysis, all that is needed is current speed and direction. Once the conversions are performed, speed is stored in the array *uterm* and direction is stored in *vterm*.

Subroutines Principal and Distr are called by Program Hrm29.f to determine the principal component current direction. Subroutine Distr makes use of a probability distribution method, while Subroutine Principal uses the Preisendorfer method.

Input arguments for Subroutine Distr include *lgr*, the number of data points, and vterm, the current directions for all data points. Subroutine Distr determines a probability frequency, and from this the principal component direction in the flood direction, *azip*, is returned.

The program then calls Subroutine Principal, whose listing is provided in Computer Listing 3.2.1. This subroutine makes use of the Preisendorfer method, which involves some statistical calculations to determine the principal component direction. The input arguments for principal include *lgr*, the number of points, *u_bar* and *v_bar*, the mean U and V current speeds, respectively. The inverse tangent function is used to calculate the principal angle, *theta* (line 54). *Beta*, which is the principal component direction, is then calculated by subtracting *theta* from 90 degrees. The routine then calculates the variance about the major axis and the minor axis (lines 94 and 97). S_{11} is the variance about the major axis, while s_{22} is the variance about the minor axis. The percentage of the variance about both the major axis and about the minor axis is then calculated (lines 103, 104). If the percentage of the variance about the major axis is less than 80% of the total variance, we conclude that the current is rotary in nature (not rectilinear). In this case, we set *beta* equal to 0.0. The return argument from principal is *azi*, short for azimuth, the principal component direction. We presently use *azi* rather than *azip* of the previous paragraph, but usually both are nearly equal.

Program Hrm29.f can be run using either the raw current data, or the filtered (quality controlled) tidal current data, which was output from Program Del_currnt.f. The example presented in Table 3.2.2, from Station 33X, makes use of the filtered data file, current14.f. The unfiltered (raw) data is in file current14.r. When harmonic analysis is performed on both the raw data, and the quality controlled data, the results are generally quite close.

Table 3.2.1 Input, Job Control, and Sample ha Control File for Harm29.f

harm.f.n

```
0      idebug
/disks/NASUSER/philr/elaware/parallel/current/filt/current14.f
 75.0   time meridian
```

harm.f.jcl

```
#!/bin/sh
set -x
  date
  date > harm.f.out
  lf95 hrm29.f jdconv.f uvcomp.f principal.org.f distr.f ncrght.f -o
currnt
  currnt < harm.f.n >> harm.f.out
  rm -f currnt *.o
  date >> harm.f.out
  date
# rm -f control.*.*.out
# rm -f data.*.*.out
```

control.033X .15.071.out

(Station 33X, 15 feet above bottom, starting at Julian Day 71)

```
data.033X.15.071.out
    1    0    6    0    1
 324.    75.   75.31555    0.    0
(11x,2f8.3)
```

1985. 3 12 9 35.000 0

Program Listing 3.2.1 Subroutine Principal

```
      subroutine principal(lg,ubar,vbar,nc_trm,beta)

c     Input Arguments :
c           lg - number of data points
c         ubar - mean U value
c         vbar - mean V value
c
c     Output Argument :
c         beta - principal component direction

*******************************************************

      parameter (ntot=25000)

      common/uvterms/utrm(ntot),vtrm(ntot)
      common/debug/idbug

*******************************************************

      if(idbug.eq.2)write(6,12)

c Initialize cumulative totals

      suu_tot = 0.0
      svv_tot = 0.0
      suv_tot = 0.0

      do 50 i=1,lg
         suu_term = (utrm(i) - ubar)**2
         svv_term = (vtrm(i) - vbar)**2
         suv_term = (utrm(i) - ubar) * (vtrm(i) - vbar)
         suu_tot = suu_tot + suu_term
         svv_tot = svv_tot + svv_term
         suv_tot = suv_tot + suv_term
   50 continue

      lgm1 = lg - 1
      suu = suu_tot/float(lgm1)
      svv = svv_tot/float(lgm1)
      suv = suv_tot/float(lgm1)

      if(idbug.eq.2)write(6,11)suu,svv,suv


   11 format(/,'Suu = ',f8.3,' Svv = ',f8.3,' Suv = ',
     *       f8.3)
   12 format(/,'Entering subroutine Principal ....')

*******************************************************

c Using inverse tangent function, calculate the angle
c theta.  Beta is then calculated from theta_deg.

