**HƯỚNG DẪN CHẠY HỆ THỐNG NGHIỆP VỤ MÔ HÌNH COUPLED CCAM**

Việc vận hành hệ thống mô hình coupled CCAM cho dự báo xoáy thuận nhiệt đới (XTNĐ) trên hệ thống tính toán hiệu năng cao (HPC) tại Trường Đại học Khoa học Tự nhiên – ĐH QGHN, bao gồm 5 bước: lấy số liệu điều kiện ban đầu và điều kiện biên dưới, xử lý chuyển đổi số liệu sang lưới bảo giác lập phương của mô hình (CC-grid), chạy mô hình, xử lý đầu ra, cuối cùng là tính toán, và đưa các kết quả lên website (http://meteo.edu.vn/en/csiro). Sơ đồ minh họa về các bước như trên hình 5, chi tiết về cách thức thực hiện trong mỗi bước như bên dưới đây.

**1. Điều kiện ban đầu và điều kiện biên dưới**

Số liệu điều kiện ban đầu được sử dụng cho mô hình CCAM là số liệu phân tích GFSnl (Global Forecast System analysis), số liệu có trong thời gian thực (real-time), với độ phân giải ngang 0,25x0,25 độ kinh vĩ, được lưu trữ luân phiên trong vòng 7 ngày trên máy chủ của NCEP (National Centers for Environmental Prediction), lấy về tại ftp://ftpprd.ncep.noaa.gov/pub/data/nccf/com/gfs/prod. Vì mô hình chạy dự báo khí hậu hạn mùa, phải tích phân liên tục trong một thời gian dài, do đó cần thiết phải cập nhật điều kiện biên dưới SST (nhiệt độ bề mặt biển) cho mô hình. Ở đây, SST được lấy từ dự báo của mô hình toàn cầu CFS (Climate Forecast System), với hạn dự báo lên tới 9 tháng, và với độ phân giải ngang 1x1 độ kinh vĩ, lấy về tại http://nomads.ncep.noaa.gov/pub/data/nccf/com/cfs/prod/cfs. Các script đã được viết để xử lý tác vụ này là (xem nội dung ở phần phụ lục):

./getgfslast.sh  
./newgetsst.sh

**2. Chuyển đổi lưới lon/lat sang lưới CC-grid của CCAM**

Để CCAM có thể đọc được các dữ liệu đầu vào, cần phải chuyển đối lưới lon/lat thông thường của số liệu sang lưới CC-grid của CCAM. Ở đây, hai bộ chương trình Fortran đã được viết với tên gọi là “cdfvidar”, và “one”, để chuyển đổi các số liệu điều kiện ban đầu và điều kiện biên dưới tương ứng. Các script được viết để thực hiện tác vụ này bao gồm:

./runcdfvidar.sh

./newmksst.sh

, thư mục lưu các file đã chuyển đổi ở tại,

/home/csiro/seas/anl/anlgfs.\*.nc

/home/csiro/seas/sst/sst.cfs.01.\*.nc

**3. Chạy mô hình CCAM**

Như đã đưa ra thông tin trong phần thiết kế thí nghiệm, CCAM sẽ được chạy dự báo cho toàn cầu với phân giải ngang khoảng 100km, sau đó được chạy chi tiết hóa cho khu vực Biển Đông Việt Nam với phân giải ngang 20km. Bộ mã nguồn mô hình được viết bằng Fortran với tên gọi “globpea”, trong đó bao gồm rất nhiều các mô-đun con, để giúp mô hình có thể chạy được, từ việc đọc số liệu đầu vào, tham số hóa các quá trình, tính toán sử dụng các node tính một cách tối ưu trên hệ thống (mô hình sử dụng số CPU là bội của 6), …, cho đến việc lưu kết quả đầu ra. Các công đoạn kiểm tra đầu vào, và chạy mô hình CCAM toàn cầu và khu vực được để trong script:

./ncheckin.sh

, xem mô hình đã chạy hay chưa bằng lệnh qstat hoặc qstat -a, nếu ở cột S trên hàng user xuất hiện chữ R thì có nghĩa là mô đang chạy. Tiếp đó, xem mô hình đang tích phân như thế nào thì dùng lệnh tail -f CFS\_2019022512/ CFS\_2019022512.20190225.1200.20190225.1200/pr.20.gfs150\_121.5\_11.5\_20km.20190225.1200 v.d. ở đây đang xem mô hình chạy tại thời điểm 12Z 25/02/2019.

