

APPENDIX C

EXAMPLE OFFLINE INTERFACE

The following C source code provides an example of an interface to the FORTRAN functions collectively known as *offline*. The file *track.c* applies the AVHT algorithm to create a DST file containing the tracks from all events with total occupancy of less than 600 active wires. It employs *Chi-Squared Cuts* of 3.0 for both heads and tails as well as *Same-Track Cuts* of 2.5 for heads and 5.0 for tails. The *Makefile* also provided was created for use on *Theory3*, an IBM RISC 6000 at Case Western Reserve University. It may need slight modifications for use on other computers.

C.1 Track.c

```

/*****
AVHT DST creation program for interface to OFFLINE.
  Creates a DST for all events with nhits <= 600 with
  Chi-Squared cuts of 3.0 on head and tails, Same-Track cuts of
  2.5 for heads and 5.0 for tails, and default MinPlanes cuts.

  By Erik Kangas
  Last updated 4/29/95
*/

#include "hougher.h"          /* Include AVHT definitions */
#include "stat.h"            /* Not needed unless doing analysis */

/*****
OFFLINE FORTRAN interface declarations
*/

extern EVTCOM event_;      /* Event common block */
void unpacker_(void);     /* Loads event infor into event_ */

/*****
Specialized tracking variables
You can put your own global variables here.

  det_name -> name of the detector data file
  dst_name -> name of the DST to be created.
  dstf     -> DST file pointer
*/

char det_name[80] = "/minimax/kangas/tracking/data/new24.dat";
char dst_name[80] = "bj677n600.dst";

```

```

FILE* dstf;

/*****
  user_init is called by offline before any tracking is done.
  This is an ideal place to setup variables, etc.
*/

void user_init_(void) {
  setup_hougher_(det_name);      /* initialize AVHT environment */
  dstf = fopen(dst_name,"w");    /* open DST file */

  /* Initialize tracking parameters */

  Verbose = 1;                  /* Ongoing display of hougher info */
  ChiCutH = 3;                  /* Chi-Squared cuts for now */
  ChiCutT = 3;
  MetCutH = 2.5;                /* Same-Track Cuts */
  MetCutT = 5.0;
}

/*****
  user_end is called by offline after any tracking
  This is an ideal place to close files, etc.
*/

void user_end_(void) {
  fclose(dstf);                /* close the DST file */
}

/*****
  user_anal is called for every event.
  you must call unpacker to put the contents of the event into event_
  common block then call LoadEventOFF to copy the common block into
  the tracking event structure. Then you can track the event or
  do whatever you like.
*/

void user_anal_(void) {

  /* You need the following section of code to load in the event */
  /* independent of what analysis or tracking you plan to do */
  /* This loads the event and checks to see if the event is */
  /* ok and if it is interesting */

  unpacker_();
  if (!event_.event_valid && !event_.partial_event) {
    printf("Error from unpacker\n");
    return;
  }
  if (event_.evtype != 1) return; /* Not an interesting event */
  LoadEventOFF(&event_,&event); /* Copy into event structure */

  /* Insert your own analysis code here. If you wish to quit before */
  /* All events are done, call user_end_ and use exit(). */

  if (event_.nwires > 600) return; /* Apply nhits cut of 600 */
  hougher_(); /* Track the event */
  WriteTEvent(dstf,tracks); /* Write results to the DST file */
}

```

C.2 Makefile

```
#
# AVHT Makefile for use with OFFLINE analysis
#
# By Erik Kangas -- April 29, 1995
#
#
# Insert the base name of your 'C' interface file below
# This is the only section you should have to modify.
# You may alter the CFLAGS variable
#

CFILE = track

#
# Following are definitions necessary for compilation / linkage
# - The path to the source files
# - Compiler directives
# - Malloctype corrects for errors in Theory3's compiler
#

HOUGHDIR = /minimax/kangas/tracking/source/
SRC = /minimax/jstreets/offline/
FSLIB_DIR = /minimax/jstreets/products/fslib/v1_4
CRNLIB = /minimax/jstreets/products/cern/v93d/lib

MALLOCTYPE=3.1; export MALLOCTYPE

CCOMP = xlc
F77 = xlf
CFLAGS = -O -I$(HOUGHDIR)
FFLAGS = -I$(SRC) -qextname

.c.o:
    $(CCOMP) -c -o $*.o $(CFLAGS) $<

.f.o:
    $(F77) -c -o $*.o $(FFLAGS) $<

#
# Define our object files
#

OBJECTS = $(CFILE).o $(HOUGHDIR)bjhough.o $(HOUGHDIR)nrc.o\
    $(HOUGHDIR)trackdef.o $(HOUGHDIR)hougher.o $(HOUGHDIR)coords.o\
    $(HOUGHDIR)stat.o

OBJECTS = $(SRC)acdevices.o $(SRC)unpacker.o $(SRC)offline.o\
    $(SRC)swbyte.o

#
# Compile / link the files. Use fortran linker.
#

$(CFILE) : $(OBJECTS) $(COBJECTS)
    $(F77) $(FFLAGS) -qextname -o $(CFILE) $(OBJECTS)\
    $(COBJECTS) $(FSLIB_DIR)/lib/fs_lib.a -L$(CRNLIB) -lmathlib\
    -lkernlib -lpacklib
```

```
#
# File dependencies
#
ALLDEP = $(HOUGHDIR)hougher.h $(HOUGHDIR)trackdef.h $(HOUGHDIR)bjhough.h\
          $(HOUGHDIR)coords.h $(HOUGHDIR)stat.h $(HOUGHDIR)nrc.h

$(CFILE).o: $(CFILE).c $(ALLDEP)

$(HOUGHDIR)bjhough.o: $(HOUGHDIR)bjhough.c $(ALLDEP)

$(HOUGHDIR)hougher.o: $(HOUGHDIR)hougher.c $(ALLDEP)

$(HOUGHDIR)stat.o: $(HOUGHDIR)stat.c $(ALLDEP)

$(HOUGHDIR)coords.o: $(HOUGHDIR)coords.c $(HOUGHDIR)coords.h $(HOUGHDIR)nrc.h

$(HOUGHDIR)nrc.o: $(HOUGHDIR)nrc.c $(HOUGHDIR)nrc.h

$(HOUGHDIR)trackdef.o: $(HOUGHDIR)trackdef.c $(ALLDEP)
```