```

```
      theta = (atan2(2.0*suv,suu-svv))/2.0
      if(idbug.eq.2)write(6,13)theta
      theta_deg = theta * 57.296
      if(idbug.eq.2)write(6,16)theta_deg

      beta = 90.0 - theta_deg
      if(idbug.eq.2)write(6,14)beta

      if(beta.lt.0.0)then
         beta = beta + 360.0
      endif

      if(beta.gt.90.0.and.beta.lt.270.0)then
         beta = beta + 180.0
         if(beta.gt.360.)beta=beta-360.0
         if(idbug.eq.2)write(6,15)beta
      endif


   13 format(' theta (radians) = ',f6.3)
   14 format(/,'The principal component direction is ',
     *       f7.3,' degrees')
   15 format(/,'The principal component direction (corrected)'
     *    ' is',/,8x,f7.3,' degrees')
   16 format(' in degrees, theta = ',f7.3)

!-------------------------------------------------------

c Calculate variance about both major and minor
c axes.  S11 is the variance about the major axis,
c s22 is the variance about the minor axis.  Smaj_axis
c is the percentage of the variance about the major axis,
c while smin_axis is the percentage of the variance about
c the monor axis.


      suplsv = suu + svv
      sumnsv = suu - svv
      theta2 = 2.0 * theta

      s11 = 0.5*((suplsv) + sqrt((sumnsv)**2 + 4.0*(suv**2)))
      if(idbug.eq.2)write(6,21)s11

      s22 = 0.5*((suplsv) - sqrt((sumnsv)**2 + 4.0*(suv**2)))
      if(idbug.eq.2)write(6,22)s22

      s12 = 0.5*((svv - suu) * sin(theta2)) + suv*cos(theta2)
      if(idbug.eq.2)write(6,23)s12

      smaj_axis = s11*100.0/(s11+s22)
      smin_axis = s22*100.0/(s11+s22)
      if(idbug.eq.2)write(6,24)smaj_axis,smin_axis

c If the percentage of the variance about the major axis is
c less than 80% of the total variance, we conclude that the
c current is rotary in nature, and we set beta = to 0.0.

```

```
      if(smin_axis.gt.20.0)then
         beta = 0.0
         nc_trm = 1
      else
         nc_trm = 0
      endif

      if(idbug.eq.2)write(6,17)


   17 format('Leaving subroutine principal....',/)
   21 format(/,'S11 = ',f8.3)
   22 format('S22 = ',f8.3)
   23 format('S12 = ',f8.3)
   24 format(/,'Major axis standard deviation = ',f9.3,
     *       /,'Minor axis standard deviation = ',f9.3)

*******************************************************

      return
      end
```

Table 3.2.2 Sample Reformatted Current Data Produced by Hrm29 (Input Current Data for Harm29, Harmonic Analysis)

54.720	146.100
48.200	140.400
47.650	143.100
39.520	144.600
33.000	161.500
23.250	154.500
12.400	144.600
11.320	130.000
8.610	247.300
8.610	334.500
16.190	330.300
24.860	317.600
22.700	316.200
25.420	306.300
27.030	302.200
28.120	312.000
35.710	317.600
36.800	326.100
41.680	316.200
40.610	313.300
43.310	307.800
42.770	327.500
43.310	321.800
39.520	317.600
43.860	312.000
53.690	324.000
53.090	323.300
52.000	313.300
49.280	305.000
46.030	319.000
40.610	309.200
39.520	323.300
33.980	323.800
28.120	323.300
18.900	328.800
8.440	284.800
13.480	206.500
12.940	222.000
17.280	200.900
31.920	182.600
31.920	165.600
45.480	174.100
48.200	155.900
53.630	161.500
63.960	157.300
72.680	148.900
82.470	153.100
89.550	153.100
94.470	153.100
97.190	150.300
99.370	150.300
102.110	148.900
103.750	153.100

3.3 Script Harm29d.f.jcl

Harm29d.f.jcl is a script (refer to Program Listing 3.3.1 below) which controls how the harmonic analysis program, Program Harm29.f, is run. Program Harm29.f is run successively for each of the data sets containing greater than 29 days of data. This is accomplished in the for loop, lines 17 through 23.