Với việc chạy mô hình CCAM tương tác khí quyển-đại dương (coupled CCAM) thì cũng vẫn cần phải sử dụng điều kiện biên dưới SST dự báo làm đầu vào cho mô hình đại dương, SST dự báo được nội suy theo phương pháp nudging, và có các thiết lập tùy chọn như bên dưới đây trong script chạy mô hình.

nriver=-1

mlo=0 # ocean (0=Interpolated SSTs, 1=Dynamical ocean)

# Downscaling mode - GCM or SST-only

nmlo=-3  
mbd\_mlo=20  
nud\_sst=1  
nud\_sss=0  
nud\_ouv=0  
nud\_sfh=0  
kbotmlo=100

# or # Downscaling CCAM

nmlo=-3  
mbd\_mlo=20  
nud\_sst=1  
nud\_sss=1  
nud\_ouv=1  
nud\_sfh=1  
kbotmlo=-1000

Thư mục và file kết quả đầu ra của mô hình lưu trên hệ thống có dạng: /work/users/csiro/seas/run/CFS\_YYYYMMDDHH/CFS\_YYYYMMDDHH.FYYYFM01.FH00.AYYYAMAD.AH00/llggfs96\_0.\_0.\_105.0km.\*.nc (dự báo toàn cầu) và llggfs150\_121.5\_11.5\_20km.\*.nc (dự báo khu vực). Trong đó, YYYYMMDDHH là thời điểm lúc lấy liệu SST dự báo, FYYYFM tháng được dự báo, và AYYYAMAD.AH là thời điểm lúc lấy số liệu phân tích GFSnl. File kết quả này đã được chuyển đổi sang lưới lon/lat thông thường bằng chương trình ở trong bước 4.

**4. Chuyển đổi đầu ra của CCAM sang lưới lon/lat thông thường**

Mặc định các biến kết quả đầu ra của mô hình sẽ vẫn nằm trên lưới CC-grid, nên để có thể đọc được các biến này trên các công cụ phân tích và xử lý dữ liệu như GrADS, Ferret NOAA, CDO, …, cần phải chuyển đổi ngược lại về lưới lon/lat kinh vĩ thông thường. Một bộ chương trình Fortran đã được viết có tên gọi “pcc2hist” sẽ giúp xử lý việc này. Script chạy tác vụ là: ./cc\_2lonlat.sh

Lưu ý: do bước này đã được lồng vào trong script chạy mô hình (trong bước 3), vì vậy chỉ chạy nếu cần chuyển đổi cho một trường hợp cụ thể khác.

**5. Tính toán, và đưa các kết quả lên website**

Và cuối cùng là công đoạn tính toán từ đầu ra của mô hình coupled CCAM, bao gồm các việc như dò tìm XTNĐ, tổ hợp dự báo, vẽ hình, biểu đồ, …, sau đó đưa các kết quả lên trang web http:/meteo.edu.vn/en/csiro. Cần chạy các script như sau:

1. Vẽ hình các dự báo cho một trạm cụ thể, v.d. Hà Nội (105.8342, 21.0278): ./mkplots.sh
2. Dò tìm XTNĐ: ./TC\_Detection.sh; và vẽ hình TCs tracking: TCcomb+plttracks.sh

Thư mục kết quả dò tìm TCs \*.txt: /work/users/csiro/TC-detection

1. Tính toán tổ hợp: tổ hợp dự báo cho mỗi biến v.d. u, v 10m, mslp, rainfall, ./mkens.sh; chỉ số ENSO ./mknino34.sh
2. Đưa kết quả lên website: cp -rf \*.gif \*.png /mnt/WEB\_DAT/CSIRO/

Tải xuống điều kiện ban đầu là của GFSnl

Tải xuống điều kiện biên dưới SST của CFS

Xử lý chuyển đổi lưới lon/lat sang CC-grid của CCAM

Xử lý đầu ra mô hình, chuyển đổi về lưới lon/lat thông thường

Tính toán từ đầu ra mô hình: dò tìm XTNĐ, tổ hợp dự báo, …, biểu diễn kết quả trên bản đồ, biểu đồ

