Instructions for Running TRACK Program:

Author: Minmin Fu

Date : 10/20/2020

These notes come from discussions with Zhiyuan Li (personal communication on 10/20/2020), discussions with Yangcheng Luo, and detailed instructions from Dr. Kevin Hodges.

Modules used on NCAR Cheyenne:

Currently Loaded Modules:

1) intel/19.0.5

2) ncarcompilers/0.5.0

3) mpt/2.22

4) netcdf/4.7.3

5) python/3.7.5

6) nco/4.7.9

7) ncarenv/1.3

8) tmux/2.9a

9) ncview/2.1.7

10) ncl/6.6.2

Step 1:

Download latest version of track from:

www.nerc-essc.ac.uk/~kih/TRACK/Track.html

Using the login 'track' and password 'katia' the latest version is 1.5.2.

Step 2:

Unzip the tar file

gunzip TRACK-1.5.2.tar.gz

Step 3:

make sure '.' is in your PATH environment variable:

export PATH=${PATH}:.

set the following environment variables:

export CC=gcc

export FC=gfortran

export ARFLAGS=

export NETCDF=[path to your netcdf installation]

Can add this to ~/.bash\_profile

load\_track() {

export PATH=${PATH}:.

export CC=gcc

export FC=gfortran

export ARFLAGS=

}

Step 4:

Recommendation for Cheyenne Users:

Edit Makefile\_linux.in file by changing this line (May also be able to solve this problem by using GNU rather than intel compilers)

NETLIB = -L/glade/u/apps/opt/intel/2019u5/compilers\_and\_libraries/linux/lib/intel64 -lintlc -lsvml -limf -L${NETCDF}/lib -lnetcdf -Wl,-rpath,${NETCDF}/lib

This adds the intel64 library so the compiler can link necessary intel math libraries (libintlc, etc).

Build track.

master -build -i=linux -f=linux

Build tools

make -f Makefile\_linux.in utils

Step 5:

Put necessary files into directories.

Put the zone.dat0, adapt.dat0 and

initial.st files in the TRACK/data directory. Put the RUNDATIN file in

the indat directory and edit it and change the value currently set as

path to the full path of the initial.st file. Put the other files in

the main TRACK directory.

Tips:

RUNDATIN.MSLP\_A file is used to track CYCLONES

RUNDATIN.MSLP file is used to track ANTICYCLONES

I prefer to run the anticyclones and cyclones in separate folders, although they can be done at the same time.

6 hourly instantaneous data is preferred, with SLP being the necessary variable.

Put your netcdf data file in the indat directory or a symbolic link to it there.

Step 6:

First run the cholesky step to setup the spectral filtering using from the TRACK directory:

bin/track.linux -i [netcdf file] -f chol < cholesky.in

this step only needs to be run once as long as the grid resolution or spectral truncation does not change.

Step 7:

Next run the spectral filtering:

bin/track.linux -i [netcdf file] -f filt < specfilt.in

this will produce the file specfil.filt\_band000 in the outdat

directory, this should be *renamed to* the indat directory, e.g.

file\_filt.dat

Check that it is actually using SLP when running. If not, you may need to modify one of the parameters in specfilt.in or use ncks to extract only the relevant variable from the .nc file you are using.

Step 8:

run the feature tracking using:

master -c=[outdir] -d=now -e=track.linux -i=file\_filt.dat -f=[EXT] -j

=RUN\_AT.in -n=1,32,12 -o=[outpath] -r=RUN\_AT\_ -s=RUNDATIN.MSLP

where [outdir] is a unique directory created by the script under the

path specified by [outpath], the [EXT] can be a unique extension for

intermediary output files so that parallel runs don't interfer. The -n

controls the length of the run. You will get some warnings about

missing files but these can mostly be ignored.

Note: it does not run very fast at all, and may need to be run overnight.

This can all be scripted.

If this works you should get the directory [outpath]/[outdir] with the

files ff\_trs\_neg and tr\_trs\_neg files which contain filtered tracks and

all tracks respectively. The filtered tracks are those that last longer

than 1 day and travel further than 1000km, these settings can be

modified if required.

The files in .nc form will also show up in the outdat directory!

N parameter refers to beginning timestep, length of window (in timesteps), and number of windows. There is a small overlap between each window. Check inside each directory in the output directory to see exactly which timesteps each window covered.

It was advised that -n=1,122,240 would well for analyzing 20 years of 6-hourly data all at once, although different variations of these numbers would give similar results.

Example Batch Script:

1 #!/bin/bash -l

2 #SBATCH -J run

3 #SBATCH -N 1

4 #SBATCH -c 1

5 #SBATCH --mem=30G

6 #SBATCH -t 24:00:00

7 #SBATCH -A UHAR0005

8 #SBATCH -p dav

9 #SBATCH -o slurm-%A-%a.out

10 #SBATCH --mail-type=ALL

11 #SBATCH --mail-user=minminjfu@gmail.com

12

13 export TMPDIR=/glade/scratch/$USER/tmp

14 mkdir -p $TMPDIR

15

16 ### Directories

17 export OUTDIR=/glade/u/home/mjfu/zwork/clean\_TRACK/TRACK\_OUTPUT

18 export FNAME=Modern\_W50\_NH\_positive\_20y

19 ### Source

20 source ~/.bash\_profile

21 load\_track

22

23 master -c=$FNAME -d=now -e=track.linux -i=file\_filt.dat -f=ext -j=RUN\_AT.in -n=1,122,240 -o=$OUTDIR -r=RUN\_AT\_ -s=RUNDATIN.MSLP

24

25 cp ./outdat/\*.nc $OUTDIR/$FNAME

26