The output from Program Harm29.f, run in this script format, will be a number of output files with calculated harmonic constituents in standard NOS format. The naming convention for these output files is *data.(station name).(depth).(Julian start day).out.cons*. These files, containing the calculated tidal current constituents, are concatenated into a cumulative output file, *total.f.ha*. The concatenation is performed in line 25. *Total.f.ha* contains the calculated tidal current constituents for each depth at every station, for every data set greater than 29 days. To see a sample output file, refer to Table 3.3.1.

Program Listing 3.3.1 Harmonic Analysis Job Control File (harm29d.f.jcl)

```
 1  #!/bin/sh
 2  set -x
 3  date
 4
 5  dir='/disks/NASUSER/philr/delaware/parallel/current/harm29/analysis'
 6    dirp="$dir/analysis.f"
 7
 8    lf95 harm29d.f -o $dirp/harm29
 9    cd $dirp
10
11    rm -f ha29.f.process *.cons total.f.ha
12
13    date > $dir/ha29.f.process
14    date > $dir/ha29.f.out
15
16    LIST=`ls -ct control*`
17    for F in $LIST
18    do
19    echo "${F}" >> $dir/ha29.f.process
20    $dirp/harm29 < ${F} >> $dir/ha29.f.out
21    date >> $dir/ha29.f.process
22    date >> $dir/ha29.f.out
23    done
24
25    cat *.cons > total.f.ha
26    rm -f $dirp/harm29
27    date
```

Table 3.3.1 Sample Harmonic Analysis Output File (Station 33X)

data.033X.15.071.out.cons

```
Harmonic Analysis of Data in
29-Day H.A.  Beginning  3-12-1985  at Hour  9.58  along 324 degrees
 -6442
       186427255912886282719652235310099 997 64591007 7232 920 14183184
       2    0   0 10781305    0   0 38132380  3371285    0   0 26142146
       3  3111073  6052683    0   0  513 958  5711035    0   0    0   0
       4    0   0    0   0    0   0  275 887 1403 882  7602816  1032838
       5  188 844 3343 991    0   0    0   0 28102298    0   0 35052849
       6  976 995    0   0
Harmonic Analysis of Data in
29-Day H.A.  Beginning  3-12-1985  at Hour  9.58  along  54 degrees
 -2083
       1 25351582 15683122 16812028  8942043 21592896 10681922 1464 980
       2    0   0  3311132    0   0  3261968  199 871    0   0  2242475
       3   462165   182296    0   0   761982   842104    0   0    0   0
       4    0   0    0   0    0   0   411870  2071862   933060   133184
       5   281801  2962034    0   0    0   0  2401974    0   0  4273247
       6  6863083    0   0
```

4. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

Two sets of programs have been developed to analyze the NOS 1984-1985 Delaware River and Bay Circulation Survey data. Prior to the analysis several reformatting steps were necessary to convert from DOS to UNIX file formats and to process single line files. Since these irregularities may be characteristic of all previous circulation surveys, the methods we used may be applicable in each case.

The first two programs are used to plot both AML and GRUNDY CTD vertical profiles and station locations. CTD depths are then checked with nautical chart depths to insure proper station location. Noisy CTD profiles were then edited when possible. The AML CTD files were rewritten to remove the extraneous data at the end of the casts.

The second two programs are used to analyze the CT/Current data files. The first program is used to plot and filter the salinity, temperature, and current speed and direction time series as well as further clip current directions. The second program is used to determine the principal current direction using the Preisendorfer scheme and to prepare the control and data files for use in the NOS 29-day harmonic analysis program. A script was written to allow performance of the harmonic analysis for order 100 current time series in one job. All constituent files are then combined into a single file. Harmonic analysis was performed on both the filtered and unfiltered current files and resulted in nearly the same results.