Đưa các kết quả, hình vẽ lên website, http://meteo.edu.vn/en/csiro

*Hình 1: Sơ đồ vận hành hệ thống mô hình coupled CCAM trong dự báo nghiệp vụ*

**Tài liệu tham khảo**

1. https://confluence.csiro.au/display/CCAM
2. https://bitbucket.csiro.au/projects/CCAM
3. www.hpc.csiro.au/users/244528/ccaminstall
4. http://www.hpc.csiro.au/users/72365/

**Phụ Lục**

**P1**

getgfslast.sh:

#!/bin/sh

ofile=gfsanl\_$yyyy$mm$dd$hh.grb2

if [[ -f $ofile ]]; then

echo "Skip downloading GFSnl ... $ofile"

exit

fi

# check if the datetime is current or not

if [[ $(expr `date +%s` - `date -d "$yyyy$mm$dd $hh" +%s`) -le `expr 7 \\* 24 \\* 3600` ]]; then

$wget ftp://ftpprd.ncep.noaa.gov/pub/data/nccf/com/gfs/prod/gfs.$yyyy$mm$dd/$hh/gfs.t${hh}z.pgrb2.0p25.f000 -O $ofile

else

$wget https://nomads.ncdc.noaa.gov/data/gfsanl/$yyyy$mm/$yyyy$mm$dd/gfsanl\_4\_$yyyy$mm${dd}\_${hh}00\_000.grb2 -O $ofile

fi

exit 0

newgetsst.sh:

#!/bin/sh

ofile=tmpsfc.01.$S\_DATE.daily.grb2

if [[ ! -f $ofile ]]; then

$wget https://nomads.ncdc.noaa.gov/modeldata/cfsv2\_forecast\_ts\_9mon/$yyyy/$yyyy$mm/$yyyy$mm$dd/$S\_DATE/$ofile

else

echo "Skip downloading SST of CFS ... $ofile"

fi

**P2**

runcdfvidar.sh:

#!/bin/bash

echo "runcdfvidar to interpolate GFSnl grib file to model grid"

# init

zsfil=topout192\_0.\_0.\_52.5km

# link data

ln -sf $CCAM\_ROOT/seas/topo/$zsfil .

ls -ltr ../P1/gfs\*grb2

lastgrb=`ls -1tr ../P1/gfs\*grb2|tail -1`

echo "lastgrb=$lastgrb"

if [[ ! -e $lastgrb ]];then exit;fi

ymd=`cdo showdate $lastgrb 2>/dev/null|sed 's/-//g'|sed 's/ //g'`

utc=`cdo showtime $lastgrb 2>/dev/null|cut -d: -f1|sed 's/ //g'`

export utc

dti=$ymd.$utc

echo "dti=$dti"

vfil=anlgfs.$dti.nc

echo "vfil=$vfil `ls -l $vfil`"

if [[ ! -e $vfil ]];then

lastnc=`echo $lastgrb|sed 's/grb2$/nc/'`

echo "lastnc=$lastnc"

if [[ ! -e $lastnc ]];then

cat << EOF > g2n.nml

&splice

outputunit = "pres", "hPa"

levinvert = .TRUE.

splicename = "HGT", "TMP", "UGRD", "VGRD", "RH", "TMP", "PRMSL", "HGT", "LAND", "PRES"

spliceunit = "ISBL", "ISBL", "ISBL", "ISBL", "ISBL", "SFC", "MSL", "SFC", "SFC", "SFC"

newname = "hgt", "temp", "u", "v", "rh", "tss", "mslp", "zs", "land", "ps"

/

EOF

ls -l $lastgrb

echo "g2n -f nc -i $lastgrb -o $lastnc < g2n.nml"

g2n -f nc -i $lastgrb -o $lastnc < g2n.nml

fi

if [[ ! -e $lastnc ]];then

echo "$lastnc does not exist - exit"

exit

fi

cat << EOF > cdf.anl.nml

&gnml

inf='$lastnc'

zsfil='$zsfil'

vfil=$vfil

io\_out=1

inzsavn=-1 inlsavn=-1

nrh=0 mxcyc=20 debug=f nvsig=4 in=40 iout=20 notop=f

oform=f oesig=f ptop=0.