The above set of programs can be easily applied to analyze data from the other eight NOS circulation surveys. All data should be available from either CO-OPS or NODC. In addition, with minor modification, the programs should be used to plot and quality control the present Delaware PORTS data for use in model forcing and validation.

With respect to future work, it would be useful to incorporate the following analysis steps:

1) For CTD, it would be useful to develop methods to predict the entire field over depth of salinity and temperature from a distinct set of CTD profiles using the natural neighbor interpolation scheme. The predicted salinity and temperature fields will be very useful for understanding estuarine density characteristics as well as provide hydrodynamic model simulation initial conditions. In the Long Island Sound Study (Schmalz, 1994), this was performed on a monthly basis using patched bilinear interpolation.

2) For current, salinity, and temperature time series as a second step in the final harmonic analysis, it would be useful to perform spectral analysis.

3) It would also be useful to assess the ability of predictions of current at one location from both observed currents and water levels at another location or from water levels at the same location.

5. REFERENCES

1. Loeper, Thomas, 2006. Analysis of CTD Data from the 1984-1985 Delaware River and Bay Circulation Survey, **NOAA Technical Memorandum, NOS CS**, Silver Spring, MD.

2. Klavans, A. S., P.J. Stone, and G.A. Stoney, 1986. Delaware River and Bay Circulation Survey: 1984-1985, **NOS Oceanographic Circulation Survey Report No. 9**, Rockville, MD.

3. CO-OPS, Information Systems Division, 1999. CO-OPS Historical Circulation Survey Data (1973-1985): Inventory and General Users Guide, **NOS CO-OPS, Information Systems Division, Internal Report**, Silver Spring, MD.

4. Schmalz, R.A., 1994. Long Island Sound Oceanography Project Summary Report, Volume 1: Application and Documentation of the Long Island Sound Three-Dimensional Circulation Model, **NOAA Technical Report NOS OES 003**, Silver Spring, MD.

5. Schmalz, R.A., C.L. Sun, and E.J. Wei, 1994. Long Island Sound Oceanography Project Summary Report, Volume 3: Scientific Publications, **NOAA Technical Report NOS OES 003**, Silver Spring, MD.

6. Schmalz, R.A., 2000. High Resolution Houston Ship Channel ADCP and CTD Survey: Model-Data Intercomparisons, **NOAA Technical Report NOS CS 6**, Silver Spring, MD.

7. Zervas, C. E., 1999. Tidal Current Analysis Procedures and Associated Computer Programs, **NOAA Technical Report NOS CO-OPS 0021**, Silver Spring, MD.

APPENDIX A. DIRECTORY STRUCTURE

A sample directory structure is given for performing the CTD and CT/current analysis work. Each program is contained and executed in separate directories for clarity.

```
drwxrwxr-x    7 philr    philr         8192 Jan 20 17:21 .
drwxrwxr-x    5 philr    philr         4096 Jan 20 20:13 ..
drwxrwxr-x    2 philr    philr         4096 Jan 18 19:31 delctd84
drwxrwxr-x    2 philr    philr         4096 Jan 20 15:43 delcurrnt
drwxr-xr-x    2 philr    philr         4096 Jan 18 19:32 gr_delctd
drwxrwxr-x    4 philr    philr        12288 Jan 20 17:15 harm29
drwxrwxr-x    2 philr    philr         8192 Jan 20 15:53 hrm29

docu/delctd84:
total 56
drwxrwxr-x    2 philr    philr         4096 Jan 18 19:31 .
drwxrwxr-x    7 philr    philr         8192 Jan 20 17:21 ..
-rwxr-xr-x    1 philr    philr          231 Jan 18 18:19 cntrl.1
-rwxr-xr-x    1 philr    philr        15749 Jan 18 19:26 del_ctd84.f
-rwxr-xr-x    1 philr    philr          539 Jan 18 19:30 delprof.jcl
-rwxr-xr-x    1 philr    philr         2803 Jan 18 18:22 hlrange.f
-rwxr-xr-x    1 philr    philr         1116 Jan 18 18:22 jdconv.f
-rwxr-xr-x    1 philr    philr         3472 Jan 18 18:22 prof.sub.f
-rwxr-xr-x    1 philr    philr         2611 Jan 18 18:22 rho.f
-rwxr-xr-x    1 philr    philr         2015 Jan 18 18:22 salin.f

docu/delcurrnt:
total 56
drwxrwxr-x    2 philr    philr         4096 Jan 20 15:43 .
drwxrwxr-x    7 philr    philr         8192 Jan 20 17:21 ..
-rwxr-xr-x    1 philr    philr          538 Jan 20 15:09 current.jcl
-rwxr-xr-x    1 philr    philr         1120 Jan 20 15:09 currnt.n
-rwxr-xr-x    1 philr    philr        14639 Jan 18 19:58 del_currnt.f
-rwxr-xr-x    1 philr    philr         2619 Jan 18 19:58 filt.pp.f
-rwxr-xr-x    1 philr    philr         1119 Jan 18 19:58 jdconv.f
-rwxr-xr-x    1 philr    philr          351 Jan 18 19:58 ncrght.f
-rwxr-xr-x    1 philr    philr         6182 Jan 18 19:58 plt_del.f

docu/gr_delctd:
total 44
drwxr-xr-x    2 philr    philr         4096 Jan 18 19:32 .
drwxrwxr-x    7 philr    philr         8192 Jan 20 17:21 ..
-rwxr-xr-x    1 philr    philr          188 Jan 18 17:51 graph.1e.n
-rwxr-xr-x    1 philr    philr          355 Jan 18 18:15 graph.jcl
-rwxr-xr-x    1 philr    philr         6752 Jan 18 18:10 gr_delctd.f
-rwxr-xr-x    1 philr    philr         2840 Jan 18 18:10 mapgr.f
-rwxr-xr-x    1 philr    philr          461 Jan 18 18:10 maptrn.f
-rwxr-xr-x    1 philr    philr         2419 Jan 18 18:10 precon.f
-rwxr-xr-x    1 philr    philr         1555 Jan 18 18:10 sub.f
```

```
docu/harm29:
total 1844
drwxrwxr-x    4 philr    philr       12288 Jan 20 17:15 .
drwxrwxr-x    7 philr    philr        8192 Jan 20 17:21 ..
drwxrwxr-x    2 philr    philr        4096 Jan 20 17:14 control
drwxrwxr-x    2 philr    philr        4096 Jan 20 17:15 data
-rwxr-xr-x    1 philr    philr     1649819 Jan 20 17:15 ha29.f.out
-rwxr-xr-x    1 philr    philr        2668 Jan 20 17:15 ha29.f.process
-rwxr-xr-x    1 philr    philr       83824 Jan 20 16:02 harm29d.f
-rwxr-xr-x    1 philr    philr         451 Jan 20 17:18 harm29d.f.jcl
-rwxr-xr-x    1 philr    philr      101192 Jan 20 17:15 total.f.ha


docu/harm29/control:
total 380
drwxrwxr-x    2 philr    philr        4096 Jan 20 17:14 .
drwxrwxr-x    4 philr    philr       12288 Jan 20 17:15 ..
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0001.05.094.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0001.63.094.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.05.115.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.05.198.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.22.066.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.22.115.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.22.176.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.22.270.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.47.115.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0002.47.270.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0003.05.094.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0003.05.293.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0003.15.094.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0003.15.198.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0003.15.293.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0004.06.094.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0004.06.198.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0005.05.094.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0005.05.293.out