calout=f

in=40 ints=0 inzs=15

opre=f

ntimes=1

splineu=f splinev=f splinet=f

zerowinds=f

kl=18

sgml=0.00445816 0.0216049 0.0541838 0.100137 0.157407 0.223937 0.297668 0.376543 0.458505 0.541495 0.623457 0.702332 0.776063 0.842593 0.899863 0.945816 0.978395 0.995542

kl=27 sgml=0.00445801 0.0251319 0.0599152 0.106927 0.164285 0.230109 0.302518 0.379629 0.459563 0.543664 0.617614 0.680235 0.733264 0.778168 0.816194 0.848394 0.875662 0.898752 0.918305 0.934863 0.948884 0.960757 0.970812 0.979326 0.986536 0.992641 0.997811

&end

EOF

echo "cdfvidar < cdf.anl.nml > log.cdfvidar.$utc"

cdfvidar < cdf.anl.nml #> log.cdfvidar.$utc

fi

exit 0

newmksst.sh:

#!/bin/bash

echo "run newmksst.sh to interpolate SST of CFS grib file to model grid"

one=$CCAM\_ROOT/src/onen/one2

topo=topout96\_0.\_0.\_105.0km

ln -sf $CCAM\_ROOT/seas/topo/$topo .

if [[ ! -e $topo ]];then

echo "$topo does not exist, exit"

ls -l $topo

exit

fi

# takes monthly SST files (netcdf)

# interpolates SST files to CCAM grid

# using provided CCAM topo data (topo below)

# for use by CCAM with namip option

#sst.cfs.01.2017082000.nc

# link data

ls -ltr ../P1/tmpsfc.\*.grb2

ifile=`ls -1tr ../P1/tmpsfc.\*.grb2|tail -1`

lastsst=sst.cfs.01.$S\_DATE.nc

if [[ ! -e $ifile ]];then exit;fi

if [[ ! -e $lastsst ]]; then

# convert grib2 to netcdf

echo "wgrib2 $ifile -netcdf $$"

wgrib2 $ifile -netcdf $$ > /dev/null 2> /dev/null

cdo -O seltimestep,1/11 -settunits,months -monmean $$ $lastsst 2>/dev/null

# remove grib2 file

rm $$

fi

symd=`cdo -showdate -seltimestep,1 $lastsst|sed 's/-//g'|sed 's/ //g'`

sh=`cdo -showtime -seltimestep,1 $lastsst|cut -d: -f1-2|sed 's/://g'|sed 's/ //g'`

echo "symd=$symd sh=$sh"

lyyy=`echo $symd|cut -c1-4`

lm=`echo $symd|cut -c5-6`

flm=`expr $lm + 0`

cyyy=`date +%Y`

cm=`date +%m`

fcm=`expr $cm + 0`

echo "cyyy=$cyyy cm=$cm lyyy=$lyyy lm=$lm"

#iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii

if [[ $cyyy -eq $lyyy && $fcm -eq $flm ]];then

echo "already have this ym"

else

echo "do we need new sst?"

fi

#iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii

echo "###################################> lastsst=$lastsst"

ofile=ts.$topo.$lastsst

rm $ofile

echo "###################################> ofile=$ofile"

#iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii

if [[ ! -e $ofile ]];then

echo "ofile=$ofile does not exist - create"

syear=`echo $symd|sed 's/ //g'|cut -c1-4`

smth=`echo $symd|sed 's/ //g'|cut -c5-6`

cdo -r copy $lastsst rsst.nc

########################################################

# now interpolate sst to CCAM grid for all times in file

########################################################

cat <<EOF > ts.nml

&gnml

inf="rsst.nc"

zsfil="$topo" inzs=15

namein="TMP\_surface" varun="K"

nameout="tos"

ofile="isst.nc"

ifill=0

ntimes=99999

sdiag=f

ids=1,ide=10,jds=1,jde=10

&end

EOF

echo "$one < ts.nml > log.one.$lastsst"