```

```
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0005.15.094.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0005.15.198.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0005.15.293.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0011.10.293.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0012.50.294.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0012.90.294.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0016.10.067.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0016.10.168.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0016.25.067.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0016.25.283.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0016.35.067.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0016.35.168.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0016.35.283.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0017.10.067.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0017.10.168.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0017.50.108.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0017.50.168.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0017.50.254.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0017.75.108.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0018.06.262.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0019.05.262.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0019.26.262.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0021.05.262.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0022.05.130.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0022.05.198.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0022.20.130.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0022.20.198.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0022.20.262.out
```

```
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0023.08.095.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0023.08.255.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0023.26.068.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0023.26.278.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0024.07.130.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0024.07.198.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0024.07.262.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0025.05.262.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0030.27.130.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0032.05.136.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0033.05.197.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0033.36.068.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0033.36.117.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0033.36.167.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0041.14.166.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0042.17.166.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0043.19.059.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0047.28.086.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0047.28.166.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0049.18.064.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0051.05.249.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0051.08.059.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0051.08.336.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0051.28.059.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0051.28.129.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0051.28.227.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0052.14.066.out
-rwxr-xr-x    1 philr    philr          189 Jan 20 17:10
control.0057.10.294.out
```

```
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0058.16.070.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0058.21.070.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0059.35.070.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0060.30.072.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0061.13.072.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0134.10.151.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0151.25.151.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0152.25.151.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0154.05.227.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0154.25.059.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0154.25.172.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0154.25.271.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.0155.07.166.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.023X.05.067.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.023X.26.067.out
-rwxr-xr-x    1 philr    philr         189 Jan 20 17:10
control.033X.15.071.out

docu/harm29/data:
total 18144
drwxrwxr-x    2 philr    philr        4096 Jan 20 17:15 .
drwxrwxr-x    4 philr    philr       12288 Jan 20 17:15 ..
-rwxr-xr-x    1 philr    philr      131208 Jan 20 17:11
data.0001.05.094.out
-rwxr-xr-x    1 philr    philr      131180 Jan 20 17:11
data.0001.63.094.out
-rwxr-xr-x    1 philr    philr      233240 Jan 20 17:11
data.0002.05.115.out
-rwxr-xr-x    1 philr    philr      230412 Jan 20 17:11
data.0002.05.198.out
-rwxr-xr-x    1 philr    philr      237048 Jan 20 17:11
data.0002.22.066.out
-rwxr-xr-x    1 philr    philr      233240 Jan 20 17:11
data.0002.22.115.out
-rwxr-xr-x    1 philr    philr      319144 Jan 20 17:11
data.0002.22.176.out
-rwxr-xr-x    1 philr    philr      267736 Jan 20 17:11
data.0002.22.270.out
-rwxr-xr-x    1 philr    philr      233240 Jan 20 17:11
data.0002.47.115.out
-rwxr-xr-x    1 philr    philr      269752 Jan 20 17:11
data.0002.47.270.out
```

```
-rwxr-xr-x    1 philr    philr      123900 Jan 20 17:11
data.0003.05.094.out
-rwxr-xr-x    1 philr    philr      132944 Jan 20 17:11
data.0003.05.293.out
-rwxr-xr-x    1 philr    philr      123900 Jan 20 17:11
data.0003.15.094.out
-rwxr-xr-x    1 philr    philr      121436 Jan 20 17:11
data.0003.15.198.out
-rwxr-xr-x    1 philr    philr      132944 Jan 20 17:11
data.0003.15.293.out
-rwxr-xr-x    1 philr    philr      123956 Jan 20 17:11
data.0004.06.094.out
-rwxr-xr-x    1 philr    philr      117432 Jan 20 17:11
data.0004.06.198.out
-rwxr-xr-x    1 philr    philr      123956 Jan 20 17:11
data.0005.05.094.out
-rwxr-xr-x    1 philr    philr      132608 Jan 20 17:11
data.0005.05.293.out
-rwxr-xr-x    1 philr    philr      123956 Jan 20 17:11
data.0005.15.094.out
-rwxr-xr-x    1 philr    philr      121212 Jan 20 17:11
data.0005.15.198.out
-rwxr-xr-x    1 philr    philr      132580 Jan 20 17:11
data.0005.15.293.out
-rwxr-xr-x    1 philr    philr      124208 Jan 20 17:11
data.0011.10.293.out
-rwxr-xr-x    1 philr    philr      122024 Jan 20 17:11
data.0012.50.294.out
-rwxr-xr-x    1 philr    philr      122024 Jan 20 17:11
data.0012.90.294.out
-rwxr-xr-x    1 philr    philr      305592 Jan 20 17:11
data.0016.10.067.out
-rwxr-xr-x    1 philr    philr      346892 Jan 20 17:11
data.0016.10.168.out
-rwxr-xr-x    1 philr    philr      651812 Jan 20 17:11
data.0016.25.067.out
-rwxr-xr-x    1 philr    philr      197652 Jan 20 17:11
data.0016.25.283.out
-rwxr-xr-x    1 philr    philr      405776 Jan 20 17:11
data.0016.35.067.out
-rwxr-xr-x    1 philr    philr      463680 Jan 20 17:11
data.0016.35.168.out
-rwxr-xr-x    1 philr    philr      197652 Jan 20 17:11
data.0016.35.283.out
-rwxr-xr-x    1 philr    philr      165396 Jan 20 17:11
data.0017.10.067.out
-rwxr-xr-x    1 philr    philr      246988 Jan 20 17:11
data.0017.10.168.out
-rwxr-xr-x    1 philr    philr      206472 Jan 20 17:11
data.0017.50.108.out
-rwxr-xr-x    1 philr    philr      124768 Jan 20 17:11
data.0017.50.168.out
-rwxr-xr-x    1 philr    philr      314608 Jan 20 17:11
data.0017.50.254.out
-rwxr-xr-x    1 philr    philr      365064 Jan 20 17:11
data.0017.75.108.out
```

```
-rwxr-xr-x    1 philr    philr      121520 Jan 20 17:11
data.0018.06.262.out
-rwxr-xr-x    1 philr    philr      121100 Jan 20 17:11
data.0019.05.262.out
-rwxr-xr-x    1 philr    philr      121100 Jan 20 17:11
data.0019.26.262.out
-rwxr-xr-x    1 philr    philr      121184 Jan 20 17:11
data.0021.05.262.out
-rwxr-xr-x    1 philr    philr      141092 Jan 20 17:11
data.0022.05.130.out
-rwxr-xr-x    1 philr    philr      119420 Jan 20 17:11
data.0022.05.198.out