$one < ts.nml #> log.one.$lastsst

ncatted -O -a units,time,m,c,"minutes since 1970-01-01 00:00:00" isst.nc nsst.nc

cdo settaxis,${syear}-${smth}-15,00,43920minutes nsst.nc fasst.nc

cp fasst.nc asst.nc

ncdump -c asst.nc|grep time:unit

cdo sinfon asst.nc

if [[ ! -e asst.nc ]];then exit;fi

ntime=`cdo ntime asst.nc`

echo "ntime=$ntime"

yy=$syear

rm kdate.\*

for ts in `seq 1 $ntime`

do

m=`expr $ts - 1`

mn=`expr $smth + $m`

if [[ $mn -gt 12 ]];then

mn=`expr $mn - 12`

yy=`expr $syear + 1`

fi

if [[ $mn -lt 10 ]];then mn=0$mn;fi

echo cdo -O setrtoc,0,99999999,${yy}${mn}15 -selname,kdate -seltimestep,$ts asst.nc kdate.$ts

cdo -O setrtoc,0,99999999,${yy}${mn}15 -selname,kdate -seltimestep,$ts asst.nc kdate.$ts

done

cdo -O -mergetime kdate.\* kdate

cdo -O -mulc,0. -selname,ktime asst.nc ktime

cdo -O -mulc,0. -selname,mtimer asst.nc mtimer

cdo -O chname,mtimer,timer mtimer timer

echo "cdo -O merge kdate ktime mtimer replace"

cdo -O -merge kdate ktime mtimer timer replace

cdo -O -replace asst.nc replace $ofile

ncatted -O -a units,time,m,c,"minutes since ${syear}-${smth}-15 00:00:00" $ofile a.nc;mv a.nc $ofile

echo "file $ofile has all months of SST anom for this file on CCAM grid"

ncdump -c $ofile|grep time:unit

########################################################

fi # [[ ! -e $ofile ]];then

exit 0

**P3**

ncheckanl.sh

#!/bin/bash

rundir=`pwd`

#hfree=`df --output=pcent /home|tail -1|cut -d\% -f1|sed 's/ //g'`

#echo "hfree=$hfree"

#if [[ $hfree -gt 95 ]];then exit;fi

#echo "okay hfree=$hfree"

echo "get last mean SST file to get year and month of prediction start"

echo "use second to last file as this is for first year of prediction"

lastts=`ls -1 $rundir/../P2/ts.topout96\_0.\_0.\_105.0km\*.nc|tail -1`

echo "lastts=$lastts"

if [[ ! -e $lastts ]];then exit;fi

lasttsym=`echo ${lastts##\*/}|cut -d\_ -f4|cut -d. -f6`

lyyy=`echo $lasttsym|cut -c1-4`

lm=`echo $lasttsym|cut -c5-6`

ym=$lyyy$lm

ym=$lasttsym

echo "lyyy=$lyyy lm=$lm ym=$ym"

lastanl=`ls -1tr ../P2/anl\*nc|tail -1`

ls -ltr $lastanl

ln -sf $lastanl .

numanl=`ls -1tr anl\*nc|wc -l`

echo "numanl=$numanl"

if [[ $numanl -eq 0 ]];then exit;fi

# now for files to pass to run\_ccam.sh

anlfile=`ls -1tr anl\*nc|tail -1`

anl=`echo $anlfile|cut -c4-6`

utch=`echo $anlfile|cut -d. -f3`

etype=CFS\_$ym

echo "anlfile=$anlfile anl=$anl utch=$utch"

if [[ $anl == "" ]]; then exit ;fi

if [[ $utch == "" ]]; then exit ;fi

echo "sbatch run1.sh $anl $utch $etype $ym $lastts 1.0"

sbatch run1.sh $anl $utch $etype $ym $lastts 1.0

**P5**

mkplots.sh:

#!/bin/bash

rundir=`pwd`

#hfree=`df --output=pcent /home|tail -1|cut -d\% -f1|sed 's/ //g'`

#echo "hfree=$hfree"

#if [[ $hfree -gt 95 ]];then exit;fi

#echo "okay hfree=$hfree"

echo "get last mean SST file to get year and month of prediction start"

echo "use second to last file as this is for first year of prediction"

lastts=`ls -1 $rundir/../P2/ts.topout96\_0.\_0.\_105.0km\*.nc|tail -1`

echo "lastts=$lastts"

if [[ ! -e $lastts ]];then exit;fi

lasttsym=`echo ${lastts##\*/}|cut -d\_ -f4|cut -d. -f6`

lyyy=`echo $lasttsym|cut -c1-4`

lm=`echo $lasttsym|cut -c5-6`

ym=$lyyy$lm

ym=$lasttsym

echo "lyyy=$lyyy lm=$lm ym=$ym"

lastanl=`ls -1tr ../P2/anl\*nc|tail -1`

ls -ltr $lastanl

ln -sf $lastanl .