-rwxr-xr-x    1 philr    philr      141092 Jan 20 17:11
data.0022.20.130.out
-rwxr-xr-x    1 philr    philr      119420 Jan 20 17:11
data.0022.20.198.out
-rwxr-xr-x    1 philr    philr      121268 Jan 20 17:11
data.0022.20.262.out
-rwxr-xr-x    1 philr    philr      404180 Jan 20 17:11
data.0023.08.095.out
-rwxr-xr-x    1 philr    philr      313740 Jan 20 17:11
data.0023.08.255.out
-rwxr-xr-x    1 philr    philr      327320 Jan 20 17:11
data.0023.26.068.out
-rwxr-xr-x    1 philr    philr      221424 Jan 20 17:11
data.0023.26.278.out
-rwxr-xr-x    1 philr    philr      141092 Jan 20 17:11
data.0024.07.130.out
-rwxr-xr-x    1 philr    philr      117348 Jan 20 17:11
data.0024.07.198.out
-rwxr-xr-x    1 philr    philr      121324 Jan 20 17:11
data.0024.07.262.out
-rwxr-xr-x    1 philr    philr      121324 Jan 20 17:11
data.0025.05.262.out
-rwxr-xr-x    1 philr    philr      149212 Jan 20 17:11
data.0030.27.130.out
-rwxr-xr-x    1 philr    philr      148092 Jan 20 17:11
data.0032.05.136.out
-rwxr-xr-x    1 philr    philr      234108 Jan 20 17:11
data.0033.05.197.out
-rwxr-xr-x    1 philr    philr      172032 Jan 20 17:11
data.0033.36.068.out
-rwxr-xr-x    1 philr    philr      157724 Jan 20 17:11
data.0033.36.117.out
-rwxr-xr-x    1 philr    philr      355516 Jan 20 17:11
data.0033.36.167.out
-rwxr-xr-x    1 philr    philr      121436 Jan 20 17:11
data.0041.14.166.out
-rwxr-xr-x    1 philr    philr      121492 Jan 20 17:11
data.0042.17.166.out
-rwxr-xr-x    1 philr    philr      134092 Jan 20 17:11
data.0043.19.059.out
-rwxr-xr-x    1 philr    philr      117180 Jan 20 17:11
data.0047.28.086.out
-rwxr-xr-x    1 philr    philr      121324 Jan 20 17:11
data.0047.28.166.out
```

```
-rwxr-xr-x    1 philr    philr      132468 Jan 20 17:11
data.0049.18.064.out
-rwxr-xr-x    1 philr    philr      279608 Jan 20 17:11
data.0051.05.249.out
-rwxr-xr-x    1 philr    philr      226520 Jan 20 17:11
data.0051.08.059.out
-rwxr-xr-x    1 philr    philr      119504 Jan 20 17:11
data.0051.08.336.out
-rwxr-xr-x    1 philr    philr      226520 Jan 20 17:11
data.0051.28.059.out
-rwxr-xr-x    1 philr    philr      289436 Jan 20 17:11
data.0051.28.129.out
-rwxr-xr-x    1 philr    philr      439544 Jan 20 17:11
data.0051.28.227.out
-rwxr-xr-x    1 philr    philr      124292 Jan 20 17:11
data.0052.14.066.out
-rwxr-xr-x    1 philr    philr      117348 Jan 20 17:11
data.0057.10.294.out
-rwxr-xr-x    1 philr    philr      204932 Jan 20 17:11
data.0058.16.070.out
-rwxr-xr-x    1 philr    philr      204932 Jan 20 17:11
data.0058.21.070.out
-rwxr-xr-x    1 philr    philr      205128 Jan 20 17:11
data.0059.35.070.out
-rwxr-xr-x    1 philr    philr      200816 Jan 20 17:11
data.0060.30.072.out
-rwxr-xr-x    1 philr    philr      213332 Jan 20 17:11
data.0061.13.072.out
-rwxr-xr-x    1 philr    philr      185584 Jan 20 17:11
data.0134.10.151.out
-rwxr-xr-x    1 philr    philr      282184 Jan 20 17:11
data.0151.25.151.out
-rwxr-xr-x    1 philr    philr      185472 Jan 20 17:11
data.0152.25.151.out
-rwxr-xr-x    1 philr    philr      118384 Jan 20 17:11
data.0154.05.227.out
-rwxr-xr-x    1 philr    philr      225092 Jan 20 17:11
data.0154.25.059.out
-rwxr-xr-x    1 philr    philr      308924 Jan 20 17:11
data.0154.25.172.out
-rwxr-xr-x    1 philr    philr      175588 Jan 20 17:11
data.0154.25.271.out
-rwxr-xr-x    1 philr    philr      124376 Jan 20 17:11
data.0155.07.166.out
-rwxr-xr-x    1 philr    philr      153076 Jan 20 17:11
data.023X.05.067.out
-rwxr-xr-x    1 philr    philr      189728 Jan 20 17:11
data.023X.26.067.out
-rwxr-xr-x    1 philr    philr      174300 Jan 20 17:11
data.033X.15.071.out

docu/hrm29:
total 60
drwxrwxr-x    2 philr    philr        8192 Jan 20 15:53 .
drwxrwxr-x    7 philr    philr        8192 Jan 20 17:21 ..
-rwxr-xr-x    1 philr    philr        1609 Jan 20 15:49 distr.f
-rwxr-xr-x    1 philr    philr         251 Jan 20 15:55 harm.f.jcl
```

```
-rwxr-xr-x    1 philr    philr         101 Jan 20 15:54 harm.f.n
-rwxr-xr-x    1 philr    philr       13676 Jan 20 15:49 hrm29.f
-rwxr-xr-x    1 philr    philr        1119 Jan 20 15:49 jdconv.f
-rwxr-xr-x    1 philr    philr         351 Jan 20 15:49 ncrght.f
-rwxr-xr-x    1 philr    philr        3563 Jan 20 15:49 principal.org.f
-rwxr-xr-x    1 philr    philr         531 Jan 20 15:49 uvcomp.f
```