numanl=`ls -1tr anl\*nc|wc -l`

echo "numanl=$numanl"

if [[ $numanl -eq 0 ]];then exit;fi

# now for files to pass to run\_ccam.sh

anlfile=`ls -1tr anl\*nc|tail -1`

anl=`echo $anlfile|cut -c4-6`

utch=`echo $anlfile|cut -d. -f3`

etype=CFS\_$ym

echo "anlfile=$anlfile anl=$anl utch=$utch"

if [[ $anl == "" ]]; then exit ;fi

if [[ $utch == "" ]]; then exit ;fi

echo "sbatch run1.sh $anl $utch $etype $ym $lastts 1.0"

sbatch run1.sh $anl $utch $etype $ym $lastts 1.0

TC\_Detection.sh:

#!/bin/bash

#SBATCH --job-name=TC\_Detection.sh

#PBS -l cput=25:00:00

#PBS -l walltime=25:00:00

#PBS -l vmem=2gb

#wdir=`pwd`

sstymdh=$1

symd=$2

sh=$3

vymd=$4

vh=$5

dom=$6

echo "TC\_Detection.sh sstymdh=$sstymdh symd=$symd sh=$sh vymdh=$vymd vh=$vh dom=$dom"

res=20km; [[ $dom == 96\_0.\_0.\_105.0km ]] && res=105km

# other arch ddir=/mnt/archive/users/csiro/CFS\_$sstymdh/CFS\_$sstymdh.$vymd.$vh.$symd.$sh

#ddir=/work/users/csiro/seas/run/CFS\_$sstymdh/CFS\_$sstymdh.$vymd.$vh.$symd.$sh

ddir=/work/users/csiro/seas/run/CFS\_$sstymdh/CFS\_$sstymdh.$vymd.$vh.$symd.$sh

tcfile=llggfs$dom.CFS\_$sstymdh.$symd.$sh.$vymd.$vh.nc

ifile=$ddir/$tcfile

ls -l $ifile; [[ ! -e $ifile ]] && exit

ofile=$wdir/TC.$res.$sstymdh.$symd.$sh.$vymd.$vh.txt

ls -l $ofile; [[ -e $ofile ]] && exit

#############################

cd $wdir

#############################

year=`echo $vymd|cut -c1-4`

mon=`echo $vymd|cut -c5-6`

echo "year=$year mon=$mon"

### ifile is too big for TC detection, it takes too long.

/bin/rm $tcfile $$.nc

# ncks -O -F -d lev,2,5,1 -v temp,u,v,zg,psl,zs $ifile $$.nc

#need plevs 850 700 500 300

cdo -sellevel,850.,700.,500.,300. $ifile $$lev.nc

cdo -selname,psl,zs $ifile $$sfc.nc

cdo merge $$lev.nc $$sfc.nc $$.nc

rm $$lev.nc $$sfc.nc

ncpdq -U $$.nc $tcfile

################

[[ ! -e $tcfile ]] && exit

ncdump -c $tcfile > dump.$$

grep UNLIMITED dump.$$>jj.$$

cat jj.$$ | cut -d"(" -f2-2 > $$.txt

nrec=`cat $$.txt | cut -d" " -f1-1`

nd=`expr $nrec / 4`

echo "nd=$nd nrec=$nrec"

cat << END1 >info.tcdetect

&DATETIMES

S\_DATE = ${year},

${mon},

01,

00,

E\_DATE = ${year},

${mon},

${nd},

18

IFILE = "${tcfile}"

/

&DOMAINS

NX = 334,

RX0 = 60.70279,

DX = 0.35721,

NY = 144,

RYFILE = "info.ry",

INPUT\_DIR ="1998"

/

&CRITERIA

VorCrit = 1.e-5,

TCorecrit = -1.,

H500crit = -1,

OCScrit = 5.,

SLPcrit = -5.,

CPradius = 300.,

Cradius = 300.,

DEBUG = F

&end

END1

echo "tcdetect.exe"

tcdetect.exe

echo "/bin/mv TCs.txt $ofile"

/bin/mv TCs.txt $ofile

echo "rm $tcfile $$.nc dump.$$ $$.txt jj.$$"

rm $tcfile $$.nc dump.$$ $$.txt jj.$$